

Voting Theory for Democracy

Using The Economics Pack
Applications of Mathematica
for Direct Single Seat Elections

Thomas Cool, February 2001

<http://www.dataweb.nl/~cool>

Applications of *Mathematica*

© Thomas Cool, 2001

Published by Thomas Cool *Consultancy & Econometrics*

Rotterdamsestraat 69, NL-2586 GH Scheveningen, The Netherlands

<http://www.dataweb.nl/~cool>

NUGI 681, 854

ISBN 90-804774-3-5

Mathematica is a registered trademark of Wolfram Research, Inc.

Aims of this book when you are new to the subject

The following should have been achieved when you finish this book.

- You will better understand the major topics in Voting Theory - Direct Single Seat Elections.
- You can better decide what particular voting scheme is suited for your purposes.
- When a voting scheme is given to you then you can better determine your voting strategy.
- You can read this book *as it is*, thus also without *Mathematica*. Without ever running a program, you will still benefit from the discussion. However, if you have a computer and practice with the programs, then you end up being able to run the voting routines and interpret their results. Note that the software can be downloaded freely from the internet and be inspected; however, if you want to run it, then you need a licence.

Aims of this book when you are an advanced reader

The following should have been achieved when you finish this book.

- One of the aims of this book has been to develop voting theory from the bottom up so that new readers in the subject get a clear view on it. When you read this book in this way as well, then you will benefit from those aims (see above) and be able to discuss voting theory in this fashion as well.
- You will better understand the similarities and differences of voting and games or matches. The ranking of the candidates is conditional to who participates in the tournament, or the ranking of the items is conditional to the budget. This would refocus your attention to decisions on the budget.
- You will better understand the distinction between voting and deciding, so that an individual vote is a decision too, but an aggregate vote result does not necessarily render an aggregate decision.

- You will refocus to the problems of cheating ("strategic voting") as the root cause for the paradoxes of voting. You will be more challenged to look for schemes that reduce the possibility for cheating.

Note that this book has two agenda's: First to develop voting theory from the bottom up, referring to cheating and sensitivity to the budget. Secondly, to solve the confusions generated by Arrow's theorem. The first objective is more permanent while the second objective is more transient. Once researchers have adopted that solution, there remains little value in teaching new students about old confusions. Yet for the moment it still is an objective:

- You can explain to others that Arrow's verbal explanations of his Impossibility Theorem do not match its mathematics, and, that deontic logic (the logic of morals) shows those verbal statements to be incorrect. You will probably reconsider your view on Arrow's Theorem, and start to see that it is rather irrelevant for group decision making.

Keywords

Social choice, social welfare, welfare economics, economic policy, decision theory, political economy, politics, game theory, testing, matching, ranking, rating, risk, certainty equivalence, general philosophy

Abstract

- The possibility to cheat on a vote causes us to use only ordinal information, but this again causes paradoxes.
- By comparing voting with games and matches, we find a structural identity that allows us to better deal with the voting paradoxes.
- A distinction is made between voting and deciding, so that an individual vote is a decision too, but an aggregate vote result does not necessarily render an aggregate decision.
- From this distinction it follows that Arrow's (1951) impossibility theorem is rather irrelevant for group decision making. Moreover, Arrow's verbal explanations of the theorem appear not to match its mathematics, and deontic logic (the logic of morals) shows those verbal statements to be incorrect.
- Sen's theorem on the impossibility of a Paretian liberal suffers from the same problem, i.e. that the mathematics do not fit the verbal explanations around it.
- The scientific method *by definition* models dynamic reality by rational mechanisms. Social choice can be regarded as rational by definition, and thus the main focus would be on the design of proper procedures - which is the moral issue.
- The *Mathematica* programs develop some consistent constitutions for group decision making and social welfare. You might like some of these. These constitutions violate Arrow's Axiom of Pairwise Decision Making (*APDM* a.k.a *IIA*), but still can be reasonable and morally desirable. The programs become available within *Mathematica* and within The Economics Pack - Cool (1999, 2001) available at <http://www.dataweb.nl/~cool> - and then evaluating:

`Needs["Economics`Pack`"]`

`Economics[Voting]`

Note: You can read this book *as it is*, thus also without *Mathematica*. Without ever running a program, you will still benefit from the discussion. Note that a search on the internet shows the existence of also other voting programs.

Preface

This book originates from the need for an empirical social welfare function in my practical work. But then there was Kenneth Arrow's Impossibility Theorem that created some doubts, and a research proposal was in fact rejected - or rather blocked without rational discussion - with reference to that Theorem. Of course, the practical person proceeds anyhow, but I thought it useful and proper to closely investigate this Theorem. It appeared that the mathematics of the Theorem differs from the common interpretation that is given to the Theorem. I have translated this common interpretation into formulas, making the interpretation now exact, and then it is possible to show, in exact mathematics, that that common interpretation is erroneous. The math of this refutation can be understood at the undergraduate level. Most authors appear not to understand the matter, and most textbooks are plain wrong.

Cool (1999, 2001), "The Economics Pack", contains programs in *Mathematica* for various schemes for Direct Single Seat Elections. This book uses these programs to develop Voting Theory from the bottom up. The focus is on Voting Theory, not on the programs. But if you have these programs available, then you can have a hands-on experience, verify the conclusions, and try your own cases. These programs can be used at an undergraduate level as well, so that the analysis becomes even more accessible.

It must be acknowledged that Voting Theory can become very abstract, so that some of the more fundamental conclusions of this book will require attention again at a more advanced level. If you are such an advanced student then you are free to start with the later chapters. However, given the apparently widespread misunderstandings, you are well advised to work through the book sequentially. It will not hurt to act as if you are new to the subject. In fact, this book follows the strategy to first develop the theory from the bottom up, and only then discuss the theory, so that both ways unite in a clear departure from current misconceptions. You will miss out on that development if you would jump too many chapters.

I thank a former colleague at CPB for discussion on my work in 1990. I also like to thank P. Ruys and A. Storcken of KUB, at that time, for being so kind to subject themselves to the earlier versions of my analysis. I also thank F. van der Blij, while, more recently, B. van Velthoven of RUL graciously gave some of his time. The responsibility for this work remains my own.

Brief contents

19	1. Voting theory and programs
47	2. Getting started
51	3. Items, voters, preferences and morals
85	4. Basic schemes
143	5. Combined schemes
165	6. Strategic voting
193	7. Probability
247	8. Measuring utility
289	9. Theoretical base
343	10. Evaluation of Arrow' s Theorem
387	11. Conclusion
389	Appendix : Hicks 1981
392	Literature
395	Routine Definition Index

Table of contents

19	1. Voting theory and programs
21	1.1 Introduction
21	1.1.1 Social welfare
22	1.1.2 Democratic context
24	1.1.3 Arrow's Impossibility Theorem
25	1.1.4 Cheating
26	1.1.5 The importance of ties
27	1.1.6 Conditions for using this book
28	1.1.7 Structure
29	1.2 If you are new to the subject
29	1.2.1 Aims of this book
29	1.2.2 You can directly use the main result
31	1.2.3 Outline conclusions
33	1.2.4 Problems in voting
34	1.2.5 Undemocratic solutions
35	1.2.6 How to proceed
36	1.3 For the advanced reader
36	1.3.1 Aims of this book
36	1.3.2 Arrow's Impossibility Theorem
38	1.3.3 Pairwise decision making
39	1.3.4 Analogy
39	1.3.5 Cheating vs Arrow's Theorem
41	1.3.6 Example constitutions
42	1.3.7 How to proceed
44	1.4 Overview for both
44	1.4.1 Structure of the book
45	1.4.2 Use of The Economics Pack
46	1.4.3 <i>Mathematica</i>
46	1.4.4 A guide
47	2. Getting started
47	2.1 The first line
47	2.2 The second line
48	2.3 The third line
49	2.4 Using the palettes
49	2.5 All in one line
51	3. Items, voters, preferences and morals
53	3.1 Introduction
54	3.2 Items
54	3.2.1 Using default Items

54	3.2.2	Using CreateNames
55	3.2.3	Arbitrary names
55	3.2.4	Role of the Status Quo
57	3.3	Voters
58	3.4	Preferences
58	3.4.1	Summary
58	3.4.2	Measurement scales
59	3.4.3	The Pref[...] object
60	3.4.4	Preferences as lists of numbers
62	3.4.5	Matrices of preferences
63	3.4.6	Fast entry of preferences
64	3.4.7	Predefined and random preferences
64	3.4.8	Preferences over subsets of items
66	3.4.9	Conversion between Pref and List
66	3.4.10	Sorting items
68	3.5	Morals
68	3.5.1	Introduction
69	3.5.2	Setting values manually
70	3.5.3	Using SetDeontic
72	3.5.4	Objects and Q's with the same structure
73	3.5.5	Universe
74	3.5.6	The difference between Is and Ought
77	3.6	VoteMargin object
77	3.6.1	The vote matrix
80	3.6.2	The VoteMargin[...] object
81	3.6.3	Row sum property for votes
82	3.6.4	Subroutines
84	3.7	Plotting tools
85		4. Basic schemes
87	4.1	Introduction
87	4.1.1	Basic voting schemes
87	4.1.2	Order of discussion
89	4.2	A voting example case
89	4.2.1	Summary
89	4.2.2	Example
91	4.2.3	To ordinal preferences
93	4.3	Pareto
93	4.3.1	Summary
93	4.3.2	Concept
93	4.3.3	Pareto routine
94	4.3.4	Efficiency pairs
95	4.4	Plurality
95	4.4.1	Summary
95	4.4.2	Concept
96	4.4.3	Plurality routine
96	4.4.4	The winner need not be among the ' first two '
97	4.4.5	Plurality fails at ties
98	4.5	Borda
98	4.5.1	Summary
98	4.5.2	Concept
98	4.5.3	The Borda routine

99	4.5.4	Borda neglects the status quo
100	4.5.5	Preference reversals
101	4.5.6	Borda Fixed Point
104	4.5.7	Another example of preference reversal
105	4.5.8	A non – majority Plurality winner and BordaFP
107	4.6	Approval
107	4.6.1	Summary
107	4.6.2	Concept
107	4.6.3	The Approval routine
109	4.6.4	The ToApproval routine
109	4.6.5	Relation to other schemes
111	4.7	Condorcet
111	4.7.1	Summary
111	4.7.2	The concept
112	4.7.3	The pairwise majority routine
114	4.7.4	Binary method versus count method
115	4.7.5	BordaFP solution for the Condorcet cycle
118	4.7.6	Condorcet and Plurality
119	4.7.7	Ties amongst Condorcet winners
119	4.7.8	Subroutines ToVoteMargin and FromVoteMargin
120	4.7.9	Appendix : Other subroutines
123	4.8	Comparing BordaFP and Condorcet
123	4.8.1	Introduction
123	4.8.2	Margin count and Borda
124	4.8.3	Properties due to current programming
125	4.8.4	Example to explain these properties
127	4.8.5	Ties
129	4.8.6	Donald Saari' s approach
132	4.8.7	Dependence on the budget
135	4.9	Voting and graphs
135	4.9.1	Introduction
135	4.9.2	Short introduction to graphs
137	4.9.3	VoteMargin to graphs
138	4.9.4	The Condorcet cycle
140	4.9.5	Graph Intersection and Pareto improvements

5. Combined schemes

143	5.1	Introduction
145	5.1.1	Introduction
146	5.1.2	The Frerejohn and Grether paradox
149	5.1.3	The Majority routine
152	5.2	Pareto Majority
152	5.2.1	Pareto (efficiency) majority
153	5.2.2	On an example given by Sen
154	5.2.3	Example of dependence of the budget
158	5.3	Pareto Pairwise
158	5.3.1	Using the count to break ties
158	5.3.2	The classic Condorcet case
158	5.3.3	Another example
159	5.3.4	Random
161	5.4	Pareto Plurality
162	5.5	Pareto Approval

165

6. Strategic voting

167

6.1 Introduction

169

6.2 Cardinal utility

171

6.3 Cheating

171

6.3.1 Possibility

171

6.3.2 Cheating with the intensity

173

6.3.3 Cheating by order

174

6.3.4 Dealing with cheating

175

6.4 Pareto Majority

175

6.4.1 The Pareto criterion

176

6.4.2 Pareto and costs

177

6.4.3 Pairwise cheating in the second stage

180

6.4.4 Comparing Condorcet and BordaFP

182

6.4.5 A choice on principle

183

6.4.6 A choice on balancing properties

185

6.5 Excursion to equity

185

6.5.1 Introduction

186

6.5.2 Dividing a cake fairly

187

6.5.3 Absolute levels

189

6.5.4 Relative positions

190

6.5.5 Subsistence

192

6.6 Conclusion

193

7. Probability

195

7.1 Introduction

195

7.1.1 The Rasch – Elo index

196

7.1.2 Other angles

197

7.1.3 Testing in general: matching, ranking and rating

198

7.1.4 Consequences of this definition

199

7.1.5 Structure of the discussion

201

7.2 Item Response matrix

201

7.2.1 Definition

201

7.2.2 Random generator

202

7.2.3 Sorted matrices

202

7.2.4 Recovering the probabilities

204

7.3 IR seen as matches

204

7.3.1 Introduction

206

7.3.2 The importance of a tie

207

7.3.3 Rating of difficulty of questions

208

7.3.4 Expected result

209

7.4 Binomial model for multiple choice tests

209

7.4.1 Introduction

209

7.4.2 Probability of passing by just guessing

211

7.4.3 Inappropriate to define competence

213

7.5 Basic concepts

213

7.5.1 Definition of Odds

213

7.5.2 Definition of Logit – and relation to Logistic

214

7.5.3 Logistic function and difference in competence

216

7.5.4 The probability model

217

7.5.5 Probability distance

220	7.6	The Rasch – Elo model
220	7.6.1	The Rasch – Elo or Item Response model
221	7.6.2	Multiplicative odds
227	7.6.3	The assumption of independence
229	7.6.4	'Observational simplicity'
230	7.7	Cycling in matches
234	7.8	The direct approach
234	7.8.1	Using Pr rather than Logistic[d]
234	7.8.2	Derivation from the Logistic
235	7.8.3	For estimation
235	7.8.4	Estimating matches
238	7.8.5	Application to voting
240	7.9	Conclusion for voting
240	7.9.1	Impression on ranking
241	7.9.2	Satisfying the assumptions
241	7.9.3	The paradoxes
242	7.9.4	Evaluation
245	7.10	Appendix

247 8. Measuring utility

249	8.1	Introduction
249	8.1.1	Introduction
251	8.1.2	Structure of the discussion
252	8.2	Development of probability
252	8.2.1	The Pr object
253	8.2.2	Bayes
254	8.2.3	Pr is orderless
255	8.2.4	Prospects
256	8.2.5	Valuing prospects
258	8.2.6	Subroutines
259	8.2.7	Random drawing from Prospects
266	8.3	Risk
266	8.3.1	Prospects and risk
267	8.3.2	Theoretical definition
269	8.3.3	Tests in <i>Mathematica</i>
271	8.3.4	Risk model
271	8.3.5	Risket
272	8.3.6	Prospect plotting
274	8.4	Certainty equivalence
274	8.4.1	The price of a lottery ticket
274	8.4.2	A standard approach
279	8.4.3	A non – standard approach
282	8.4.4	Justification for the non – standard approach
283	8.4.5	Comparing the methods
285	8.4.6	Application to voting
285	8.4.7	A note on independence

289 9. Theoretical base

291	9.1	Theory overview
291	9.1.1	Introduction
292	9.1.2	Notation

293	9.1.3	Implied aggregate ordering
297	9.2	The solution to Arrow's difficulty in social choice
297	9.2.1	Introduction
300	9.2.2	Basic concepts
304	9.2.3	Restatement of Arrow's Theorem
306	9.2.4	A lemma
307	9.2.5	Rejection of the Arrow Moral Claim (AMC)
307	9.2.6	Rejection of the Arrow Reasonableness Claim (ARC)
308	9.2.7	Selection of the culprit axiom
309	9.2.8	Examples of consistent constitutions
309	9.2.9	A reappraisal of the literature
312	9.2.10	Conclusion
313	9.3	Constitution and SWF – GM
313	9.3.1	Ordering vs Choice Set
314	9.3.2	Implied order
314	9.3.3	Conditional generator
315	9.3.4	Ratio of taking subsets
316	9.3.5	Conditional generator and SWF – GM
317	9.4	Renaming and rejecting APDM
317	9.4.1	Introduction
317	9.4.2	The axiom
318	9.4.3	Why we can reject APDM
319	9.4.4	On an example by Sen
319	9.4.5	Discussion
320	9.4.6	Survival
321	9.5	Subset Consistency and Fully – Matched – ness
321	9.5.1	Introduction
321	9.5.2	Condition Alpha : Subset Consistency
325	9.5.3	Condition Beta : Fully Matched
325	9.5.4	Reproduction of Sen (1970 : 39) on APDM
328	9.6	The possibility of a Paretian Liberal
328	9.6.1	Introduction
329	9.6.2	The moral situation
332	9.6.3	Restatement of the example
337	9.6.4	Restatement of theorem and proof
340	9.6.5	Evaluation

10. Evaluation of Arrow's Theorem

343	10.1	Introduction
345	10.1.1	Introduction
345	10.1.2	Looking back at (1990 g)
346	10.1.3	Evaluation
347	10.2	Definition & Reality methodology
347	10.2.1	Mankind as the SWF – GM
348	10.2.2	The world is given
348	10.2.3	Rational reconstruction of the paradoxes
349	10.2.4	Axiomatic method
349	10.2.5	Axiomatic method & empirical claim
350	10.2.6	Does APDM belong to the definition of rationality ?
350	10.2.7	Other elements for empirical work
351	10.3	Arrow 1950
351	10.3.1	The Forsythe – Borda paradox

352	10.3.2	Arrow 1950 paradox
353	10.3.3	Cool 1990 (a)
353	10.3.4	Cool 1990 (b) using the Pareto SWF – GM
354	10.3.5	Cool 1990 : note of caution on Pareto
354	10.3.6	Cool 2000
356	10.4	Frerejohn and Grether paradox
356	10.4.1	Proposal vs alternatives
356	10.4.2	The Frerejohn and Grether paradox
356	10.4.3	Cool 1990
357	10.4.4	Cool 1990 on pairwise – ism
358	10.5	Proof particulars
358	10.5.1	Voting vs deciding
358	10.5.2	Theorem that voting is not deciding
359	10.5.3	Axioms for voting (as differing from deciding)
359	10.5.4	Solution approach
360	10.5.5	Reaction to Sen on APDM
361	10.5.6	Ex falso sequitur quodlibet
363	10.6	Deontic logic vs preference
363	10.6.1	Domain of the preference orderings
364	10.6.2	Constitutional process
364	10.6.3	Deontic logic vs preference
364	10.6.4	Degrees of moral obligation
366	10.7	Some literature on Arrow's Theorem
366	10.7.1	Duncan Black
366	10.7.2	Robert Dahl 1956
366	10.7.3	Jan Tinbergen 1956
367	10.7.4	Leif Johansen 1969
367	10.7.5	Amartya Sen 1986
367	10.7.6	Representative agent 1989
367	10.7.7	Dale Jorgenson 1990
368	10.7.8	Researchers Anonymous 1990
369	10.7.9	Mas – colell, Whinston and Green 1995
369	10.7.10	Conclusion
370	10.8	Sen
370	10.8.1	APDM
373	10.8.2	Pareto and the status quo
375	10.8.3	In “Development as freedom”
379	10.9	Miscellaneous sources for confusion
379	10.9.1	Introduction
379	10.9.2	Arrow and “Arrow”
379	10.9.3	Keeping the logic straight
380	10.9.4	Axiomatic analogy
380	10.9.5	Major paradox of voting
380	10.9.6	Refutation of the verbal claims
381	10.9.7	It is not just rejection of the axioms
381	10.9.8	Mental blocks
382	10.9.9	Evaluating Saari's approach

387	11. Conclusion
389	Appendix : Hicks 1981
392	Literature
395	Routine Definition Index

1. Voting theory and programs

1.1 Introduction

1.1.1 Social welfare

The analysis of social welfare is the key subject of economics. Part of social welfare is determined by the market mechanism in which transactions are conducted with money. Each market transaction is Pareto improving - and a Pareto improvement is defined as a change such that some people advance while nobody sees his or her position deteriorating. Each market transaction is voluntary, and if the participants would not see an improvement, they would not partake in it. This property is so interesting that we would like to see it also in other aspects of social welfare. One such other part of social welfare depends upon group decisions in which voting occurs. This, voting, is the topic of this book. We see it as one of the ways how people aggregate their preferences to arrive at a social optimum. For example, in democracies, decisions on taxes or on government expenditures are influenced by the ballot. Note that there is a third aspect of social welfare, which would consist of plain talk, social conventions, psychology etcetera - but this we do not deal with here (see however Aronson (1992)).

The present book is oriented at clarification of Direct Single Seat Elections. Thus we exclude topics like proportional representation for Multiple Seat Elections and other issues of Social Choice Theory. The elections studied here are also Direct, so that votes directly affect the final choice. In an Indirect system, voters would elect an intermediate body of representatives, who then would enter into discussion and another voting round. An indirect system would introduce all kinds of complexities that we currently do not want to look into.

Though we will not study the indirect system, it may be mentioned that it is a very interesting one. In the U.S., voters elect an electoral body which then elects the President. This setup currently has rather technical flavour, but it could be developed into something of real meaning. Such an electoral body could employ more complicated voting techniques that would be infeasible for the whole population - and such a layered approach would keep matters simple for the average voter and save costs too, while it would also allow for a greater degree of sophistication in the final election. In a way, such a system already exists in many European countries, where voters elect a Parliament that

then elects the Prime Minister. This European system warrants that Parliament and Prime Minister have the same electoral base, which prevents that both claim to be elected by the people - but with opposing views. This short discussion, however, only indicates the complexities, which, again, will not be the topic of this book.

Our intended audience is a general public of economists and related professions and students in those fields. We assume that you know what a utility function is (or at least how economists use the construct), or that you are willing to develop your knowledge of such things alongside with working on this book.

1.1.2 Democratic context

In some countries, dictators hold elections and get elected with 99% of the vote. It is dubious what the value of such exercises are. In old British universities it apparently has been the practice, when it had to be decided who would be appointed professor, to have a normal vote first, and then a second round where everyone voted for the winner of the first round, so that the final decision was 'unanimous'. The catalogue of human customs is endless.

This book will assume that voting takes place in a democratic context in which voting is not a mere ritual, in which voting is about real issues, and where the voters are free.

This assumption is not without meaning and not without consequences. One of the pitfalls of Voting Theory is that the theory can become very abstract and lose sight of essential properties of voting as a mechanism in democracy and for democracy.

One such abstract theoretical question then can be: "If the majority prevails, what is to stop it from exploiting the minority?" Well, if we study voting with the assumption of democracy in the background, then such a question is rather out of focus. The question mistakes a technical formulation for a moral principle, and forgets the context in which the technique is applied.

It appears that authors and students of Voting Theory indeed tend to confuse the techniques with the proper moral context. For this reason it is useful to refer to Hart (1961, 1997). Hart's book is advised reading in general on the relationship of law and morals, but the following points can be recalled here

usefully. When we wonder why people should live together, form a social group, and install a system of justice, then Hart calls attention to these five 'truisms' (p194-198)

- (i) *Human vulnerability*. "There are species of animals whose physical structure (including exoskeletons or a carapace) renders them virtually immune from attack by other members of their species and animals who have no organs enabling them to attack. If men were to lose their vulnerability to each other there would vanish one obvious reason for the most characteristic provision for law and morals: *Thou shalt not kill*."
- (ii) *Approximate equality*. "Even the strongest must sleep at times and, when asleep, loses temporarily his superiority. This fact of approximate equality, more than any other, makes obvious the necessity for a system of mutual forbearance and compromise which is the base of both legal and moral obligation."
- (iii) *Limited altruism*. "But if men are not devils, neither are they angels; and the fact that they are a mean between these two extremes is something which makes a system of mutual forbearance both necessary and possible."
- (iv) *Limited resources*. "It is a merely contingent fact that human beings need food, clothes, and shelter; and these do not exist at hand in limitless abundance; but are scarce, have to be grown or won from nature, or have to be constructed by human toil. These facts alone make indispensable some minimal form of the institution of property (though not necessarily individual property), and the distinctive kind of rule which requires respect for it."
- (v) *Limited understanding and strength of will*. "'Sanctions' are therefor required not as the normal motive for obedience, but as a *guarantee* that those who would voluntarily obey shall not be sacrificed to those who would not. To obey, without this, would be to risk going to the wall."

Hart usefully adds: "The simple truisms that we have discussed (...) are of vital importance for the understanding of law and morals (...)". Of course, it is another matter how such a system of law evolves into a democracy. However, a democracy still is subject to above 'truisms', and having them recalled here, should protect us against thinking that the technical formulation of a voting rule would be the only rule relevant for its social application.

In other words, if we evaluate the voting schemes below, then some theoretical questions might pop up, purely from the technical formulation of the schemes - like for example the question why the majority would not exploit the minority. Such theoretical questions however can distract from the real purpose why we study voting - and such questions should not be mistaken for the true questions that are relevant for an evaluation. It can happen that a majority exploits a minority, but if they do so, then they surely do not need a voting rule to do so.

Some of the questions that have been generated by technique are Arrow's Impossibility Theorem and Sen's Theorem of the Impossibility of a Paretian Liberal. We will show below that there is a difference between the math on one hand and the verbal explanations and the intended applications on the other hand. These thus are typical examples of misguided theorising.

1.1.3 Arrow's Impossibility Theorem

One of the key topics of our discussion will be Kenneth Arrow's (1951) Impossibility Theorem on constitutions. Arrow claims that there are some axioms that are each reasonable and morally desirable when considered by themselves separately, but that generate an inconsistency when we try to combine them. Thus it would be impossible to attain an ideal situation. In 1951 Arrow wrote:

"If consumers' values can be represented by a wide range of individual orderings, the doctrine of voters' sovereignty is incompatible with that of collective rationality."

Over the years that suggestion has grown into a claim, and this has made the logical and moral fixture ever and ever greater. This interpretation of these axioms and the repetition of this by other authors, has created an amazing tension within economic theory and the profession. Many see collective rationality and consumer sovereignty as either innocuous or necessary, but then they apparently want something impossible, and thus they have conceptual difficulty with Arrow's result.

Dictatorship is one possible conclusion that some people draw from this. This need not be a dictatorship by one person, but it rather would be seen as the imposition of one moral view on society, so that social welfare would no longer be sensitive to the flux of individual opinion. In this way, it has become a key issue in Social Choice Theory to determine whether the social optimum is

'given' to mankind or still can depend upon personal opinions. This book, then, rejects the assumption of a dictatorship - which explains why the title uses the label "for democracy". Our objective is to help you to find your preferred voting scheme to express your views. The use of *Mathematica* programs enhances your abilities to do so.

This book will accept the pure mathematics of Arrow's Theorem that cause the contradiction, but we will reject the claim that the axioms would be reasonable or morally desirable. The explanation of this will take some of your time, but you are advised to follow the discussion closely, since you should base your opinion on what is reasonable or morally desirable on your *own* evaluation rather than on some authority.

For some people, Arrow's Theorem seems to support the notion that there would not be an ideal system for social decisions. And if there is no ideal, then one would conclude to value-relativism. In itself, value-relativism is an attractive proposition to the sceptical mind. However, it would be a confusion to say that Arrow's Theorem proves value-relativism. Arrow's axioms are not reasonable and neither morally desirable, so they cannot be used to disprove an ideal. Value-relativism can be accepted, but it would be based on the notion that we respect people and their views. Once we accept value-relativism, then people are free to pursue their own ideals. Which then still may exist (be consistent).

1.1.4 Cheating

Sometimes it is said that the basic problem in Voting Theory is caused by Arrow's Theorem. This is not true. The basic problem is caused by the possibility that people can cheat with their vote. When we use money in the market place then cheating is controlled by the police, and it is generally possible to verify whether a banknote is forged or not. For a vote, we cannot look into your heart, and we have to presume that you vote for what you stand for. There are various ways to do something about cheating in voting - like having people stand up and having them explicitly say their vote (which uses the penalty of reputation). However, this does not always work, while secret ballots are an important good, and strategic voting - a nice word for cheating - then is possible and will affect the result.

The possibility of cheating also shows why the assumption of 'cardinal utility' has limited value. With cardinality, the preferences of the people become like

weights, that we can put on a balance and simply add up (or Nash multiply). If people would not cheat, we could simply ask everyone's preference (and check that it is measurable if they say so). But if someone could misrepresent a weight for a preferred choice, then the total would not be true. Since we as economists assume that people are rational, we must presume that people will cheat (at times). And thus the assumption of (cardinal) interpersonal comparable preferences, which seemed so promising, meets with a problem. Part of the problem is also that cardinal utility, if it exists, must be measured by someone, perhaps by some bureaucratic institute, and this could cause some new problems of its own. If people would not cheat, perhaps that institute might.

The various voting schemes have been proposed precisely since the possibility of cheating is such a problem. The schemes generally limit the impact of cheating. How they do that, will be discussed below. Secondly, these imposed limitations also create their own paradoxes of voting.

1.1.5 The importance of ties

Ties can be a crucial issue for voting theory, but not always. When we have a tie based on indifference, where nobody cares, then we may as well flip a coin. Ties only become crucial when there are strongly opposing views. These then would be the hard choices, where always someone has to suffer. Looking at practical situations in reality, we find that different cultures adopt different solutions. In the U.S., it is more common that the majority takes advantage of its position. In Holland, the solution often is to talk longer, look for compromises, do more research, etcetera. Indeed, in general, a good tie-breaker could be to let the status quo persist, until a solution is found that is acceptable to all - though the day of reckoning of course cannot always be postponed.

One property of tie-breaking rules is that they might make the decisions more sensitive to the actual budget under consideration. For example, in one case there is a clear preference for topic *B*, which then is preferred over status quo *A*. But in a slightly different case, the group is richer, and can also consider possibility *C*. Now, however, a tie occurs, and because of the tie-breaking rule the status quo *A* persists. Clearly, this is 'paradoxical', since on one hand the group has become richer and on the other hand it selects an item that earlier was considered inferior. In some respects, Arrow's "Impossibility Theorem" codifies this property and turns it into a conclusion that there exists no good general decision method. My view is to reject the usefulness of such

mathematics, since it adds nothing to the observed problem, since it suggests a criterion for 'goodness' that is not relevant in practice so that it becomes misleading, and since it freezes one assumption while it neglects the fact that we, once we observe such a tie, have more options open to solve it, depending upon time and circumstances.

It appears that Voting Theory can only provide suggestions for solution approaches. In the end, the group itself must decide what to do in actual situations. Yet, the fact that theory cannot advise on a clear 'universal' tie-breaking rule, and the fact that theory cannot decide for you, should not cause you to conclude that there would exist no ideal. What you consider ideal, namely, is up to you.

1.1.6 Conditions for using this book

The basic requirement for using this book is that you have at least a decent undergraduate level of understanding of mathematics and economics or are willing to work up to that level along the way.

You can read this book *as it is*, thus also if you do not have *Mathematica*. Even without ever running a program, you will still benefit from the discussion.

Yet, if you have *Mathematica* and want to run the programs, then this book assumes that you have worked with *Mathematica* for a few days. You must be able to run *Mathematica*, understand its handling of input and output, and its other basic rules. Note that *Mathematica* closely follows standard mathematical notation. There are some differences with common notation though since the computer requires very strict instructions. Note also that *Mathematica* comes with an excellent Help function that starts from the basic "Getting Started" and works up to the most advanced levels. There are also many books that give an introduction.

When you want to run the voting programs, you should also have a working copy of The Economics Pack, applications for *Mathematica*, by the same author and available as Cool (1999, 2001).

1.1.7 Structure

This book allows for both beginner and advanced readers. Chapter 1.2 starts for the beginning readers. Advanced readers would tend to start with chapter 1.3.

If you have done the beginner chapters and have become interested in voting theory, then you should study some of the serious textbooks in the field (advised are Mueller (1989) and Sen (1970).) After that, you would benefit from chapter 1.3 as well. But if you are new to the field, you should not bother with chapter 1.3 just now.

Once you have mastered these issues, you will find the more complex Parts 9 and 10 of the book that may require more work and some additional study using the library. This part of the book would be directly interesting for advanced students. But even if you are an advanced student, then you are still advised to work your way up, since some points are rather subtle and easily overlooked, particularly in relation to the new programs that are presented here.

Since various *Mathematica* programs are provided, you can have an hands-on experience, and this will allow you to better understand the issues. Since both beginner and advanced readers will be new to the specific formats of these programs, these chapters are advised reading for all.

1.2 If you are new to the subject

1.2.1 Aims of this book

This is only to remind you of the aims set out on the first page of the book.

1.2.2 You can directly use the main result

Since *Mathematica* is so easy to use, you can directly use the main routine and main result of this book. A small example gives a direct introduction into the voting issues and it shows how you can apply the routines. (See Part 2 "Getting Started" first if you really want to run the programs.)

Suppose that there are four friends, **Charlene**, **Chuck**, **John** and **Sue**. Suppose that the group wants to decide about studying or not. The main alternatives are partying, travelling, playing music, or watching a TV movie combined with some study. Chuck wants to study because there are soon exams, but he would accept a TV break. He actually thinks that playing some music would be relaxing for his nerves about the exams. Charlene is the party animal, and the others have mixed preferences. The following is a quick implementation in terms of the voting routines provided in the Economics Pack.

- First you specify what items the vote is about. It is optional to sort them.

```
Items = {travel, study, party, TV, music} // Sort;
```

- You also have to specify what the status quo is. If the group cannot come to an agreement, the status quo persists. In this case, the status quo might consist of an old plan dating from last week. You should be aware that specifying the status quo *after* the vote is often considered unfair.

```
StatusQuo[] = TV;
```

- Each person specifies his or her preferences. The order is like "smaller than" (\leq), meaning that the first item is least preferred and that the last item is most preferred. For example:

```
Charlene = Pref[study, travel, TV, music, party];
Chuck = Pref[party, travel, TV, study, music];
John = Pref[travel, music, study, TV, party];
Sue = Pref[travel, study, TV, music, party];
```

- These Pref objects must be transformed into the Preferences matrix. The order of voters is arbitrary, but now becomes fixed.

```
SetPreferences[{Charlene, Chuck, John, Sue}];
```

- There could be political factions with different numbers of votes. Unless such different votes have been assigned, SetPreferences assigns equal votes to everyone. Let us check this.

```
Votes
```

$$\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}$$

- And that is it. You can call a vote.

```
Vote[]
```

```
{StatusQuo → TV, Pareto → {TV}, Select → TV}
```

Some readers would already have guessed this result. In this case the added value could look small. However, one of the advantages of using this whole setup is that you can analyse how the vote came about. It appears that Chuck blocks a change from the status quo. The (current) Vote rule is that minorities can block a change that is a deterioration for them from the status quo. Another value of the implementation in *Mathematica* is that there are more possible voting schemes, and that you are confronted with the question what scheme to use.

- Under Plurality voting, the group would have had a party (with 3/4 majority).

```
Plurality[]
```

$$\left\{\text{Sum} \rightarrow \begin{pmatrix} \text{music} & \frac{1}{4} \\ \text{party} & \frac{3}{4} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{1}{4} & \text{music} \\ \frac{3}{4} & \text{party} \end{pmatrix}, \text{Max} \rightarrow \left\{\text{party}, \frac{3}{4}\right\}, \text{Select} \rightarrow \text{party}\right\}$$

- If travel had been the status quo (e.g. the plan that they had made last week) then there would be three alternatives that would be *both* acceptable to all *and* an improvement for someone (Pareto points). A majority vote on these improving points would mean that the preferences on these improving items would have been weighed by their rank-order, and this would result into playing music.

`Vote[travel]`

`{StatusQuo → travel, Pareto → {music, travel, TV}, Select → music}`

1.2.3 Outline conclusions

This example allows some early conclusions on the content and relevance of this book:

- Voting occurs everywhere. Groups are everywhere, and groups have to make decisions continuously. Formal voting might be a rare occasion, but another view is that voting occurs so often that we hardly notice it unless we declare a formal occasion.
- The voting routines discipline us on the aspects involved in voting. We must decide on the Items, the Status Quo, the number of voters, their weights, their Preferences, and this all apart from issues concerning the voting scheme itself.
- Often, the major result of such a process is that we start thinking about what the status quo actually is, and what the alternatives could be. Often we have impressions about people's preferences, but rather than simply assuming these and voting on these, the major effort consists in formulating a proposal that gains more support. For example, one possible rule is that people can propose their own candidates - normally their favorite but also compromise candidates. And the naming of candidates is already an indication of preference. Another important issue is what the real structure of preferences is. Are preferences merely "more is better" or aspire people at a balance between challenges and capacities ?
- The Pareto principle appears to be quite important for voting. If the group doesn't want that Chuck drops out, the group has to allow him to veto something that he regards as a (possible) deterioration. The principle of safeguarding minority rights is that majority voting should only be applied to points that are Pareto improving. This is also where the appeal of majority voting comes from. Majority voting can help to resolve the

indecision about what to select from various Paretian points. It would be a misconception to think that majority voting would be acceptable by itself for non-Paretian points. (That, namely, would be a political view that is not necessarily accepted by the minority.)

- The properties of the voting schemes are quite varied. For example, for Plurality voting it suffices that everyone mentions his or her most preferred choice. For the (default) Vote scheme, everyone has to order all items in their order of preference. The latter is more labourious, it can have more errors, and people might be seduced to cheat on their true preferences. So you have to learn to balance the pro's and con's.
- Having these various routines available, you can quickly run alternative schemes, and judge their properties. This will help you to determine what scheme suits your purposes.
- For practical purposes, we currently are only interested in finding the winner, and we are not interested in fully ordering the candidates. (Though see the advanced discussion.)
- As said before: we will consider Direct Single Seat Elections here. Thus we don't consider Indirect or Multiple Seats cases.
- Note that voting theory disregards the use of prices as an instrument for decision making. However, compensation payments are allowed to construct package deals - which could introduce the price mechanism via the back door. Keynes once compared the stock market to a beauty contest: where the voters are mainly trying to predict what the other voters will do. This angle we shall not pursue here.
- *Mathematica* is a nice environment to discuss voting theory. It takes away all the tedious computation, and it allows you to concentrate on the argument. It is another question whether *Mathematica* is a good environment to do the calculations for actual voting situations. Presumably, there can be occasions where *Mathematica* could be used, but, given human psychology, a quick adoption by our Parliaments or shareholder meetings can be doubted.

1.2.4 Problems in voting

One of the important challenges in Voting Theory is that some situations can be very paradoxical. This may have caused that Voting has called the attention of various interesting historical figures, like the Marquis de Condorcet 1785 (known from the French Revolution), his opponent J.C. de Borda 1781, and Charles Dodgson 1876 (a.k.a. Lewis Carroll, the author of "Alice in Wonderland").

In 1785 the Marquis de Condorcet discovered the existence of the paradoxes of voting. Let us consider "the Condorcet case", with a Parliament consisting of three parties and three topics on ballot, while the numbers of seats and the preferences are as in Table 1.

Parliament decides to vote first on the pair $\{A, B\}$, which gives B as the winner, so that $A < B$. Then the pair $\{B, C\}$ is taken, and C is the winner, so that $B < C$. Then, to round it off, the pair $\{C, A\}$ is taken, and A is the winner, so that $C < A$. Collecting all results, we get $A < B < C < A$, which situation is called a "cycle".

Table 1: Condorcet 1785

Party	Seats	Low	Mid	High		A	B	B	C	C	A
Red	25	A	B	C			25		25	25	
Green	35	C	A	B			35	35			35
Blue	40	B	C	A		40			40		40
Total	100					40	60	35	65	25	75
					Win		B		C		A

What would you do in a situation like this ?

NB. If this is the first time that you have heard about this kind of problem, or if you have not yet really thought about it, then you should put this book to a rest for a moment, and write down your own possible solutions. You should really do this, since this is a nice opportunity to match up your intuitions with those of some Nobel Prize winners. Then, it would be nice if you would consider the case of a dogfood experiment as well. The first day the dog can choose from dogfoods A and B , the next day from B and C , and the third day from A and C . He chooses in a cycle as above. Do you explain this by the conclusion that the dog is confused ?

In 1950 Kenneth Arrow first posed a similar problem and then in 1951, in his "Social Choice and Individual Values", proved a theorem that certain axioms

result into a contradiction and thus cause an impossibility. The logic of the theorem is sound, has been tested by many, and can be accepted by us. Arrow also claimed that his theorem would mean that there would not exist social welfare functions that are both *reasonable* and *morally desirable*. This claim has caused quite some confusion in the literature.

Condorcet's paradox is clearly remarkable, but are you willing to conclude that you cannot find a voting method that you would consider reasonable and morally desirable ? Many Nobel Prize winning authors claim that there would not be any such method.

The message of this book however is that you can breath freely again. However, since you have to accept the impossibilities of Arrow's Theorem, you must think through carefully what you want to make of this. The impossibility means that you always must reject one of the axioms. That is true for a fact. The problem is to decide which of these. This book provides a suggestion and arrives at a result that many would consider both reasonable and morally desirable. But be aware that we cannot decide for you. The book only tries to *help* you with the decision process that you and your group have to go through. The main conclusion remains that it is up to you and your group what you consider reasonable and morally desirable. Theory cannot decide this for you. Theory can clarify the aspects that affect your decision, but cannot take that decision away from you.

But now back to the Condorcet case: how would you solve it ?

1.2.5 Undemocratic solutions

Some people grow so wary of the voting paradoxes that they resort to undemocratic methods to solve them. An amazingly popular conclusion is to accept dictatorship. This is perhaps too simple an example, since it is so easy to reject. Let us make the problem a bit more difficult. Consider the following system:

1. The chairperson assigns a number to all voters in the meeting, starting with 1 for himself or herself, 2 for the next in line, etcetera. (This may be done randomly.) The chairperson then proposes an item.
2. The person next in line may propose an alternative. If he or she does not propose one, then the next in line may propose an alternative. Etcetera. If

the middle voter does not propose an alternative, then the item under proposal is elected.

3. If the proposal meets an alternative, then a simple pairwise majority vote is held, and the winner becomes the new proposal. Alternatives may again be suggested, starting with the chairperson. Alternatives that have already been rejected may not be proposed again.

Can you pinpoint, exactly, why this system is undemocratic ?

1.2.6 How to proceed

If you think that you are an advanced student in Voting Theory, then you should continue with chapter 1.3.

Otherwise, if you are beginning, you should continue with chapter 1.4 and work up to and including Part 6. If you know basic probability theory, then you can continue with parts 7 and 8. Parts 9 and 10 would be off-limits for a while. You should practice a lot, and use the examples in this book and find other ones that teach you about the different properties of the various voting schemes. Only afterwards, and only if you are willing to enter into the advanced level, then you could start reading chapter 1.3, and then continue with Part 9 and 10. But you should also use other textbooks on voting theory, since the text in those parts presumes some knowledge. It is good to read those other textbooks, since reading these will clarify to you that those other books *don't give you a hold* on the problem while *this book does*.

1.3 For the advanced reader

1.3.1 Aims of this book

This is only to remind you of the aims set out on the first page of the book.

1.3.2 Arrow's Impossibility Theorem

It is said that one of the major intellectual results of 20th century would be Arrow's Theorem. In 1951, Kenneth Arrow formulated a set of axioms that many would consider reasonable and morally desirable, and he then showed that these axioms result into a contradiction. The conclusion would be that there would exist no good Social Welfare Function Generating Mechanisms (SWF-GM) - and by implication constitutions - and this is indeed accepted by many.

The following terms will be used:

- The Social Welfare Function (SWF) is of the Bergson-Samuelson type, and is directly defined over the commodity domain.
- The Social Welfare Function Generating Mechanism (SWF-GM) is of the Arrow type, it is defined over the preferences of the individual agents and it constructs the aggregate preference.
- A constitution (Social Decision Function (SDF)) determines the best element(s) in the budget set. These elements form the Choice Set of the budget.
- A Constitutional Ordering (CO) is an ordering that arises from applying the constitution (SDF) on subsets of the budget or to subsequent budgets. Note that there are various types of such CO's, and the most important one is the one conditional to the existing budget.

In the literature, the word 'constitution' sometimes is used for the SWF-GM, but given the conventional concept of a constitution it is better to associate the word with the SDF. The existence of a SWF-GM is a sufficient but not necessary condition for a constitution (SDF). Thus it could be argued that Arrow's analysis would not be relevant for constitutions. However, Arrow speaks about real constitutions himself, and his theorem clearly points to inconsistencies for CO's as well. If we adopt, for theory's sake, the additional requirement for

constitutions that their CO's should generate an ordering, then the situation is equivalent to Arrow's Theorem, and then no 'reasonable and morally desirable' constitution would exist. Once this distinction is clear, we may as well focus on the SWF-GM again.

Can we really hold that there would exist no reasonable and morally desirable SWF-GMs ? In my analysis the body of current economic analysis on this topic is rather misguided. While Arrow's Theorem is mathematically sound, there still is the matter of interpretation. It is just an assumption that the axioms would be reasonable and morally desirable. Considering them carefully, it appears that we can reject that view. It is possible to define good SWF-GMs and constitutions, i.e. reasonable and morally desirable. This books implements a couple of them.

The basic view is that we live in a dynamic world and that the budget changes regularly. There could occur historic preference reversals, but anyone who knows that particular historical development need not consider this unreasonable nor morally bad. Since the scientific method *by definition* models dynamic reality by rational mechanisms, social choice can be regarded as rational by definition, and thus the main focus would be on the design of proper procedures - which is the moral issue. The main conclusion remains that it is up to you and your group to decide what you consider reasonable and morally desirable. Theory cannot decide this for you. Theory can clarify the aspects that affect your decision, but cannot take that decision away from you.

Arrow's imperative impossibility and cynical implications thus are replaced with the freedom to choose from a wide range of possibilities, while maintaining our reasonableness and moral integrity.

This conclusion is radically different from Arrow's conclusion. Arrow suggests that you have to settle for a suboptimal situation. However, in my analysis you could get precisely what you want, given the properties of reality and group decision making. Thus, in my analysis, the suggestion that you lose something valuable is absent. The discussion in this book shows that a scheme like Pareto-Majority (with a Fixed Point Borda for the Pareto points) might be acceptable to classical liberals, namely in that it protects minority rights. This still has attractive properties for us. But this is just an example, and for you and your group the conclusion could be different.

Arrow's presentation of his theorem has had a rather negative impact on economic theory. Many authors have concentrated on the 'impossibility' that it created. There has been a percolation into all of economics, where teachers have been radiating to their students that 'the ideal is impossible'. Nobody explicitly says so, but a bright student can hear his or her teachers thinking: "If you aspire at the ideal, you don't understand mathematics."

For some of us it may well feel like freedom regained when it is realised that Arrow just has the wrong perspective. Kenneth Arrow recieved a Nobel Prize for his work - also in many other areas in economics - and we are accustomed to attach great value to authority. But his presentation of the problem is incorrect. What is important is whether his assumptions are relevant. And then it appears that they don't apply to group decision making.

1.3.3 Pairwise decision making

We reject Arrow's Axiom of Pairwise Decision Making (*APDM*). Arrow himself called this the "Axiom of Independence of Irrelevant Alternatives" (*AIIA*). This new name *APDM* however is much clearer about what the axiom really means in normal English. Since this renaming is a significant departure from the common literature, I have added a separate chapter (9.4) to clarify this choice of words.

The constitutions programmed in *Mathematica* violate *APDM*. Thus, we do not only reject *APDM* for SWF-GM's but also for constitutions (SDFs). *APDM* is a wrong way to deal with the conditional dependence of orderings on the budget set.

A realistic decision maker cannot accept this axiom. For example, in a choice among three possibilities *A*, *B* and *C* the group choice on only two items, such as $\{A, B\}$, *must* include the votes on the other item, and thus the choice is not independent of these. The reason for that dependence is that if one can determine a voting cycle then the group decision is indifference rather than some preference. We see here the subtle difference between *voting* and *deciding*. It is no problem that voting patterns show a cycle (or indecision). What counts is that the constitution results in a consistent decision. (Perhaps we should speak about Aggregate Decision Theory - but we keep the name Voting Theory.)

Note that since we reject Arrow's axiom, we are consistent. The "voting paradoxes" are paradoxes and no real contradictions. (The dictionary has "paradox" = "*seeming* contradiction".)

1.3.4 Analogy

There is an easy analogy for our rejection of *APDM*.

Consider a person who can have a modest income (M) or who can be well off ($\neg M$, with \neg negation). With a modest income he prefers to stay at home (H) for the holiday, and to spend the money on more expensive dinners (D). If the person is well to do, then he prefers a holiday in a foreign country ($\neg H$) but he also wants to save money on expensive restaurants ($\neg D$). Thus the decision on $a = \{H, D\}$ versus $b = \{\neg H, \neg D\}$ depends upon the income. At first it might look strange that a relatively rich person might eat in cheap restaurants, but once we understand that he has other bills to pay, the paradox is solved. Economists have learned to present such situations in such manner that the emphasis is not on the paradox, but on the rationality of the situation. For example, the person has the choice between $\{H, D, \text{leisure}, M\}$ and $\{\neg H, \neg D, \text{work}, \neg M\}$. My proposal is that economists adopt the same attitude with regard to the voting paradoxes. For a group decision, there can be a dependence on the budget set. Let us accept this and stop saying that this would be 'paradoxical'.

Note that the mathematics of 'individual object dependence' is the same as the mathematics of 'social subject dependence'. In a machine, the relation of two parts can depend upon a third part. In voting, the relative positions of two candidates might depend upon the budget of available candidates. In a chess match, the ranking can depend upon the participants of the tournament. Once it is recognised that the mechanisms can be the same, there is no need to call it 'paradoxical'.

1.3.5 Cheating vs Arrow's Theorem

It is sometimes thought that all problems in voting are caused by Arrow's theorem. This however is a misunderstanding. The problems in voting are not caused by Arrow's Theorem but by the possibility of cheating.

There is a notion that Arrow's axiom of *APDM* has merit since it blocks cardinality and hence cheating. But to reject cheating we do not need *APDM*.

Thus we have $cardinality \Rightarrow \neg APDM$, or $APDM \Rightarrow \neg cardinality$, since $APDM$ uses orderings only; but we do not have the converse $\neg APDM \Rightarrow cardinality$, and thus there still is a world to choose from. The basic reason to reject $APDM$ is also that it destroys ordinal information - see Parts 9 and 10 below.

Voting procedures are introduced not only to aggregate preferences, but also to do it in such manner that cheating is limited and that the outcome is as true as possible. The conditions that are required to limit cheating can be so severe that the voting procedure that is used (and that results into a winner only) can be of less use if we would want to find the whole social ordering. But it is not said that we would be interested in finding the whole ordering - so we might well accept those limitations.

Since Arrow presented his theorem in 1951, the voting schemes have been judged increasingly in terms of their effectiveness in generating a social ordering. The real question is different, however. Voting procedures are not targetted at finding the whole ordering independent from the budget. For voting, cheating is the problem and not Arrow's axioms.

For example, any constitution (Social Decision Function (SDF)) can create a (subset-consistent) ordering that is conditional on the budget set. So there is an ordering, from the winner down to the last candidate, and if we take any subset of this conditional order (conditional to the same budget set) then the ranking does not change (since the budget hasn't). But there is little use for it, since decision making concentrates on finding the winner only.

Because of the dependence on the budget, the ordering can change when the budget changes. This should not be surprising, since that is a possibility when we aggregate votes. Changes in the budget can have dramatic effects. Note though that the use of the Pareto-Majority scheme reduces such effects of budget changes. In connection to this, it should be remarked that the common explanation on Arrow's Theorem tends to confuse preference reversals on subsets within a budget set with preference reversals of budget changes. In this book, this distinction is clearly made.

A major conclusion is that a society that wants to maintain consistency over a prospective budget that is larger than the budget that it actually has, might want to determine the ranking of the possible decisions that depends upon that larger prospective budget. In practice this already happens when governments

make long term budget forecasts. Of course, this at best only reduces surprises. But we can identify voting schemes that reduce the likelihood of such surprises.

1.3.6 Example constitutions

Consider these three constitutions - included in the `Constitutions[]` call:

- **Pareto (efficiency) majority:** Only those items under choice are considered that benefit some and that are not to the disadvantage of anyone. Note that efficiency depends *absolutely* on the Status Quo, and is not *relative*. If there are more such items without a clear order, then a Fixed Point Borda majority decision is used. Ties on Borda Fixed Points are broken with the Condorcet margin count.
- **Borda (-majority):** each voter gets N (say $N = 100$) points, and may distribute these across the items under choice. The item with the highest sum of points is selected.
- **Pairwise majority:** Items are brought to the floor in pairs, and decided upon by normal majority of "pro" and "contra". If a cycle occurs then there is indifference (a deadlock).

The first is my suggestion for a good standard, the other two are the basic schemes much discussed in the literature but which only provide raw components for Pareto Majority.

A basic assumption is that there is always a Status Quo to which alternatives are compared. This is often neglected in theory, but it is important. It relates to the distinction between Statics and Dynamics. In dynamics we study the change of a situation. Static theory is only relevant as a stepping stone for dynamics. Thus we should include a status quo if we want to translate our results to dynamics.

When there is a deadlock or indifference over the whole set, then the Status Quo is maintained. Then you have to provide additional decision rules, such as random selection (throwing dice) etcetera. Since this book cannot decide how you solve your deadlocks, the general rule is that the status quo is maintained.

In all cases, we concentrate on picking the winner, rather than constructing the collective welfare index - thus we use a constitution (Social Decision Function (SDF)). Not deriving the full ordering is a matter of efficiency.

This efficiency however should not be confused with the content of the argument. It would be a confusion to reason: "Sen (1970) already explained that Arrow's Theorem is less relevant for constitutions (SDFs). These routines work for this reason. This author does not really discuss SWF-GM on orderings. Rejection of *APDM* is unimportant." Thinking like this would be a confusion since it should be understood that we have to reject *APDM* to solve the paradoxes, *also* for constitutions (SDFs). And next, we can show that each consistent constitution (SDF) can be used to create a SWF-GM (i.e. the whole ordering conditional on the budget set).

The Voting packages provide their own solution routines. Note that voting situations can be represented by Graphs, and that Steve Skiena's great package `DiscreteMath`Combinatorica`` deals with these. The `Voting`` package also provides routines to translate to Graphs, and one may benefit from those plotting and solution routines.

1.3.7 How to proceed

Even though you are an advanced reader, I still would kindly ask you to start at the very beginning anyway and work your way up from this beginning till the end, though not skipping the difficult parts. You will benefit also from the introduction to the beginning student above, since it provides a quick example how you can use the routines. You would have to look at the first chapters anyway, since they provide details about the implementation in *Mathematica*.

In particular, I draw your attention to:

- The combination of *first* Pareto and *only then* a simple scheme would be a very relevant condition for an acceptable scheme.
- The Fixed Point Borda scheme, where only winners are accepted who also win from the alternative winner when they would not participate in the budget set. Where the Condorcet scheme uses pairs and settles ties with voting scores, the Fixed Point Borda uses rank-orders and *only then* checks with pairwise comparisons for potential winners.
- When you reconsider the familiar voting paradoxes, keep in mind that there is the difference between preference reversal within the budget set and the preference reversals from budget changes.

- Consider the voting schemes on (a) how they control the potential impact of cheating, rather than on (b) how they might establish the whole ordering over the whole budget set.

During your reading, you will notice that there are various subtle novel points along the way. One of my ideas is that Voting Theory has been on a wrong footing since Arrow's 1951 result, and the only way to regain a proper footing is to tell the story as it should be told. When there are hundreds of misunderstandings, then it is difficult to exactly state for each different reader how his or her particular misunderstanding can be solved. In a state of general confusion it is better to start "from scratch", and tell the story like one would tell it to a new student. This also explains why this book has this integration between the beginning and the advanced sections.

Of course, once you have worked yourself up to Parts 9 and 10, then only you, the advanced reader, will truly benefit from the main intellectual result of this book. This Part gives a foundation and justification what we tell to the beginning students. You will find that this result is quite challenging. The arguments in the introductory part may look easy or simple, but that is just the phrasing that I chose, and the argument is quite abstract. The accessible phrasing makes that the argument can be followed by the average student, but, in my experience, it takes a more abstract mind to really understand the arguments. There still is value in a good education.

1.4 Overview for both

1.4.1 Structure of the book

The book basically has:

1. The basic elements: the items, voters, preferences and morals.
2. The basic voting schemes: these tend to neglect whether the items are Paretian or not.
3. The combined schemes: which first select the Paretian points, and then apply the basic scheme.

You can set the `Vote[]` routine to the scheme of your own choice, so that you have a short command available. When starting up the packages, the default `Vote[]` routine is `ParetoMajority[]`. This first finds the Pareto points and then applies the fixed point Borda scheme. But you can Clear it and redefine it as you wish.

(Note that, on one hand, this book advocates a clear distinction between *voting* and *deciding*, and now we use a command `Vote[]` that would actually give the decision... Well, once it has been accepted and understood that voting fields do not yet give a decision, we might as well use the short word that everybody is used to.)

Subsequently, there are the probability approaches. This contains material that is normally missing from the standard probability courses but that still is very important to understand more about the world. Part 7 contains a discussion on the relationship of voting and games or matches. We discuss the logit model for the theory of testing, matching, ranking and rating, and determine the conditions under which there would be a Rasch - Elo rating for voting. (This is like the Elo rating for chess players.) Part 8 considers whether utility functions can be recovered from probability experiments, as is sometimes suggested in the literature. If we could measure utility objectively, then we would not need voting. Giving proper definitions of risk and certainty equivalence, we find that the scope to determine cardinal utility is limited, and that the normal explanation in the literature actually is off-track.

For the advanced reader, there is an additional part with high theory in Parts 9 and 10. This high theory justifies what the book sells to the novice readers.

Readers new to the subject who want to understand this part, should use the library, but are advised to have these chapters available to guide them through the arguments.

Note: You are advised to use the internet as well. Voting Theory is an interesting subject and various people and organisations have devoted attention to it. A search engine will quickly generate results. [Www.britannica.com](http://www.britannica.com) has some information on Borda, Condorcet and other historical figures. Some schools put out summary reviews. Other researchers pose problems - a challenge for the programs below. [Www.siam.org](http://www.siam.org) has some interesting reviews. The list is large.

1.4.2 Use of The Economics Pack

Voting Theory has been one of the key topics of research that lead to the development of The Economics Pack, applications of *Mathematica*. It is assumed throughout that you have a copy of The Economics Pack available if you want to run the software.

The Economics Pack itself has been developed for economics, business and finance in general. The software was written while doing research, giving practical decision support and teaching, and it has proven its usefulness many times over. The software is of a basic rather than a grand nature, but it provides a working environment that many will enjoy to have. The applications of the Pack may help you to get the job done, to get a feel of the discussed problems, or to get a refresher of economics. The software can also be used as a reliable base to create programs of a higher complexity. The name "The Economics Pack" does not mean that you could solve any economic problem with this, but it does mean that when you start doing economics, then you are likely to want to have these tools at your disposal.

1.4.3 Mathematica

Mathematica is a language to do mathematics with the computer. Note that mathematics itself is a language that generations of geniusses have been designing to state their theorems and proofs. This elegant and compact language is now being implemented on the computer, and this creates an incredible powerhouse that will likely grow into one of the revolutions of mankind - something that can be compared to the invention of the wheel or the alphabet; at least, it registers with me like that. Note that, actually, it is not the invention of precisely the wheel that mattered, since everybody can see roundness like in irisses, apples or in the Moon; it was the axle that was the real invention. In the same way next generations are likely to speak about the 'computer revolution', but the proper revolution would be this implementation of mathematics.

1.4.4 A guide

Since *Mathematica* is such an easy language to program in, it also represents something like a pitfall. It is rather easy to prototype the solution to a problem, or to write a notebook on a subject. But it still appears to be hard work to maintain conciseness, to enhance user friendliness and to document the whole.

Keep in mind the distinction between **(a)** an economic problem, **(b)** how a solution routine has been programmed, **(c)** the way how to use the routines.

This book focusses on (a) the economic problem of voting. It however also provides a guide on (c) but neglects (b). Thus, the proper focus is on the *why*, i.e. the content of Voting Theory, for which we want to apply these routines. But this also requires that we explain *how* to use them. If you want to know more about how the routines have been programmed, then you might use the routine `ShowPrivate[]`.

2. Getting started

The Economics Pack becomes fully available by the single command `<<Economics`All``. It is good practice however to use a few separate command lines to better control the working environment. Three lines can be advised in particular.

2.1 The first line

You start by evaluating:

```
Needs["Economics`Pack`"]
```

This makes the `Economics[]` command available by which you can call specific packages and display their contents. Before you use this, read the following paragraphs first.

2.2 The second line

`CleanSlate`` is a package provided with *Mathematica* that allows you to reset the system. You thus can delete some or all of the packages that you have loaded and remove other declarations that you have made. The only condition is that `CleanSlate`` resets to the situation that it encounters when it is first loaded. You would normally load `CleanSlate`` after you have loaded some key packages that you would not want to delete. The `ResetAll` command is an easy way to call `CleanSlate``. Your advised second line is:

```
ResetAll
```

```
ResetAll
```

ResetAll calls CleanSlate, or if necessary loads it.
This means that your notebook does not have to distinguish
between calling CleanSlate` and evaluating CleanSlate[]

Note that if you first load CleanSlate` and then the Economics Pack, then the ResetAll will clear the Pack from your working environment, and thus also remove ResetAll. If you would happen to call ResetAll again after that, then the symbol will be regarded as a Global` symbol.

2.3 The third line

After the above, you could evaluate EconomicsPack to find the list of packages.

```
EconomicsPack
```

Select the package of your interest, load it, and investigate what it can do. For example:

```
Economics[Voting]
```

You can suppress printing by an option `Print → False`. You can call more than one package in one call. If you want to work on another package and you want to clear the memory of earlier packages, simply call ResetAll first. This also resets the In[] and Out[] labels.

```
Economics[xi, ...]
```

shows the contents of xi` and if needed loads the package(s).
Input xi can be Symbol or String with or without back-
apostrophe. To prevent name conflicts,
Symbols are first removed. Economics[] doesn't need the Cool`,
Varianed` etc. prefixes

```
Economics[All]
```

assigns the Stub-
attribute to all routines in the Pack (except some packages)

```
EconomicsPack
```

gives the list {directory → packages}

Note: Economics[x, Out → True] puts out the full name of the context loaded.

This book will use basically these packages:

```
Economics[Voting, Logic, Logic`Deontic, CES, AGE, Economic`Fairness, Logit,  
Probability, Risk]
```

Voting uses the following subpackages:


```
Economics[Voting`Common, Voting`Utilities, Voting`Formats, Voting`Graphics,
Voting`Borda, Voting`Approval, Voting`Plurality, Voting`Pareto,
Voting`Pairwise, Voting`Theory]
```

2.4 Using the palettes

The Pack comes with some palettes. These palettes have names and structures that correspond to the chapters in The Economics Pack itself.

- The master palette is "TheEconomicsPack.nb" and it provides the commands above and allows you to quickly call the other palettes or to go to the guide under the help function.
- The other palettes have "TEP_" as part of their name, so that they can easily be recognised as belonging to the Pack. These "TEP_" palettes contain blue buttons for loading the relevant packages and grey buttons for pasting commands.
- The exception here is "TEP_Arrowise.nb" that only deals with the package for making arrow diagrams.

The voting palette is part of the TEP_Economics palette.

Note: For *Mathematica* 3.0 (but not for 4.0) the palettes have to be copied (but only once) from the Economics directory to the *Mathematica* system directory. You would do (only once):

```
InstallEconomicsPalettes[];
```

You may have to restart *Mathematica* 3.0 before the palettes are shown in the palettes menu.

2.5 All in one line

You can also load the Pack by the following single line.

```
<<Economics`All`
```

This evaluates Needs["Economics`Pack`"] and Economics[All], and opens the palettes. It does not call ResetAll, however.

3. Items, voters, preferences and morals

3.1 Introduction

In this part we have to define:

1. How to represent the items or candidates.
2. How to represent the voters.
3. How to represent the preferences of the voters.

These basic concepts are covered in the `Voting`Common`` package:

```
Economics[Voting`Common]
```

Cool`Voting`Common`

BestInPref	NumberOfItems	PrefPairs	VoteMargin
CheckVote	NumberOfVoters	ProperPrefsQ	Voter
FixedListQ	Pref	StatusQuo	Votes
Items	Preferences	ToPref	VotingQ

This part also uses some of the utilities and formats:

```
Economics[Voting`Utilities]
```

Cool`Voting`Utilities`

AppendPref	ExamplePrefs	PairwiseToMatrix	SetPreferences
Condorcet	ItemOrderQ	PairwiseToVoteMargin	SetRandomPreferences
DefaultItems	NPairs	PrefSubset	SetRandomVoteMargin
DefineFast	NPrefOrderQ	SelectPreferences	StrictRisingPrefsQ
EqualVotes	Pairs	SetFirstValue	TakePref

```
Economics[Voting`Formats]
```

Cool`Voting`Formats`

FromVoteMargin	ListToPref	PrefToList	VoteMatrix
GraphToPref	ListToPrefOrderQ	ToVoteMatrix	VoteMatrixToPref
GroupOrderQ	ListToVoteMargin	VoteMarginToGraph	
InvertGraph	ListToVoteMatrix	VoteMarginToOrderQ	
ListToGraph	PrefOrderQ	VoteMarginToPref	

3.2 Items

3.2.1 Using default Items

There are the items to be voting about:

<code>Items</code>	a list. You would set your own <code>Items = {...}</code> . The default has <code>NumberOfItems</code> elements of the Alphabet
<code>NumberOfItems</code>	Must be set to the number of Items considered
<code>DefaultItems[(n)]</code>	for <code>n</code> a Blank, takes <code>NumberOfItems</code> , for <code>n</code> a Number sets <code>NumberOfItems</code> , sets the items to A, B, C, ... and <code>StatusQuo[] := First[Items]</code>
<code>StatusQuo[]</code>	gives the item that represents the status quo. By default the first of <code>Items</code>

Note that the default items are Strings "A", "B", ... and that *Mathematica* does not normally print "s.

- If we have five candidates:

```
DefaultItems[5]

{A, B, C, D, E}

% // FullForm

List["A", "B", "C", "D", "E"]

NumberOfItems

5
```

- `DefaultItems` sets `Items` to a procedure. `Items` uses the current value of `NumberOfItems`.

```
ShowPrivate["Items"]

Cool`Voting`Common`Private`

Items takes NumberOfItems elements of the
  Alphabet. Note that you can set your own Items = {...}

Items := CreateNames[NumberOfItems]
```

3.2.2 Using CreateNames

`CreateNames` for $n \leq 26$ gives single capitals, thereafter double capitals.

`CreateNames[n_Integer]` creates a list of n names, using the alphabet (capitals)

`CreateNames[n_Integer, labels_List, proc_:StringJoin]`

creates a list of n names, using the labels as the elements rather than the alphabet, and using proc as the operation of concatenation (e.g. ToProperName)

```
NumberOfItems = 10; Items
```

```
{A, B, C, D, E, F, G, H, I, J}
```

```
NumberOfItems = 30; Items
```

```
{AA, AB, AC, AD, AE, AF, AG, AH, AI, AJ, AK, AL, AM, AN,  
AO, AP, AQ, AR, AS, AT, AU, AV, AW, AX, AY, AZ, BA, BB, BC, BD}
```

3.2.3 Arbitrary names

Of course, you are free to define your own names as well. These can be Symbols as well as Strings. Make sure however that the Items do not have values (wrong would be e.g. Washington = 5), and that the NumberOfItems fits your list.

```
Items = {Washington, Jefferson, Madison, Franklin, Adams};  
NumberOfItems = Length[Items]
```

```
5
```

3.2.4 Role of the Status Quo

The default assumption is that the first candidate is also the status quo.

```
StatusQuo[]
```

```
Washington
```

- You can also define a different status quo, but you should make sure that it is in the list of Items.

```
StatusQuo[] = Items[[3]]
```

```
Madison
```

The Social Choice literature tends to neglect the issue of a status quo. Improvement then is not judged from the status quo, but abstractly comparing

arbitrary points. The difference in views is the one between absolute and relative improvement.

Note that there are two important perspectives on the status quo:

- If we include the status quo, then the real decision is only about proposals that are an improvement from the status quo. This is essentially a luxury situation, and every voter can feel relatively relaxed.
- If we exclude the status quo, then the voting problem becomes a hard choice, where one person has to suffer for the advancement of another person.

Note that these two problems have the same technical appearance. In both cases the vote is, say, on B , C and D , (with A in the background as the status quo, or not accepted as such). If we would analyse these two problems as the same 'technical problem', then we make a serious error. Using only the 'technical perspective' tends to emphasise the 'hard choices', since it is less obvious that there is a luxury interpretation. The suggestion of the technical perspective thus can be quite misleading. For this reason it is advisable to *always* include a status quo, just to safeguard psychological accuracy.

Of course, once this is understood to the core of our bones, then we might neglect the status quo at times, since it would be obvious that we are only discussing luxury questions. If it is assumed that we only regard points that are better than the status quo anyway, then most texts of the Social Choice literature again become relevant, namely for the second step in the decision process, how to select from various possible improvements.

A problem of course is, how to judge whether something is an improvement or not. In general the voting scheme will determine this from the preference lists of the individuals. But this then is an important feature of the scheme.

For single seat elections there are two obvious possibilities for a status quo. The first possibility is a vacancy, the second possibility is that the original dignitary remains in function or that there is some designated successor. For elections such as for the U.S. Presidency, the ballots don't show 'Vacancy'. It is assumed then that U.S. citizens have accepted, by becoming citizens, that there will be no vacancy. For our discussion we however will include the possibility of a vacancy, since it is useful to be explicit about the role of the status quo.

3.3 Voters

A voter does not have to be a single individual, but can also represent a party with a certain percentage of the vote. Each voter is associated with a preference ordering.

<code>NumberOfVoters</code>	must be set to the number of voters
<code>Votes</code>	gives the list of votes per voter. The sum must add to unity. The default for 3 voters is <code>PM[{.25, .35, Rest}]</code>
<code>EqualVotes[(<i>m</i>)]</code>	for <i>m</i> a Blank, takes <code>NumberOfVotes</code> , for <i>m</i> a Number sets <code>NumberOfVotes</code> , and sets <code>Votes</code> to a list of equal votes $1/m$

Note: PM is the probability measure input facility of `Statistics`Common``.

- If we have ten voters with "one person, one vote":

```
EqualVotes[10]
```

$$\left\{ \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10}, \frac{1}{10} \right\}$$

```
NumberOfVoters
```

```
10
```

- Of course, you can assign your own scheme.

```
vlis = {10, 33, 21, 90};
NumberOfVoters = Length[vlis];
Votes = vlis / Add[vlis]
```

$$\left\{ \frac{5}{77}, \frac{3}{14}, \frac{3}{22}, \frac{45}{77} \right\}$$

3.4 Preferences

3.4.1 Summary

Items can be ordered by the degree of preference attached to them, and we will use the Pref object to hold such an ordering. But we can also use numbers to indicate the value that we attach to the items. A list of numbers then is the most general representation, where the number can express the order or the intensity. There are some useful utilities, such as conversion between formats, creation of preference matrices, and selection of subsets of items.

3.4.2 Measurement scales

For measurement, there are the following scales:

- Nominal scale: The data are mere labels or categories, used to identify an attribute of the observation. Example: car names, nationalities, religions.
- Ordinal scale: The data can be *ordered*. Example: The order in which children have been born.
- Interval scale: There is a fixed unit of measurement, so that the *distance* between observations has meaning and can be compared to other distances. Example: Temperature in degrees Celsius: a rise of 5 degrees is 5 times the rise of 1 degree.
- Ratio scale: An interval scale with a meaningful *zero point*. Example: length or weight.

Economists have an ongoing debate whether preferences as experienced by human beings are merely ordinal or have a stronger measurement scale. This discussion basically is about interpersonal comparison of utility. With ordinality, one tends to reject interpersonal comparability - though the assumption of "one person, one vote" still implies some comparability. The strongest assumption is called *cardinal* utility: when utility is not only a ratio scale for each individual, but can also be added over individuals (or Nash multiplied).

Pareto 1897 is known for interpersonal incomparability, with the associated concepts of ordinal utility, Pareto-optimality and unanimity voting - or the mundane *if you can't beat them, join them*. It is less well known that Pareto also

acknowledged cases of comparability, with additive cardinality. If there is cardinal utility, then simple weighed addition (Nash: multiplication) obviously results into some total ('social') utility. One of the first modern researchers on social welfare, Ramsey, was a strong advocate for such (intergenerational) equality. Tinbergen (1985) shows a similar preference for measurability and numerical aggregation.

For the most of this book we will assume only ordinal preference. Cardinality will feature mostly in the discussion about cheating. Part 8 will consider the question whether cardinal utility can be recovered from probability experiments.

One of the pitfalls in working with ordinal preferences is to interpret an order like {1, 2, 3, 4, 5} still as something cardinal. It seems that 5 is much further from 1 than 3. Yet, it is crucial that we disregard such notions, since ordinal data lack any information about intensities. It is important to keep this in mind, especially when judging on the performance of the various voting schemes.

3.4.3 The Pref[...] object

The Pref object collects the items in their order of preference (for a voter). The order is like 'less than' (<). $\text{Pref}[A, B]$ means $A < B$. Items for which there is indifference can be put within a list, so that $\text{Pref}[A, \{B, C\}]$ means $A < B = C$.

`Pref[x1, ..., xn]`

gives a preference order from the lowest preferred x_1 to the highest preferred x_n . The position in the order in fact gives the ordinal preference value. Elements of equal preference are put in sublists, such as $\text{Pref}[x_1, \{x_2, x_3\}]$

- This Pref object tells us that D is the best item, B the worst, while the voter is indifferent for A and C inbetween.

```
pr = Pref["B", {"A", "C"}, "D"]
```

```
Pref(B, {A, C}, D)
```

The Pref[...] object has not been taken as the basic programming object since it gives less information, and since the size of the gap between the various alternatives can best be put into numbers. However, for pairwise majority voting, the Pref[...] object does good service (see below).

Note that the Pref object uses $<$ and not $>$. The best element comes last and does not come first. The reason is that the position is an indication of the value, and a higher value is taken as an indication of higher preference, since utility functions are rising as well. Of course, in a text we still can write (1: $x > y$) meaning that voter 1 prefers x over y , and (2: $y > x$) meaning that voter 2 feels conversely, so that ($x = y$) or that the aggregate is indifference or indecision. But for the implementation in *Mathematica* we must write {Pref[y, x], Pref[x, y]}.

However, the computer is supposed to make life easier rather than complicated, so, it took me a day, compliments to *Mathematica*, but the routine ToPref recognises simple inequality schemes, and helps to construct Pref objects.

<code>ToPref[ineqs]</code>	uses the inequalities to create a proper Pref object. There is no check on cycles, while \leq and \geq generate two Pref objects
----------------------------	--

■ These are examples.

```
Clear[a, b, c]
```

```
ToPref[a > b > c]
```

```
Pref(c, b, a)
```

```
ToPref[a == b == c]
```

```
Pref({a, b, c})
```

```
ToPref[a >= b == c]
```

```
Pref({b, c}, a) ∨ Pref({a, b, c})
```

```
ToPref[a <= b, d >= c]
```

```
Pref(a, b, c, d) ∨ Pref(a, b, {d, c}) ∨ Pref({a, b}, c, d) ∨ Pref({a, b}, {d, c})
```

3.4.4 Preferences as lists of numbers

The standard representation of preferences will use numbers. Each number refers to the value attached to the item of the same position. When we transform above Pref object for example:

- In above example pr , item D is most preferred, so it gets value 4. B is worst and gets value 1. A and C divide the sum of their places 2+3. (See also the section below.)

```
DefaultItems[4]
```

```
{A, B, C, D}
```

```
PrefToList[pr]
```

```
 $\left\{\frac{5}{2}, 1, \frac{5}{2}, 4\right\}$ 
```

```
% // Fraction
```

```
 $\left\{2\frac{1}{2}, 1, 2\frac{1}{2}, 4\right\}$ 
```

- Back to the Pref format.

```
ListToPref[%%]
```

```
Pref(B, {A, C}, D)
```

Since the Pref object provides only ordinal information, the routine PrefToList can use only the positional data. You, as a user, however, can provide all kinds of other lists.

You can control the measurement scale by setting the N option of ProperPrefsQ. The default option is:

```
Options[ProperPrefsQ]
```

```
{N → Automatic}
```

In general, let a voter assign k points over n items. Possible option settings are:

- Ordinal scale: $N \rightarrow \text{Automatic}$. Then $k = 1 + \dots + n = 1/2 n (n + 1)$. The preferences are basically permutations of $1, \dots, n$. But also equal values for indifference are allowed (generally entered as average values of the occupied positions).
- Interval or ratio scale: $N \rightarrow k$. Use e.g. $k = 100$ to impose equal voting power.
- Cardinality: $N \rightarrow \text{Infinity}$. Thus a ratio scale with comparability over individuals. No meaningful sum.

- Thus, if a voter thinks that D is a real lousy item, and if cardinal utility is allowed:

```
SetOptions[ProperPrefsQ, N → ∞];
lis = {2, 1, 2, -∞};
```

`ProperPrefsQ[matrix]` tests whether the preference matrix satisfies the conditions required for the current setting of the N option. This topic brings us to considering matrices of preferences.

<code>ProperPrefsQ[mat]</code>	Option $N \rightarrow k$ determines this test: Ordinality for $k =$ Automatic, and each row sum then should equal $1 + \dots + n = n(n+1)/2$. Interval/ratio scale for k a number, and cardinality for k Infinity.
<code>StrictRisingPrefsQ[mat]</code>	gives True if the rows are permutations of $\{1, \dots, n\}$

3.4.5 Matrices of preferences

Each voter (party) is associated with a preference ordering on n items. If we have m voters, then we get a $\{m, n\}$ matrix.

- The default preferences are from the `Condorcet[]` routine. This assumes 3 items and 3 voters.

```
Condorcet[]; Preferences
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

- Using `SetPreferences`, you can not only define Preferences, but also set the number of items and number of voters. If the implied number of voters differs from the current number, then it is assumed that these new voters will have equal votes. Note: This current call uses the options of `ProperPrefsQ`, that we set above in section 3.4.4.

```
{{1, 2, 3, 4}, {4, 2, 1, 3}} // SetPreferences
```

```
{Number of Voters → 2, Number of items → 4,
  Votes are nonnegative and add up to 1 → True, Preferences fit the numbers of Voters and Items → True,
  Type of scale → Cardinal, Preferences give a proper ordering → True,
  Preferences add up to → {10}, Items → {A, B, C, D}, Votes → {1/2, 1/2}}
```

Thus, if you are in doubt, `Preferences // SetPreferences` should work.

- Since above setting uses cardinal utility, we are free to define any value.

```
{{1, 2, 3, -4}, {4, 200, 1, 30000}} // SetPreferences
```

ProperPrefsQ::pos : Proper Preference matrix should better contain only nonnegative numbers

{Number of Voters → 2, Number of items → 4,

Votes are nonnegative and add up to 1 → True, Preferences fit the numbers of Voters and Items → True,

Type of scale → Cardinal, Preferences give a proper ordering → False,

Preferences add up to → {2, 30205}, Items → {A, B, C, D}, Votes → $\left\{\frac{1}{2}, \frac{1}{2}\right\}$

- Note: Resetting to ordinal utility.

```
SetOptions[ProperPrefsQ, N → Automatic];
```

Preferences

Preferences is a {NumberOfVoters, NumberOfItems} matrix (list of lists) for the values assigned to the items, in the order of Items. A higher value means a higher priority. Thus {{1, 2}, {1, 2}} means that there are two voters and that both assign a higher value to B rather than A

SetPreferences[x]

checks on x, sets NumberOfVoters and NumberOfItems. It assigns equal voting power if the existing votes don't match.

3.4.6 Fast entry of preferences

There is a fast way to define a preference matrix. Suppose that the items are *U*, *V*, *W*, *X*, *Y*, and *Z*, while the group size is 60. A possible preference situation is as follows.

```
DefineFast[{25 UVWXYZ, 33 XUVYZW, 2 WVXZYU}]
```

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 6 & 1 & 4 & 5 \\ 6 & 2 & 1 & 3 & 5 & 4 \end{pmatrix}$$

Votes

$$\left\{\frac{5}{12}, \frac{11}{20}, \frac{1}{30}\right\}$$

```
DefineFast[{n ABC, m CAB, ...}]
```

is a quick way to allocate $n + m + \dots$ votes over items A, B, C, \dots . Preferences, Votes and Items are set

3.4.7 Predefined and random preferences

There are some predefined preferences and there are random preferences.

- This routine presumes NumberOfItems = 3 and NumberOfVoters = 3.

```
ExamplePrefs[1]
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

- This routine redefines NumberOfItems and NumberOfVoters.

```
SetRandomPreferences[4, 6]
```

$$\begin{pmatrix} 3 & 2 & 6 & 5 & 4 & 1 \\ 4 & 1 & 6 & 2 & 3 & 5 \\ 6 & 5 & 3 & 2 & 4 & 1 \\ 2 & 5 & 6 & 3 & 4 & 1 \end{pmatrix}$$

```
{NumberOfVoters, NumberOfItems}
```

```
{4, 6}
```

ExamplePrefs[n]	for n =1,2 give example Preferences for 3 voters and 3 items (with e.g. weighed voting)
SetRandomPreferences[]	sets the Preferences to random orderings
SetRandomPreferences[n]	sets the number of items to n , and then generates random preferences
SetRandomPreferences[m, n]	sets the numbers of voters and items to m and n resp., and then generates random preferences

3.4.8 Preferences over subsets of items

Below we shall meet the problem of considering preferences over subsets of items. The routines of TakePref and SelectPreferences then are useful.

`SelectPreferences` sets `Preferences` and `Items`. It keeps the original order of the `Items`.

```
DefaultItems[6]; SetRandomPreferences[4, 6]
```

$$\begin{pmatrix} 2 & 1 & 3 & 5 & 6 & 4 \\ 4 & 5 & 1 & 2 & 3 & 6 \\ 2 & 6 & 4 & 3 & 5 & 1 \\ 4 & 2 & 6 & 1 & 5 & 3 \end{pmatrix}$$

```
SelectPreferences[{"F", "A"}]
```

{Number of Voters \rightarrow 4, Number of items \rightarrow 2,
 Votes are nonnegative and add up to 1 \rightarrow True, Preferences fit the numbers of Voters and Items \rightarrow True,
 Type of scale \rightarrow Ordinal, Preferences give a proper ordering \rightarrow True,
 Preferences add up to \rightarrow {3}, Items \rightarrow {A, F}, Votes \rightarrow $\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}$ }

Preferences

$$\begin{pmatrix} 1 & 2 \\ 1 & 2 \\ 2 & 1 \\ 2 & 1 \end{pmatrix}$$

Items

{A, F}

<code>SelectPreferences[sel_List]</code>	is the same as setting <code>Preferences = TakePref[sel]</code> and setting the <code>Items</code> to the selected items
<code>TakePref[sel_List]</code>	reduces <code>Preferences</code> to only the selected items mentioned in <code>sel</code> . <code>Items</code> are used for sorting <code>sel</code>
<code>TakePref[x_List?MatrixQ, sel_List (, i)]</code>	reduces a preference matrix <code>x</code> to only the selected items <code>sel</code> (default <code>i = Items</code>)
<code>TakePref[x_List?VectorQ, sel_List (, i)]</code>	reduces a preference list <code>x</code> to only the selected items <code>sel</code> (default <code>i = Items</code>)
<code>TakePref[x_Pref, y_List (, i)]</code>	does the same for <code>Pref</code> objects (default <code>i = Items</code>)
<code>TakePref[{x_Pref}, sel_List (, i)]</code>	does the same for <code>Pref</code> objects (default <code>i = Items</code>)

The method of reduction is controlled by the `N` option of `Options[ProperPrefsQ]`: `N → Automatic` means Ordinality, `N → number` means an Interval or Ratio scale, `N → Infinity` means Cardinality

The order of the original `Items` is kept, but `sel` may be entered in arbitrary order. See `Results[TakePref]` to find the new order of the items. See `DefaultItems` if you want to restore the default items

3.4.9 Conversion between Pref and List

The following routines allow a conversion between the `Pref` and `List` representations. We have seen an example above, in section 3.4.4.

<code>ListToPref[p_List]</code>	turns a list into a <code>Pref</code> object, using <code>Items</code>
<code>PrefToList[Pref[...]]</code>	turns a <code>Pref</code> object into a list again

3.4.10 Sorting items

If you are interested in sorting elements, there is also a format `ListToPref[Order, list (, q)]` that uses subroutines:

```
ListToPrefOrderQ[preference List, q:PrefOrderQ]
```

uses the preference list to define the q sorting criterion, with default PrefOrderQ

PrefOrderQ PrefOrderQ is a head only, that may get a definition by ListToPrefOrderQ. If an order is defined, then it can be used for Sort. PrefOrderQ[i,j] must give False if the first element in a pair {i,j} comes before the last, according to the stated preference list. A pair is assumed to consist of elements of Items

```
DefaultItems[3]; ListToPrefOrderQ[{3, 1, 2}];
```

??PrefOrderQ

PrefOrderQ is a head only. See ListToPrefOrderQ. If an order is defined, then it can be used for Sort. It gives False if the first element in a pair comes before the last, according to the stated preference list. A pair is assumed to consist of elements of Items

```
PrefOrderQ[A, A] = True
```

```
PrefOrderQ[A, B] = False
```

```
PrefOrderQ[A, C] = False
```

```
PrefOrderQ[B, A] = True
```

```
PrefOrderQ[B, B] = True
```

```
PrefOrderQ[B, C] = True
```

```
PrefOrderQ[C, A] = True
```

```
PrefOrderQ[C, B] = False
```

```
PrefOrderQ[C, C] = True
```

3.5 Morals

3.5.1 Introduction

Morals have the same structural form as Preferences:

Preferences	Morals
Better	Ought
Indifference	Freedom
Worse	Not Allowed

Morals enter the discussion on voting since people have morals or principles and they want voting procedures to reflect these. How it will be decided what the rules will be, is part of the *constitutional process*. The possible constitutional amendments on voting rules then become themselves the items of discussion and voting. The preferences then often take a strong form, such that people are unwilling to consider other aspects before some principles have been accepted first. Such an ordering is also called *lexicographic* - taken from the analogy of a dictionary where words are ordered such that for example a *p* is always before a *u*.

The prime subject of the theory of morals is that there is a gap between Is and Ought. This principle is not self-evident. People tend to confuse reality with what should be. Once you are aware of the distinction, it seems pretty obvious - yet confusion creeps up at unexpected moments anyway. Some countries in the world for example have a death penalty, and the citizens of those states are used to the idea - which may cause some of them to think that this is how it *should* be. But a 'should' can never be derived from an 'is'.

The 'logic of morals' is called 'deontic logic'. The most important axiom is that if something is morally imperative, then also all its implications are morally imperative. If a person drowns, and if accidental deaths ought to be prevented, then we should try to save that person. This deontic axiom will play a key role in understanding the constitutional process on voting rules, and hence it is useful to develop that subject a bit more.

Economics[Logic`Deontic]

Cool`Logic`Deontic`

Allowed	FreedomQ	NotAllowedQ	SetDeontic	ToOught
AllowedQ	MoralConclude	NotOught	ToAllowed	
DeonticAxiom	MoralSelect	Ought	ToFreedom	
Freedom	NotAllowed	OughtQ	ToNotAllowed	

SetDeontic["Explain"]

SetDeontic[u, o, na] simplifies the following steps:

The user has to set

Universe[] = {p, ¬p, q, ...} where each p has a ¬p (not p)

Ought[] = Ought[{...}] with a selection from the universe, for the Op

NotAllowed[] = NotAllowed[{...}] with another selection, for the ¬Ap

Then ToAllowed[] and ToFreedom[] give what is allowed and what is free to choose

The crucial idea is that $Op \Leftrightarrow \neg A\neg p$.

The universe consists of three disjoint sets: Ought, Freedom and NotAllowed. The Universe,

Allowed and Freedom objects read as Or[], the Ought and NotAllowed objects read as

And[]. Ought, Freedom and NotAllowed may also be seen as Better, Indifferent and Worse.

SetDeontic[Universe] creates the universe from the binary states, and selects the Ought[Universe] cases

3.5.2 Setting values manually

By first setting some values manually, we will better understand the components.

- Required are some undeclared Symbols. Each represents some statement, like p = "This person drowns", q = "I help".

```
symb = {p, q, r, s, t, v}
```

```
{p, q, r, s, t, v}
```

- The elements of the universe should also contain the negations - like $\neg p$ = "This person does not drown".

```
u = Universe[] = Flatten[FromEvent /@ symb]
```

```
{p, ¬p, q, ¬q, r, ¬r, s, ¬s, t, ¬t, v, ¬v}
```

- $O\neg p$ means "This person should not drown". Let us also take Or for some r .

```
o = Ought[] = Ought[{p, r}]
```

```
Ought({p, r})
```

- Let us declare that t and v are not allowed: $\neg At \ \& \ \neg Av$.

```
na = NotAllowed[] = NotAllowed[{t, v}]
```

```
NotAllowed({t, v})
```

The key concept is $O(\neg p) \Leftrightarrow \neg Ap$. For example: You should not smoke \Leftrightarrow It is not allowed that you smoke. (An ethical principle is stronger than a health warning !)

- It turns out that we did not properly state what ought to happen. We forgot $\neg t$ and $\neg v$.

```
too = ToOught[na]
```

```
Ought({¬ t, ¬ v})
```

- And neither were we specific on what is not allowed. We forgot $\neg p$ and $\neg r$.

```
ToNotAllowed[o]
```

```
NotAllowed({¬ p, ¬ r})
```

<code>ToAllowed[]</code>	derives what is allowed from what is not allowed
<code>ToFreedom[]</code>	derives what is subject to free choice from <code>Ought[]</code> and <code>NotAllowed[]</code>
<code>ToNotAllowed[x_Ought]</code>	derives what is not allowed if x Ought
<code>ToOught[x_NotAllowed]</code>	derives what Ought if x is NotAllowed

Note that only the last two require an input. They must be called before the first two can be called.

3.5.3 Using SetDeontic

The routine `SetDeontic` helps us to consistently define the realms of the discussion. Hence, properly redefining `Ought` and `NotAllowed`.

```
SetDeontic[syms, {p, r}, {t, v}]
```

```
{{p, ¬ p, q, ¬ q, r, ¬ r, s, ¬ s, t, ¬ t, v, ¬ v}, Ought({p, r, ¬ t, ¬ v}),  
  NotAllowed({¬ p, ¬ r, t, v}), Allowed({p, q, r, s, ¬ q, ¬ s, ¬ t, ¬ v}), Freedom({q, s, ¬ q, ¬ s})}
```

```
SetDeontic[U_List, O_List, NA_List]
```

The universe elements are defined as the elements in U and their negations. What ought is defined as the elements in O and the negations in NA. What is NotAllowed is defined from the elements in NA and the negations of O

```
SetDeontic[Universe]
```

sets Universe[Universe] to the outer product of $\{p, \neg p\}$ for the elements in U, and sets Ought[Universe] to the list of possibilities that satisfy what ought

SetDeontic has also defined the objects Allowed and Freedom.

- Allowed is what is not NotAllowed. What ought, is also allowed. (It would be strange to say "You ought to help, but you are not allowed to help.")

```
Allowed[]
```

```
Allowed((p, q, r, s,  $\neg q$ ,  $\neg s$ ,  $\neg t$ ,  $\neg v$ ))
```

- Freedom exists where we are allowed to do things that we do not ought to do.

```
Freedom[]
```

```
Freedom((q, s,  $\neg q$ ,  $\neg s$ ))
```

3.5.4 Objects and Q's with the same structure

<code>Allowed[]</code>	should refer to an <code>Allowed[{...}]</code> object
<code>Allowed[{ ... }]</code>	is the object that contains what is allowed
<code>AllowedQ[p]</code>	is True iff <code>p</code> is an element of <code>Allowed[]</code>
<code>AllowedQ[p_List]</code>	is True iff all elements in <code>p</code> are in <code>Allowed[]</code>
<code>AllowedQ[Universe, p_List]</code>	is the same as <code>AllowedQ</code>

<code>Freedom[]</code>	should refer to a <code>Freedom[{...}]</code> object
<code>Freedom[{ ... }]</code>	is the object that contains what is free to choose from
<code>FreedomQ[p]</code>	is True iff <code>p</code> is an element of <code>Freedom[]</code>
<code>FreedomQ[p_List]</code>	is True iff all elements in <code>p</code> are in <code>Freedom[]</code>
<code>FreedomQ[Universe, p_List]</code>	is True iff all elements in <code>p</code> that are not-ought are in <code>Freedom[]</code>

<code>NotAllowed[]</code>	should refer to a <code>NotAllowed[{...}]</code> object
<code>NotAllowed[{ ... }]</code>	is the object that contains what is not allowed
<code>NotAllowedQ[p]</code>	is True iff <code>p</code> is an element of <code>NotAllowed[]</code>
<code>NotAllowedQ[p_List]</code>	is True iff all elements in <code>p</code> are in <code>NotAllowed[]</code>
<code>NotAllowedQ[Universe, p_List]</code>	is True iff some elements in <code>NotAllowed[]</code> also occur in <code>p</code>

<code>Ought[]</code>	should refer to an <code>Ought[{...}]</code> object
<code>Ought[{ ... }]</code>	is the object that contains what ought
<code>OughtQ[p]</code>	is True iff <code>p</code> is an element of <code>Ought[]</code>
<code>OughtQ[p_List]</code>	is True iff all elements in <code>p</code> are in <code>Ought[]</code>
<code>OughtQ[Universe, p_List]</code>	is True iff all elements in <code>Ought[]</code> also occur in <code>p</code>

Note: Also defined has been Not-Ought, since sometimes there is linguistic confusion with Ought-Not (when people want to emphasise something, for example). NotOught ($\neg O$) = Freedom *or* NotAllowed (just the complement).

NotOught[]

NotOught($\{q, s, t, v, \neg p, \neg q, \neg r, \neg s\}$)

Freedom[] || NotAllowed[]

(Freedom($\{q, s, \neg q, \neg s\}$) \vee NotAllowed($\{\neg p, \neg r, t, v\}$))

NotOught[]	derives for which it is not said that it ought (Freedom or NotAllowed)
NotOught[{...}]	is the object that contains what not ought

Other functions for NotOught are not available.

3.5.5 Universe

Above gives just the elements of the universe. The real universe is a logical combination of some if its elements. Possible states of the world are for example $p \ \& \ q \ \& \ r$, but also $\neg p \ \& \ \neg q \ \& \ r$. Given our elements, we must take all possible combinations of $\{p, \neg p\}$, $\{q, \neg q\}$, etcetera. Rather than using the symbol '&' we will use lists. Thus a list $\{p, \neg q, r\}$ is the same as the assertion that $p \ \& \ \neg q \ \& \ r$, with all these phenomena occuring at the same time. The universe of all such possible combinations is Universe[Universe]. SetDeontic[Universe] will create this universe. However, mainly interesting is Ought[Universe] that gives the list of possible states that satisfy what ought. The latter hence is also put out by SetDeontic[Universe].

- This gives the possible combinations that satisfy what ought.

SetDeontic[Universe]

$$\begin{pmatrix} p & q & r & s & \neg t & \neg v \\ p & q & r & \neg s & \neg t & \neg v \\ p & \neg q & r & s & \neg t & \neg v \\ p & \neg q & r & \neg s & \neg t & \neg v \end{pmatrix}$$

```
MoralSelect[lis_List?MatrixQ, q]
```

selects from the matrix using criterion *q*. The latter must be defined for *q*[Universe, ...] – which is the case for *q* = AllowedQ, FreedomQ, NotAllowedQ and OughtQ

```
MoralSelect[q]
```

uses Universe[Universe],
and for *q* = OughtQ it gives Ought[Universe]

Note that the *q*[Universe, ...] criteria have different meanings for elements or a state of the universe.

3.5.6 The difference between Is and Ought

Above we took *p* = "This person drowns", *q* = "I help". Above universe suggests that it still would be allowed that a person drowns but is not helped. The deontic axiom however suggests: If someone is drowning and can probably be saved by helping, and if you consider that this person should not drown, then you should save him or her.

There are two ways to manipulate logical statements that contain Ought. One way is to use a replacement rule, the other is to use the MoralConclude[] command. Both are weak routines, but the first is weakest.

```
MoralConclude[argument]
```

supplements Conclude with
the Deontic Axiom ($Op \ \& \ p \Rightarrow q \Rightarrow Oq$)

```
DeonticAxiom
```

gives the Deontic Axiom in rule
format ($Ought[p_]\ \& \ p_ \Rightarrow q_ \Rightarrow Ought[q]$)

MoralConclude can best be used in combination with the function Conclude of The Economics Pack. Conclude is further not explained here. The DeonticAxiom can be combined with Infer, idem.

Let us further develop the issue by clear words rather than *p* and *q*.

- Let us consider two statements. The first is philosophical since it exactly copies the structure of the axiom.

```
stat1 = Ought[¬drown] && (¬drown ⇒ help)
```

```
(Ought(¬drown) ∧ (¬drown ⇒ help))
```

- Using a replacement rule now is fast and right on target.

```
stat1 /. DeonticAxiom
```

```
Ought(help)
```

- The second statement is more practical and messes up the neat structure of the philosophical argument. (1) It states the conclusion when one would not help - and some people are slow to draw a conclusion. (2) It clarifies that helping implies getting wet oneself. And perhaps there is danger that one drowns oneself. (3) The idea that the victim should not drown comes only as a late realisation.

```
stat2 = (¬help ⇒ drown) && (help ⇒ getwet) && Ought[¬drown]
```

```
((¬ help ⇒ drown) ∧ (help ⇒ getwet) ∧ Ought(¬ drown))
```

- Replacement now gets us nowhere. See the discussion in The Economics Pack on the difficulty of using replacing rules (the axiomatic method).

```
stat2 /. DeonticAxiom
```

```
((¬ help ⇒ drown) ∧ (help ⇒ getwet) ∧ Ought(¬ drown))
```

Let us now use the Conclude and MoralConclude routines.

- Note that we first initialise Conclude[] - this sets Conclusions = {}. Subsequent calls give only the news. Then, the logical conclusions from the first statement are not impressive.

```
Conclude[]; Conclude[stat1]
```

```
{{drown ∨ help}, Ought(¬ drown)}
```

- New conclusions from the second statement are neither impressive. Note that And and Or are not Orderless - see the discussion in The Economics Pack how you can deal with that.

```
Conclude[stat2]
```

```
{{help ∨ drown}, (¬ help ∨ getwet)}
```

- This would be the moral conclusion however.

```
MoralConclude[stat2]
```

```
{Ought(getwet), Ought(help)}
```

Some philosophers argue that, since getting wet cannot be a strong moral imperative, the deontic axiom only has limited application. Yet in this case it spells out what should be done.

3.6 VoteMargin object

3.6.1 The vote matrix

The vote matrix V has elements $V[i, j]$ with the (relative) score in favour of i in the comparison with j . The row sum $V[i]$ then gives the total score for i , for all comparisons. Note that the matrix has the property that $V[j, i] = 1 - V[i, j]$.

- The following is an example fractional vote matrix. We need only define upper-diagonal elements, since the diagonal is zero and since the other elements can be derived. You can use this routine directly if you have raw data on pairwise vote results.

```

      □ .1 .5
pwwdata = □ □ .7;
      □ □ □

```

```
PairwiseToMatrix[pwwdata]
```

$$\begin{pmatrix} 0 & 0.1 & 0.5 \\ 0.9 & 0 & 0.7 \\ 0.5 & 0.3 & 0 \end{pmatrix}$$

The vote matrix also becomes interesting when we consider 'irrational' vote results - also for single individuals. A preference representation by a list of numbers is always 'rational' in the sense that there will be no cycle. A cycle would occur if someone would prefer $A > B$, $B > C$ but $C > A$ again. We would say that this person is undecided or indifferent (and perhaps confused between $>$ and \geq). But, technically, such a cycle cannot be represented by a list of numbers, and hence we have to look for other ways of representation.

- We can express indifference or indecision by a simple list of numbers.

```
DefaultItems[3]; PrefToList[Pref[{"C", "B", "A"}]]
```

```
{2, 2, 2}
```

- A cycle itself cannot be a simple list of numbers.

```
PrefToList[Pref["A", "C", "B", "A"]]
```

```
PrefToList::frq : Some items used more than once
```

```
(2 A)
```

Preference cycles however can arise when group votes are aggregated. Group indecision or indifference can show up as cycles in pairwise voting. To represent such cycles, we can use a matrix of pairwise comparisons.

- Transforming a single preference list into a vote matrix uses "one person, one vote".

```
ListToVoteMatrix[{1, 2, 3}]
```

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

- A cycle $A > B$, $B > C$ and $C > A$, for a single person, would result into the following vote matrix.

```
v = {{0, 1, 0}, {0, 0, 1}, {1, 0, 0}}
```

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

- With `Cycle → True` the cycle is put into the `Pref` object. It has not been defined, however, what this object now would represent.

```
VoteMatrixToPref[v, Cycle → True]
```

```
VoteMarginToPref::cyc : Cycle {B, A, C, B}
```

```
Pref(B, A, C, B)
```

- The default conclusion is indifference (indecision).

```
VoteMatrixToPref[v]
```

```
VoteMarginToPref::cyc : Cycle {B, A, C, B}
```

```
Pref({A, B, C})
```

`ListToVoteMatrix[preference_List]`

makes a matrix of pairwise vote results,
with $V[i, j] = 1$ if $x[[i]] > x[[j]]$, 0 if $<$, and 1/2 if $=$.

`VoteMatrixToPref[matrix]`

turns a vote matrix into a
Pref object. If Option Cycle \rightarrow True,
then the Pref object may contain a cycle

If we have Votes and Preferences, then we can have a pairwise vote, and generate the aggregate vote matrix. This issue will be discussed in more detail in the section on the PairwiseMajority routine.

- Compare this vote matrix with the Condorcet case in section 1.2.4.

`Condorcet[]; VoteMatrix[]`

$$\begin{pmatrix} 0 & 0.4 & 0.75 \\ 0.6 & 0 & 0.35 \\ 0.25 & 0.65 & 0 \end{pmatrix}$$

`VoteMatrix[p:Preferences, v:Votes]`

determines pairwise vote matrix for
numeric preferences using ListToVoteMatrix

Theory generally makes a distinction between indifference and incompleteness (see Sen (1970:3)). Indifference would exist if both $A \leq B$ and $B \leq A$ are asserted. Incompleteness would exist when neither are asserted. This distinction does not help much when there are cycles $A < B$ and $B > A$, which can occur in particular when aggregating preferences. For the aggregate, *indecision* might mean that there are strong emotions involved, and *indifference* might mean that nobody cares. Indeed, when there is a tie, it thus is a good idea to look how that tie is being caused and whether there are preference intensities. In that respect, ties are a key subject of Voting Theory. But when we do not specifically discuss the subject of tie-breaking, then we may equate $tie = indifference = indecision$.

3.6.2 The VoteMargin[...] object

A single vote is 1, 0 or a fraction for indifference. An aggregate vote result will give more fractional data. The VoteMargin object is a good tool to deal with such fractions. Rather than using vote matrix V , we will use the matrix of margins, $P = V - V'$, where V' is the transpose of V . The elements thus are $P[i, j] = V[i, j] - V[j, i]$ which is the margin of the votes in favour of i over the votes in favour of j . This VoteMargin matrix is 'negative symmetric, in that $P[j, i] = -P[i, j]$. The advantages of using this matrix are:

1. It is easier to check that $P[j, i] = -P[i, j]$ rather than $V[j, i] = 1 - V[i, j]$.
2. If $P[i, j] > 0$ then i wins, if $P[i, j] < 0$ then it loses, and otherwise there is a tie.
3. The row sums of the margins $P[i]$ are as informative as the row sums of the votes $V[i]$.

- Above we used the raw data `pwdata`. Turning these raw data into the matrix of margins $V[i, j] - V[j, i]$.

```
vm = PairwiseToVoteMargin[pwdata]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & -0.8 & 0. \\ 0.8 & 0 & 0.4 \\ 0. & -0.4 & 0 \end{pmatrix} \right)$$

Note: The words VoteMatrix and VoteMargin look very much alike. It has been a deliberate programming decision to choose this so. It forces us to very clearly understand their differences and to be specific in their use. The VoteMargin has been made a special object to emphasise this.

```
VoteMargin[ {row1, row2, ..., rown} ]
```

the outcomes of pairwise comparisons of n items. (a)
For votes: if $V[i, j]$ are the votes for i in the match with j ,
then $P[i, j] = V[i, j] - V[j, i]$. (b) Another application
of the object uses also the intensities of the preferences

Thus each element $\text{VoteMargin}[[i, j]]$ is the outcome of a preference consideration. Assumed is that 0 means Indifference, and it applies to the diagonal (in the plot from bottom left to top right). A positive value means that i is more than j , a negative value conversely. The size of the value may matter, depending upon the application. If $\text{VoteMargin}[i, j] + \text{VoteMargin}[j, i] \neq 0$, then the preference pairs are 'irrational', as sometimes happens in experiments. (Note that it is useful to keep the word 'irrational' between quotes, since science by definition will try to find a rational explanation for what happens.) Evaluate `VoteMargin["Explain"]`.

- Did you see that the latter has a cycle ?

VoteMarginToPref[vm]

VoteMarginToPref::cyc : Cycle {C, A, C}

Pref({A, C}, B)

`VoteMarginToPref[pp_VoteMargin, i: Items]`

changes a VoteMargin object into a Pref object, while checking for cycles

In the default situation (Cycle → False) indecision is represented by indifference. Alternatively (Cycle → True) indecision can be shown by a cyclic Pref object - for the cycle only. Messages → True (default) or False (optional) control the printing of messages on cycles. Note: The search of cycles uses the routine FindCycle of the Combinatorica' package. Therefor, the VoteMargin object is first transformed into a graph, using VoteMarginToGraph. The default SameQ option however is taken from FromVoteMargin. See the discussion on the pairwise voting routines.

- We can also go back to the votes again.

ToVoteMatrix[vm]

$$\begin{pmatrix} 0 & 0.1 & 0.5 \\ 0.9 & 0 & 0.7 \\ 0.5 & 0.3 & 0 \end{pmatrix}$$

`ToVoteMatrix[v_VoteMargin]`

returns the VoteMatrix

3.6.3 Row sum property for votes

The row sum of a vote matrix gives all votes flowing to that item. This could be used as an indication of aggregate preference. The choice does not change if we use margins instead. The item with the maximal sum of margins is the same as the item with the maximal sum of votes.

VoteMargin["Explain"]

If we have a pairwise vote between items i and j , the votes for i can be recorded in $V[i, j]$ and the votes for j can be recorded in $V[j, i]$. The row sums then give the total votes going to each item.

The VoteMargin matrix then is $P = V - V'$ (with V' the transpose of V). An example V is:

$$\begin{pmatrix} 0 & V[1, 2] & V[1, 3] \\ V[2, 1] & 0 & V[2, 3] \\ V[3, 1] & V[3, 2] & 0 \end{pmatrix}$$

The VoteMargin matrix can be interpreted as the lead of the winner over the loser. The matrix is symmetric apart from the change of signs, i.e. $P[i, j] = -P[j, i]$.

$$\begin{pmatrix} 0 & V[1, 2] - V[2, 1] & V[1, 3] - V[3, 1] \\ V[2, 1] - V[1, 2] & 0 & V[2, 3] - V[3, 2] \\ V[3, 1] - V[1, 3] & V[3, 2] - V[2, 3] & 0 \end{pmatrix}$$

Why do we use this? Basically, we want to determine the item with the highest vote count, so we take the sum of each row in V , and then take the result with the highest value. Let \sum stand for the sum running over the columns j . Suppose that A is the winner, then $\sum V[A, j] > \sum V[i, j]$ for all $i \neq A$.

The same condition holds for P . Using $V[i, j] + V[j, i] = 1$:

$$\sum P[A, j] > \sum P[i, j] \text{ for all } i \neq A \Leftrightarrow$$

$$\sum (V[A, j] - V[j, A]) > \sum (V[i, j] - V[j, i]) \text{ for all } i \neq A \Leftrightarrow$$

$$\sum (V[A, j] - (1 - V[A, j])) > \sum (V[i, j] - (1 - V[i, j])) \text{ for all } i \neq A \Leftrightarrow$$

$$\sum 2 V[A, j] > \sum 2 V[i, j] \text{ for all } i \neq A$$

Thus we basically work with the V matrix, but the P matrix just looks neater.

3.6.4 Subroutines

The following have been used in the sections above.

```
PairwiseToMatrix[x_?MatrixQ]
```

takes an incomplete matrix x with pairwise vote results, and completes it;
 x should have values $0 \leq x[[i,j]] \leq 1$ only above the diagonal,
 which gives the percentage of votes for i in the duel with j

```
PairwiseToVoteMargin[x_?MatrixQ]
```

applies `PairwiseToMatrix`, and turns the result into a `VoteMargin` object

Other subroutines are:

```
VoteMarginToOrderQ[pref_VoteMargin, q]
```

translates the VoteMargin object into a sorting order criterion q , that can be used for Sort. For example $q = \text{PrefOrderQ}[i]$ for the i th voter

```
SetRandomVoteMargin[n:NumberOfItems, type:Integer, ran_List: {-1, 1}]
```

creates a $n \times n$ matrix of Random[type, ran] values, though with diagonal 0. The Head VoteMargin is added, to distinguish this matrix from the normal preference ordering that is represented by a List

- This creates a VoteMargin object from a vote matrix.

```
v = ListToVoteMatrix[{1, 2, 3}]
```

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

```
VoteMargin[v - Transpose[v]]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \right)$$

- This works directly.

```
ListToVoteMargin[{1, 2, 3}]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \right)$$

```
ListToVoteMargin[pref_List]
```

changes a preference list into a VoteMargin[...] object

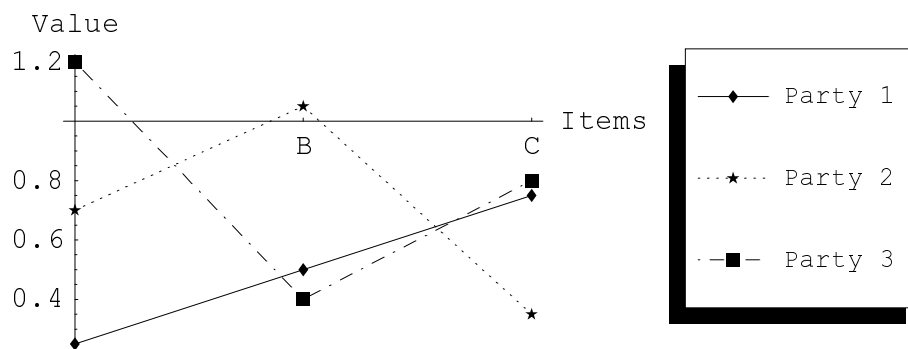
3.7 Plotting tools

Duncan Black proved: If we plot the preferences along an axis and if we can find a plotting order such that all plots are single-peaked, then there will be no cycle. But if all plots contain at least one preference (which can be different for each plot) such that there are at least two peaks, then a cycle becomes possible.

- The following gives Duncan Black's plot of the Condorcet example that we have been using. The default plots Votes * Preferences.

```
Condorcet[]
```

```
DuncanBlackPlot[TextStyle → {FontSize → 11},
  PlotLegend → {"Party 1", "Party 2", "Party 3"},
  LegendShadow → {-0.05, -0.05}];
```



```
DuncanBlackPlot[x, plotting opts]
```

plots the preferences x with the items on the x-axis and the values on the y-axis. May be used to see single-peakedness. Default for x are the Preferences weighed by the Votes

These routines give density plots.

```
PreferencesPlot[p:Preferences, plotting opts]
```

gives a density plot of p (default not weighed with Votes)

```
VoteMarginPlot[pp_VoteMargin, opts___Rule]
```

density plot of
a VoteMargin object

4. Basic schemes

4.1 Introduction

4.1.1 Basic voting schemes

We better have some experience with the basic voting schemes before we continue with theory and with the question what would be the best scheme.

To make sure that the discussion below starts with the default situation, we call the Condorcet routine.

Condorcet[]

Preferences

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

Votes

{0.25, 0.35, 0.4}

4.1.2 Order of discussion

A logical order of discussion would be:

- Pareto: select only those items that nobody vetos. Voters thus say 0 or 1 per item.
- Approval: consider those items that everyone approves of - and select the item with the widest approval base. Voters thus say 0 or 1 per item.
- Plurality: select the item with the most votes. Each voter has only 1 vote, and supposedly votes for the most favourite candidate. If it is imposed that the item must have more than 50% of the vote, then we call this Majority Plurality.
- Condorcet: the items are voted on in pairwise fashion. Select the item that wins all pairwise votes.
- Borda: the items are 'ordered by merit' by each voter, preference numbers are assigned, and summed per item. The item with the highest count is selected. Note: Here the discussion on the preference measurement scale returns strongly.

The order of discussion below however is a bit dictated by the properties of the routines. Approval voting can use the Borda routines, so it is discussed only after these. It also appears useful to discuss Borda before Condorcet.

While discussing these basic voting schemes, we shall be using one common 'voting example case' to show how the same case can be decided differently depending upon the scheme used. It is useful to take a more elaborate example, where the individual preferences depend upon multidimensional utility comparisons. This helps to emphasise the point that the problem matter is a real one, and that the different voting results are not merely academic.

4.2 A voting example case

4.2.1 Summary

Voters can use a Constant Elasticity of Substitution (CES) utility function to score items on the various dimensions, and arrive at a single ranking order.

`Economics [CES]`

Cool`CES`

[Ces](#) [CESlimitQ](#) [CheckES](#) [Results\\$CES](#)
[CES](#) [CESterm](#) [DCES](#) [S](#)
[CESContours](#) [CEStermRule](#) [Options\\$CES](#) [SetOptionsCESSector](#)

4.2.2 Example

Suppose that there are two presidential candidates, *B* and *C*. The status quo *A* means that the office will remain vacant. Hence:

`DefaultItems[3]`

{A, B, C}

`StatusQuo[]`

A

Suppose that there are three voters (different from *B* and *C*). Suppose that only national security and the economy are relevant. Each voter then determines for each of the candidates his or her competence levels on national security and the economy on a scale from 0 to 100. Each voter has a different opinion, and a possible result is the following.

```
Scores[1] = {{10, 10}, {40, 80}, {80, 60}};
Scores[2] = {{50, 50}, {40, 67}, {80, 40}};
Scores[3] = {{80, 20}, {90, 20}, {35, 45}};
```

We can plot these scores in the national security and economy competence space.

```
toText[x_] := MapThread[Text, {(# <> ToString[x])& /@ Items, Scores[x]}]
```

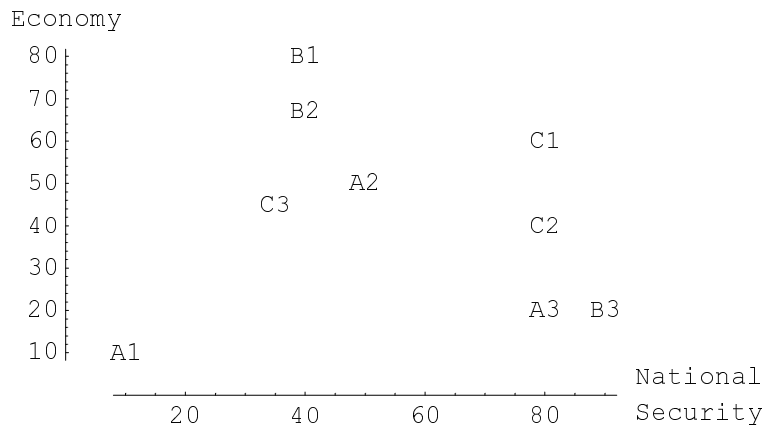
```

gr = Union @@ Array[toText, 3]

{Text[A1, {10, 10}], Text[A2, {50, 50}], Text[A3, {80, 20}], Text[B1, {40, 80}],
 Text[B2, {40, 67}], Text[B3, {90, 20}], Text[C1, {80, 60}], Text[C2, {80, 40}], Text[C3, {35, 45}]}

ps = Show[Graphics[gr], Axes → True, AxesOrigin → {0, 0}, AxesLabel →
{"National\nSecurity", "Economy"}];

```



To compare these options, each voter can weigh these scores in a utility function. Voters will disagree about the attributes, about the scores in those dimensions, and about the weights in each private utility function. Yet, each can use this scheme to find a single ranking order. Let us assume that the voters use Constant Elasticity of Substitution (CES) functions to balance the scores.

```
Voter[1][{ns_, ec_}] = CES[1, {.3, .7}, {ns, ec}, .5]
```

$$\frac{1}{\left(\frac{0.3}{ns^{.1}} + \frac{0.7}{ec^{.1}}\right)^{.1}}$$

```
Voter[2][{ns_, ec_}] = CES[1, {.7, .3}, {ns, ec}, .7]
```

$$\frac{1}{\left(\frac{0.7}{ns^{0.428571}} + \frac{0.3}{ec^{0.428571}}\right)^{2.333333}}$$

```
Voter[3][{ns_, ec_}] = CES[1, {.7, .3}, {ns, ec}, 1.5]
```

$$(0.3 ec^{0.333333} + 0.7 ns^{0.333333})^3$$

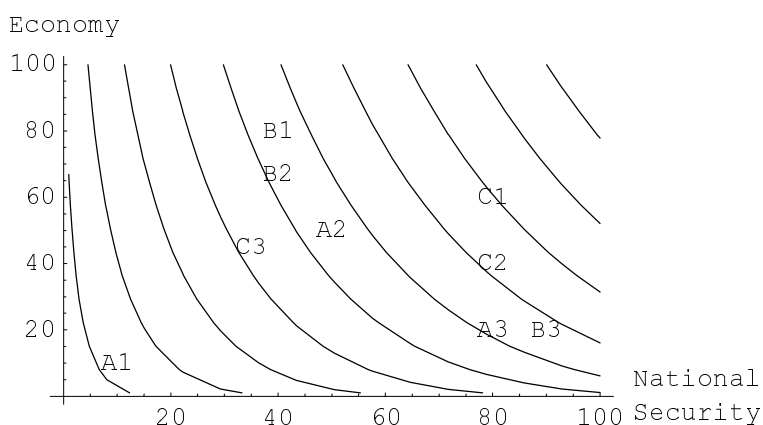
We can find the indifference contours as follows, for example for voter 3:

```

cp = ContourPlot[Voter[3][{x, y}], {x, 1, 100}, {y, 1, 100}, ContourShading
→ False, FrameLabel → {"National Security", "Economy"}, DisplayFunction →
Identity];

```

```
Show[ps, cp, DisplayFunction -> $DisplayFunction];
```



Hence, instead of a Preferences matrix we now get a matrix of utility scores per voter.

```
Utility[x_] := Voter[x] /@ Scores[x]
```

```
uts = Array[Utility, 3]
```

```
{ 10.,    61.5385  64.8649 }
{ 50.,    46.1554  63.5389 }
{ 56.2054  61.6911  37.8253 }
```

Note: The term *logrolling* is used when a proposal on subject X is combined with a proposal on subject Y, to gather sufficient votes for the two of them. Above discussion shows that logrolling is the same as using a more-dimensional utility function. Legislative procedures may cause us to think that X and Y are different subjects and thus need to be considered separately, but the economic approach would be to look for optimal combinations. Logrolling thus is sometimes depicted as cheating, but, it thus isn't.

4.2.3 To ordinal preferences

Above matrix of utility levels suggests an interval/ratio measurement level. Perhaps that is the true state of the world. Normally we assume an ordinal level only, and that can be created as follows.

■ Create Pref objects.

```
SetOptions[ProperPrefsQ, N -> Automatic]
```

```
{N -> Automatic}
```

```
ord = ListToPref /\@ uts
```

```
{Pref(A, B, C), Pref(B, A, C), Pref(C, A, B)}
```

- And turn these into lists again. The 'basicExample' matrix will be used in subsequent chapters again.

```
basicExample = PrefToList /\@ ord
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 2 & 3 & 1 \end{pmatrix}$$

```
EqualVotes[]; SetPreferences[%]
```

```
{Number of Voters → 3, Number of items → 3,
```

```
  Votes are nonnegative and add up to 1 → True, Preferences fit the numbers of Voters and Items → True,
```

```
  Type of scale → Ordinal, Preferences give a proper ordering → True,
```

```
  Preferences add up to → {6}, Items → {A, B, C}, Votes →  $\left\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right\}$ }
```

4.3 Pareto

4.3.1 Summary

The Pareto package gives the absolute improvements from the status quo, including the status quo itself.

```
Economics[Voting`Pareto]
```

Cool`Voting`Pareto`

[EfficiencyPairs](#) [ListToOrderedPairs](#) [NOrderPair](#) [Pareto](#)

4.3.2 Concept

A situation is Pareto optimal if any change would come at the deterioration of some voter, even if some others advance.

In the national security - economy competence space above, any point above i 's indifference contour of a $x(i, j)$ point is Pareto Optimising from $x(i, j)$, for voter i . For example, B3 is on a higher contour for 3 than A3, and this again is higher than C3, meaning that voter 3 would accept only B3 over the status quo.

Pareto advanced his criterion not as a moral condition but rather as a criterion for efficiency and equilibrium. Yet, since the discussion concerns morals, it is difficult to disregard the moral implications. Note that a voter may always have some deeper reasons for blocking a proposal. Suppose that a proposal is made that everyone improves by \$1 but the King by \$1 million. Everyone thus seems to improve. But some voters may think that there is a relative deterioration, and thus vote for the status quo and keep the money in the treasury chest. Hence, before we would arrive at the opinion that those voters would be irrational, we should check the reasoning.

4.3.3 Pareto routine

In the above preferences, there is no Pareto improvement from the status quo.

```
Pareto[]
```

```
{A}
```

You could verify this by plotting the contours for all voters.

<code>Pareto[x_List (, sq)]</code>	gives the items that are Pareto optimising or indifferent to sq. If x is not specified, Preferences are taken. If sq is not specified, then StatusQuo[] is taken. If x is a matrix of preference lists, then the intersection is taken
------------------------------------	--

4.3.4 Efficiency pairs

Social Choice literature tends to neglect the issue of a status quo. Pareto improvement then is not judged absolutely from the status quo, but comparing points relative to each another. The following is an example where there is no absolute improvement from the status quo, but where there is a relative improvement from *B* to *C*.

```
{{1, 2, 3}, {3, 1, 2}, {2, 1, 3}} // SetPreferences;
```

```
Pareto[]
```

```
{A}
```

```
EfficiencyPairs[]
```

```
(B C)
```

4.4 Plurality

4.4.1 Summary

Plurality voting can violate Pareto optimality, while it does not generate sufficient information about which are the two main contenders.

Economics[Voting`Plurality]

Cool`Voting`Plurality`

[ParetoPlurality](#) [Plurality](#)

4.4.2 Concept

In plurality voting each voter gets one vote, and supposedly votes for his or her favourite. The item with the most votes wins. If it is imposed that the item must have more than 50% of the vote, then we call this Majority Plurality.

A rationale for plurality voting is: Once an election is organised, it is often clear who would be the two main contenders. Everyone who wants some influence on this choice, can choose from these two. Everyone who does not care, can vote for any other than these two or not vote at all. If neither of the two main contenders gets more than 50% of the vote, then there is ample reason to neglect the votes on the non-main-contenders, for people deliberately did not vote for them. Neglecting these self-declared irrelevant votes, the contender with the most votes can be said to win.

Of course, this rationale depends upon clarity who the two main contenders are. Sometimes repeated elections are used to generate this clarity, until a candidate has a plurality vote of more than 50%. This might be a good method to force the reluctant voters to break the indecision. Though, again, voters could be in a swing mood, so that there is no convergence.

A problem with this kind of scheme is that it is difficult to program in a computer: it is not obvious how voters would adjust their votes dynamically. Some would stick with their original candidate, others would switch to one of the two main contenders.

4.4.3 Plurality routine

To return to the basic example: nobody appears to vote for the status quo, and C has most votes. A clear majority.

```
EqualVotes[]; basicExample // SetPreferences;
```

```
Plurality[]
```

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} B & \frac{1}{3} \\ C & \frac{2}{3} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{1}{3} & B \\ \frac{2}{3} & C \end{pmatrix}, \text{Max} \rightarrow \left\{ C, \frac{2}{3} \right\}, \text{Select} \rightarrow C \right\}$$

`Plurality[p:Preferences, v:Votes, i:Items]`

gives the plurality result. The item with the highest count is given, and it is checked whether it receives more than half of the vote

Note that Plurality voting thus neglects the Pareto condition. We had found that C is not Pareto improving from the status quo. In other words, 2/3 of society neglects the 1/3 veto implied by the vote for B.

4.4.4 The winner need not be among the 'first two'

In some cases, the two first items found by Plurality do not succeed in the end. So it need not be obvious which items are 'the first two'. (Example taken from D. Davison's page at <http://www.mich.com/~donald/dispute.html>.)

```
DefineFast[{40 CBDA, 18 ACDB, 17 ABDC, 16 ABCD, 9 DBCA}]
```

$$\begin{pmatrix} 4 & 2 & 1 & 3 \\ 1 & 4 & 2 & 3 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 3 & 4 \\ 4 & 2 & 3 & 1 \end{pmatrix}$$

A and B are the 'first two'. But we will see below that D is strong winner under some schemes.

```
Plurality[]
```

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} A & \frac{49}{100} \\ B & \frac{9}{50} \\ C & \frac{17}{100} \\ D & \frac{4}{25} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{4}{25} & D \\ \frac{17}{100} & C \\ \frac{9}{50} & B \\ \frac{49}{100} & A \end{pmatrix}, \text{Max} \rightarrow \left\{ A, \frac{49}{100} \right\}, \text{Select} \rightarrow \{\} \right\}$$

4.4.5 Plurality fails at ties

Plurality is weaker at solving ties than other systems.

Condorcet[] ; EqualVotes[]

$$\left\{ \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right\}$$

Plurality[]

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} A & \frac{1}{3} \\ B & \frac{1}{3} \\ C & \frac{1}{3} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{1}{3} & A \\ \frac{1}{3} & B \\ \frac{1}{3} & C \end{pmatrix}, \text{Max} \rightarrow \left\{ \{A, B, C\}, \frac{1}{3} \right\}, \text{Select} \rightarrow \{\} \right\}$$

4.5 Borda

4.5.1 Summary

The Borda scheme seems attractive, but appears to violate Pareto and Plurality notions, while it is also subject to preference reversal. The Fixed Point Borda however solves these latter two objections.

`Economics[Voting`Borda]`

Cool`Voting`Borda`

[Borda](#) [BordaComplement](#) [BordaFP](#) [BordaFPQ](#)
[BordaAnalysis](#) [BordaField](#) [BordaFPLocalSet](#) [BordaXvsXCom](#)

4.5.2 Concept

Each voter orders the items by their merit, preference numbers are assigned, and summed per item. The item with the highest count is selected.

4.5.3 The Borda routine

Let us reset the parameters to the example voting case.

```
EqualVotes[]; basicExample // SetPreferences;
```

```
Borda[]
```

```
C
```

```
BordaAnalysis[]
```

$$\left\{ \text{Select} \rightarrow C, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{ \frac{5}{3}, 2, \frac{7}{3} \right\}, \text{Position} \rightarrow (3), \text{Ordering} \rightarrow \begin{pmatrix} \frac{5}{3} & A \\ 2 & B \\ \frac{7}{3} & C \end{pmatrix} \right\}$$

<code>Borda[<i>p:Preferences</i>, <i>v:Votes</i>, <i>i:Items</i>]</code>	chooses the items with the maximum in the BordaField $v . p$
<code>BordaField[<i>p:Preferences</i>, <i>v:Votes</i>]</code>	applies the Votes to Preferences: $v . p$
<code>BordaAnalysis[<i>p:Preferences</i>, <i>v:Votes</i>, <i>i:Items</i>]</code>	analyses the situation for a Borda type of vote: 1) the selected items 2) the BordaField 3) the positions of the maxima 4) the items sorted from lowest to highest weighed vote 5) whether the selection are fixed points

4.5.4 Borda neglects the status quo

This example shows that Borda can accept a change from a status quo that is not Pareto improving.

```
DefaultItems[]; EqualVotes[];

SetPreferences[{{3, 2, 1}, {1, 3, 2}}];
```

The classical liberal will hold that there is no Paretian improvement from status quo A .

```
Pareto[]

{A}
```

Borda's scheme would always take B , even if A was the status quo.

```
BordaAnalysis[]
```

$$\left\{ \text{Select} \rightarrow B, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{ 2, \frac{5}{2}, \frac{3}{2} \right\}, \text{Position} \rightarrow (2), \text{Ordering} \rightarrow \begin{pmatrix} \frac{3}{2} & C \\ 2 & A \\ \frac{5}{2} & B \end{pmatrix} \right\}$$

This also gives the classical liberal answer to the discussion by Sen, "Collective choice and social welfare", North-Holland 1979, p48. Sen here neglects the issue of the status quo.

4.5.5 Preference reversals

There is preference reversal when addition or elimination of irrelevant items can change the outcome of a decision. In the following example, C and D could be said to be irrelevant. Plurality would select A as the clear winner, but Borda selects either A or B , depending upon whether C and D have been included or not.

```
EqualVotes[]; DefaultItems[];

SetPreferences[{{4, 3, 2, 1},
                {4, 3, 2, 1},
                {4, 3, 2, 1},
                {1, 4, 2, 3},
                {1, 4, 2, 3}}];
```

```
Plurality[]
```

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} A & \frac{3}{5} \\ B & \frac{2}{5} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{2}{5} & B \\ \frac{3}{5} & A \end{pmatrix}, \text{Max} \rightarrow \left\{ A, \frac{3}{5} \right\}, \text{Select} \rightarrow A \right\}$$

A fan of the Borda approach would argue that Plurality fails here.

```
Borda[]
```

```
B
```

There are two sides of the coin:

- B is best to the minority that considers A to be really bad. The majority seems rather indifferent between A and B , so, why not allow the minority their best?
- Alternatively, C and D are rather irrelevant candidates, and the true choice is only between A and B . We then have only 1 and 2 scores, and the 'wide gap' between A and B disappears.

And, if we eliminate C and D , then Borda agrees with Plurality.

```
SelectPreferences[{"A", "B"}];
```

```
Preferences
```

$$\begin{pmatrix} 2 & 1 \\ 2 & 1 \\ 2 & 1 \\ 1 & 2 \\ 1 & 2 \end{pmatrix}$$

Borda[]

A

Note, though, that we use only ordinal data, and hence we cannot really argue that 'B considers A to be really bad'. My solution to the issue is the concept of the Borda Fixed Point.

4.5.6 Borda Fixed Point

Above preference reversal is easy to judge upon since A is also the plurality winner. However, there can be some cases where preference reversal is less obvious to settle. The concept of a fixed point winner however solves the case in general.

- We have to redefine the case since SelectPreferences above changed all parameters.

```
EqualVotes[]; DefaultItems[];

SetPreferences[{ {4, 3, 2, 1},
                  {4, 3, 2, 1},
                  {4, 3, 2, 1},
                  {1, 4, 2, 3},
                  {1, 4, 2, 3} }];
```

Regard the situation when item X would not participate. Define the "Borda Complement" as all winners when X does not participate: $BC[X] = \text{Borda}[\text{Items} \setminus \{X\}]$. Each alternative winner would certainly be an *interesting* candidate.

- The Borda complement of B is:

```
BordaComplement["B"]
```

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 4

A

As a next step, we compare X to its alternative winners (when X does not participate). There is a fixed point when an item wins from its complements, i.e. $X = \text{Borda}[\{X, BC[X]\}]$. Taking all these fixed points then surely selects the *not-irrelevant* items from the merely interesting ones. Having selected all fixed points, we can use Borda again to find the final winner from those (thus using $\text{Borda}[\{\text{FPs}\}]$ to find the overall winner).

- If we match B with the alternative winner (when B does not participate), then we find that B loses - and hence it is not a fixed point.

BordaXvsXCom["B"]

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 4

CheckVote::adj : NumberOfItems adjusted to 2

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

A

- We can test all items whether they are Borda fixed points.

BordaFPQ /@ Items

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 4

CheckVote::adj : NumberOfItems adjusted to 2

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

{True, False, False, False}

The general fixed point Borda routine is BordaFP.

BordaFP[$p:Preferences,$
 $v:Votes, i:Items$]

first collapses to the set of fixed point winners,
then applies Borda to this selection

Note: A fixed point A wins from the alternative winner B - with the alternative defined as the match when A does not participate. BordaFPQ tests whether a point is a Borda fixed point.

BordaFP[]

BordaFP::chg : Borda gave {B}, Fixed Point is {A}

A

BordaAnalysis[]

$\left\{ \text{Select} \rightarrow B, \text{BordaFPQ} \rightarrow \{\text{False}\}, \text{WeightTotal} \rightarrow \left\{ \frac{14}{5}, \frac{17}{5}, 2, \frac{9}{5} \right\}, \text{Position} \rightarrow (2), \text{Ordering} \rightarrow \begin{pmatrix} \frac{9}{5} & D \\ 2 & C \\ \frac{14}{5} & A \\ \frac{17}{5} & B \end{pmatrix} \right\}$

We can show that BordaPF always finds the Plurality winner - so that we do not need a special check on that.

Theorem: A Majority Plurality winner is also found by BordaFP.

Proof: A Majority Plurality winner wins the duel with the Borda winner. Hence it is a fixed point. The Borda winner then is not a fixed point, and is no longer considered. Hence the former wins by BordaFP. QED

Perhaps it is useful to note the following points. (1) BordaFP makes controlled use of the ordinal data. The Borda winner is used to find a starting point of the search. Alternative winners are selected by the Borda count. Each alternative winner again is tested on being a fixed point - and thus generates its own alternative again. However, being a fixed point is established by a pairwise vote. Then the Borda scheme is applied again if there are more fixed points to choose from. The ordinal data thus are used to select candidates, but the basic test is the pairwise vote. (2) The search of the fixed points starts from the first Borda result. This seemed efficient from a programming point of view. Perhaps it would have been better to test all points, using `Select[Items, BordaFPQ]`. (3) BordaFP basically has no 'cycle'. The local set consists of fixed points that are connected in that the alternative winner becomes another candidate to testing on being a fixed point. A larger set points to the possibility of pairwise cycles, but does not represent a cycle in terms of fixed points themselves. Thus for BordaFP the difference between voting and deciding is much less dramatic than for Condorcet, and limited to the problem of breaking 'real' ties.

Subroutines are:

```
BordaFPQ[x, p_List: Preferences, v_List: Votes, i_List: Items]
```

gives True iff

$x == \text{BordaXvsXCom}[x, p, v, i]$ (or `MemberQ` for indifference lists)

```
BordaComplement[x, p_List: Preferences, v_List: Votes, i_List: Items]
```

gives the Borda winner when x is neglected,

i.e. $C[x] = \text{Borda}[\text{Items} \setminus \{x\}]$. `TakePref` is used to find the complement, and this depends upon `Options[ProperPrefsQ]`

```
BordaXvsXCom[x, p_List: Preferences, v_List: Votes, i_List: Items]
```

tests x with its `BordaComplement`,

i.e. evaluates $\text{Borda}[x, C[x]]$ where $C[x] = \text{Borda}[i \setminus \{x\}]$.

If $x == \text{BordaXvsXCom}[x]$ then there is a fixed point, see `BordaFPQ`

```
BordaFPLocalSet[x, p: Preferences, v: Votes, i: Items]
```

identifies the local set of fixed points starting at x . No such set gives `{}`. The routine works from x using `BordaComplement`

See `Results[BordaFPLocalSet]` for the preference decisions at the considered points and `Results[BordaFPLocalSet, All]` for all points that the routine has looked at. Use `Select[Items, BordaFPQ]` to test whether there would be more fixed points.

4.5.7 Another example of preference reversal

```
DefaultItems[]; EqualVotes[];
```

```
{{3, 2, 1}, {3, 2, 1}, {1, 3, 2}} // SetPreferences;
```

Borda's scheme in this case results into indifference between A and B .

```
Borda[]
```

```
{A, B}
```

If we would hold a second round, however, between these supposedly equal candidates, then A would be chosen.

```
SelectPreferences[%];
```

```
CheckVote::adj : NumberOfItems adjusted to 2
```


Preferences

$$\begin{pmatrix} 2 & 1 \\ 2 & 1 \\ 1 & 2 \end{pmatrix}$$

Borda[]

A

This way of presentation, with voting rounds, suggests that the method could be acceptable, since it allows some convergence.

An alternative presentation however would show 'divergence'. Start with *A* and *B*, and conclude that *A* is better. Then include *C*, which is dominated by both *A* and *B*. But now it appears that *A* and *B* have an equal score. Thus including an inessential *C* changes dominance into indifference. Van den Doel & Van Velthoven, "Democratie en welvaartstheorie", Samson 1990, p110, present this example in this order. In my view, the way of presentation that emphasises convergence is more useful. Rather than creating the possible misunderstanding that preference reversal is a wholly incurable problem, we should emphasise that such problems can also be solved.

The way to solve this issue is to use the condition of a fixed point.

```
{{3, 2, 1}, {3, 2, 1}, {1, 3, 2}} // SetPreferences;
```

BordaFP[]

BordaFP::chg : Borda gave {A, B}, Fixed Point is {A}

A

Below we will discuss preference reversal from the angles of cheating and of changes in the budget.

4.5.8 A non-majority Plurality winner and BordaFP

The example of section 4.4.4 shows that a plurality winner that has less than 50% still can be defeated by a BordaFP.

```
DefineFast[{40 CBDA, 18 ACDB, 17 ABDC, 16 ABCD, 9 DBCA}]
```

$$\begin{pmatrix} 4 & 2 & 1 & 3 \\ 1 & 4 & 2 & 3 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 3 & 4 \\ 4 & 2 & 3 & 1 \end{pmatrix}$$

```
Plurality[]
```

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} A & \frac{49}{100} \\ B & \frac{9}{50} \\ C & \frac{17}{100} \\ D & \frac{4}{25} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{4}{25} & D \\ \frac{17}{100} & C \\ \frac{9}{50} & B \\ \frac{49}{100} & A \end{pmatrix}, \text{Max} \rightarrow \left\{ A, \frac{49}{100} \right\}, \text{Select} \rightarrow \{\} \right\}$$

- We could run Borda again, and that would give D too. However, there is little reason to trust the Borda routine anymore. BordaFP securely gives us a fixed point winner.

```
BordaFP[]
```

D

4.6 Approval

4.6.1 Summary

A problem is how to turn a preference list into an approval list. The default implementation is that everything less than the status quo is rejected (gets a 0) while everything that is at least as good as the status quo is accepted (gets a 1). But perhaps the average or median is a better cut-off point - and it could depend per voter.

```
Economics[Voting`Approval]
```

Cool`Voting`Approval`

[Approval](#) [ParetoApproval](#) [ToApproval](#)

4.6.2 Concept

Approval voting allows voters to enter only 0's (rejects) and 1's (accepts). For example, on the ballot you cross out all names for the items that you want to reject, and then only the remaining items are counted. Technically, we can use the Borda routine to add the counts and to find the item with the widest support.

4.6.3 The Approval routine

It is not straightforward to translate a preference list into an approval list. Should we take the Average or Median value, or some other criterion, like 2/3 ? The default takes the first element as the norm, assuming that this is the status quo. But you are free to set this differently.

- Returning back to the basic example.

```
DefaultItems[3]; EqualVotes[];
basicExample // SetPreferences;
```

```
Options[ToApproval]
```

```
{Min → First}
```

ToApproval[]

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

- As you may have guessed, setting the status quo as the norm, causes it to have the widest approval base. The status quo is only challenged when it is Pareto inoptimal, but then there is no clear mechanism to break the tie.

Approval[BordaAnalysis]

$$\left\{ \text{Select} \rightarrow A, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{1, \frac{2}{3}, \frac{2}{3}\right\}, \text{Position} \rightarrow (1), \text{Ordering} \rightarrow \begin{pmatrix} \frac{2}{3} & B \\ \frac{2}{3} & C \\ 1 & A \end{pmatrix} \right\}$$

`Approval[f, p:Preferences, v:Votes, i:Items, opts]`

calls function `f` with `ToApproval[p, opts]` and unchanged `v` and `i`. You can use `f = Borda, BordaFP` and `BordaAnalysis`

`ToApproval[p_List: {}, opts]`

turns preference matrix `p` into an approval matrix with 0's (rejects) and 1's (accepts) only. Option `Min → c` (default First) determines: $p_{ij} = 0$ if $p_{ij} < c[p_i]$ else 1, where p_i is the i th row, and `c` the criterion function.

`ToApproval[p_List?VectorQ, c]`

applies criterion `c` to the single preference vector `p`

When `Preferences = ToApproval[]`, then you can use `Borda, BordaFP` and `BordaAnalysis` for the results

- This creates a random approval matrix.

Preferences = Table[Random[Integer], {NumberOfVoters}, {NumberOfItems}]

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

- Be sure to call BordaAnalysis now, and not Approval[BordaAnalysis].

`BordaAnalysis[]`

$$\left\{ \text{Select} \rightarrow \text{B}, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{ \frac{2}{3}, 1, \frac{1}{3} \right\}, \text{Position} \rightarrow (2), \text{Ordering} \rightarrow \begin{pmatrix} \frac{1}{3} & \text{C} \\ \frac{2}{3} & \text{A} \\ 1 & \text{B} \end{pmatrix} \right\}$$

4.6.4 The ToApproval routine

Note that the ToApproval routine also works for vectors. Sometimes this is the quickest way to understand what the meaning of a cutoff rule is.

- A combination of steps.

```
DefaultItems[4]; pr = Pref["B", "A", "C", "D"];
lis = PrefToList[pr]
```

`{2, 1, 3, 4}`

- Take anything from C onwards - this depends upon lis !

```
ToApproval[lis, #[[3]]&]
```

`{0, 0, 1, 1}`

- Application of the Median for a larger list.

```
ToApproval[{1, 4, 2, 1, 5, 5}, Median]
```

`{0, 1, 0, 0, 1, 1}`

4.6.5 Relation to other schemes

While the Pareto rule sees the 0's as veto's, and does not know what to do with the 1's, the Approval rule does not regard the 0's as veto's and simply adds the 1's. A combination would be Pareto-Approval, in which the 0's are veto's and the 1's are added.

Perhaps Approval voting could help in generating information about the strongest candidates - where Plurality fails. Such a primary round would not select the two best ones, but only provides information on approval, and people would be free to vote for their true candidate in the proper voting round. But such a scheme would fail e.g. if there are three candidates who each get 1/3 of approval.

Below we will discuss strategic voting, but not do this for Approval voting, precisely since it is so difficult to determine a cut-off point. One comment on strategic voting however can be made here. If you vote strategically, then you should withhold approval for less favorites, reducing their chances. If you accept a lesser item, while others strategically reject your preferred item, then you may help defeat your most preferred choice. The ultimate position is that you reject everything except your most preferred choice - which would be Plurality voting again.

4.7 Condorcet

4.7.1 Summary

Pairwise voting was strongly supported by the Marquis de Condorcet. The Condorcet winner is the item that wins all its duels with the others. If there is a tie or cycle, then the margin of winning (the vote margin count) can be used. The same solution however is also found by BordaFP.

`Economics[Voting`Pairwise]`

Cool`Voting`Pairwise`

PairsToPaths	PairwiseMajority	ScoreBoard	UseFindCycle
Pairwise	PairwiseVoter	ScoreVote	VoteMarginToBinary
PairwiseField	PPath	ToVoteMargin	VoteMarginToCount

4.7.2 The concept

All possible pairs of the items are formed and subjected to a vote. With n items, there are $n(n-1)/2$ votes. The item that wins all pairwise votes is selected.

Note that Pareto-optimality can be determined in pairwise fashion. Namely, all items can be dropped that do not survive the pairwise comparison with the status quo. This philosophy of pairwise comparison now is repeated with a pluralistic interpretation.

Also, for a single individual preference it is true that each pairwise comparison can be made without considering the other items. It is now suggested that this property should also hold for the aggregate. However, the occurrence of cycles is a severe test to this suggestion.

Cycles however are merely a test to this idea and not necessarily a contradiction. The key notion is that a pairwise *voting field* is not yet a *decision*. One of the major pitfalls in Voting Theory is to think that a voting field - or its VoteMargin transformation - already provides the decision, so that a cycle would imply an irrational decision. If you fell into that trap, the crucial step out of it then is to see the difference between *voting* and *deciding*. The VoteMargin object is only the input for the final decision process - and not its end.

Technically, in a voting field we can have $<$, but in a decision it can become \leq , and for a cycle we then can get $=$.

There are two methods to select the winner:

- The binary approach: each pairwise win is counted as a 1 and a loss is a 0. The item with $n - 1$ wins is called the Condorcet winner. (Note: A tie between two items can be seen as a win, and contribute to being a Condorcet winner, or it can be counted as a loss, causing that there is no such winner.) If there is no Condorcet winner, then there likely is a cycle, and we can choose between the highest sum of wins or revert to the margin count.
- The margin count approach: for each pair the shares of votes *pro* and *contra* are recorded - and the winning *margins* give a VoteMargin object. Then the highest row sum gives the winner.

The standard pairwise approach uses the binary approach, and uses the count only in case of ties and cycles.

4.7.3 The pairwise majority routine

Let us reset the parameters to the example voting case.

```
EqualVotes[]; DefaultItems[3];
basicExample // SetPreferences;

Preferences

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 2 & 3 & 1 \end{pmatrix}$$

```

There are two possible calls for PairwiseMajority.

- PairwiseMajority[...] directly constructs the VoteMargin object and works from there.
- PairwiseMajority[Show, ...] records the various steps: (1) the matrix of voters and the pairs under consideration, (2) the matrix of votes cast per voter and pair, (3) the total vote per pair (row sums).

In both cases the binary result is indicated by **1** and the count results is indicated by *N*.

In the example presidential voting case, we have the pairwise vote results $B > A$, $C > A$ and $C > B$, giving a direct order $C > B > A$, with C the Condorcet winner. Note that C wins the vote, but not unanimously, so C is not Paretian.

PairwiseMajority[Show]

$$\left\{ \text{Outer} \rightarrow \begin{pmatrix} \{1, \{A, B\}\} & \{1, \{A, C\}\} & \{1, \{B, C\}\} \\ \{2, \{A, B\}\} & \{2, \{A, C\}\} & \{2, \{B, C\}\} \\ \{3, \{A, B\}\} & \{3, \{A, C\}\} & \{3, \{B, C\}\} \end{pmatrix}, \text{Pairwise} \rightarrow \begin{pmatrix} \{0, \frac{1}{3}\} & \{0, \frac{1}{3}\} & \{0, \frac{1}{3}\} \\ \{\frac{1}{3}, 0\} & \{0, \frac{1}{3}\} & \{0, \frac{1}{3}\} \\ \{0, \frac{1}{3}\} & \{\frac{1}{3}, 0\} & \{\frac{1}{3}, 0\} \end{pmatrix} \right\}$$

$$\text{Sum} \rightarrow \begin{pmatrix} \{A, B\} & \{\frac{1}{3}, \frac{2}{3}\} \\ \{A, C\} & \{\frac{1}{3}, \frac{2}{3}\} \\ \{B, C\} & \{\frac{1}{3}, \frac{2}{3}\} \end{pmatrix}, \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & -\frac{1}{3} & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & 0 \end{pmatrix} \right)$$

$$\begin{aligned} 1 &\rightarrow \{\text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{0, 1, 2\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow C, \\ &\text{Pref} \rightarrow \text{Pref}(A, B, C), \text{Find} \rightarrow C, \text{LastCycleTest} \rightarrow \text{False}, \text{Select} \rightarrow C\}, \\ N &\rightarrow \left\{ \text{Sum} \rightarrow \left\{ -\frac{2}{3}, 0, \frac{2}{3} \right\}, \text{Pref} \rightarrow \text{Pref}(A, B, C), \text{Select} \rightarrow C \right\}, \text{All} \rightarrow C \end{aligned}$$

Output $1 \rightarrow \{\dots\}$ uses the binary count to determine the Condorcet winner. A pairwise win (a positive value in the VoteMargin object) gives 1, a loss 0. This either gives the Condorcet winner or the Status Quo. Output $N \rightarrow \{\dots\}$ means that the numeric values of the VoteMargin object are used, and the highest row sum gives the winner. Output indicated by All gives either the Condorcet winner or the margin count winner. The latter thus uses the information on the margins of winning rather than the Status Quo.

Subresults are:

1. (1a) Sum \rightarrow the row sums after 1 / 0 transformation, (1b) Max \rightarrow the maximal value of this. If this equals (NumberOfItems - 1), then there is a Condorcet winner, (1c) Pref \rightarrow gives the Pref object (1d) Find \rightarrow takes the last element in Pref, (1f) LastCycleTest \rightarrow True / False if the latter is not unique, (1g) Select \rightarrow either the unique Condorcet winner or the Status Quo.
- N. (Na) Sum \rightarrow the row sums of the VoteMargin object, (Nb) Pref \rightarrow gives the Pref object, (Nc) Select \rightarrow the item with the highest preference.

<code>PairwiseMajority[</code>	works from the <code>VoteMargin</code> object
<code>p:Preferences, v:Votes]</code>	
<code>PairwiseMajority[</code>	takes the <code>PairwiseField[]</code> of <code>NPairs[]</code>
<code>Show, p:Preferences,</code>	
<code>v:Votes]</code>	

Note: The occurrence of indifference causes output of a List. Note: You can evaluate `VoteMargin["Explain"]`.

4.7.4 Binary method versus count method

The following is an example that *A* can win with less votes than *B*. Thus:

1. *A* wins pairwise from *B* and *C*,
2. but still *the number of votes for A < the number of votes for B*

- Set the votes and preferences.

```
Votes = { .26, .26, .48 };
```

```
SetPreferences[{ {3, 2, 1}, {3, 2, 1}, {1, 3, 2} }];
```

- The binary method gives *A* as the winner, but the count *B*.

```
PairwiseMajority[Show]
```

$$\left\{ \text{Outer} \rightarrow \begin{pmatrix} \{1, \{A, B\}\} & \{1, \{A, C\}\} & \{1, \{B, C\}\} \\ \{2, \{A, B\}\} & \{2, \{A, C\}\} & \{2, \{B, C\}\} \\ \{3, \{A, B\}\} & \{3, \{A, C\}\} & \{3, \{B, C\}\} \end{pmatrix}, \text{Pairwise} \rightarrow \begin{pmatrix} \{0.26, 0\} & \{0.26, 0\} & \{0.26, 0\} \\ \{0.26, 0\} & \{0.26, 0\} & \{0.26, 0\} \\ \{0, 0.48\} & \{0, 0.48\} & \{0.48, 0\} \end{pmatrix} \right\}$$

$$\text{Sum} \rightarrow \begin{pmatrix} \{A, B\} & \{0.52, 0.48\} \\ \{A, C\} & \{0.52, 0.48\} \\ \{B, C\} & \{1., 0\} \end{pmatrix}, \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & 0.04 & 0.04 \\ -0.04 & 0 & 1. \\ -0.04 & -1. & 0 \end{pmatrix} \right),$$

1 → {StatusQuo → A, Sum → {2, 1, 0}, Max → 2, Condorcet winner → A,

Pref → Pref(C, B, A), Find → A, LastCycleTest → False, Select → A},

N → {Sum → {0.08, 0.96, -1.04}, Pref → Pref(C, A, B), Select → B}, All → A}

- We can understand this, since *A* is the Plurality winner too.

```
Plurality[]
```

$$\left\{ \text{Sum} \rightarrow \begin{pmatrix} A & 0.52 \\ B & 0.48 \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} 0.48 & B \\ 0.52 & A \end{pmatrix}, \text{Max} \rightarrow \{A, 0.52\}, \text{Select} \rightarrow A \right\}$$

- We find that Borda makes the same error as the count method - since it neglects the fixed point condition.

BordaAnalysis[]

$\left\{ \text{Select} \rightarrow B, \text{BordaFPQ} \rightarrow \{\text{False}\}, \text{WeightTotal} \rightarrow \{2.04, 2.48, 1.48\}, \text{Position} \rightarrow (2), \text{Ordering} \rightarrow \begin{pmatrix} 1.48 & C \\ 2.04 & A \\ 2.48 & B \end{pmatrix} \right\}$

- But BordaFP finds the Plurality and the Condorcet winner.

BordaFP[]

BordaFP::chg : Borda gave {B}, Fixed Point is {A}

A

VoteMarginToBinary[<i>vm_VoteMargin</i>]	determines the amount of wins per item, and declares the Condorcet winner for the item that wins all duels
VoteMarginToCount[<i>vm_VoteMargin</i>]	adds the rows of the VoteMargin object, and determines the Item with the highest value

4.7.5 BordaFP solution for the Condorcet cycle

Consider the Condorcet paradox, discussed in the introduction - section 1.2.4.

Condorcet[] ; Preferences

$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$

Votes

{0.25, 0.35, 0.4}

This is the classic example where a cycle $B > A > C > B$ arises. For the binary method, there is no clear solution, and the status quo is selected. The margin count method now can solve the deadlock by giving the solution with the highest margin - in this case A as well.

PairwiseMajority[Show]

VoteMarginToPref::cyc : Cycle {C, A, B, C}

$$\left\{ \text{Outer} \rightarrow \begin{pmatrix} \{1, \{A, B\}\} & \{1, \{A, C\}\} & \{1, \{B, C\}\} \\ \{2, \{A, B\}\} & \{2, \{A, C\}\} & \{2, \{B, C\}\} \\ \{3, \{A, B\}\} & \{3, \{A, C\}\} & \{3, \{B, C\}\} \end{pmatrix}, \text{Pairwise} \rightarrow \begin{pmatrix} \{0, 0.25\} & \{0, 0.25\} & \{0, 0.25\} \\ \{0, 0.35\} & \{0.35, 0\} & \{0.35, 0\} \\ \{0.4, 0\} & \{0.4, 0\} & \{0, 0.4\} \end{pmatrix} \right.$$

$$\text{Sum} \rightarrow \begin{pmatrix} \{A, B\} & \{0.4, 0.6\} \\ \{A, C\} & \{0.75, 0.25\} \\ \{B, C\} & \{0.35, 0.65\} \end{pmatrix}, \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -0.2 & 0.5 \\ 0.2 & 0 & -0.3 \\ -0.5 & 0.3 & 0 \end{pmatrix},$$

1 → {StatusQuo → A, Sum → {1, 1, 1}, Max → 1, No Condorcet winner → {A, B, C},
 Pref → Pref({A, B, C}), Find → {A, B, C}, LastCycleTest → True, Select → A},
 N → {Sum → {0.3, -0.1, -0.2}, Pref → Pref(C, B, A), Select → A}, All → A}

We can better understand the situation by regarding the Borda fixed points. We find that all points are fixed points.

BordaFPQ /@ Items

CheckVote::adj : NumberOfItems adjusted to 2

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 2

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

{True, True, True}

BordaFP selects A as well, but now since it has the highest normal Borda value of all fixed points.

BordaFP[]

BordaFP::set : Local set found: {A, B, C}

BordaFP::chg : Borda gave {A}, Fixed Point is A

A

- A has the highest normal Borda value of all fixed points.

BordaAnalysis[]

$$\left\{ \text{Select} \rightarrow A, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \{2.15, 1.95, 1.9\}, \text{Position} \rightarrow (1), \text{Ordering} \rightarrow \begin{pmatrix} 1.9 & C \\ 1.95 & B \\ 2.15 & A \end{pmatrix} \right\}$$

- If the scores would be the same, then all points are selected.

EqualVotes[]

$$\left\{ \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right\}$$

BordaFP[]

{A, B, C}

Note: Perhaps it is useful to add the following observation on this case. We already noted that part of the attractiveness of the BordaFP approach is that it allows us to determine which items are relevant and which are not. We have not developed a concept and program like this for Condorcet yet, since it seems quite unnecessary to do so. Yet, using the BordaFP approach, we can discuss a small 'paradox'. Suppose that you accept above solution *A* for this variant of the Condorcet paradox (with these vote weights). You might then argue that apparently *C* is less relevant, drop it from the list and propose a new vote. Then: if you drop *C*, then the new comparison is between *A* and *B*, and then *B* will be selected: paradox ! However, this 'paradox' can directly be solved: the argument that *C* would be 'non-essential' now has no force since *C* is a fixed point. If *A* were dropped, then *C* would be selected.

- If we drop *C* then *B* would be selected.

Condorcet[] ; SelectPreferences[{"A", "B"}];

CheckVote::adj : NumberOfItems adjusted to 2

Preferences

$$\begin{pmatrix} 1 & 2 \\ 1 & 2 \\ 2 & 1 \end{pmatrix}$$

BordaFP[]

B

- But we cannot drop *C*, since it is an essential item.

Condorcet[] ; SelectPreferences[{"B", "C"}];

CheckVote::adj : NumberOfItems adjusted to 2

Preferences

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 1 & 2 \end{pmatrix}$$

BordaFP[]

C

The pairwise scheme and BordaFP will be compared in more detail below.

4.7.6 Condorcet and Plurality

It is interesting to compare the Pairwise scheme with the Plurality scheme.

- The Majority Plurality winner that wins from all other items, naturally is also a Condorcet winner.
- Conversely, however, the Condorcet winner, that wins from all pairs, need not be a Majority Plurality winner. The example from section 4.4.4 namely is a counter-example, where there is a Condorcet winner that is not found by Plurality.

DefineFast[{40 CBDA, 18 ACDB, 17 ABDC, 16 ABCD, 9 DBCA}]

$$\begin{pmatrix} 4 & 2 & 1 & 3 \\ 1 & 4 & 2 & 3 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 3 & 4 \\ 4 & 2 & 3 & 1 \end{pmatrix}$$

PairwiseMajority[]

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} \begin{pmatrix} 0 & -\frac{1}{50} & -\frac{1}{50} & -\frac{1}{50} \\ \frac{1}{50} & 0 & \frac{4}{25} & -\frac{23}{50} \\ \frac{1}{50} & -\frac{4}{25} & 0 & -\frac{12}{25} \\ \frac{1}{50} & \frac{23}{50} & \frac{12}{25} & 0 \end{pmatrix} \end{pmatrix} \right\},$$

1 → {StatusQuo → A, Sum → {0, 2, 1, 3}, Max → 3, Condorcet winner → D,

Pref → Pref(A, C, B, D), Find → D, LastCycleTest → False, Select → D},

N → {Sum → { $-\frac{3}{50}$, $-\frac{7}{25}$, $-\frac{31}{50}$, $\frac{24}{25}$ }, Pref → Pref(C, B, A, D), Select → D}, All → D}

4.7.7 Ties amongst Condorcet winners

If a pairwise comparison results into a tie, then neither item can be, strictly speaking, a Condorcet winner. But we can imagine that two items win from all others, and come out in a tie too when they are compared. We shall call such items Condorcet winners too. (This means that when we transform the `VoteMargin` object into a `Vote Matrix`, that we use the option `SameQ → False`, which therefor is the default setting.)

- In this case, *A* and *B* would be Condorcet winners

```
DefaultItems[]; {{3, 2, 1}, {2, 3, 1}} // SetPreferences;
```

- This is the default situation.

```
p = PairwiseMajority[]
```

```
VoteMarginToPref::cyc : Cycle {B, A, B}
```

```
VoteMarginToBinary::dif : Selection A differs from Condorcet winning {A, B}
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix} \right), \right.$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 2, 0\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow \{A, B\},$
 $\text{Pref} \rightarrow \text{Pref}(C, \{A, B\}), \text{Find} \rightarrow \{A, B\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A\},$
 $N \rightarrow \{\text{Sum} \rightarrow \{1, 1, -2\}, \text{Pref} \rightarrow \text{Pref}(C, \{A, B\}), \text{Select} \rightarrow \{A, B\}, \text{All} \rightarrow \{A, B\}\}$

- If you would hold that a tie cannot be counted as a win, then there would be no Condorcet winners.

```
p = PairwiseMajority[SameQ → True]
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix} \right), \right.$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{1, 1, 0\}, \text{Max} \rightarrow 1, \text{No Condorcet winner} \rightarrow \{A, B\},$
 $\text{Pref} \rightarrow \text{Pref}(C, \{A, B\}), \text{Find} \rightarrow \{A, B\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A\},$
 $N \rightarrow \{\text{Sum} \rightarrow \{1, 1, -2\}, \text{Pref} \rightarrow \text{Pref}(C, \{A, B\}), \text{Select} \rightarrow \{A, B\}, \text{All} \rightarrow \{A, B\}\}$

4.7.8 Subroutines ToVoteMargin and FromVoteMargin

`ToVoteMargin` is a routine that is much employed in `PairwiseMajority`, in order to determine the results of pairwise voting.

<code>ToVoteMargin[p:Preferences, v:Votes]</code>	determines the <code>VoteMargin</code> object for numeric preferences, with <code>VoteMatrix</code>
<code>ToVoteMargin[!N]</code>	can be used for algebraic preferences, but this then relies on defined <code>Preferences</code> and <code>Votes</code>
<code>VoteMatrix[p:Preferences, v:Votes]</code>	determines the pairwise vote matrix for numeric preferences using <code>ListToVoteMatrix</code>

```
DefaultItems[]; Votes = {.4, .6};
{{1, 2, 3}, {3, 1, 2}} // SetPreferences;
```

```
VoteMatrix[]
```

$$\begin{pmatrix} 0 & 0.6 & 0.6 \\ 0.4 & 0 & 0 \\ 0.4 & 1. & 0 \end{pmatrix}$$

```
vm = ToVoteMargin[]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & 0.2 & 0.2 \\ -0.2 & 0 & -1. \\ -0.2 & 1. & 0 \end{pmatrix} \right)$$

When you have algebraic preferences, then this approach works.

```
DefaultItems[]; Votes = {.4, .6};
{{1+v, 2-v, 3}, {3, 1, 2}} // SetPreferences;
```

ProperPrefsQ::pos : Proper Preference matrix should better contain only nonnegative numbers

```
ToVoteMargin[!N]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & \text{If}[v+1 == 2-v, 0., \text{If}[v+1 < 2-v, -0.4, 0.4]] - 0.6 & 0 \\ -\text{If}[v+1 == 2-v, 0., \text{If}[v+1 < 2-v, -0.4, 0.4]] - 0.6 & 0 & 0.6 - \text{If}[2-v == 3, 0., \text{If}[2-v < 3, -0.4, 0.4]] - 0.6 \\ -\text{If}[v+1 == 3, 0., \text{If}[v+1 < 3, -0.4, 0.4]] - 0.6 & 0.6 - \text{If}[2-v == 3, 0., \text{If}[2-v < 3, -0.4, 0.4]] - 0.6 & 0 \end{pmatrix} \right)$$

Conversely, having a vote margin, the binary Condorcet count requires us to reconstruct the voting matrix again, now with 1 for a win and 0 for a loss.

<code>FromVoteMargin[vm_VoteMargin, opts]</code>	changes a positive number into 1 and a negative number into 0
---	--

If the option `SameQ` → `False` then off-diagonal 0's (that indicate draws) are counted as wins too, giving 1 (which is the default situation, allowing for multiple Condorcet winners). Otherwise these are counted as losses, giving 0. This subroutine is used in `PairwiseMajority`, and the setting of the option thus affects its behaviour.

4.7.9 Appendix: Other subroutines

The following subroutines are only mentioned for some technical completeness.

<code>Pairs[i:Items]</code>	gives the list of unordered pairs in i
<code>NPairs[n:NumberOfItems]</code>	gives the list of unordered pairs for the set {1, ..., n}
<code>Pairwise[preference list, pair_List]</code>	gives the element from pair that has highest preference. It is assumed that pair is a list of two numbers, where the numbers give the positions of items in Items
<code>Pairwise[preference list, v]</code>	assigns votes v to the position with highest preference

Note: Pairwise uses the majority rule, but you are free to give another implementation.

```
Pairwise[{3, 2}, {1, 2}]
```

```
1
```

```
Pairwise[{3, 2}, vote]
```

```
{vote, 0}
```

<code>PairwiseVoter[n_Integer, pair_List]</code>	performs the pairwise vote for voter n using his preferences and voting power
<code>PairwiseField[list of pairs]</code>	applies PairwiseVoter to all voters, for that list of pairs
<code>GroupOrderQ[x, y]</code>	is set by PairwiseField[] and gives the implied preference order. The order is the same as LessEqual: the last position in the list is the most important

Note: A pair here is a list of two numbers, where the numbers give the positions of items in Items.

- There are 3 pairs when there are 3 items.

```
NPairs[]
```

$$\begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 2 & 3 \end{pmatrix}$$

Subroutines are:

<code>PPath[{a, b}]</code>	gives the list of pairs that together form a path from <i>a</i> to <i>b</i> . PairsToPaths has to be performed first.
<code>PairsToPaths[pairs_List]</code>	chains pairs to form paths. Output are the pairs that form beginning and end of the longest paths, and with the intermediate pairs thus eliminated. Note: cycles are not looked for. The result is stored in PPath

<code>ListToOrderedPairs[preferece list, items]</code>	changes the preference list into a list of ordered pairs
<code>NOrderPair[preferece list, {i, j}]</code>	makes the ordered pair for numbered items <i>i</i> and <i>j</i> , based on the preference list for one voter. If there is indifference between item1 and item2, both {item1, item2} and {item2, item1} occur. There is a strict preference if the opposite pair does not occur.

Note that *p:Preferences* indicates a matrix and *preference_List* indicates a vector.

<code>NPrefOrderQ[pair_List, preference_List]</code>	can be used for Sort, and gives False if the first element in pair comes before the last, according to the stated preference list. Pair is assumed to consist of numbers
--	--

- Preference list {4, 1, 3, 2} means that the second element has the least value, then comes the last element, then the third, while the first element has highest value. Sorting:

```
Sort[{1, 2, 3, 4}, NPrefOrderQ[{{#}}, {4, 1, 3, 2}] & ]
{2, 4, 3, 1}
```

4.8 Comparing BordaFP and Condorcet

4.8.1 Introduction

It would appear that the BordaFP and the Condorcet (PairwiseMajority) rules both are good contenders for the selection as *the* best rule. We should look for examples where the one performs better than the other. If all examples point to one method, then we could have found a good procedure.

We should realise that there is actually a close connection between the Borda count and the pairwise method. Consider the table below. Assume a voter with preference $A > B > C$. In the pairwise votes $\{A, B\}$ and $\{A, C\}$ the voter will vote for A , and hence this will collect 2 points. Similarly, B will collect 1 point from $\{B, C\}$, and C will collect nothing. The Borda ranking $\{2, 1, 0\}$ thus summarises the results from pairwise comparisons too. We only added $\{1, 1, 1\}$ since we started counting at 1 rather than 0.

$A > B > C$	$\{A, B\}$	$\{A, C\}$	$\{B, C\}$	Total
A	1	1	0	2
B	0	0	1	1
C	0	0	0	0

It has been suggested in the literature that Borda already knew this connection. Given this structural identity, can we still say that there is a difference between the Borda count and the Condorcet pairwise comparisons? Well, there still is a difference in how we deal with these data. Section 4.7.4 had an example where a Condorcet winner could still lose in the Borda count.

4.8.2 Margin count and Borda

For the *pure* cycle of 3 items as in the Condorcet case, we can prove that Borda and Condorcet give the same result, for any distribution of votes.

Condorcet[]; Preferences

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

- Let us take above cycle, and use abstract votes.

```
Clear[p, q]; Votes = {p, q, 1 - p - q};
```

- The pairwise vote matrix and row sums.

```
VoteMatrix[]
```

$$\begin{pmatrix} 0 & -p-q+1 & 1-p \\ p+q & 0 & q \\ p & 1-q & 0 \end{pmatrix}$$

```
pw = Plus @@ Transpose[%]
```

```
{-2 p - q + 2, p + 2 q, p - q + 1}
```

- The Borda sum.

```
bor = Votes . Preferences // Simplify
```

```
{-2 p - q + 3, p + 2 q + 1, p - q + 2}
```

- The difference between the vote count and Borda sum.

```
bor - pw
```

```
{1, 1, 1}
```

This means that the maximal item in the margin count will also be the maximal item in the Borda sum, for any distribution of votes. The schemes will select the same item.

Given the connection between the Borda count and pairwise voting, we may expect this property holds for cycles in general. However, I am not familiar with a theorem on this.

4.8.3 Properties due to current programming

When we compare the pairwise method and the BordaFP method, then we should distinguish the general properties from the particular properties of the current programming implementations.

- BordaFP uses fixed points, but PairwiseMajority has not been programmed for its own fixed points.
- BordaFP uses the Borda count again over a set of fixed points. It does not take the Borda count over the whole budget in order to protect itself from

less essential items. But PairwiseMajority simply is not programmed for a similar protection. If we test PairwiseMajority on the set of Borda fixed points, then we find that it can find the same solution. (Namely, see the former section 4.8.2.)

Yet, when it comes down to choosing, the particular properties of these routines allow us to judge that BordaFP likely is better than PairwiseMajority, even if we assume that the latter could be amended for its own fixed points.

4.8.4 Example to explain these properties

The properties discussed above can be shown by the following case. This case has no Condorcet winner but a deadlock on three alternatives. The PairwiseMajority margin count leads to the selection of item z (which would also be selected by Borda), while BordaFP selects y . The major cause for this difference is that BordaFP collapses to the fixed points. BordaFP does this automatically - if we manually collapse for PairwiseMajority, then we find that this routine also selects y .

```

Clear[x, y, z, u, v];
Items = {u, v, x, y, z};
Votes = {.4, .4, .2};

lis = {Pref[u, x, y, z, v],
       Pref[u, v, z, x, y],
       Pref[u, y, v, z, x]}

{Pref(u, x, y, z, v), Pref(u, v, z, x, y), Pref(u, y, v, z, x)}

lis // SetPreferences; Preferences

$$\begin{pmatrix} 1 & 5 & 2 & 3 & 4 \\ 1 & 2 & 4 & 5 & 3 \\ 1 & 3 & 5 & 2 & 4 \end{pmatrix}$$


```

PairwiseMajority and BordaFP give out messages on cycles and local sets that seem a bit confusing. They also select different points

- PairwiseMajority puts out a cycle message on $\{v, x, y\}$, and this has been produced by FindCycle of the `Combinatorica`` package. Yet, the Pref object includes z in the cycle. This is for the reason that z has the same amount of wins as x . Note that PairwiseMajority has not been programmed to specifically recognise fixed points.
- Since there is a cycle and no clear Condorcet winner, PairwiseMajority suggests the status quo u but also offers the margin count winners $\{y, z\}$.

PairwiseMajority[]

VoteMarginToPref::cyc : Cycle {y, v, x, y}

$$\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & -1. & -1. & -1. & -1. \\ 1. & 0 & -0.2 & 0.2 & -0.2 \\ 1. & 0.2 & 0 & -0.6 & 0.2 \\ 1. & -0.2 & 0.6 & 0 & -0.2 \\ 1. & 0.2 & -0.2 & 0.2 & 0 \end{pmatrix} \right),$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow u, \text{Sum} \rightarrow \{0, 2, 3, 2, 3\}, \text{Max} \rightarrow 3, \text{No Condorcet winner} \rightarrow \{x, z\},$

$\text{Pref} \rightarrow \text{Pref}(u, \{v, x, y, z\}), \text{Find} \rightarrow \{v, x, y, z\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow u\},$

$N \rightarrow \{\text{Sum} \rightarrow \{-4., 0.8, 0.8, 1.2, 1.2\}, \text{Pref} \rightarrow \text{Pref}(u, \{v, x\}, \{y, z\}), \text{Select} \rightarrow \{y, z\}, \text{All} \rightarrow \{y, z\}\}$

- The message by BordaFP is on $\{x, y, z\}$. But then it applies Borda to this selection, and finds y .

BordaFP[]

BordaFP::set : Local set found: {x, y, z}

BordaFP::chg : Borda gave {y, z}, Fixed Point is y

y

- If we call BordaAnalysis, then we see that y and z are equal candidates, and both fixed points. BordaAnalysis however is basically Borda - it does no active search for fixed points and thus does not see x .

BordaAnalysis[]

$\{\text{Select} \rightarrow \{y, z\}, \text{BordaFPQ} \rightarrow \{\text{True}, \text{True}\},$

$$\text{WeightTotal} \rightarrow \{1., 3.4, 3.4, 3.6, 3.6\}, \text{Position} \rightarrow \begin{pmatrix} 4 \\ 5 \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} 1. & u \\ 3.4 & v \\ 3.4 & x \\ 3.6 & y \\ 3.6 & z \end{pmatrix}$$

- If we would collapse for PairwiseMajority too, then it would also select y . This we however already clarified in 4.8.2.

SelectPreferences [{**x**, **y**, **z**}] ; **Preferences**

CheckVote::adj : NumberOfItems adjusted to 3

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

PairwiseMajority []

VoteMarginToPref::cyc : Cycle {z, x, y, z}

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -0.6 & 0.2 \\ 0.6 & 0 & -0.2 \\ -0.2 & 0.2 & 0 \end{pmatrix}, \right.$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow x, \text{Sum} \rightarrow \{1, 1, 1\}, \text{Max} \rightarrow 1, \text{No Condorcet winner} \rightarrow \{x, y, z\},$

$\text{Pref} \rightarrow \text{Pref}(\{x, y, z\}), \text{Find} \rightarrow \{x, y, z\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow x\},$

$N \rightarrow \{\text{Sum} \rightarrow \{-0.4, 0.4, 0.\}, \text{Pref} \rightarrow \text{Pref}(x, z, y), \text{Select} \rightarrow y\}, \text{All} \rightarrow y\}$

We conclude, at this point, that BordaFP seems to have all the good points of Condorcet's PairwiseMajority, while the demands on the voters are less severe. Pairwise voting on n items requires $n(n-1)/2$ pairwise votes, while BordaFP requires only a permutation of n . While the calculation is more complex, its steps can be demonstrated and easily understood.

4.8.5 Ties

As said, ties are a crucial issue for voting when there are strongly opposing views. If we need to find out about the intensities of feelings, then the Borda scheme provides us with the rankings per voter, and we can do research on them. The Condorcet pairwise comparisons do not immediately provide much information. A 50/50 result may be caused by strongly or mildly opposing views.

- Consider some opposing views.

DefaultItems[4] ; **EqualVotes** [] ;

{**Pref** @@ **Items**, **Pref** @@ **Reverse**[**Items**]}

{**Pref**(A, B, C, D), **Pref**(D, C, B, A)}

% // **SetPreferences** ;

- The vote matrix shows "indifference" everywhere. This matrix thus is less informative, since on $\{B, C\}$ the opposing views do not differ much, and on $\{A, D\}$ the rankings differ a lot.

VoteMatrix[]

$$\begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}$$

- There are no margins of victory.

ToVoteMargin[]

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \right)$$

If the matrix of margins shows only zero's, then we know that all preferences are opposed, and we can infer that also the rankings would be like that. To some extent, the pairwise information allows us to work our way back to the information provided by the Borda scheme. But clearly we lose information somewhere, since the vote matrix only provides information on n items, and there would be m voters.

A bottom line however remains that even a Borda scheme does not provide essential data on the intensities. Borda uses only ordinal rankings, and not interval-ratio ratings. The "strongly opposing views" on $\{A, D\}$ might concern something trivial as how we should colour 1 particular grain of sand in the whole of the Sahara.

It follows that Borda's approach would be more informative for settling ties than Condorcet's pairwise approach, but only to a limited extent. Also, while we would ask voters to provide us with their Borda information, we still could calculate the vote margin counts, if these are less sensitive to cheating. It is a different thing to use the Borda information to look for compromises, and to use the margin to break the tie if no such compromise appears possible.

4.8.6 Donald Saari's approach

On March 4th 2000, Voting Theory hit the columns of The Economist, in the article "The mathematics of voting; Democratic symmetry" p97:

(...) In a paper just published in the journal Economic Theory, Donald Saari, a mathematician at Northwestern University in Evanston, Illinois, claims to have got to the root of the problem. It is, he says, all to do with symmetry (...). Essentially, says Dr Saari, voting paradoxes arise when the system fails to respect natural cancellations of votes. In a two-candidate contest, for example, nobody would deny that the candidate with the most first-preference votes should win. One way to explain this is that votes of the form AB (ie, candidate A is preferred to candidate B) should cancel out votes of the form BA. if this leaves a surplus of A then A wins. These cancellations are a form of reflectional symmetry. But votes in a three-candidate election should cancel out, too (....) This is a form of rotational symmetry, since the three votes form a rotating cycle. Taking these two symmetries into account, it is possible to characterise all paradoxes for a three-candidate election under any voting procedure. Dr Saari's results can also be generalised for elections with more than three candidates using more complicated, but closely related symmetries. It is thus possible to evaluate the "fairness" of different voting systems. (...) The fairest voting system, says Dr Saari, would respect both symmetries, (....)

This argument is more abstract, and we can better use Saari's own example, and run our routines. Saari (S&C:6) discusses the following example, where he starts with a clear case, and then adds a cycle to show his reasoning.

- There are 48 voters.

```
Clear[A, B]; Items = {A, B, C}; a = {20, 28}; Votes = a / Add[a]
```

$$\left\{ \frac{5}{12}, \frac{7}{12} \right\}$$

```
pr = {ToPref[A > B > C], ToPref[B > A > C]}
```

```
{Pref(C, B, A), Pref(C, A, B)}
```

```
PrefToList[%]
```

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 1 \end{pmatrix}$$

```
% // SetPreferences;
```

- B wins in all schemes (when we neglect the status quo issue).

Constitutions[]

$\{ \text{Borda} \rightarrow B, \text{ParetoMajority} \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Pareto} \rightarrow \{A\}, \text{Select} \rightarrow A \},$

$\text{PairwiseMajority} \rightarrow \{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & -\frac{1}{6} & 1 \\ \frac{1}{6} & 0 & 1 \\ -1 & -1 & 0 \end{pmatrix} \right), 1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{1, 2, 0\},$

$\text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow B, \text{Pref} \rightarrow \text{Pref}(C, A, B), \text{Find} \rightarrow B, \text{LastCycleTest} \rightarrow \text{False}, \text{Select} \rightarrow B \},$

$N \rightarrow \{ \text{Sum} \rightarrow \{ \frac{5}{6}, \frac{7}{6}, -2 \}, \text{Pref} \rightarrow \text{Pref}(C, A, B), \text{Select} \rightarrow B \}, \text{All} \rightarrow B \} \}$

- Next, a cycle of opposing votes is added.

a = {20, 28, 9, 9, 9}; Votes = a / Add[a]

$\left\{ \frac{4}{15}, \frac{28}{75}, \frac{3}{25}, \frac{3}{25}, \frac{3}{25} \right\}$

pr = ToPref /@ {A > B > C, B > A > C, A > B > C, B > C > A, C > A > B}

$\{ \text{Pref}(C, B, A), \text{Pref}(C, A, B), \text{Pref}(C, B, A), \text{Pref}(A, C, B), \text{Pref}(B, A, C) \}$

PrefToList[%]

$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 1 \\ 3 & 2 & 1 \\ 1 & 3 & 2 \\ 2 & 1 & 3 \end{pmatrix}$

% // SetPreferences;

- Borda still selects B , but suddenly A has become the Condorcet winner.

Constitutions[]

$\{ \text{Borda} \rightarrow B, \text{ParetoMajority} \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Pareto} \rightarrow \{A\}, \text{Select} \rightarrow A \},$

$\text{PairwiseMajority} \rightarrow \{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & \frac{1}{75} & \frac{13}{25} \\ -\frac{1}{75} & 0 & \frac{19}{25} \\ -\frac{13}{25} & -\frac{19}{25} & 0 \end{pmatrix} \right), 1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 1, 0\},$

$\text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow A, \text{Pref} \rightarrow \text{Pref}(C, B, A), \text{Find} \rightarrow A, \text{LastCycleTest} \rightarrow \text{False}, \text{Select} \rightarrow A \},$

$N \rightarrow \{ \text{Sum} \rightarrow \{ \frac{8}{15}, \frac{56}{75}, -\frac{32}{25} \}, \text{Pref} \rightarrow \text{Pref}(C, A, B), \text{Select} \rightarrow B \}, \text{All} \rightarrow A \} \}$

Saari's reasoning now is, that the votes that cause the cycle can be considered to be irrelevant - since they 'cancel' each other. (It is a matter of words to call this 'symmetry'.)

However, let us see what BordaFP has to say on this.

- BordaFP selects A , since it wins from the alternative winner (B).

BordaFP[]

BordaFP::chg : Borda gave {B}, Fixed Point is {A}

A

BordaFPQ /@ Items

CheckVote::adj : NumberOfItems adjusted to 2

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 2

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

{True, False, False}

BordaAnalysis[] // N

{Select → B , BordaFPQ → {False},

WeightTotal → {2.26667, 2.37333, 1.36}, Position → (2.), Ordering → $\begin{pmatrix} 1.36 & C \\ 2.26667 & A \\ 2.37333 & B \end{pmatrix}$ }

My reasoning is - contrary to Saari - that the added votes cannot be neglected. If we consider the fixed points, then the addition has an effect, since when we consider a winner and its alternative winner, in this case A and B only, then the added votes are in favour of A .

SelectPreferences[{A, B}];

CheckVote::adj : NumberOfItems adjusted to 2

Preferences

$\begin{pmatrix} 2 & 1 \\ 1 & 2 \\ 2 & 1 \\ 1 & 2 \\ 2 & 1 \end{pmatrix}$

Also, item C is a typical example of an irrelevant item that can cause a preference reversal in Borda voting. When we discussed Borda, we observed its sensitivity to preference reversal, and we introduced BordaFP to deal with that. So Saari's reasoning that the votes 'should cancel' is not 'obvious' (from this

point of view). While Saari would hold that C is of vital importance since it shows a cycle for some voters, I would hold that C could be neglected since it is not a fixed point.

Basically, we here have the choice whether we attach more importance to the voters or to the items. Saari says that the items are more important, since he cancels the votes of 27 voters and keeps C in the race. I would say that the votes are important, that item C is a less relevant item, and that the proper question is whether the winner is a convincing winner. Of course, C can become an important item, if we add other voters. But then the argument is that those voters count, rather than C . (When we would add an item D such that the first voter has $A > B > C > D$, the second voter $D > B > A > C$, and D part of the cycle for the other voters, then this distinction might be less clear.)

See section 10.9.9 for further notes that lead to a rejection of Saari's approach. Saari's result is actually deeper, since he would appear to prove that "symmetry" \Leftrightarrow Borda. See Saari's papers at http://www.math.nwu.edu/~d_saari/. Assuming equivalence to Borda (which I did not check, though) then there would be no need for implementation of this "cycle or rotational symmetry" in a *Mathematica* program. Given my general rejection of his approach, I have not developed this angle further.

4.8.7 Dependence on the budget

Let the budget first consist of status quo A and alternative B only.

- BordaFP and PairwiseMajority select B.

```
DefaultItems[]; EqualVotes[];
{{1, 2}, {1, 2}, {2, 1}} // SetPreferences;

BordaFP[]

B

All /. PairwiseMajority[]

B
```

The next year, the group is richer, and option C becomes available. The three voters retain their old preference order, but include C in different positions.

```
new = {AppendPref[{1, 2}, 3],
       AppendPref[{1, 2}, 0],
       AppendPref[{2, 1}, 1.5]}
```

CheckVote::adj : NumberOfItems adjusted to 3

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

```
% // SetPreferences;
```

```
BordaFP[]
```

```
{A, B, C}
```

```
All /. PairwiseMajority[]
```

VoteMarginToPref::cyc : Cycle {C, A, B, C}

```
{A, B, C}
```

You may have noticed that the new preferences actually are from the Condorcet case. While cycles generally can be solved by declaring a tie (and then applying some tie-breaking rule), they retain their main paradoxical character from critical budget changes. BordaFP can solve preference reversals when the budget has been given, but it cannot foresee that there could be a future budget change that would cause such a preference reversal. In this example case, a clear preference $B > A$ is changed into $A = B = C$, for the same people, and only by including C. If the group would adopt the tie-breaking rule to resort to the status quo, then now, while it has become richer, it should accept the status quo A even though it was considered inferior before.

It is a property of the ordinal approach that this can happen. The ordinal approach does not use the information about the preference intensities, and thus the weights for the various items depend upon the budget of considered items. In a cardinal approach, the weights would remain fixed, and the deduction then of course would not change. Budget dependence thus is the price for trying to prevent cheating.

In this case, a group, in real practice, who notes that preference $B > A$ is changed into $A = B = C$, would locate the problem and reopen negotiations. In such negotiations, the group would very likely use some of the available cardinal information that is avoided in the official voting rules. Some people would hotly advocate one solution, others might put up a show of disappointment or disgust. Eventually the group could decide that the earlier B

> A was a mistake, or it could be creative and find another alternative that is an acceptable improvement over the status quo. Rather than adhering to the strict rules of these programmed routines, with their limited logic, the group would use common sense as well. The rules would eventually be used for the official decision and registration thereof, as society eventually adheres to proper procedure. But the use of the voting rules is embedded in a larger structure, which makes that the limitations to the ordinal approach are less severe than perhaps originally thought.

Note, incidently, that C here is a Borda Fixed Point, so that we did not add an irrelevant item. The budget dependence of BordaFP is limited to such fixed points - which are the strong contenders. A budgetting process that wants to reduce future preference reversals thus can be advised to try to forecast potential strong contenders, and to include them, prospectively, in current decision making.

4.9 Voting and graphs

4.9.1 Introduction

A *graph* is a collection of some points (vertices) with some connecting lines (edges). Not all points need to be connected. In a directed graph the connections are depicted with arrows, which indicates a relation of dominance. Graphs can be handled by Steve Skiena's `DiscreteMath`Combinatorica`` package. Note that the `Voting`` package has already called this package.

For voting, we are free to define dominance as *winning* ($>$) or as *losing* ($<$). There are two reasons to take the second approach. The first is that we already have adopted the $<$ relation for the `Pref` object. The second reason is that incoming arrows are easier to count (visually) than outgoing arrows. A consequence is that we must transpose the `VoteMargin` object when we turn it into a `Graph` object.

- We don't evaluate this, since the list of routines is quite long.

```
Contents["DiscreteMath`Combinatorica`"]
```

<code>InvertGraph[g_Graph]</code>	transposes the adjacency matrix, so that the $>$ relation becomes the $<$ relation, and the To arrows become From arrows
<code>GraphToPref[g_Graph]</code>	assumes a $<$ directed graph where indifferent points are connected by arrows, and turns this into a <code>Pref</code> object

4.9.2 Short introduction to graphs

- This tells us what a graph is.

`?Graph`

```
Graph[g,v] represents a graph object where g is
an adjacency matrix and v is a list of vertices. More...
```

- This is a graph in which only B (second row) sends out something to A and C .

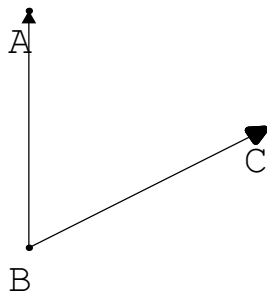
```
gr = Graph[{{0, 0, 0}, {1, 0, 1}, {0, 0, 0}},
           {{-0.5, 0.866}, {-0.5, -0.866}, {1., 0}}]
```

$$\text{Graph}\left(\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866 \\ -0.5 & -0.866 \\ 1. & 0 \end{pmatrix}\right)$$

If we want to plot a graph, then Skiena provides the routines `ShowGraph` and `ShowLabeledGraph`. It turns out that we want to plot something else, namely Labeled Directed Graphs, and thus `Voting` provides for a new routine to do so.

- Above graph now is called a Labeled Directed Graph.

```
ShowPrefGraph[gr];
```



- We translate this in the following `Pref` object.

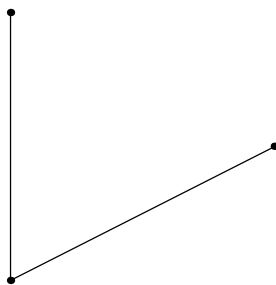
```
GraphToPref[gr]
```

```
VoteMarginToPref::cyc : Cycle {C, A, C}
```

```
Pref(B, {A, C})
```

- `ShowGraph` plots the same graph as an Undirected and Unlabeled Graph.

```
ShowGraph[gr];
```



ShowPrefGraph[
graph, opts]

shows a graph in a typical preference analysis situation:
as a labeled directed graph. Adjust with options

4.9.3 VoteMargin to graphs

- A VoteMargin object normally is negative symmetric.

```
vm = ListToVoteMargin[{1, 5/2, 5/2}]
```

$$\text{VoteMargin}\left(\begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}\right)$$

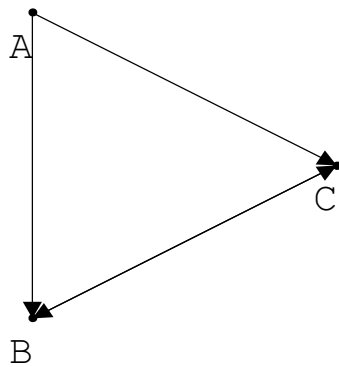
- This is its graph: nonnegative elements become 1, the negative elements become zero, and it is transposed. A lot of work indeed, but needed to get the arrows right.

```
gr = VoteMarginToGraph[vm]
```

$$\text{Graph}\left(\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ 1. & 0 \end{pmatrix}\right)$$

- As we knew from *vm*, *A* sends to all, while *B* and *C* send to each other. The move from *A* to one of the others is improving.

```
ShowPrefGraph[gr];
```



In the transformation to the graph, all negative elements are set to zero, and all positive elements are set to 1. Off-diagonal zero elements in the VoteMargin object, that denote indifference, can be represented at least in two ways in graphs. This is controlled by the option SameQ:

- With the option `SameQ → False` (default): all off-diagonal zero's are set to 1, so that all items can be plotted separately, and indifference is given by two arrows 'to and fro'.
- With the option `SameQ → True`: indifference is represented by a value 0 in the adjacency matrix. In that case the adjacency values represent the 'distances' between points, so two indifferent items are plotted at the same vertex.

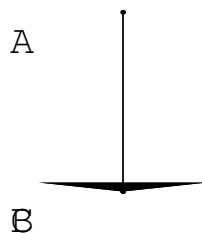
- The indifference between *B* and *C* now means that they are the same point.

```
gr2 = VoteMarginToGraph[vm, SameQ → True]
```

$$\text{Graph}\left(\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ -0.5 & -0.866025 \end{pmatrix}\right)$$

- This arrow is a bit peculiar, but it *is* an arrow. Also *B* and *C* are printed across one another.

```
ShowPrefGraph[gr2];
```



```
VoteMarginToGraph[  
x_VoteMargin, opts]
```

turns a `VoteMargin` object into a `Graph`, which may provide a bridge to the routines in `DiscreteMath`Combinatorica``, such as `ShowGraph`.

4.9.4 The Condorcet cycle

Consider the Condorcet case.

```
Condorcet[]; Preferences
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

```
e = PairwiseMajority[]
```

```
VoteMarginToPref::cyc : Cycle {C, A, B, C}
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -0.2 & 0.5 \\ 0.2 & 0 & -0.3 \\ -0.5 & 0.3 & 0 \end{pmatrix}, \right.$$

```
1 → {StatusQuo → A, Sum → {1, 1, 1}, Max → 1, No Condorcet winner → {A, B, C},
  Pref → Pref({A, B, C}), Find → {A, B, C}, LastCycleTest → True, Select → A},
N → {Sum → {0.3, -0.1, -0.2}, Pref → Pref(C, B, A), Select → A}, All → A}
```

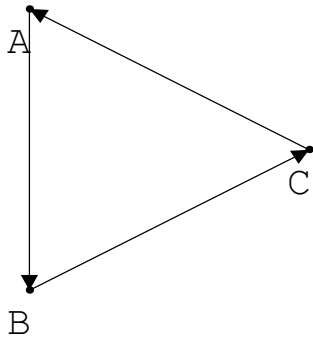
```
f = VoteMargin /. e
```

$$\text{VoteMargin} \begin{pmatrix} 0 & -0.2 & 0.5 \\ 0.2 & 0 & -0.3 \\ -0.5 & 0.3 & 0 \end{pmatrix}$$

```
g = VoteMarginToGraph[f]
```

$$\text{Graph} \left(\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ 1. & 0 \end{pmatrix} \right)$$

```
ShowPrefGraph[g];
```



- In this case, we can find that there is cycle.

```
FindCycle[g, Directed]
```

```
{3, 1, 2, 3}
```

Note that, if a graph matrix is difficult to read, then TransitiveReduction helps to clean up the matrix, so that we can spot cycles easier.

Remember that when we work with single lists, then there will be no cycles.

Since looking for cycles is an important activity in voting, a routine has been written so that that Steve Skiena's FindCycle is directly called for Preferences and Votes.

UseFindCycle[<i>p:Preferences, v:Votes, opts</i>]	transforms to a Graph and uses FindCycle of the Combinatorica` package
UseFindCycle[<i>x_VoteMargin, opts</i>]	does the same

See VoteMarginToGraph for both. The graph is available in Results[UseFindCycle].

These are subroutines that we do not evaluate.

```
ShowListGraph[x_List, opts]

    performs ListToVoteMargin, VoteMarginToGraph,
    and ShowPrefGraph. The options apply to these and
    Show. For two elements you may want to adjust the PlotRange

ListToGraph[x_List, opts___Rule]

    performs VoteMarginToGraph[ListToVoteMargin[x], opts]
```

4.9.5 Graph Intersection and Pareto improvements

?GraphIntersection

GraphIntersection[g, h] gives the graph defined
by the edges which are in both graph g and graph h. [More...](#)

- For these two voters, the moves from *A* to *B* and from *B* to *C* are *relatively* Pareto optimising.

```
a = ListToVoteMargin[{1, 2, 2}]
b = ListToVoteMargin[{1, 1, 2}]
```

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \right)$$

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \right)$$

```
ag = VoteMarginToGraph[a]
bg = VoteMarginToGraph[b]
```

$$\text{Graph}\left(\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ 1. & 0 \end{pmatrix}\right)$$

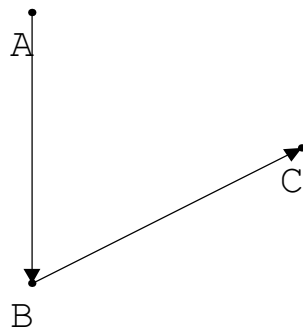
$$\text{Graph}\left(\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ 1. & 0 \end{pmatrix}\right)$$

- The intersection reveals this PO situation.

```
gi = GraphIntersection[ag, bg]
```

$$\text{Graph}\left(\begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -0.5 & 0.866025 \\ -0.5 & -0.866025 \\ 1. & 0 \end{pmatrix}\right)$$

```
ShowPrefGraph[TransitiveReduction[gi]];
```



- Which gives just the *group* Pref object that we expected.

```
GraphToPref[gi]
```

```
Pref(A, B, C)
```


5. Combined schemes

5.1 Introduction

5.1.1 Introduction

In this part, we combine some basic schemes.

Where the basic schemes still are deficient, is that they violate the Pareto condition. Hence it is straightforward to take the combinations where *first* the Pareto points are selected, and *only then* the particular schemes are applied. In fact, it can be argued that this would be the position of the classical liberal. In the viewpoint of the classical liberal, majority voting would only be acceptable to solve an indecision about a collection of Pareto optimising points. (Note that chapter 9.6 rejects Sen's argument on the 'impossibility of a Paretian liberal'.)

A Majority Plurality winner is also a Condorcet winner. A combination might be to first hold a Plurality round, and only use pairs if there is no Majority winner. Given that such winners are more the exception than the rule, this however is a rather Byzantine construction.

As said, the number of pairs, used by Condorcet, rises quadratically, which would put great demands on voters. BordaFP only requires an individual ordering. Since BordaFP contains the Majority Plurality and Binary Condorcet cases, there seems little need to look for other combinations than Pareto and BordaFP.

It may be noted however that BordaFP sometimes gives a tie between fixed points, while the Condorcet margin count, over the whole budget, still indicates a difference. This margin count then could be used to settle the tie. This scheme we shall call the `Majority` scheme, and it will be discussed below.

The combination of Pareto and Majority appears to be strongest. For this reason we start with this.

(1) `ParetoMajority` first collapses the preferences to the Pareto points, and then it applies Borda while it also takes account of the fixed point condition. Sets of the Fixed Points are decided upon first with Borda, and if that does not help with the Condorcet margin count over the whole budget. `SetOptions[ParetoMajority, N → ...]` allows you to control how the collapsed preference should look. A value `Automatic` gives ordinality, a value `Infinity`

maintains the original values (presuming cardinality), a fixed value gives this total (for interval or ratio scale).

(2) `ParetoBorda` uses Borda on the original scores. When interval, ratio or cardinal scales are used, then there is more room for cheating in the second round. Since there is no fixed point condition, preference reversal is possible.

(3) `ParetoPairwise` first collapses the preferences and then applies pairwise voting to the Pareto Points. (The first collapse is to ordinality, but that does not affect pairwise voting.)

(4) `ParetoPlurality` first collapses to the Pareto points, and then applies the Plurality rule. Again: if there is a Majority Plurality winner, this is also found by `ParetoMajority`. If there is no Majority Plurality winner, then you may have to accept a non-BordaFP solution.

(5) `ParetoApproval` first collapses to the Pareto points, and then applies Approval voting. The problem remains that there is no clear rule for changing an ordinal preference into an Approval statement.

The routine `Constitutions[]` calls `ParetoMajority`, `Borda` and `PairwiseMajority`.

<code>Constitutions[]</code>	calls the <code>ParetoMajority</code> , <code>Borda</code> and <code>PairwiseMajority</code> constitutions while using current values of <code>Preferences</code> and <code>Votes</code>
------------------------------	--

Technical note: `ParetoBorda` and `ParetoPairwise` may redefine global variables like `Preference` or `Items`, and thus have not been included in `Constitutions[]`. A redefinition of the variables by one routine would hinder the other. `ParetoMajority`, `Borda` and `PairwiseMajority` have also been programmed with different algorithms, so they can provide information on different angles.

5.1.2 The Frerejohn and Grether paradox

The Frerejohn & Grether paradox (F&G) appears to be illuminating for various angles of interest, and it will be used more often in this book. See also Sen (1986:1103). Three preference orderings (1: $x > y > z > w$), (2: $y > z > w > x$) and (3: $z > w > x > y$), would, with pairwise majority vote without reflection, give ($x > y > z > w > x$). Let us see how the Fixed Point Borda deals with this.

■ Define the case.

```
EqualVotes[]; Clear[w, x, y, z]; Items = {w, x, y, z};

{Pref[w, z, y, x], Pref[x, w, z, y], Pref[y, x, w, z]} // SetPreferences;
```

Preferences

$$\begin{pmatrix} 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \\ 3 & 2 & 1 & 4 \end{pmatrix}$$

- Pairwise majority gives a cycle of all four, but z would have the highest margin.

PairwiseMajority[]

VoteMarginToPref::cyc : Cycle {x, w, y, x}

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & \frac{1}{3} & -\frac{1}{3} & -1 \\ -\frac{1}{3} & 0 & \frac{1}{3} & -\frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & 0 & \frac{1}{3} \\ 1 & \frac{1}{3} & -\frac{1}{3} & 0 \end{pmatrix} \right\},$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow w, \text{Sum} \rightarrow \{1, 1, 2, 2\}, \text{Max} \rightarrow 2, \text{No Condorcet winner} \rightarrow \{y, z\},$
 $\text{Pref} \rightarrow \text{Pref}(\{w, x, y, z\}), \text{Find} \rightarrow \{w, x, y, z\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow w\},$

$N \rightarrow \{\text{Sum} \rightarrow \{-1, -\frac{1}{3}, \frac{1}{3}, 1\}, \text{Pref} \rightarrow \text{Pref}(w, x, y, z), \text{Select} \rightarrow z\}, \text{All} \rightarrow z\}$

- BordaFP shows that x, y and z form a fixed point set, which explains why w would drop out. But BordaFP cannot settle the tie that exists in this set of fixed points, since Borda on it again gives a tie.

lis = BordaFP[]

BordaFP::set : Local set found: {x, y, z}

BordaFP::chg : Borda gave {z}, Fixed Point is {x, y, z}

{x, y, z}

- Of course, F&G did not specify what the status quo was.

Pareto[]

{w, z}

Note that the BordaFP set is $\{x, y, z\}$ while the Borda solution *over the whole budget set* is z . We could settle the fixed point tie by taking the Borda point, if it belongs to the fixed point set. On the other hand, we note that z also has the highest Condorcet margin count. It seems - but this is intuition only - that the latter is more robust against cheating on ties. A proposal is to take the Condorcet margin count as a final tie breaker. Note that we must take the margin count *over the whole budget*, since the margin count over the set of fixed points gives indecision as well (Borda and the margin count are then the same).

The F&G paradox thus causes some fundamental questions: (1) For the *pure* 3 items cycle we already have shown that the BordaFP set is the same as a Condorcet cycle. Is this always the case ? (2) When BordaFP settles for a tie, can we then take the Condorcet margin count to break it ? Or would it be that the Borda count does not really differ from the Condorcet margin count, also for the whole budget set (and not just for cycles) ? (3) While the above assumes non-cheating, how does this work out with cheating ?

Since Condorcet relies on pairs and since the number of pairs rises quadratically, the Borda scheme is more economical in general. But when ties arise, for likely small numbers of items, then the Condorcet margin count over the whole budget would provide additional information, and thus it might be used to settle those ties. That margin count can easily be calculated from the preference information, without any additional burden to the voters. (If it would turn out that this approach would be equivalent to Borda - and we are speaking here only about tie-breaking while using the whole budget set - then it would still be nice to keep Condorcet's name in here as well, in memory of his contribution.)

We should be critical about how to establish the margin count. If we reduce the problem to the cycle only, then the margin count evaporates.

- Reset the problem to the cycle only.

`SelectPreferences[1is]; Preferences`

$$\begin{pmatrix} 3 & 2 & 1 \\ 1 & 3 & 2 \\ 2 & 1 & 3 \end{pmatrix}$$

- Determine the VoteMargin object.

`ToVoteMargin[]`

$$\text{VoteMargin} \left(\begin{pmatrix} 0 & \frac{1}{3} & -\frac{1}{3} \\ -\frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & 0 \end{pmatrix} \right)$$

- While the Condorcet margin count seemed to be in favour for z , this advantage now has disappeared. We cannot use the margin count *of the cycle* to settle the tie.

`VoteMarginToCount[%]`

`{Sum → {0, 0, 0}, Pref → Pref({x, y, z}), Select → {x, y, z}}`

The proper consideration is rather that *the whole budget* consists of $\{w, x, y, z\}$, so that dropping w is a needless dis-informative act, that destroys information. Thanks to w we know more about the preference order between for example x and z . Thus, we could reasonably use the margin count of the whole budget. This would create some dependence of the final solution on the budget, but, since this only holds for ties, it could well be accepted. A budget-dependent tie-breaker is better than no tie-breaker, especially since it depends upon the preferences.

5.1.3 The Majority routine

The Majority routine works like the others.

`Majority[p:Preferences, v:Votes, i:Items]`

applies BordaFP[p, v, i], and if the solution has a cycle,
then breaks the tie with the Condorcet margin count on the whole budget set

- Reconsider the Frerejohn & Grether paradox. We have to redefine it, since we used SelectPreferences above.

`EqualVotes[]; Clear[w, x, y, z]; Items = {w, x, y, z};`

`{Pref[w, z, y, x], Pref[x, w, z, y], Pref[y, x, w, z]} // SetPreferences;`

`Preferences`

$$\begin{pmatrix} 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \\ 3 & 2 & 1 & 4 \end{pmatrix}$$

- Solve it, while neglecting the status quo and Pareto issues.

Majority[]

BordaFP::set : Local set found: {x, y, z}

BordaFP::chg : Borda gave {z}, Fixed Point is {x, y, z}

$$\left\{ \text{BordaFP} \rightarrow \{x, y, z\}, \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & \frac{1}{3} & -\frac{1}{3} & -1 \\ -\frac{1}{3} & 0 & \frac{1}{3} & -\frac{1}{3} \\ \frac{1}{3} & -\frac{1}{3} & 0 & \frac{1}{3} \\ 1 & \frac{1}{3} & -\frac{1}{3} & 0 \end{pmatrix} \right\},$$

$$N \rightarrow \left\{ \text{Sum} \rightarrow \left\{ -1, -\frac{1}{3}, \frac{1}{3}, 1 \right\}, \text{Pref} \rightarrow \text{Pref}(w, x, y, z), \text{Select} \rightarrow z \right\}, \text{Select} \rightarrow z \}$$

We can note two key properties:

- If the Borda winner is also a Borda Fixed Point, and if the Borda winner is also the Condorcet margin count winner, then the Majority scheme gives the same result as Borda. The premisses however are not always true.
- The Majority result is less dependent on the budget because of BordaFP. But its tiebreaker is fully dependent on it, since the Borda fixed point set does not provide enough information to break the tie and since we thus deliberately consider the whole budget. If you want to break the tie in this manner, then you must be sure that you have included all important items.

Thus note: A key motivation in voting theory for democracies is that we want the results to be dependent upon the preferences of the individuals. This can be called the First Principle. One important consequence of the First Principle is that results will also depend upon the budget. Different budget items trigger different preferences, and if we allow only ordinal information to deter cheating, then results will be conditional to the budget. Some authors then also impose axioms (in particular the *APDM* that will be discussed below) which effectively kills the dependence on the budget. But this then leads to an inconsistency. And trying to impose such an axiom is inconsistent to start with, since the dependence on the budget is one important *aspect* of that First Principle. It is only for individuals that we hypothesise the independence of preferences on the budget, but for the aggregate we cannot exclude a dependence on *a priori* grounds. So the imposition of such axioms (and *APDM* in particular) is inconsistent with the First Principle. Subsequently, a tie-breaking rule that introduces another dependence upon the budget could again

cause 'paradoxes' - but accepting that rule would be exactly what we wanted in the first place, namely dependence on individual preferences.

5.2 Pareto Majority

5.2.1 Pareto (efficiency) majority

If $B > A$ and $C > A$ are both Paretian improvements (from the Status Quo A), while there is no clear efficiency preference on $\{B, C\}$, then there might still be a deadlock. The ParetoMajority rule solves this deadlock by Fixed Point Borda majority voting. This itself already uses Borda over the set of fixed points. If there still is a tie, then the Condorcet margin count on the whole budget is used. A final deadlock of indifference by still remaining equal scores is left to the user. You may select the Status Quo, use dice, etcetera.

- ParetoMajority[] applied to the Condorcet situation gives:

```
Condorcet[]; ParetoMajority[]

{StatusQuo → A, Pareto → {A}, Select → A}
```

- An application to a random set of preferences.

```
DefaultItems[]; pr = SetRandomPreferences[4, 6];
SetFirstValue[2]
```

$$\begin{pmatrix} 2 & 5 & 4 & 1 & 3 & 6 \\ 2 & 6 & 5 & 4 & 1 & 3 \\ 2 & 4 & 3 & 6 & 5 & 1 \\ 2 & 5 & 3 & 6 & 4 & 1 \end{pmatrix}$$

```
ParetoMajority[]

{StatusQuo → A, Pareto → {A, B, C}, Select → B}
```

```
ParetoMajority[
p:Preferences, v:Votes,
i:Items, s:StatusQuo[]]
```

first selects the Pareto points that dominate the Status Quo, and then applies the BordaFP rule. The Condorcet margin count of the whole budget is applied to final ties

Since Borda does not use fixed points, it is a less relevant second step. But it has been included here for completeness.

```
ParetoBorda[
p:Preferences, v:Votes]
```

first selects the EfficiencyPairs that dominate the Status Quo, and then applies the (plain) Borda majority rule to those

5.2.2 On an example given by Sen

Sen (1970:48) gives the following example. Assume that the status quo is C.

```
DefaultItems[3]; EqualVotes[];
SetPreferences[{ {3, 2, 1}, {1, 3, 2} }];
StatusQuo[] = "C"
```

C

■ Pareto Majority gives:

```
ParetoMajority[]
```

```
{StatusQuo → C, Pareto → {B, C}, Select → B}
```

Thus, there is clear solution. Only when your frame of mind consists of pairwise voting without a status quo, then you have the experience of paradox. Then, namely, the resulting social index is either intransitive or there is a cycle. The Binary PairwiseMajority routine cannot decide in that case, and selects the status quo. The Count rule would always select *B*, whatever the status quo. (It can be assumed that it is only used when the Binary situation gives a deadlock.)

```
PairwiseMajority[Show]
```

```
VoteMarginToPref::cyc : Cycle {B, A, B}
```

```
VoteMarginToBinary::dif : Selection C differs from Condorcet winning {A, B}
```

$$\left\{ \text{Outer} \rightarrow \begin{pmatrix} \{1, \{A, B\}\} & \{1, \{A, C\}\} & \{1, \{B, C\}\} \\ \{2, \{A, B\}\} & \{2, \{A, C\}\} & \{2, \{B, C\}\} \end{pmatrix}, \text{Pairwise} \rightarrow \begin{pmatrix} \{\frac{1}{2}, 0\} & \{\frac{1}{2}, 0\} & \{\frac{1}{2}, 0\} \\ \{0, \frac{1}{2}\} & \{0, \frac{1}{2}\} & \{\frac{1}{2}, 0\} \end{pmatrix} \right\}$$

$$\text{Sum} \rightarrow \begin{pmatrix} \{A, B\} & \{\frac{1}{2}, \frac{1}{2}\} \\ \{A, C\} & \{\frac{1}{2}, \frac{1}{2}\} \\ \{B, C\} & \{1, 0\} \end{pmatrix}, \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \right)$$

```
1 → {StatusQuo → C, Sum → {2, 2, 1}, Max → 2, Condorcet winner → {A, B},
```

```
Pref → Pref({A, B, C}), Find → {A, B, C}, LastCycleTest → True, Select → C},
```

```
N → {Sum → {0, 1, -1}, Pref → Pref(C, A, B), Select → B}, All → B}
```

5.2.3 Example of dependence of the budget

It is suggested here that ParetoMajority has the best papers to be generally accepted for common applications (exceptions excluded of course). At the same time, dependence on the budget is the main drawback of ordinal voting schemes. It is useful to show how these two points combine. We use the default `Vote[]` routine (at setup ParetoMajority), and compare it to the performance of Borda. Suppose that individuals 1, 2 and 3 compare a status quo w with three clear possible improvements x, y and z .

- Set Items and Votes. Use `DefaultVotes` to reset the Status Quo.

```
EqualVotes[]; DefaultItems[];
Clear[w, x, y, z]; Items = {w, x, y, z};
```

- 1 and 2 agree, while 3 takes an opposing view.

```
{a = {1, 2, 3, 4}, a, {1, 4, 3, 2}} // SetPreferences;
```

Preferences

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 4 & 3 & 2 \end{pmatrix}$$

- The 2/3 majority on the Pareto points cause z to be selected.

Vote[]

```
{StatusQuo → w, Pareto → {w, x, y, z}, Select → z}
```

- The group preference order can be determined.

```
v1 = VoteToPref[]
```

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::adj : NumberOfItems adjusted to 2

CheckVote::adj : NumberOfItems adjusted to 1

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

$$\left\{ \text{StatusQuo} \rightarrow \begin{pmatrix} w & w \\ w & x \\ w & y \\ w & z \end{pmatrix}, \text{Pref} \rightarrow \text{Pref}(w, x, y, z) \right\}$$

```
b1 = BordaAnalysis[]
```

CheckVote::adj : NumberOfItems adjusted to 4

$$\left\{ \text{Select} \rightarrow z, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{ 1, \frac{8}{3}, 3, \frac{10}{3} \right\}, \text{Position} \rightarrow (4), \text{Ordering} \rightarrow \begin{pmatrix} 1 & w \\ \frac{8}{3} & x \\ 3 & y \\ \frac{10}{3} & z \end{pmatrix} \right\}$$

It turns out that voter 3 is unhappy with the situation. Motivated by the bad outcome he or she starts spending a lot of money looking for another alternative, and indeed, succeeds in finding - say in a foreign country - item u that would be a Pareto improvement on z . Voters 1 & 2 consider the situation, and, while basically accepting u , they come up with alternative v that would indeed be better for 3 but that they themselves still prefer. The situation becomes:

```
EqualVotes[]; Clear[w, x, y, z, u, v]; Items = {w, x, y, z, u, v};
```

- 1 and 2 agree, while 3 takes another view.

```
{b = {1, 2, 3, 4, 5, 6}, b, {1, 6, 3, 2, 5, 4}} // SetPreferences;
```

Preferences

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 6 & 3 & 2 & 5 & 4 \end{pmatrix}$$

- The 2/3 majority on the Pareto points cause v to be selected.

```
Vote[]
```

```
{StatusQuo → w, Pareto → {u, v, w, x, y, z}, Select → v}
```

- The group preference order can be determined.

v2 = VoteToPref[]

CheckVote::adj : NumberOfItems adjusted to 5

CheckVote::adj : NumberOfItems adjusted to 4

CheckVote::adj : NumberOfItems adjusted to 3

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

$$\left\{ \text{StatusQuo} \rightarrow \begin{pmatrix} w & w \\ w & x \\ w & y \\ w & z \\ u & u \\ w & v \end{pmatrix}, \text{Pref} \rightarrow \text{Pref}(w, x, y, z, u, v) \right\}$$

b2 = BordaAnalysis[]

CheckVote::adj : NumberOfItems adjusted to 6

{Select → v, BordaFPQ → {True},

$$\text{WeightTotal} \rightarrow \left\{ 1, \frac{10}{3}, 3, \frac{10}{3}, 5, \frac{16}{3} \right\}, \text{Position} \rightarrow (6), \text{Ordering} \rightarrow \begin{pmatrix} 1 & w \\ 3 & y \\ \frac{10}{3} & x \\ \frac{10}{3} & z \\ 5 & u \\ \frac{16}{3} & v \end{pmatrix}$$

Let us now compare the group preferences and the Borda rankings.

- The Pref's give the same ranking.

Pref /. {v1, v2}

{Pref(w, x, y, z), Pref(w, x, y, z, u, v)}

- Borda however shows a preference reversal for $\{x, y\}$. And while first $x < z$, now $x = z$.

Ordering /. {b1, b2}

$$\left\{ \begin{pmatrix} 1 & w \\ \frac{8}{3} & x \\ 3 & y \\ \frac{10}{3} & z \end{pmatrix}, \begin{pmatrix} 1 & w \\ 3 & y \\ \frac{10}{3} & x \\ \frac{10}{3} & z \\ 5 & u \\ \frac{16}{3} & v \end{pmatrix} \right\}$$

It would be difficult to argue that any of the considered alternatives to the status quo would be 'irrelevant'. Both x , y and z are important since that is how the discussion started, u is important since it is a Paretian improvement on all these, and v of course is the winner.

The conclusion is that the Borda ranking is much more sensitive to the budget than Pref, and that Pref is much less sensitive. Pref, as calculated by `VoteToPref[]`, is protected against big surprises, since the order has been found by successively eliminating the winners of the subsets. It is not guaranteed that this will never cause a surprise, but such surprises will be much less frequent than with Borda. Such surprises will occur, when the budget changes such that new BordaFP items are included that start causing ties.

5.3 Pareto Pairwise

5.3.1 Using the count to break ties

Since pairwise voting is now applied to only Pareto points, we can be more relaxed about using the Count approach.

It may be that the concept of the Condorcet winner derives its appeal from mimicking Pareto optimality - while it need not be Pareto optimising, at least it wins all its duels. This view is a bit one-sided, since it does not explain why the margins should be neglected. Yet, now however all points already are Paretian, and there is no need for such mimicking anyhow. It may be considered more important now to count all votes, which means the margins by which items win their duels.

5.3.2 The classic Condorcet case

The Pareto-Pairwise scheme first selects all Pareto optimising points from the status quo and then submits these to Pairwise voting.

```
Condorcet[] ; ParetoPairwise[]

{StatusQuo → A, Pareto → {A}, Select → A}
```

```
ParetoPairwise[p:Preferences ,  
v:Votes , i:Items , sq:Automatic]
```

first selects the Pareto points,
and then applies the PairwiseMajority rule

ParetoPairwise may adjust Preferences and Items.

5.3.3 Another example

The Condorcet case is already an example how the Count can be used to break deadlocks. The following is another example. This gives a clear Pareto improvement from status quo *E* to *D*. But from *D* onwards, there is a cycle {*A*, *B*, *C*} as above. A move to any of these would be improving, but the Binary method has no way to determine which of these three to select. So the Count can be taken.

```
DefaultItems[]; EqualVotes[];
SetPreferences[2 + {{3, 2, 1, 0, -1}, {1, 3, 2, 0, -1}}];
Preferences
```

$$\begin{pmatrix} 5 & 4 & 3 & 2 & 1 \\ 3 & 5 & 4 & 2 & 1 \end{pmatrix}$$

```
ParetoPairwise["E"]
```

CheckVote::set : Items set to values at routine call

VoteMarginToPref::cyc : Cycle {B, A, B}

VoteMarginToBinary::dif : Selection E differs from Condorcet winning {A, B}

$$\left\{ \text{Pareto} \rightarrow \{A, B, C, D, E\}, \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & -1 & 0 & 1 & 1 \\ -1 & -1 & -1 & 0 & 1 \\ -1 & -1 & -1 & -1 & 0 \end{pmatrix} \right\},$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow E, \text{Sum} \rightarrow \{4, 4, 3, 1, 0\}, \text{Max} \rightarrow 4, \text{Condorcet winner} \rightarrow \{A, B\},$

$\text{Pref} \rightarrow \text{Pref}(E, D, \{A, B, C\}), \text{Find} \rightarrow \{A, B, C\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow E\},$

$N \rightarrow \{\text{Sum} \rightarrow \{2, 3, 1, -2, -4\}, \text{Pref} \rightarrow \text{Pref}(E, D, C, A, B), \text{Select} \rightarrow B\}, \text{All} \rightarrow B\}$

5.3.4 Random

When generating a random matrix, it is useful to set the first value - taken as the status quo - to a lower value, since otherwise the Pareto condition leaves little to choose from.

```
DefaultItems[]; EqualVotes[];
SetRandomPreferences[4, 6]; pr = SetFirstValue[2]
```

$$\begin{pmatrix} 2 & 5 & 4 & 3 & 1 & 6 \\ 2 & 6 & 4 & 3 & 5 & 1 \\ 2 & 4 & 1 & 6 & 3 & 5 \\ 2 & 3 & 1 & 4 & 5 & 6 \end{pmatrix}$$

ParetoPairwise[]

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::set : Items set to values at routine call

VoteMarginToPref::cyc : Cycle {D, B, D}

VoteMarginToBinary::dif : Selection A differs from Condorcet winning {B, D}

$\left\{ \text{Pareto} \rightarrow \{A, B, D\}, \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \right\},$

$1 \rightarrow \{\text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{0, 2, 2\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow \{B, D\},$
 $\text{Pref} \rightarrow \text{Pref}(A, \{B, D\}), \text{Find} \rightarrow \{B, D\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A\},$

$N \rightarrow \{\text{Sum} \rightarrow \{-2, 1, 1\}, \text{Pref} \rightarrow \text{Pref}(A, \{B, D\}), \text{Select} \rightarrow \{B, D\}, \text{All} \rightarrow \{B, D\}\}$

5.4 Pareto Plurality

The Pareto-Plurality scheme first selects all Pareto optimising points from the status quo and then submits these to Plurality voting.

```
EqualVotes[]; DefaultItems[];
SetRandomPreferences[4, 6]; pr = SetFirstValue[2]
```

$$\begin{pmatrix} 2 & 5 & 4 & 3 & 6 & 1 \\ 2 & 4 & 1 & 5 & 6 & 3 \\ 2 & 5 & 3 & 6 & 4 & 1 \\ 2 & 6 & 4 & 1 & 5 & 3 \end{pmatrix}$$

```
ParetoPlurality[]
```

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::set : Items set to values at routine call

$$\left\{ \text{Pareto} \rightarrow \{A, B, E\}, \text{Sum} \rightarrow \begin{pmatrix} B & \frac{1}{2} \\ E & \frac{1}{2} \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{1}{2} & B \\ \frac{1}{2} & E \end{pmatrix}, \text{Max} \rightarrow \left\{ \{B, E\}, \frac{1}{2} \right\}, \text{Select} \rightarrow \{\} \right\}$$

`ParetoPlurality[p:Preferences,
v:Votes, i:Items, sq:Automatic]`

first selects the Pareto points,
and then applies the plurality rule

- Note that this solution is also found by ParetoMajority[].

```
DefaultItems[]; ParetoMajority[]
```

CheckVote::adj : NumberOfItems adjusted to 6

CheckVote::adj : NumberOfItems adjusted to 3

BordaFP::set : Local set found: {B, E}

$$\left\{ \text{StatusQuo} \rightarrow A, \text{Pareto} \rightarrow \{A, B, E\}, \text{BordaFP} \rightarrow \{B, E\}, \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \right.$$

$$\left. N \rightarrow \{ \text{Sum} \rightarrow \{-2, 1, 1\}, \text{Pref} \rightarrow \text{Pref}(A, \{B, E\}), \text{Select} \rightarrow \{B, E\}, \text{Select} \rightarrow \{B, E\} \} \right\}$$

5.5 Pareto Approval

The Pareto-Approval scheme first selects all Pareto optimising points from the status quo and then submits these to Approval voting.

Approval[Borda] must be called if the Preferences are not binary. If we use a binary preference matrix then we can use the Borda routine directly. For a binary preference matrix, the Borda routine will select the items with a full column of 1's. These will also be Pareto improving. The difference between Borda and Pareto-Approval thus only arises if all items have at least one 0 somewhere.

- The random generator has been run till each column has at least one 0.

```
EqualVotes[4]; DefaultItems[6];
Preferences = Table[Random[Integer], {NumberOfVoters}, {NumberOfItems}]
```

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

```
b = Borda[]
```

```
{C, D, E}
```

```
p = Pareto[]
```

```
{A, C, E, F}
```

- The Pareto-Approval point is both Pareto and Borda.

```
Intersection[b, p]
```

```
{C, E}
```

- We can show this also in this way.

```
SelectPreferences[p];
```

CheckVote::adj : NumberOfItems adjusted to 4

BordaAnalysis[]

{Select → {C, E}, BordaFPQ → {True, True},

$$\text{WeightTotal} \rightarrow \left\{ \frac{9}{4}, \frac{11}{4}, \frac{11}{4}, \frac{9}{4} \right\}, \text{Position} \rightarrow \begin{pmatrix} 2 \\ 3 \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{9}{4} & A \\ \frac{9}{4} & F \\ \frac{11}{4} & C \\ \frac{11}{4} & E \end{pmatrix}$$

Above approach has been implemented in the routine.

```
ParetoApproval[p:Preferences, v:Votes, i:Items, sq:Automatic]
```

first selects the Pareto points and then applies BordaAnalysis. The p matrix must be 0 and 1 only – which differs from Approval

```
EqualVotes[4]; DefaultItems[6];
```

```
Preferences = Table[Random[Integer], {NumberOfVoters}, {NumberOfItems}]
```

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

ParetoApproval[]

CheckVote::adj : NumberOfItems adjusted to 3

CheckVote::set : Items set to values at routine call

{Pareto → {A, B, E}, Select → B, BordaFPQ → {True},

$$\text{WeightTotal} \rightarrow \left\{ \frac{15}{8}, \frac{9}{4}, \frac{15}{8} \right\}, \text{Position} \rightarrow (2), \text{Ordering} \rightarrow \begin{pmatrix} \frac{15}{8} & A \\ \frac{15}{8} & E \\ \frac{9}{4} & B \end{pmatrix}$$

6. Strategic voting

6.1 Introduction

Now that we are familiar with the basic voting schemes and some combinations of those, we can enter into a more fundamental discussion of why we would use such schemes in the first place. This then is the fundamental insight and definition: the basic problem and subject matter for Voting Theory is: to deal with the issues of comparability of utility and the problems of cheating about preferences.

The problem of comparability of utility and the problem of cheating actually are very much the same problem. Comparability of utility is not a sufficient condition to solve voting problems, since people could cheat. If people would not cheat, then we could ask whether their utilities are comparable, and if so, solve the issue by simply adding utilities (or have some Nash multiplication). If people are honest but utilities incomparable, then we should wonder why we would have a system of 'one person, one vote' anyway. Theories of altruism and sympathy suggest that utility is comparable to some level, and the main reason why we are hesitant to go further than 'one person, one vote' is that we take into account that people could cheat.

One angle to the problem is that voting could be used to determine the weights in cardinal aggregation. But there is quite a difference between a simple summation with (unitary) weights, and the case where a majority determines what the weights shall be for the minority. In the past it was rather the minority who determined the weight of the majority.

The best analytical position likely is to presume that there are some basic processes at the cardinal level, that use force and power, in which people compare their utility with those of others, and that, by evolution and social development, result into a system of justice, in which voting schemes are used as more democratic ways to settle issues. Voting schemes thus serve specific objectives, and, with the lack of objective ways to determine cardinal utility, their prime function is to balance fairness with the risks of cheating.

Hence, if we want to judge on the performance of the voting schemes, we should be clear about what they are used for, and it turns out that the focus is precisely that balance.

The steps of reasoning thus are:

1. We start with cardinal utility.
2. This does not work when there is no objective measure and when there is cheating.
3. NB. There can also be unwanted redistribution effects, when one group exploits another. There thus is a prime motivation to find an acceptable solution.
4. Hence the classical liberal position of Pareto.
5. Then there is a second stage, to choose from various Paretian points. Now there is cheating in the second stage.
6. For the second stage: Borda's scheme does not work. Pairwise Majority has some drawbacks. But Majority seems to work acceptably.
7. Consider the costs of decision making in general.

This Part of the book provides the details of this line of reasoning. Subsequently, Parts 7 and 8 complete the picture by relating voting with the theory of games, and showing that measuring cardinal utility is problematic. Parts 9 and 10 then conclude the matter, by showing that this approach also solves the problem that Arrow's Theorem created in the literature on voting.

6.2 Cardinal utility

Let us return to the basic example in chapter 4.2: there are two candidates for President while the status quo would be a vacancy. The voters have utility functions on some attributes, and we can determine the preference schemes. In chapter 4.2 we only used the ordinal preferences, but we also could assume that the utility functions are cardinal. The assumption of cardinality is that all utilities are perfectly compatible, comparable and addable, while they have a common natural zero point. When all voters have equal weight - which is a political decision apart from cardinality - then it suffices to add the votes (or Nash multiply them). Then the best selection is the candidate with the highest sum (product).

- To reproduce this in *Mathematica*, you may have to run that chapter again.

```
DefaultItems[3]

{A, B, C}

uts = Array[Utility, 3]


$$\begin{pmatrix} 10. & 61.5385 & 64.8649 \\ 50. & 46.1554 & 63.5389 \\ 56.2054 & 61.6911 & 37.8253 \end{pmatrix}$$

```

- Adding the votes:

```
total = Plus @@ uts

{116.205, 169.385, 166.229}
```

- And the cardinal winner is ...

```
Extract[Items, Position[total, Max[total]]]

{B}
```

If we had multiplied the utilities and selected the highest product, then this generally means that we maximise the minimal value in the product (though this need not always be the case).

In both cases, the status quo has no special position, and there is some redistribution of some kind. Redistribution hence is a subject that is linked with voting. (From some books on voting it might seem as if the subject can be treated without consideration of redistribution, but this is only valid under

specific assumptions - which assumptions basically are equivalent to explicitly disregarding redistribution.)

If the assumption of cardinality can be made, then its application is straightforward and justified. It would be wrong, in itself, to use another scheme.

If we would have used ParetoMajority instead:

```
SetOptions[ProperPrefsQ, N → Automatic];
```

```
EqualVotes[3]
```

$$\left\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right\}$$

```
Preferences = basicExample
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 2 & 3 & 1 \end{pmatrix}$$

```
Vote[]
```

```
{StatusQuo → A, Pareto → {A}, Select → A}
```

There is no uniform improvement on the status quo. The views on B and C differ too much.

When we compare the cardinal approach and the ordinal approach, then it appears that, if we would attach great value to the status quo, then we still might design a procedure that *first* selects all Paretian points, and *only then* applies a cardinal scheme.

This approach however loses its appeal, once it is realised that cardinality loses out anyhow because of the possibility of cheating. Ordinality destroys information on the intensities on the preferences - this is the price that we pay for the ordinal approach. The only reason why we are willing to pay that price is that we do not have an objective measure for cardinality - which means that there is the possibility of cheating.

6.3 Cheating

6.3.1 Possibility

One theoretical position is to take votes at their face value: if people vote in a certain way, apparently these are their preferences. But this runs counter to another economic axiom: in some cases it could be rational to cheat.

A nice word for cheating would be strategic voting. But cheating and deceit are clearer terms.

There would be proper strategy if we *allow* people to use such strategies. In some respects, it is not bad in itself to instruct people to make the best of their limited voting power. In this framework, strategic behaviour would re-introduce some elements of veto-power that the scheme otherwise would deny. But normally cheating is not allowed.

6.3.2 Cheating with the intensity

The problem with cardinal utility is that voters may misstate the intensity of their preference.

Consider the basic example of chapter 4.2. For example, voter 2 notes that his choice C is not selected, and then he may increase the intensity of his preferences.

```
Voter[2][{ns_, ec_}] = CES[3, {.7, .3}, {ns, ec}, .7]
```

$$\frac{3}{\left(\frac{0.7}{ns^{0.428571}} + \frac{0.3}{ec^{0.428571}} \right)^{2.33333}}$$

```
total2 = Plus @@ Array[Utility, 3]
```

```
{216.205, 261.696, 293.307}
```

```
Extract[Items, Position[total2, Max[total2]]]
```

```
{C}
```

We can also use the voting routines to show the possibility of cheating with cardinal utility.

- Set the N option of ProperPrefsQ to Infinity.

SetOptions[ProperPrefsQ, N → Infinity]

$\{N \rightarrow \infty\}$

For example, let the status quo be *B*, let two people favour a change to *C*, and let one person favour a change to *A* - and give this person the possibility to wildly exaggerate her preference for *A*.

EqualVotes[]; StatusQuo[] = "B";

{{1, 2, 3}, {1, 2, 3}, {1000, 2, 1}} // SetPreferences

ProperPrefsQ::row : Proper Preference matrix row sums {6, 1003} should better all equal 6

$\{ \text{Number of Voters} \rightarrow 3, \text{Number of items} \rightarrow 3, \}$

Votes are nonnegative and add up to 1 → True, Preferences fit the numbers of Voters and Items → True,
Type of scale → Ordinal, Preferences give a proper ordering → False,

Preferences add up to → {6, 1003}, Items → {A, B, C}, Votes → $\left\{ \frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right\}$

- If we would use the Borda count, then *A* would be chosen - perhaps due to cheating.

Borda[]

A

- Pairwise voting filters intensity out - useful if voter 3 had been cheating.

PairwiseMajority[]

$\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & -\frac{1}{3} & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & 0 \end{pmatrix} \right), \}$

1 → {StatusQuo → B, Sum → {0, 1, 2}, Max → 2, Condorcet winner → C,
Pref → Pref(A, B, C), Find → C, LastCycleTest → False, Select → C},

$N \rightarrow \left\{ \text{Sum} \rightarrow \left\{ -\frac{2}{3}, 0, \frac{2}{3} \right\}, \text{Pref} \rightarrow \text{Pref(A, B, C)}, \text{Select} \rightarrow C \right\}, \text{All} \rightarrow C \}$

- Of course, moving to *C* is not Pareto. So a classical liberal would prefer *B*.

ParetoMajority[]

{StatusQuo → B, Pareto → {B}, Select → B}

The above thus shows that all solutions - *A*, *B* and *C* - have something to say for them.

- Make sure that we use ordinality again.

```
SetOptions[ProperPrefsQ, N → Automatic]

{N → Automatic}
```

6.3.3 Cheating by order

Limiting preference expressions to ordinality helps to limit the effects of cheating on intensity, but it also changes the form of cheating. People would still be free to re-arrange the preference order, to achieve a better result.

- Assume a Borda scheme.

```
EqualVotes[]; DefaultItems[];
pr = {{1, 2, 3}, {3, 2, 1}, {1, 2, 3}};
pr // SetPreferences;

Borda[]

C

StrategicPref[Borda, 2]

StrategicPref::str : Iter 2: A strategic vote will give item B in the solution

{Borda → {C}, (2 3 1) → (B C)}
```

If all ties would be settled by flipping a coin, then the probability that *B* is selected has been increased from 0 to 50%.

```
StrategicPref[c, n_Integer, pp:Preferences]
```

looks for schemes of deceitful voting under constitution *c*.
For *c* = BordaFP, Borda, PairwiseMajority

Uses default Votes and Items, for ordinal preferences only. Note that this is a limited routine: it assumes that the other voters don't cheat ...

6.3.4 Dealing with cheating

Sometimes there are ways to deal with cheating. Consider an example from another area. Division of a cake may cause people to cheat. A solution is, when dividing a cake between two persons, to let one person make the cut, and to give the other person first choice. This is not fail-safe, since one would always prefer the other to make the cut - since cuts never are perfect. So flip a coin first. If there still is lack of trust, both persons could flip a coin, where neither can influence the flip of the other - and two equal results (HH, TT) make one the cutter and two different (HT, TH) make the other the cutter. And there are other devices, until a level of trust has been reached.

The different voting methods have been created because they deal differently with cheating. The classical liberal position, expressed in ParetoMajority, then is a way to deal with cheating in voting.

Lately, there have been proposals for 'declared strategy voting' (DSV). Voters would submit their strategy, rather than a simple vote. Of course, voters could also submit their true preference ordering, if they can assume safely that a programme would maximise their utility. Such schemes are complex, since all voters are voting strategically at the same time.

In the mean time, we better try to understand why Pareto Majority has been seen as a way to deal with cheating.

6.4 Pareto Majority

6.4.1 The Pareto criterion

Cheating is a strong argument to reject decision making based on cardinal utility.

Classical liberals solved the issue by giving each individual the right to veto a change from the status quo - the Pareto rule. By consequence, a change can only take place when someone improves and nobody suffers. If there are more possible Pareto improvements on the status quo, then some system of majority voting can be used to select from these, such as Borda or PairwiseMajority. It can be argued that 'majority voting' actually derives its moral standing from the (hidden) assumption that only Pareto improving options are on ballot.

The classical liberal position obviously deals with cheating with some success. Suppose that B would be an improvement to all, but voter 1 could live with B but would actually prefer C on top of that. Voter 1 could veto B , until others are willing to accept C as well. However, if voter 1 misrepresents his veto, then the status quo endures, and he thus shoots himself in the foot.

Note, in this example, that voter 1, by blocking a proposal, may always have some deeper reasons. Recall the example is that everyone improves by \$1 but the King by \$1 million. Voter 1 thus seems to improve. But voter 1 may think that there is a relative deterioration. We already concluded that such a position need not be irrational - and it neither might be cheating.

Note also that the Paretian approach is not necessarily conservative. People are altruistic to some degree, and thus might be tolerant to a (relative) deterioration for themselves. Note, that this argument might also be turned around, in that majority voting might be defended, saying that people are altruistic to some degree, so that majority voting does not have to result into exploitation. It might well be a matter of personal opinion what has the greatest risk.

Note also that the information requirements for *absolute* Paretian improvements are very limited. We just need veto (1, 0) information. If we want to know about *relative* Paretian improvements, then we need all ordinal data.

Note, finally, that the Pareto criterion not only helps for cheating on existing items, but it may also help for cheating on the budget itself - since inclusion or exclusion of an item can affect the decision. Suppose that there is no veto allowed. Let (1: $A > B$), (2: $B > A$) and (3: $A > B$), so that there is a two thirds majority for A . Suppose that B is the status quo, so that person 2 experiences a deterioration. But we assumed that 2 cannot veto this. Let person 2 see the light, and look for an item C . Such a proposal could e.g. be that voter 1 should give \$1 million to voter 3 - clearly a proposal that 1 would reject but that 3 would enjoy - but voter 1 would not be able to veto it. Then we get (1 & 2: $B > C$) while 1 disagrees with 3 since (3: $C > A$). With C added, and 2 possibly cheating with (2: $C > A$), there would arise the Condorcet case (1: $A > B > C$), (2: $B > C > A$) and (3: $C > A > B$) with indecision or indifference. If we assume that no decision is taken in case of a cycle, then the introduction of C would be equivalent to a veto. It is more economical to directly grant veto rights on the status quo, since it saves time on the discussion about such C 's. (Though it can be a good exercise to try to think up such C 's.)

6.4.2 Pareto and costs

It has been argued that Paretian veto power comes with large costs. A discussion of costs is by Buchanan & Tullock, "The calculus of consent", Michigan 1962. Methods like Borda and Pairwise Majority thus are often seen as schemes that have been proposed as alternative to veto power, in order to reduce the costs of collective decision making.

Historically, it can be doubted whether the schemes by Borda and Pairwise Majority were based solely on cost considerations. A classical liberal would rather hold that first a selection is made of all Pareto improving points from the status quo, and only then the schemes of Borda or Pairwise Majority are applied to choose the best from these improvements. This has a different motivation than costs. From this point of view it is only logical that the schemes of Borda and Pairwise Majority do not respect the veto power. They have entered a the debate for a different reason than costs.

Similarly, the classical liberal will reject these schemes for non-Pareto points when the argument would only be costs. These schemes would only be acceptable, for non-Pareto points, if they save so many costs that compensation payments can be made to those people who lose out. It then is difficult to understand the cost argument. Proponents for a change from the status quo

could use compensating payments to opponents. The discussion about the size of the compensation would require time, but, in that case it would be better to design a rule for time management rather than abolish veto powers overall.

The classical liberal also wants a real payment of compensation. Kaldor and Hicks have advanced the notion that sometimes payments need not be paid out, but that it suffices that the possibility of payment is shown in theory. Unfortunately, the literature calls this the Neo-Paretian criterion instead of the Kaldor-Hicks criterion. It violates the Pareto criterion, so it is strange that the Kaldor-Hicks criterion should be named after Pareto. The Kaldor-Hicks criterion seems relevant, if 100 million tax payers each could receive a penny, but the government does not actually pay this since this would be too costly. It *seems* relevant, but that does not mean that necessarily *is* relevant. A classical liberal would hold on to the idea that a proposal should also pay for the costs of actually paying its compensations. One practical solution is to lump all payments together, e.g. in the annual budget.

6.4.3 Pairwise cheating in the second stage

Let us presume that the Pareto criterion is used for the first stage. Then we need to decide how to proceed with the Paretian points in the second stage. It would be nice if we could stay close to the Paretian principle. The Pareto rule indeed has some key properties:

1. The Pareto points can be established by pairwise comparison of each point with the status quo.
2. There is some hierarchy. If item *B* loses from the status quo, it drops out.
3. In those comparisons there is 'one person, one vote'.

Continued use of this philosophy for the second stage seems rational. We wonder how this would look like.

(Ad 1) Using pairwise comparisons could be a psychological method to induce the voters to focus their attention to only the pair under consideration. People might be induced to disregard the impact on the other items.

However, it is more common in economics to assume that people are rational, and to design mechanisms that enhance rationality.

(Ad 2) Should we vote on all pairs, or is it sufficient to take a hierarchy ? This now is a question on efficiency, not on cheating. For example, if there are 4 candidates, then there are a Binomial(4, 2) = 6 different pairs, but a hierarchy of 3 seems enough if we could match the winner of $\{A, B\}$ with the winner of $\{C, D\}$.

It turns out that this issue of hierarchy is based on a confusion. A hierarchy only applies to identifying the Pareto optimising candidates, but cannot be used for the final comparison of those Pareto winners. Sometimes we are able to pair up $\{A, B\}$ such that we can predict that the loser will certainly lose also from all others. What we are trying to do then is to identify the Pareto winners. It would be a confusion to think that we could use this scheme for a winning hierarchy. Thus pairwise comparisons can only be argued for with the argument of cheating, and not on hierarchy. If we use the binary method, then cycles are possible. Thus we are forced to vote on all possible pairs - in order to detect the cycle.

(Ad 3) The issue of 'one person, one vote' for Pareto is related to the issue of using the Pairwise Binary method rather than the Pairwise Margin Count method. Having 'one person, one vote' already limits the possibility to abuse the intensity, but there is more to it. The Binary method limits the impact of organised group strategy. When a vote on $\{A, B\}$ gives the voting scores $\{v, w\}$ - for example $\{60\%, 40\%\}$ with the margins $\{v-w, w-v\} = (0.2, -0.2)$ - then accounting a win as 1 and a loss as 0 (the binary method) reduces the impact of the full scores. The binary method derives partly from being able to make simple counts so that the method is transparant to everyone. But the more basic reason is that *parties* (co-ordinated groups rather than single individuals) might misstate a preference on one candidate to favour their real preference.

The literature - see Mueller (1989:395) who refers to Gibbard 1973 and Satterthwaite 1975 - suggests that pairwise comparisons, and apparently using the binary method, would be strategy proof, though under certain conditions. There are obvious limits to strategy-proofness: (1) We have seen above that some cheating on including or excluding items in the budget still could be possible. (2) And the binary method can have cycles - which is why we would want to use the margin count method to break the tie, which however gives more room for cheating. (The voting literature is too negligent on these two topics.)

Consider the following example. The binary count gives Condorcet winners A and C , though B should be included because of the cycle. To break the tie, the margin count gives C . However, voter 4 prefers A . By voting strategically, the margin count evaporates, and A , B and C form a full tie. If dice are used to settle the deadlock, the probability of A has risen from 0 to 33%. (Of course 4 then also runs the risk that B is chosen, which has lowest rank. But perhaps the difference in intensity between B and C is small while the preference for A could be large.)

```
EqualVotes[]; DefaultItems[];
pr = {{1, 2, 3}, {3, 2, 1}, {1, 2, 3}, {3, 1, 2}};
pr // SetPreferences;
```

- StrategicPref advises that 4 votes according to {3, 2, 1}.

```
StrategicPref[PairwiseMajority, 4]
```

VoteMarginToPref::cyc : Cycle {B, A, B}

VoteMarginToBinary::dif : Selection A differs from Condorcet winning {A, C}

StrategicPref::str : Iter 1: A strategic vote will give item A in the solution

$$\left\{ \text{PairwiseMajority} \rightarrow \left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{pmatrix}, \right. \right.$$

$$1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 1, 2\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow \{A, C\},$$

$$\text{Pref} \rightarrow \text{Pref}(\{A, B, C\}), \text{Find} \rightarrow \{A, B, C\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A\},$$

$$N \rightarrow \left\{ \text{Sum} \rightarrow \left\{ 0, -\frac{1}{2}, \frac{1}{2} \right\}, \text{Pref} \rightarrow \text{Pref}(B, A, C), \text{Select} \rightarrow C\right\}, \text{All} \rightarrow C\}, (3 \ 2 \ 1) \rightarrow (A \ B \ C) \}$$

- If the count is used, and 4 votes strategically according to above scheme, then A enters the choice.

```
PairwiseMajority[{{1, 2, 3}, {3, 2, 1}, {1, 2, 3}, {3, 2, 1}}]
```

VoteMarginToPref::cyc : Cycle {B, A, B}

VoteMarginToBinary::dif : Selection A differs from Condorcet winning {A, B, C}

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \right.$$

$$1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 2, 2\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow \{A, B, C\},$$

$$\text{Pref} \rightarrow \text{Pref}(\{A, B, C\}), \text{Find} \rightarrow \{A, B, C\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A\},$$

$$N \rightarrow \{ \text{Sum} \rightarrow \{0, 0, 0\}, \text{Pref} \rightarrow \text{Pref}(\{A, B, C\}), \text{Select} \rightarrow \{A, B, C\}\}, \text{All} \rightarrow \{A, B, C\} \}$$

All in all, trying to extend the Pareto philosophy to the second stage appears to have only a limited effect. These methods only work to the extent that people are not as rational as economic theory assumes that they are. It is good to remember that honesty in voting probably also has other sources than only these voting methods. (Yet it is important to see why these schemes were proposed: namely to deal with cheating. So we should judge their effectiveness in terms of this objective.)

6.4.4 Comparing Condorcet and BordaFP

It is useful to compare pairwise voting ('Condorcet') with BordaFP on their sensitivity to cheating. The current implementation of the routine StrategicPref uses the Margin Count, and that of course allows more room for cheating.

6.4.4.1 When there is a fixed point winner

When there is a Condorcet winner, StrategicPref can be used with BordaFP, to show that cheating can be prevented. Since the StrategicPref implementation for PairwiseMajority still uses the Margin Count, the call with PairwiseMajority still shows room for cheating. Thus there is a difference, but this is also caused by the implementation.

- Let us consider a Majority Plurality winner, that then is a Condorcet winner too.

```
Votes = { .26, .26, .48 };
```

```
SetPreferences[{ {3, 1, 2}, {3, 1, 2}, {1, 3, 2} }];
```

- The StrategicPref implementation for PairwiseMajority assumes the Margin Count method. In that case voter 3 can achieve an improvement from A to C.

```
StrategicPref[PairwiseMajority, 3]
```

StrategicPref::str : Iter 2: A strategic vote will give item C in the solution

{PairwiseMajority →

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & 0.04 & 0.04 \\ -0.04 & 0 & -0.04 \\ -0.04 & 0.04 & 0 \end{pmatrix}, 1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 0, 1\}, \text{Max} \rightarrow 2, \right.$$

Condorcet winner → A, Pref → Pref(B, C, A), Find → A, LastCycleTest → False, Select → A),
 $N \rightarrow \{ \text{Sum} \rightarrow \{0.08, -0.08, 0.\}, \text{Pref} \rightarrow \text{Pref}(B, C, A), \text{Select} \rightarrow A \}, (1 \ 2 \ 3) \rightarrow (C) \}$

- BordaFP always selects A , and whatever voter 3 does, it is irrelevant. In a sense, voter 3 is wholly irrelevant.

StrategicPref[BordaFP, 3]

BordaFP::chg : Borda gave {C}, Fixed Point is {A}

StrategicPref::non : Strategy useless, iter 3: item A is the best result, also honestly

{BordaFP → {A}, Out → {1, 3, 2}}

(* Results[StrategicPref, BordaFP] *)

6.4.4.2 When there is a tie

BordaFP and PairwiseMajority react basically the same when there is no Condorcet winner, and when we allow for tie-breaking rules. We can usefully consider the Condorcet case as an example.

Condorcet[] ; Preferences

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

- Since there is no Condorcet winner, the binary method halts. If we use the margin count, then 1 can avoid a less preferred A and cause a more preferred B .

StrategicPref[PairwiseMajority, 1]

VoteMarginToPref::cyc : Cycle {C, A, B, C}

StrategicPref::str : Iter 2: A strategic vote will give item B in the solution

$$\left\{ \text{PairwiseMajority} \rightarrow \left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \left(\begin{pmatrix} 0 & -0.2 & 0.5 \\ 0.2 & 0 & -0.3 \\ -0.5 & 0.3 & 0 \end{pmatrix} \right) \right\} \right.$$

$1 \rightarrow \{ \text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{1, 1, 1\}, \text{Max} \rightarrow 1, \text{No Condorcet winner} \rightarrow \{A, B, C\},$

$\text{Pref} \rightarrow \text{Pref}(\{A, B, C\}), \text{Find} \rightarrow \{A, B, C\}, \text{LastCycleTest} \rightarrow \text{True}, \text{Select} \rightarrow A \},$

$N \rightarrow \{ \text{Sum} \rightarrow \{0.3, -0.1, -0.2\}, \text{Pref} \rightarrow \text{Pref}(C, B, A), \text{Select} \rightarrow A \}, \text{All} \rightarrow A \}, (1 \ 3 \ 2) \rightarrow (B) \}$

- The conclusion is the same for BordaFP. Apparently, the items are fixed points, and cheating then has some effect.

StrategicPref[BordaFP, 1]

$$\left\{ \text{BordaFP} \rightarrow \{A\}, \begin{pmatrix} 1 & 3 & 2 \\ 2 & 3 & 1 \end{pmatrix} \rightarrow \{B, B\} \right\}$$

6.4.5 A choice on principle

If we consider the various arguments, then it appears that cheating cannot be fully eliminated. We want the decision to be based on preferences, and then we also adopt tie-breakers based on preferences: and thus we simply cannot avoid some influence of cheating. In the face of inevitable defeat, we should accept that defeat, rather than trying to act as if a solution still would be possible. Given that cardinal measures cannot be observed objectively, it is an improvement to accept ordinality, but this improvement works only to some extent.

A choice can be made on the base of principle. When we start out with assigning votes to people, then we should have a good reason if we would deviate from that idea. To remain consistent, the votes should also count in the second phase. The added advantage of the Count and / or BordaFP is that there are no 'cycles' anymore. The argument that parties could co-ordinate cheating, and would be deterred by the Binary method, is less convincing. Even if the Binary method would not allow cheating, under well specified conditions, there still could be cheating on the selection of the candidates and the choice of the budget - and the method is sensitive to the budget. It would often be more relevant that the final result better reflects the votes actually cast. If cheating occurs, this fact could perhaps better be tabled on the particular issues and evidence of the day. Hence, a principled choice would be to emphasise sensitivity to preferences, and look for other ways to tackle cheating.

6.4.6 A choice on balancing properties

We could also take a vote on which voting scheme to use.

The voting schemes can be scored on their degree of sensitivity to cheating.

Less sensitive to intensity \Leftrightarrow Less room for cheating
Binary Pairwise
BordaFP
Count Pairwise
Ordinal Borda
Interval/Ratio Borda
Cardinal
More sensitive to intensity \Leftrightarrow More room for cheating

If the probability of cheating is not a function of the sensitivity only, but is also affected by other variables, then the two criteria of sensitivity to intensity and sensitivity to cheating are partly independent, and a choice can be made. The group as a whole might converge on an optimum rule. Let voters 1 and 2 assign scores for Sensitivity and Cheating - where a high score for Cheating means that it is appreciated that there is less room for cheating - and let them simply add them (in cardinal fashion).

- These number are arbitrary. Your group would have your own numbers.
(This group appears to be cheating-averse.)

```
SetOptions[ProperPrefsQ, N → Infinity];

Items = {binary, bordafp, count, ordinal, interval, cardinal};

Voter[1] = {10, 10, 20, 50, 100, 100} + {1000, 1000, 900, 100, 50, 1};
Voter[2] = {1, 1, 4, 50, 50, 200} + {1000, 800, 500, 200, 50, 10};

Array[Voter, 2] // SetPreferences;

Preferences
( 1010 1010 920 150 150 101 )
( 1001 801 504 250 100 210 )
```

- If binary is the status quo, this group would stick there.

```
StatusQuo[] = binary;

Vote[]

{StatusQuo → binary, Pareto → {binary}, Select → binary}
```

- If cardinal was the status quo, then the binary method would be chosen.

```
StatusQuo[] = cardinal;

total3 = Plus @@ Preferences

{2011, 1811, 1424, 400, 250, 311}

Extract[Items, Position[total3, Max[total3]]]

{binary}
```

- We could advise this group to use ParetoMajority. Cheating averse as they are, they still choose the binary method.

```
SetOptions[ProperPrefsQ, N → Automatic];
```

PrefToList /@ (ListToPref /@ Preferences)

$$\begin{pmatrix} \frac{11}{2} & \frac{11}{2} & 4 & \frac{5}{2} & \frac{5}{2} & 1 \\ 6 & 5 & 4 & 3 & 1 & 2 \end{pmatrix}$$

Vote[]

CheckVote::adj : NumberOfItems adjusted to 5

{StatusQuo → cardinal, Pareto → {binary, bordafp, cardinal, count, ordinal}, Select → binary}

6.5 Excursion to equity

6.5.1 Introduction

Our decision on the adoption or rejection of some voting rule generally depends on our ideas how it would affect our lives in practice. Voting, as considered just by itself, is a rather empty subject. We should not neglect what the voting is about. This insight is sufficiently important to justify a short excursion to the problem of equity.

Above, we already noted that voting necessarily depends upon notions of redistribution. When books and other expositions on voting manage to neglect the issue, then it is only by choice, but not necessarily a wise choice.

The Pareto principle does not derive from the topic of voting on itself, but derives rather from another realm of discussion. The reason why the classical liberals were so in favour of Pareto's principle, is that it gives a person the right to veto any perceived violation of his or her well-being.

(The classic example is the issue of taxation. To be sure: the issues are subtle here, and not all taxes are in violation with basic rights. For example, when a legal system for taxes has been created such that it is not known, *ex ante*, who will be taxed and who will benefit, then people can adapt their behaviour, and if they accept the legal framework, then they also accept the implied taxation.)

The point remains that the majority rule, unchecked by the Pareto precondition, would give any majority the possibility to terrorise any minority. There could be shifting majorities, and this shifting might provide another check on exploitation, but positions would likely become entrenched, and society as a whole would not show much respect for the common individual. As James Madison emphasised, democracy is not quite the majority rule, but rather the respect for minority rights.

It is useful to shortly consider these issues of equity here, since they emphasise the importance of the issues of voting, and since they place them in the context of the wider economic problem.

In dividing a cake, we already see that people can become jealous or that some divisions are inefficient. We can identify a simple solution that is efficient and

that prevents jealousy, which gives the `BalancedPareto` routine. However, this solution compares *levels* of positions. People often compare *relative* positions, taking some historical point of reference. If you favour all children in the family except one, so that the situation of it remains the same, say for ten years, that child will experience feelings of relative deterioration. In other cases, some minimal income is required for survival, and more serious ethical questions arise. Note that the `BalancedPareto` routine chooses for people. Obviously, more division rules are possible, and it is actually the group itself that has to decide what rule to select.

6.5.2 Dividing a cake fairly

The idea is to fairly divide a cake of size 1. Let $W = \{w_1, \dots, w_n\}$ be the wants or claims of n persons. These wants are limited, so that $0 \leq w_i < \infty$. Then $G = \{g_1, \dots, g_n\}$ will be what they get.

- If the sum $S = (w_1 + \dots + w_n) \leq 1$, then we can give everyone what he or she wants.
- For $S > 1$, some considerations are:
 1. Equal division - giving $1 / n$ to everyone - need not be Pareto Optimal (PO). A solution is PO if any individual improvement would be at the cost of someone else. If a person gets more than he or she wants, then the allocation is not PO, and a reallocation makes someone else better off.
 2. A proportional allocation W / S can cause jealousy. A person can become jealous if he does not get what he wants and if another person has more. An allocation is called 'balanced' if nobody is jealous.
 3. The `BalancedPareto` algorithm satisfies Pareto-optimality and non-jealousy. (It will be a good exercise if you try to find this algorithm yourself.)

We can tackle some of these issues with this package.

`Economics[Economic`Fairness]`

Cool`Economic`Fairness`

[BalancedPareto](#)
 [Jealous](#)
 [RandomGet](#)
 [RelativeJealous](#)
[ExplainFairness](#)
[Proportional](#)
[RandomWant](#)
[SatisfiedQ](#)

6.5.3 Absolute levels

You may check that this group wants more cake than the cake provides.

$$w = \left\{ \frac{1}{10}, \frac{2}{5}, \frac{1}{2}, \frac{1}{10}, \frac{1}{5} \right\};$$

Suppose that we give everyone a part of the cake that is in proportion to his or her claim. We add up all claims, find S and give everyone W / S . This is the "proportional share rule".

```
pr = Proportional[%]
```

$$\left\{ \frac{1}{13}, \frac{4}{13}, \frac{5}{13}, \frac{1}{13}, \frac{2}{13} \right\}$$

<code>RandomWant[n_Integer:5]</code>	generates a list of n random wants, default n = 5. If the sum is less than or equal to 1, a message is put out
<code>RandomGet[n_Integer:5]</code>	generates an allocation for n persons, default n = 5. The sum is 1
<code>Proportional[wants_List]</code>	gives wants / Add[wants]

The "proportional share rule" however can cause jealousy. See below the Jealousy complex. Note that the diagonal in the jealousy matrix is always False, since nobody is jealous on himself or herself. The Position key gives the positions of True in the jealousy matrix. The smallest proportion is claimed by the first person, 1/10, which is much smaller than the equal share 1/5. If the Min person does not get sufficient, then he should be jealous on all others (row of True's). In this case, the Position key shows that 1 is jealous on everyone indeed, except for the person who gets as much. The largest proportion claimed is by the third person, almost 1/2, which is clearly bigger than a equal share 1/5. The Max person should not be jealous on anyone (even when he or she does not get enough). Indeed, 3 does not occur to the left in the Position key - but it occurs on the right, since everybody is jealous on 3 (who got most since he or she claimed most).

j = Jealous[w, pr]

$\left\{ \text{Max} \rightarrow \frac{5}{13}, \text{Position}[\text{Max}] \rightarrow (3), \text{Min} \rightarrow \frac{1}{13}, \text{Position}[\text{Min}] \rightarrow \left(\frac{1}{4} \right), \right.$

$\left. \text{Jealous} \rightarrow \begin{pmatrix} \text{False} & \text{True} & \text{True} & \text{False} & \text{True} \\ \text{False} & \text{False} & \text{True} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{True} & \text{True} & \text{False} & \text{True} \\ \text{False} & \text{True} & \text{True} & \text{False} & \text{False} \end{pmatrix}, \text{Position} \rightarrow \begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 5 \\ 2 & 3 \\ 4 & 2 \\ 4 & 3 \\ 4 & 5 \\ 5 & 2 \\ 5 & 3 \end{pmatrix} \right\}$

Jealous[want, get, otherget]

declares a person jealous,
if he does not get what he wants ($\text{get} < \text{want}$),
and if another person gets
more than what he gets ($\text{otherget} > \text{get}$)

Jealous[wants_List, get_List]

gives the matrix of occurrences of jealousy for
an allocation. Matrix[i, j] is True if i is jealous on j

- The following is an allocation of the cake that is both balanced and PO.

bp = BalancedPareto[w]

$\left\{ \frac{1}{10}, \frac{3}{10}, \frac{3}{10}, \frac{1}{10}, \frac{1}{5} \right\}$

- This allocation does not mean that everybody is satisfied. Only the modest claimants will be happy.

SatisfiedQ[w, bp]

{True, False, False, True, True}

- We can check that nobody is jealous.

j = Jealous[w, bp]

$\left\{ \text{Max} \rightarrow \frac{3}{10}, \text{Position}[\text{Max}] \rightarrow \left(\frac{2}{3} \right), \text{Min} \rightarrow \frac{1}{10}, \right.$

$\left. \text{Position}[\text{Min}] \rightarrow \left(\frac{1}{4} \right), \text{Jealous} \rightarrow \begin{pmatrix} \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \end{pmatrix}, \text{Position} \rightarrow \{\} \right\}$

BalancedPareto[wants_List]	finds the allocation that is balanced (no jealousy) and Pareto optimising (PO)
SatisfiedQ[want, get]	gives True if $get \geq want$. The remainder is not consumed – we assume free disposal.
SatisfiedQ[w_List, g_List]	for lists

The algorithm is: with n persons, first allocate all who want less than $1 / n$; then allocate recursively for the remainder; and allocate $remainder / m$ for the final m remaining persons (so that they will not be jealous on each other).

6.5.4 Relative positions

When $g_i / w_i < 1$, then there is jealousy for relative positions when $g_i / w_i < g_j / w_j$. This has most meaning when the wants are determined by what people got in the former period, $W = f(G[-1])$. With balanced growth, all wants grow as fast. Alternatively, redistribution gives some winners and some losers, but the losers might not blame each other if they lose by the same proportion. Note that the winners might worry if they don't grow as much as another winner.

RelativeJealous[want_List, get_List]	calls Jealous[1, get / want]
--------------------------------------	------------------------------

- This generates some random data.

```
gold = RandomGet[]
```

$$\left\{ \frac{98}{247}, \frac{47}{247}, \frac{41}{247}, \frac{32}{247}, \frac{29}{247} \right\}$$

```
gnew = RandomGet[]
```

$$\left\{ \frac{11}{105}, \frac{1}{35}, \frac{3}{7}, \frac{3}{70}, \frac{83}{210} \right\}$$

- These are the relative changes, $r > 1$ an increase, $r < 1$ a decrease.

```
gnew / gold // N
```

```
{0.264043, 0.150152, 2.58188, 0.330804, 3.36634}
```

RelativeJealous[gold, gnew]

$$\left\{ \text{Max} \rightarrow \frac{20501}{6090}, \text{Position}[\text{Max}] \rightarrow (5), \text{Min} \rightarrow \frac{247}{1645}, \text{Position}[\text{Min}] \rightarrow (2), \right.$$

$$\left. \text{Jealous} \rightarrow \begin{pmatrix} \text{False} & \text{False} & \text{True} & \text{True} & \text{True} \\ \text{True} & \text{False} & \text{True} & \text{True} & \text{True} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \\ \text{False} & \text{False} & \text{True} & \text{False} & \text{True} \\ \text{False} & \text{False} & \text{False} & \text{False} & \text{False} \end{pmatrix}, \text{Position} \rightarrow \begin{pmatrix} 1 & 3 \\ 1 & 4 \\ 1 & 5 \\ 2 & 1 \\ 2 & 3 \\ 2 & 4 \\ 2 & 5 \\ 4 & 3 \\ 4 & 5 \end{pmatrix} \right\}$$

I have not implemented a rule that takes account of relative positions. Obviously, if everyone wants to grow as much, then the shares should remain the same, and thus there is only one satisfactory distribution. One justification for different shares could arise from the contribution to growth itself, but then we leave the realm of this simple excursion.

6.5.5 Subsistence

If all require a minimum level g_{\min} , and if $g_{\min} \leq w_i \leq 1/n$, then the BalancedPareto rule will work.

BalancedPareto has only a problem if $g_{\min} > 1/n$. Then someone has to die if the others want to survive. There are no clear rules here. Of course, there is a difference between a static and a dynamic framework. In a static framework, one might be tempted to eliminate the 'neediest'. The effect for $1/n < g_{\min} < 1/(n-1)$ however is independent of need. Considering need would be relevant if the minimum depends upon the person. Then there might be a rule that if your elimination does not help me to get my minimum, but if my elimination helps you to get your minimum, then I perhaps better go. But in a dynamic framework, a very needy person might as well concern a child that is important for future production.

w = RandomWant[]

$$\left\{ \frac{57}{250}, \frac{39}{125}, 0, \frac{39}{125}, \frac{44}{125} \right\}$$

Average income is $1/n$, and a minimum can be taken at $1/3$ of the average.

$$w2 = \text{Max}[1/3 \ 1/\text{Length}[w], \ #] \& \ /@ \ w$$

$$\left\{ \frac{57}{250}, \frac{39}{125}, \frac{1}{15}, \frac{39}{125}, \frac{44}{125} \right\}$$

$$bp2 = \text{BalancedPareto}[w2]$$

$$\left\{ \frac{57}{250}, \frac{529}{2250}, \frac{1}{15}, \frac{529}{2250}, \frac{529}{2250} \right\}$$

If the minimum is $1/n$:

$$w3 = \text{Max}[1/\text{Length}[w], \ #] \& \ /@ \ w$$

$$\left\{ \frac{57}{250}, \frac{39}{125}, \frac{1}{5}, \frac{39}{125}, \frac{44}{125} \right\}$$

$$bp3 = \text{BalancedPareto}[w3]$$

$$\left\{ \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5} \right\}$$

6.6 Conclusion

We have highlighted the classical liberal position. This limits the discussion to the Pareto items, and only then applies majority rules. The approach is rational since it prevents cheating on the selection of the Pareto points.

We have compared this classical approach with the 'plain' schemes of Borda and Pairwise Majority. The plain application of these schemes violate the condition of Pareto optimality. This is no surprise, given that they deviate from this assumption. But they also allow cheating.

How about cheating in the second round ? In itself, cheating is discouraged, since the data for the second step are also used for the first step. For example, if everyone else favours B and voter 1 can live with B but prefers C , so that in reality $status\ quo < B < C$, then voter 1 shoots himself in the foot by voting $B < status\ quo < C$ and thus by not revealing the true preference order. Thus there are some incentives for honesty - and the second round can take the advantage of that. But examples where cheating can work can also be imagined, especially when we allow for non-ordinality. Even pairwise majority, which binary method is insensitive to the intensities of preferences, can still be used in a cheating manner, since we use the margin count to break ties. Thus the classical liberal has no settled answer how to deal with cheating in the second round. Accepting defeat is more gracious than trying to deny it. Of course one can argue that the vote cast is the only real test of what people actually want. But this can also be doubted, and there is still room for research here. The conclusion is that cheating in the second round best should be discussed with the arguments of the particular issue of the day, and we should not rely on voting methods or think that we could do so.

For the second round, the Fixed Point Borda is a compromise between the preference insensitivity of binary pairwise majority (with its cycles) and plain Borda that is sensitive to preference reversals and thus also cheating.

Given the limitations of reality, the classical liberal position seems rather reasonable and morally attractive, and thus provides an obvious counterexample to Kenneth Arrow's claim that there would be no reasonable and morally desirable constitutions. It remains of course for any group itself to determine what it considers ideal for what situation.

7. Probability

7.1 Introduction

7.1.1 The Rasch - Elo index

People sometimes compare electoral campaigns with matches. Normally this has an emotional content, like 'this candidate really puts up a fight'. But a formal similarity exists when chess players can win from each other in a cycle. Sen (1970:51) gives another formal comparison:

"Two Australians may tie for the Australian championship in some game, neither being able to defeat the other, but it is perfectly possible for one of them to become the world champion alone, since he might be able to defeat all non-Australians, which the other Australian champion may not be able to do so. Similarly, two poets or scientists could get the same national honors, with only one of them receiving some international honor such as the Nobel Prize, without this appearing as irrational in some significant sense."

It appears worthwhile to discuss the similarities and differences of these phenomena.

In chess there is the Elo rating for the competence of chess players - developed by Arpad Elo. Earlier, Georg Rasch developed for psychology the Rasch rating for the level of competence of students in answering test questions. It appears that the mechanisms of these ratings are the same. We can wonder whether we can use this Rasch - Elo rating for the competence of the candidates in an election. Let us first do this, and then think about what we are doing.

Consider the Condorcet example again. The Rasch - Elo ratings of the candidates follows from the matrix of pairwise vote results.

- The pairwise vote matrix of the Condorcet case.

`Condorcet[]; v = VoteMatrix[]`

$$\begin{pmatrix} 0 & 0.4 & 0.75 \\ 0.6 & 0 & 0.35 \\ 0.25 & 0.65 & 0 \end{pmatrix}$$

- The Rasch - Elo ratings RatingP and the, similarly ordered, probabilities Pr of winning from the 'average' opponent.

$\text{estv} = \text{MatchPrToRating}[\mathbf{v}]$

$\left\{ \text{SSE} \rightarrow 1.50254, \text{RatingP} \rightarrow \{140.137, 87.6328, 72.2299\}, \right.$
 $\left. \text{Pr} \rightarrow \{0.557507, 0.48221, 0.46012\}, \text{Slope} \rightarrow \frac{\log(10)}{400} \right\}$

- The implied aggregate preference ordering.

$\text{ListToPref}[\text{RatingP} /. \text{estv}]$

$\text{Pref}(\text{C}, \text{B}, \text{A})$

Thus, we now have a Rasch - Elo rating of items (politicians), similar to the rating of chess players or the rating of students and test questions.

What does this mean ? What have we done ? We can only do this kind of thing if we have a convincing theory and statistical model. Developing this will take up the rest of Part 7.

7.1.2 Other angles

It is useful to point to two other angles on voting that hang together with the above.

- There is a probabilistic element, when we allow people to vote strategically and to forecast how others will vote.
- We should be critical of the shape of the utility functions. It often appears that people's utility depends upon the correspondence between their capacities and the challenges that they face. Are there too few challenges, then people get bored; are there too many challenges, then they get frustrated. Recently, Mihaly Csikszentmihalyi (1997) pointed to the empirical evidence of this approach.

Also these aspects point to the usefulness of considering the theory of testing in general.

7.1.3 Testing in general: matching, ranking and rating

Every student in the world will be familiar with the idea of a test. In the 1950's, the Danish statistical consultant Georg Rasch was asked by his government to test children on their reading abilities. This research resulted into what now is called Item Response Theory (IRT). A test consists of subjects responding to items (questions). Rasch distinguished between the *competence* of the student and the *challenge* of the test, and he posed the hypothesis that both can be compared in the same rating dimension space. The rating of a subject is interpreted as competence, the rating of an item (question) is interpreted as the difficulty of the question. Rasch then related the *difference* between these ratings to the probabilities of success and failure of providing the correct answer. The more competent the student, or the easier the question, the likelier it is that the proper answer is given. Rasch's work has caused a wealth of other research and practical results, e.g. for computer programs that adjust to the observed level of competence and that provide the tests that are apparently needed to guide the student onwards to the next level.

In the early 1960s as well, Arpad Elo was asked by the U.S. Chess Federation to reconsider the system that the organisation used to indicate the strength of the players. Elo came up with the same system as Rasch, apparently without communication between them. The Elo rating system now is quite famous as well.

It is useful to consider the Rasch - Elo model, and link it up with voting. Both fall under the *general definition of testing*:

Testing is to score objects on criteria, and to compare objects by means of such criteria.

In voting, each voter can be seen as a criterion, and a candidate scores (wins the vote) or not.

There is a natural progression in testing from matching to ranking and to rating. Ratings have been used for IQ, sport games, bets or gambling, Social Science Citation Index, etcetera. A recent paper of Rafiei & Mendelzon (2000) looks into the rating of internet pages. There is a link to neural networks too - where a neuron fires when a threshold is reached. Once you grow aware of it, it is everywhere.

7.1.4 Consequences of this definition

There are two obvious applications for testing: one is *matching* objects - like in marriages - and the other is to *rank* or *rate* them - like in determining the winner of a *match* (game, contest). Ranking would be for an ordinal scale only. If we have an interval scale, so that only the difference between variables has objective meaning, then the *ranking* turns into a *rating*.

Note the different meanings that we thus attach to the various words. In common language the word 'match' is used for both games and matching, i.e. there would be a 'matching' if the distance measure is zero. For us, however, these two meanings of 'match' are a bit confusing, and we should avoid the confusion. We will use the expression 'find the best *combination*' for 'matching' in the sense of pairing up.

Ranking and rating can be done deterministically or with an element of randomness. When player 1 wins against player 2, it is possible that this result is deterministic. For example, if the game is 'weight', then player 1 or player 2 is heavier, and this result will be the same in repeated trials. However, in some matches there is only a probability to win. But even with winning probabilities we still can define a *distance measure*. Interestingly, the probability distance $|p - (1 - p)|$ is the same as the VoteMargin (i.e. if we interpret a vote proportion as the probability of getting a vote from a random voter).

It is important to see that there are always criteria. Even if we organise pairwise duels, like in chess, then the comparison of the items or subjects (players) still relies on criteria. The criterion for winning in chess is to take the opponent's King. But there are more criteria for getting to that point. It may be an enormous task to further develop such criteria, and hence we can decide to skip such development, and we may only regard the outcomes of such contests. But we should be aware that this is only a simplification.

A classic example of testing is where the criteria are exam questions. People who do an exam, can be seen as being in a contest with the questions. They can also be seen as being in pairwise contests with one another to do better on the exam. This insight links 'testing with criteria' to 'pairwise matches'.

We should be aware of at least three points of uncertainty: (1) The criteria might only be an approximation to the real objective of the test. (2) The way of

aggregation might also be subject to discussion. (3) And, more in general, the scores need not be certain but can have a stochastic component. Testing quickly becomes statistical testing.

One possible type of testing is voting. A voter can give an ordinal scale which indicates that the object higher on the list wins from the object lower on the list. This uses *certainty*. Alternatively, there is only the *probability* of winning. We still could use an ordinal scale to express such a likelihood of winning (such as "A is likelier to win than B").

There are some interconnections that at first may be surprising. It is interesting to observe that students doing a test, 'vote' for the answers. If the good answer does not get any votes, then we might conclude that the test itself failed (as an instrument for differentiation). Thus:

- In voting the interest is in the winning answer.
- In testing students, at issue is rather whether the student belongs to the winning group - so this testing might be seen as inverse voting.

Another basic idea of testing is the prediction of winning. If we have three persons and we know the winning probabilities in a match between the first two persons, then we would like to make a prediction on the winning probabilities for matches with the third person. To make this prediction, we could use criteria scores on the rankings of competence of the three persons.

7.1.5 Structure of the discussion

Compared to the huge literature, the discussion below will be introductory. First we will develop the Item Response matrix and the Match probability matrix, so that above discussion becomes more concrete. Then it appears useful to investigate how you can pass a multiple choice test by just guessing. Looking into this issue makes us more aware of the aspects to take into account. Only then we get to define the basic concepts that are required for the Rasch - Elo model. After all these introductory steps it becomes relatively simple to develop that model and to show its properties.

ResetAll

Economics[Logit]**Cool`Logit`**

AverageMatchPr	IRToMatchPr	PrDistance	SequentialIR
BinomialPass	IRToMatchValue	PrIR	SortIR
CreateIR	Logit	PrIRFromGrowth	WinPr
HeuristicIR	MatchPrToRating	RandomIR	
IREquations	MatchToPr	RatingP	
IRToMatch	MatchToRating	RatingQ	

7.2 Item Response matrix

7.2.1 Definition

An Item Response Matrix gives the response $\{i, j\}$ of person i on item (question) j , for n persons and m items. A 0 is fail and 1 is pass. Intermediate values are allowed in principle, though we concentrate on $\{0, 1\}$. Such a matrix records actual winnings and losses. It is a another step to estimate the probabilities and ratings from these.

7.2.2 Random generator

RandomIR creates a $0 \mid 1$ row or matrix, with the following formats. [Note that this kind of matrix can also be used for Approval voting.]

<code>RandomIR[m]</code>	uses 50/50 for all elements in the row
<code>RandomIR[m, {x}]</code>	uses BernoulliDistribution[x], all elements (for more elements it creates a table)
<code>RandomIR[m, {}]</code>	draws a random x , and uses this for all elements
<code>RandomIR[m, Random]</code>	draws from a random Bernoulli for each element
<code>RandomIR[n, m, x_____]</code>	does so for n rows
<code>RandomIR[p matrix]</code>	uses the p_{ij} elements for Bernoulli draws per cell

- For 3 persons and 6 questions, with a 50 % chance for the correct answer.

`RandomIR[3, 6]`

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

- Similarly, with the 90 % chance of the correct answer.

`RandomIR[3, 6, {0.9}]`

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

- Similarly, for all questions the same unknown p .

```
RandomIR[3, 6, {}]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

- With $p_{i,j}$ per element itself drawn randomly.

```
RandomIR[3, 6, Random]
```

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

7.2.3 Sorted matrices

The responses can also be ordered from 0 to 1, giving the impression that the easy questions are on the left and the difficult questions are on the right. We should treat such an interpretation with care, however, since also subjects with low ability could by chance answer difficult questions, and a perfect line-up is a very unlikely outcome. But this kind of matrix can be useful to emphasise some points of analysis.

```
SortIR[RandomIR[3, 6, Random]]
```

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

If you want to specify specific probabilities:

```
lis = SortIR[(RandomIR[10, {}] & ) /@ {0.9, 0.7, 0.3}]
```

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

```
SortIR[lis]
```

sorts such that the 1's are first,
suggesting that the easy questions are
on the left and the difficult ones on the right

7.2.4 Recovering the probabilities

How can we rate the persons and questions on their probabilities of winning ?

- A quick ordering follows from the observed average probabilities of winning.

`RatingP[] = Average /@ lis`

$$\left\{1, \frac{7}{10}, \frac{3}{10}\right\}$$

`RatingQ[] = Average /@ Transpose[lis]`

$$\left\{1, 1, 1, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right\}$$

<code>RatingP[i]</code>	identifies the rating for the <i>i</i> th person
<code>RatingP[{n}]</code>	gives a list of <i>n</i> rating symbols for persons
<code>RatingQ[j]</code>	identifies the rating for the <i>j</i> th question
<code>RatingQ[{m}]</code>	gives a list of <i>m</i> rating symbols for questions (items).

These have to be set by the user.

- If these probabilities would be independent, then we get the probability matrix:

`mat = Outer[Times, RatingP[], RatingQ[]]`

$$\begin{pmatrix} 1 & 1 & 1 & \frac{2}{3} & \frac{2}{3} & \frac{2}{3} & \frac{2}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{7}{10} & \frac{7}{10} & \frac{7}{10} & \frac{7}{15} & \frac{7}{15} & \frac{7}{15} & \frac{7}{15} & \frac{7}{30} & \frac{7}{30} & \frac{7}{30} \\ \frac{3}{10} & \frac{3}{10} & \frac{3}{10} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{10} & \frac{1}{10} & \frac{1}{10} \end{pmatrix}$$

The assumptions of *averaging* and *independence* actually are unsatisfactory. The averaging causes us to give the same weight to questions that have a different degree of difficulty. The independence does not seem right, since if we have people of comparable competence, then the probability that one person answers correctly would depend upon whether the others answer correctly as well.

- We can look at other functions:

`mat = Outer[Max, RatingP[], RatingQ[]]`

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \frac{7}{10} & \frac{7}{10} & \frac{7}{10} & \frac{7}{10} & \frac{7}{10} & \frac{7}{10} & \frac{7}{10} \\ 1 & 1 & 1 & \frac{2}{3} & \frac{2}{3} & \frac{2}{3} & \frac{2}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

This does not seem satisfactory. Try some formats yourself !

7.3 IR seen as matches

7.3.1 Introduction

This chapter concerns matches as games and not in the sense of pairing up objects (like marriages for people).

Suppose that subject A has probability p to answer correctly to a question (win a voter) and subject B has probability q to answer correctly. Like in a quiz, the questions may be considered to be less relevant, but we regard A and B as actually competing with each other.

- An example with repeated draws (10 voters), and $p = 0.7$ and $q = 0.5$.

```
lis = RandomIR[10, {0.7, 0.5}]
```

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

- When both subjects answer the same - $\{0, 0\}$ or $\{1, 1\}$ - then the value of $1/2$ can be given to each. The scores then result in the following winning frequencies - where the diagonal gives half of the number of questions.

```
IRToMatch[lis]
```

$$\begin{pmatrix} 5 & \frac{11}{2} \\ \frac{9}{2} & 5 \end{pmatrix}$$

- These are the relative frequencies. The diagonal gives $1/2$, as the probability of winning from an opponent of equal strength.

```
IRToMatchPr[lis]
```

$$\begin{pmatrix} \frac{1}{2} & \frac{11}{20} \\ \frac{9}{20} & \frac{1}{2} \end{pmatrix}$$

The latter matrix falls into the general class of match probability matrices. Element $\text{Pr}[i, j]$ is the probability that subject i (chess player i) wins from subject j (chess player j). Note that in this case the probability model is more complex, since above match matrix has been constructed via interpreting the IR matrix. It is up for discussion now whether that is a sensible approach.

`IRToMatchPr[lis_List?MatrixQ]` determines the match probabilities. Equal to `MatchToPr[IRToMatch[lis]]`

We can generalise this for any bigger IR matrix.

- Suppose that 4 persons answer 10 questions. Or 4 candidates meet 10 voters, so that the matrix would be the transpose of the approval matrix.

`lis = RandomIR[4, 10]`

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

- This gives the frequencies.

`IRToMatch[lis]`

$$\begin{pmatrix} 5 & 4 & \frac{11}{2} & 4 \\ 6 & 5 & \frac{13}{2} & 5 \\ \frac{9}{2} & \frac{7}{2} & 5 & \frac{7}{2} \\ 6 & 5 & \frac{13}{2} & 5 \end{pmatrix}$$

- A Match Probability Matrix divides by the number of matches.

`prs = IRToMatchPr[lis]`

$$\begin{pmatrix} \frac{1}{2} & \frac{2}{5} & \frac{11}{20} & \frac{2}{5} \\ \frac{3}{5} & \frac{1}{2} & \frac{13}{20} & \frac{1}{2} \\ \frac{9}{20} & \frac{7}{20} & \frac{1}{2} & \frac{7}{20} \\ \frac{3}{5} & \frac{1}{2} & \frac{13}{20} & \frac{1}{2} \end{pmatrix}$$

Thus an IR matrix can be transformed into a matrix of match results by regarding each pair of rows $\{i_1, i_2\}$ as a match between persons i_1 and i_2 . Each person scores on some criteria, and we can determine the shares of winning. Also a political election might be regarded as the outcome of such a screening process (on criteria that may be unknown to us).

Doing this for matches between persons also provides a suggestion for generalising IRT. We could generalise IRT by assuming that both items and subjects are scored on such (hidden) criteria. The very fact that items and subjects have ratings that can be compared, could be caused from the existence of such (hidden) criteria.

7.3.2 The importance of a tie

The routine `IRToMatchValue` defines when there is a win, loss or tie (values 1, 0, 1/2). A Match Matrix then gives the wins of pairwise matches, i.e. the levels or frequencies. By default, the diagonal assumes a 50/50 result of a match against an opponent of equal quality, and thus it gives half of the number of plays. A Match Matrix can also be transformed into a matrix of winning probabilities (`MatchPr`).

<code>IRToMatchValue[x, y]</code>	calculates the score of a pairwise match outcome {x, y}. Default {1, 0} → 1, {0, 1} → 0 and other values 1/2. Can be redefined by the user
<code>IRToMatch[matrix, opts]</code>	translates a IR matrix into a person to person match, using <code>IRToMatchValue</code> for the item scores; it gives the levels (frequencies)
<code>MatchToPr[x_List?MatrixQ]</code>	transforms a match outcome matrix into a probability matrix

- You can redefine `IRToMatchValue` yourself, e.g. for the value for ties.

ShowPrivate[IRToMatchValue]

Cool'Logit'Private'

`IRToMatchValue[x, y]` calculates the score of a pairwise match outcome {x, y}.

Default {1, 0} → 1, {0, 1} → 0 and other values 1/2. Can be redefined by the user

`IRToMatchValue(0, 1) = 0`

`IRToMatchValue(1, 0) = 1`

`IRToMatchValue(x_, x_) := $\frac{1}{2}$`

7.3.3 Rating of difficulty of questions

Above method takes only the total scores, and does not weigh by the degree of difficulty of the questions. We can check this by looking at sorted matrices.

■ Sorting.

```
sortlis = SortIR[lis]
```

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

■ Recalculating the match probabilities on the sorted list gives the same result as above: thus the degree of difficulty has no effect.

```
sortprs = IRToMatchPr[sortlis]
```

$$\begin{pmatrix} \frac{1}{2} & \frac{2}{5} & \frac{11}{20} & \frac{2}{5} \\ \frac{3}{5} & \frac{1}{2} & \frac{13}{20} & \frac{1}{2} \\ \frac{9}{20} & \frac{7}{20} & \frac{1}{2} & \frac{7}{20} \\ \frac{3}{5} & \frac{1}{2} & \frac{13}{20} & \frac{1}{2} \end{pmatrix}$$

Alternatively, we might consider leaving out the really easy and difficult tests that have no discriminating value.

■ In the following case, there are 6 ties, earning 3 points for each. There are only 4 clear wins for person 1. Due to the half points earned by a tie, the odds are 7 to 3.

```
pers1 = {1, 1, 1, 1, 1, 1, 0, 0, 0, 0};
pers2 = {1, 1, 0, 0, 0, 0, 0, 0, 0, 0};
```

```
IRToMatch[{pers1, pers2}]
```

$$\begin{pmatrix} 5 & 7 \\ 3 & 5 \end{pmatrix}$$

```
IRToMatchPr[{pers1, pers2}]
```

$$\begin{pmatrix} \frac{1}{2} & \frac{7}{10} \\ \frac{3}{10} & \frac{1}{2} \end{pmatrix}$$

Suppose we drop the last items that were too difficult for both subjects. This approach would conform to setting `IRToMatchValue[0, 0] = 0`. Then person 1 has 5 points and person 2 has 1 points, giving odds 5/1 instead of 7/3. Quite a

difference, and just caused by ties. Whether to drop difficult questions depends upon the case, however, and it is not something that can be decided in general. The only criterion is that a test or match should measure what it intends to measure.

7.3.4 Expected result

We started with the assumption that subject A has probability p to answer correctly to a question, and that subject B has probability q to answer correctly. Let us look at a single draw (Bernoulli), and not look at the repeated draws (binomial). Then the probability that A wins from B is $(1 + p - q) / 2$.

- This assumes a joint Bernoulli model.

```
JointDensity[x_, y_] = p^x (1 - p)^(1-x) q^y (1 - q)^(1-y);
```

```
ExpScore == Sum[Sum[IRToMatchValue[x, y] JointDensity[x, y],  
  {y, 0, 1}], {x, 0, 1}] // Simplify
```

```
ExpScore ==  $\frac{1}{2} (p - q + 1)$ 
```

There would be some simplicity in observation if A 's expected score is the same as A 's probability of providing the right answer. In that case, namely, the observations on one aspect are a good estimate for the other. Then $p = (1 + p - q) / 2$. Then:

- For the Bernoulli model, the probabilities of answering correctly and winning are equal, when:

```
Solve[p == (1 + p - q)/2, q] // Simplify
```

```
{{q -> 1 - p}}
```

In this case the probability of a tie can be neglected, since the two draws of p and $q = 1 - p$ behave the same as p by itself. In all other cases, however, ties have an important impact.

- The probability of a tie has an important weight in the joint Bernoulli model.

```
Pr[tie] = JointDensity[0, 0] + JointDensity[1, 1]
```

```
(1 - p)(1 - q) + p q
```


7.4 Binomial model for multiple choice tests

7.4.1 Introduction

Multiple choice tests are a good point of reference for repeated draws. Each question has a number of possible answers, and for a good test all these answers should be equally likely if the tested subject would not know anything about the material. Of course, if the subject knows much about the material, then the answers should not be misleading either. The key idea however is that the test starts to be discriminating if the subject scores better than by just guessing. In the following we assume that each question has the same number of answers m .

7.4.2 Probability of passing by just guessing

Let a multiple choice question have m possible answers, so that the probability of guessing the right answer is $p = 1/m$. Let the test consist of $n = 10$ of such questions, all independent. Then the probability of x correct answers to the whole test is given by the binomial distribution.

- The Binomial model for n draws and x successes, where each draw has the independent probability of success p .

`PDF[BinomialDistribution[n, p], x]`

$$(1 - p)^{n-x} p^x \binom{n}{x}$$

Assume that 5.5 correct answers are sufficient to pass the test. How many possible answers should each question have ? We want to reduce the probability of passing by just guessing.

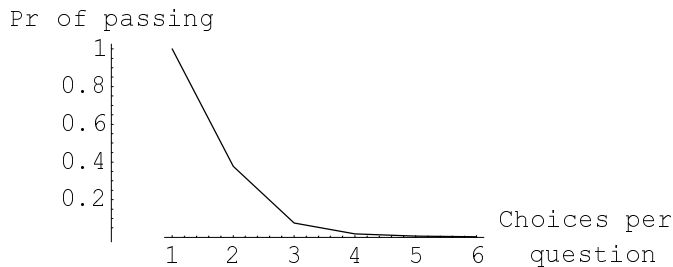
- When each question has 3 answers, 7.6% of the subjects would pass by only guessing.

`BinomialPass[3.]`

0.0765635

- With 4 multiple choice answers per test the probability of passing reduces to almost zero.

```
plot1 = PlotTable[BinomialPass[m], {m, 1, 6}, AxesOrigin → {0, 0},
  AxesLabel → {"Choices per\n question", "Pr of passing"}];
```



`BinomialPass[m, opts]` with default options $N \rightarrow 10$ and $\text{Min} \rightarrow 5.5$, gives the probability of passing a test with total score 5.5 out of 10 questions, when each question has m (multiple choice) answers, and thus probability $1/m$ of a correct answer

However, sometimes students are allowed to 'compensate' one course by another. Then a minimum score of only 4 out of 10 questions is allowed for 'passing the course with compensation'.

- If there are only 4 choices per question, then still 7.8% of the students might qualify for compensation by just guessing.

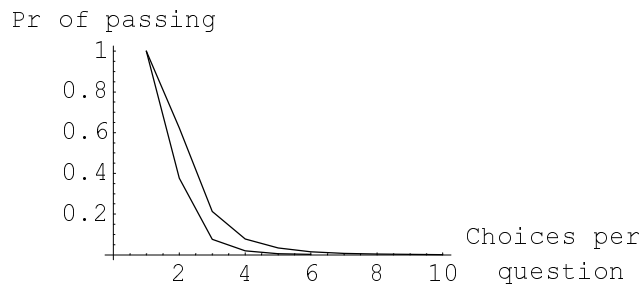
```
BinomialPass[4., Min → 4]
```

```
0.0781269
```

It follows that 6 possible choices are required.

```
plot2 = PlotTable[BinomialPass[m, Min → 4], {m, 1, 10}, AxesOrigin → {0, 0},
  AxesLabel → {"Choices per\n question", "Pr of passing"},
  DisplayFunction → Identity];
```

```
Show[plot1, plot2];
```



7.4.3 Inappropriate to define competence

The binomial model assumes constant p . In practice some multiple choice answers are likelier than others. Constant p is only relevant if you don't know anything - which is the Laplace position of ignorance. The more you know, the likelier the proper answer. In fact, university graduates can solve $1+1$ into 2 without error. It follows that the true model differs from the binomial model.

The number of questions is more related to the validity of the test than to the degree of difficulty. A test with only a limited number of choices is not a valid test. A certain number of questions is required to filter out the free riders (who can do a test several times a year). But if we want to measure the difficulty of the test and the competence of the subjects, then we have to look for other measures.

Thus, a possible definition of 'competence' as the 'share of successes minus the statistically expected probability' would be misleading.

- This would be a wrong definition of competence.

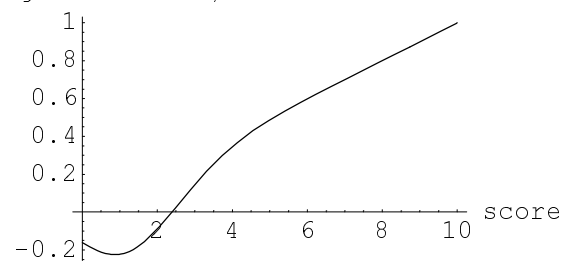
```
mis1[x_] = x/10 - PDF[BinomialDistribution[10, 1/6], x];
```

```
Plot[misl[x], {x, 0, 10}, AxesLabel → {"score",
"Corrected\nCompetence\n(Wrong definition)"}];
```

Corrected

Competence

(Wrong definition)



7.5 Basic concepts

Before we can continue with the Rasch - Elo model, we must develop some basic concepts.

7.5.1 Definition of Odds

If p is a probability of winning then the odds are $p / (1 - p)$. For example, if Trojan has a chance of 1/3 of winning the horse race, then the odds are 1 against 2 that it will win. The computation is very simple, but it will be useful to introduce a *Mathematica* symbol for it, since using the symbol conveys what we are discussing.

`Odds == Odds[p]`

$$\text{Odds} == \frac{p}{1-p}$$

`Solve[%, p]`

$$\left\{ \left\{ p \rightarrow \frac{\text{Odds}}{\text{Odds} + 1} \right\} \right\}$$

<code>Odds[p]</code>	gives the odds $p / (1-p)$
<code>Odds[p_List]</code>	maps over all entries
<code>Odds[p, i]</code>	finds the odds of the i th entry (also for Prospects p)
<code>Odds["B", p, q]</code>	finds the odds that A wins from B , when A answers a question with probability p and B similarly with q , and winning is 1 and ties are 1/2. The "B" refers to the single draw Bernoulli model
<code>FromOdds[o]</code>	returns the probability again $p = o / (1 + o)$

Input p can also be a Prospect[x, y, p].

7.5.2 Definition of Logit - and relation to Logistic

Whereas the odds range over $[0, \infty)$, the logarithm of the odds ranges over $(-\infty, \infty)$ and it is sometimes more attractive to work with.

`Logit[p] \Leftrightarrow Log[Odds[p]]`

$$\log\left(\frac{p}{1-p}\right) \Leftrightarrow \log\left(\frac{p}{1-p}\right)$$

$\text{Logit}[p]$
 $\text{Log}[p/(1-p)]$ – also for lists and prospects,

If we regard the Logit as a function of some other variable x , then we get a functional relationship that is called the Logistic. Let us assume that this variable x is scaled linearly with constant c .

$$\text{Logit}[p] == c x$$

$$\log\left(\frac{p}{1-p}\right) == c x$$

$$\text{Solve}[\%, p]$$

$$\left\{ \left\{ p \rightarrow \frac{e^{c x}}{1 + e^{c x}} \right\} \right\}$$

If we divide numerator and denominator with $e^{c x}$ then we get the common expression for the Logistic function.

$$\text{Pr} == \text{Logistic}[x, \text{Slope} \rightarrow c]$$

$$\text{Pr} == \frac{1}{1 + e^{-c x}}$$

7.5.3 Logistic function and difference in competence

Let us assume that a person (voter) has a competence rating RatingP and that the item (test question) has a rating on the degree of difficulty RatingQ . Since there is no obvious 'zero' value for these ratings, we get an *interval scale* (which is a crucial observation), and the difference in ratings $d = \text{RatingP} - \text{RatingQ}$ has real meaning and becomes the variable that determines the probability of a correct answer, or, the vote cast for the candidate.

Hence the variable x for the $\text{Logistic}[x]$ will be the difference in ratings of the subject and the item. Actually, since the rating scale makes no real distinction between subjects and items, the difference in ratings between subjects can also be used to find the probability of winning a pairwise contest.

- The Rasch - Elo model is this form of the Logistic:

$$\text{eq} = (\text{Pr} == \text{Logistic}[d, \text{Slope} \rightarrow c])$$

$$\text{Pr} == \frac{1}{1 + e^{-c d}}$$

- The probability that person i answers question j correctly.

`Pr[i, j] == Logistic[RatingP[i] - RatingQ[j], Slope → c]`

$$\Pr(i, j) == \frac{1}{1 + e^{-c(\text{RatingP}(i) - \text{RatingQ}(j))}}$$

- The general Logistic function has parameters that determine the shift of the domain, the range and its central value, and the slope.

`Logistic[d, {Right, Center, Range, Slope}]`

$$\text{Center} + \frac{\text{Range}}{1 + e^{-(d - \text{Right}) \text{Slope}}}$$

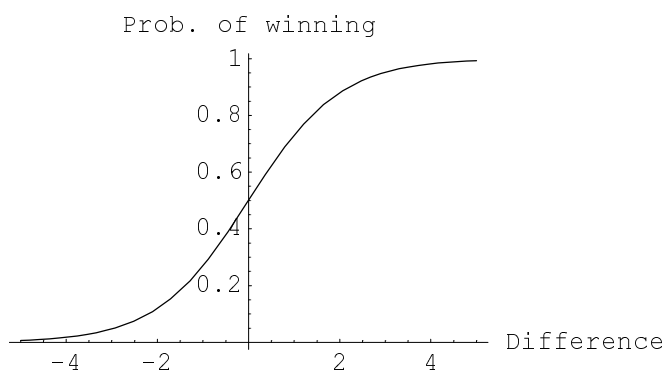
- These parameters can also be set or entered as options. Default are:

`Options[Logistic]`

`{Right → 0, Center → 0, Range → 1, Slope → 1}`

- This is the ubiquitous shape of the function.

`plot = Plot[Logistic[d], {d, -5, 5}, AxesLabel → {"Difference", "Prob. of winning"}];`



`Logistic[x, opts]` gives the Logistic function a.k.a. Sigmoid

`Logistic[x, {right, center, range, slope}]` is another input format

Default parameter values are given by `Options[Logistic]`. Note that 1 parameter can be considered redundant.

7.5.4 The probability model

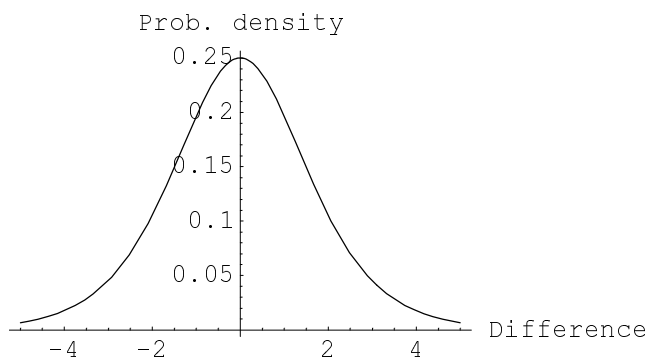
Since the Logistic function goes from 0 till 1, continuously increasing, it can be regarded as a cumulative probability distribution. Elo (1978) calls the density the Verhulst probability density. The Logistic is in fact very close to the Normal distribution (by proper choice of parameters).

- The density is:

```
ds[d_] = D[Logistic[d, Slope -> c], d]
```

$$\frac{c e^{-cd}}{(1 + e^{-cd})^2}$$

```
Plot[ds[d] /. c -> 1, {d, -5, 5}, AxesLabel -> {"Difference", "Prob. density"}];
```



- The variance is:

```
Integrate[d^2 ds[d], {d, -∞, ∞}] /. Re[c] > 0 -> True
```

$$\frac{\pi^2}{3 c^2}$$

The inverse of the Logistic gives the Logit. Thus, if we have the odds, then we can find a value for the differences in ratings. In fact, the difference in ratings is just the logarithm of the odds divided by unknown parameter c .

- This solves above equation for the difference in competence d . Note that the Log in this expression gives the Logit when we manipulate the minus signs.

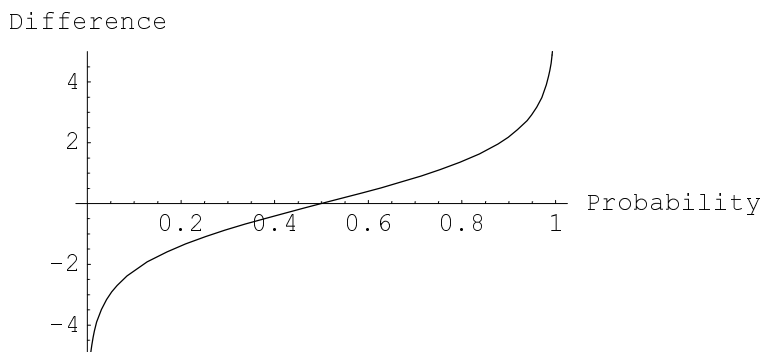
```
Solve[eq, d]
```

Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.

$$\left\{ \left\{ d \rightarrow -\frac{\log\left(-\frac{p_r-1}{p_r}\right)}{c} \right\} \right\}$$

- This plot is the inverse of the Logistic plot above.

```
Plot[Log[p/(1 - p)], {p, 0, 1}, AxesLabel → {"Probability", "Difference"},  
PlotRange → {-5, 5}];
```



7.5.5 Probability distance

When player 1 wins from player 2, it is possible that this result is deterministic. For example, if the game is 'weight', then player 1 or player 2 is heavier, and this result will be the same in repeated trials. However, in some matches there is only a probability to win. But even with winning probabilities we still can define a 'distance measure'.

(1) The basic idea is that when two players are equal - of equal strength - then the winning probabilities should be $p = 1/2$. Thus we are not considering 'matching' in the sense of two colours being exactly equal, but in the sense of being equally likely to be selected.

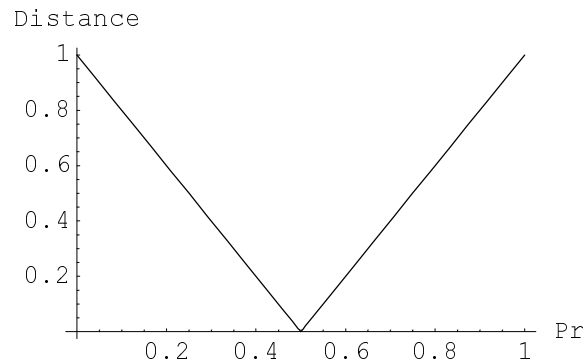
(2) If player 1 has the probability p of winning from player 2, then an obvious distance between the players would be the Euclidean distance $dist(p) = \text{Abs}[p -$

$(1 - p)] = \text{Abs}[2p - 1]$. This distance measure runs from 0 till 1. The measure is symmetric, since $\text{dist}(p) = \text{dist}(1 - p)$. Note that this function is the same as the VoteMargin except for the absolute value.

```
PrDistance[p]
```

```
|2 p - 1|
```

```
Plot[%, {p, 0, 1}, AxesLabel -> {Pr, Distance}];
```



(3) If subjects 1 and 2 play against a common opponent 3, then we have two probabilities $p = \text{Pr}[1, 3]$ and $q = \text{Pr}[2, 3] \neq 1 - p$, and then we would like to have a distance measure as well. We would then translate p and q into the probability that 1 wins from 2, $\text{Pr}[1, 2] = f(p, q)$ for some function f . Then the distance follows from the above, $\text{dist}(p, q) = \text{dist}(f(p, q))$.

Under particular assumptions (see the 'direct' approach discussed below) it is possible to find a simple expression for the implied winning probability. Then we get:

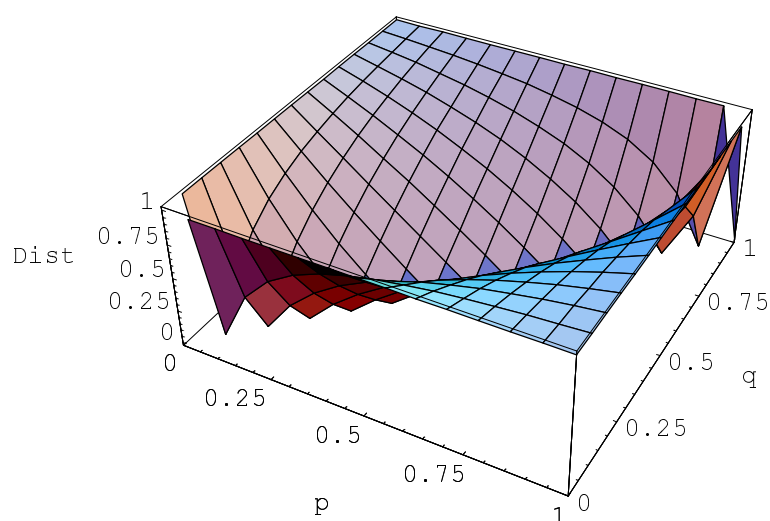
```
PrDistance[p, q]
```

$$\left| \frac{2\left(\frac{1}{q} - 1\right)}{\frac{1}{q} - 2 + \frac{1}{p}} - 1 \right|$$

```
% // Simplify
```

$$\left| \frac{p - q}{-2q p + p + q} \right|$$

```
Plot3D[%, {p, 0, 1}, {q, 0, 1}, AxesLabel -> {"\np", " q", "Dist "}]
```



<code>PrDistance[p]</code>	gives the distance measured as $\text{Abs}[p - (1-p)]$
<code>PrDistance[p, q]</code>	gives the distance for $\text{WinPr}[p, q]$ which assumes that p and q are defined on a third 'standard' player

This measure shows that 'matches' (games) also might be regarded as 'matching' (as for marriage), since we have found a distance measure that can be zero. However, the zero distance here means equal likelihood of winning (which is not the first association with marriage).

What is the usefulness of this distance measure ? If we regard only $(p - (1 - p))$ which thus has no absolute value, then we not only have the distance, but also the sense of direction. Normally, however, 'distance' is regarded as an absolute value. In that case, going from probability to distance means a loss of information (the direction). In that sense, the information in p itself is superior.

As we will see in the Rasch model, the p and q are analysed in terms of hidden competence ratings. It appears that these hidden factors add little information. We thus shall see that working with p itself would seem to be superior again.

7.6 The Rasch - Elo model

7.6.1 The Rasch - Elo or Item Response model

The Rasch - Elo, or Item Response, model is: if two variables (persons or items) have a certain rating (the rating of the person interpreted as competence, the rating of the item interpreted as the difficulty of the question), then the probability of giving the proper response depends upon the difference between these ratings. For chess players, they can take the difference of their ratings, and then determine the probabilities of winning or losing.

The probability model is decent and the statistical procedure that we apply is the basic 'mean difference test'. The common statistical textbook develops this test for the Normal distribution, but the Logistic function is very close to that distribution. The ratings that we have been discussing are regarded as mean values. The subjects or items are thought to sample around their mean, sometimes underperforming and sometimes overperforming. There is a probability that the one subject or item samples a higher value than the other subject or item, and thereby wins. That likelihood increases when the means are further apart.

- For example, for 2 persons and 3 questions, the matrix of probabilities would be:

```
CreateIR[RatingP[{2}], RatingQ[{3}]]
```

$$\begin{pmatrix} \frac{1}{1+e^{\text{RatingQ}(1)-\text{RatingP}(1)}} & \frac{1}{1+e^{\text{RatingQ}(2)-\text{RatingP}(1)}} & \frac{1}{1+e^{\text{RatingQ}(3)-\text{RatingP}(1)}} \\ \frac{1}{1+e^{\text{RatingQ}(1)-\text{RatingP}(2)}} & \frac{1}{1+e^{\text{RatingQ}(2)-\text{RatingP}(2)}} & \frac{1}{1+e^{\text{RatingQ}(3)-\text{RatingP}(2)}} \end{pmatrix}$$

- Suppose that the standard rating is 100, and let the other ratings be as follows, and now with a specific slope.

```
pmat = CreateIR[{100, 120}, {50, 100, 200}, Slope → 0.1]
```

$$\begin{pmatrix} 0.993307 & 0.5 & 0.0000453979 \\ 0.999089 & 0.880797 & 0.00033535 \end{pmatrix}$$

This output finds a clear interpretation. Question 1 is so easy, compared to the competence of the tested subjects, that the probability that it is answered correctly is more than 99%. Question 2 is harder, has the same rating as Person 1, and thus his or her probability of answering correctly is 1/2. Had the slope been 1, then the 20 points higher rating for Person 2 would have meant again a

chance of 99% of answering correctly. But the slope now reduces his or her chances to almost 88%. Question 3 has a rating that almost excludes a correct answer.

`CreateIR[p , q , $opts$]` creates the matrix from p personal ratings and q item ratings, both lists, using the options for the Logistic function

- If we use above probabilities for an exam... This draws randomly, using the probabilities in the cells of the matrix.

`RandomIR[pmat]`

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

- And again for a resit exam - person 1 now got question 2, or was it just chance ?

`RandomIR[pmat]`

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

7.6.2 Multiplicative odds

7.6.2.1 Transitive: {1, 2} and {2, 3} give {1, 3}

The Rasch - Elo model has transitively multiplicative odds. If we have the odds for the match of {1, 2} and the match of {2, 3}, then we can find the odds for the match {1, 3} by simply multiplying the odds. For example:

- If 1 has probability 1/3 to win from 2, then its odds are 1/2.
- If 2 has the probability 1/4 to win from 3, then its odds are 1/3.
- The odds that 1 wins from 3 *thus* are $1/2 * 1/3 = 1/6$. The probability that 1 wins from 3 then is 1/7.

- In formulas:

`Odds[Pr[1, 2]] Odds[Pr[2, 3]] == Odds[Pr[1, 3]]`

$$\frac{\text{Pr}(1, 2) \text{Pr}(2, 3)}{(1 - \text{Pr}(1, 2)) (1 - \text{Pr}(2, 3))} == \frac{\text{Pr}(1, 3)}{1 - \text{Pr}(1, 3)}$$

% /. {Pr[1, 2] → 1/3, Pr[2, 3] → 1/4}

$$\frac{1}{6} == \frac{\text{Pr}(1, 3)}{1 - \text{Pr}(1, 3)}$$

- We can check this property by solving the three equations for the logits.

eq = IREquations[3]

$$\left\{ \log\left(\frac{\text{Pr}(1, 2)}{1 - \text{Pr}(1, 2)}\right) == \text{Slope}(\text{RatingP}(1) - \text{RatingP}(2)), \right. \\ \left. \log\left(\frac{\text{Pr}(1, 3)}{1 - \text{Pr}(1, 3)}\right) == \text{Slope}(\text{RatingP}(1) - \text{RatingP}(3)), \log\left(\frac{\text{Pr}(2, 3)}{1 - \text{Pr}(2, 3)}\right) == \text{Slope}(\text{RatingP}(2) - \text{RatingP}(3)) \right\}$$

elim = Eliminate[eq, Array[RatingP, 3]]

$$\log\left(\frac{\text{Pr}(1, 3)}{1 - \text{Pr}(1, 3)}\right) - \log\left(\frac{\text{Pr}(2, 3)}{1 - \text{Pr}(2, 3)}\right) == \log\left(\frac{\text{Pr}(1, 2)}{1 - \text{Pr}(1, 2)}\right)$$

- Pr[1, 3] thus directly depends on the other probabilities.

Solve[elim, Pr[1, 3]]

$$\left\{ \left\{ \text{Pr}(1, 3) \rightarrow \frac{\text{Pr}(1, 2) \text{Pr}(2, 3)}{2 \text{Pr}(2, 3) \text{Pr}(1, 2) - \text{Pr}(1, 2) - \text{Pr}(2, 3) + 1} \right\} \right\}$$

7.6.2.2 Relation for the 'odds share'

Since the probabilities satisfy the relation $\text{Pr}[i, j] = 1 - \text{Pr}[j, i]$, the transitive relationship can be manipulated around. It is also important that it does not matter who we call "person 3". The general principle then becomes that if we have two, then we can find the third. The following relations are equivalent.

- Pr[1, n] and Pr[1, m] give Pr[n , m] for example for $n = 2$ and $m = 3$
- Pr[1, m] and Pr[2, m] give Pr[1, 2] for example for $m = 3$
- Pr[1, 2] and Pr[1, n] give Pr[2, n] for example for $n = 3$

This can be interpreted in two ways:

1. When we *already have* the probabilities for the matches of {1, 2} and {1, 3}, then we can derive the probability for the match {2, 3}. Person 1 has given us enough information to say something about the relative strength of his opponents.
2. If persons 1 and 2 *announce* a match, then, before we bet on the outcome, we can regard their winning probabilities against a common opponent m (for example $m = 3$). This also gives a forecast of the match.

- $\text{Pr}[1, 2]$ appears to be the 'share of the odds for 1 in the total odds to 3'. Just rework the multiplicative odds that we derived above.

```
Odds[Pr[1, 2]] * Odds23 == Odds13;
```

```
Solve[%, Pr[1, 2]]
```

$$\left\{ \left\{ \text{Pr}(1, 2) \rightarrow \frac{\text{Odds13}}{\text{Odds13} + \text{Odds23}} \right\} \right\}$$

- If we use $p = \text{Pr}[1, m]$ and $q = \text{Pr}[2, m]$, for some common opponent m , then $\text{Pr}[1, 2]$ follows as the share in the total odds. The result does not depend on other parameters.

```
Pr[1, 2] == Odds[p] / (Odds[p] + Odds[q]) // Simplify
```

$$\text{Pr}(1, 2) == \frac{p - pq}{-2q p + p + q}$$

- This surprisingly elegant result is also available as a routine.

```
Pr[1, 2] == WinPr[p, q]
```

$$\text{Pr}(1, 2) == \frac{\frac{1}{q} - 1}{\frac{1}{q} - 2 + \frac{1}{p}}$$

```
% // Simplify
```

$$\text{Pr}(1, 2) == \frac{p - pq}{-2q p + p + q}$$

Note that the outcome is symmetric if we regard $\text{WinPr}[q, p]$, and that the two probabilities add up to 1, as they should.

`WinPr[p, q]`

gives the probability that 1 wins against 2, if 1 has probability p to win from a third opponent, and if 2 has the probability q to win from the same opponent. This assumes the Rasch model

There is another small check on consistency. Since the Rasch - Elo model has also multiplicative odds, the relationship above should also satisfy this. Since p is the probability of winning of 1 from m , and since $(1 - q)$ is the probability of m to win from 2, then multiplying their odds gives the odds for 1 winning from 2. If we deduce the odds for 1 and 2 in above manner, then we get the same result.

```
Odds[p] Odds[1 - q] == Odds[WinPr[p, q]]
```

$$\frac{p(1-q)}{(1-p)q} = \frac{\frac{1}{q} - 1}{\left(1 - \frac{\frac{1}{q} - 1}{\frac{1}{q} - 2 + \frac{1}{p}}\right) \left(\frac{1}{q} - 2 + \frac{1}{p}\right)}$$

```
% // Simplify
```

```
True
```

7.6.2.3 Presence of (2 p - 1)

Though we already derived this relationship, it is interesting to show another derivation in which $(2p - 1)$ reappears (i.e. the probability distance). When we have the probabilities for the matches of $\{1, 2\}$ and $\{1, 3\}$, then we can derive the probability for the match $\{2, 3\}$.

- In this way, we see the expression $(2p - 1)$ reappearing.

```
Solve[elim, Pr[2, 3]] // Simplify
```

$$\left\{ \left\{ \Pr(2, 3) \rightarrow \frac{(\Pr(1, 2) - 1) \Pr(1, 3)}{\Pr(1, 2) (2 \Pr(1, 3) - 1) - \Pr(1, 3)} \right\} \right\}$$

- Above finding has been put in this separate function.

```
PrIR[1, 2, 3]
```

$$\Pr(2, 3) \rightarrow \frac{(\Pr(1, 2) - 1) \Pr(1, 3)}{\Pr(1, 2) (2 \Pr(1, 3) - 1) - \Pr(1, 3)}$$

7.6.2.4 Effects of education and training

If Person 1 invests in education and training, then his or her level of competence $\text{RatingP}[1]$ rises, but we should assume that these remain the same for the others. $\text{IREquations}[3]$ shows then that *all* winning probabilities of Person 1 rise, while the bilateral probabilities of the others remain the same.

Let us see how this can be consistent. From $\text{IREquations}[3]$, we see that a rise in $\text{RatingP}[1]$ has a positive effect on the Logit, so that there is a multiplicative effect on the odds. Since the slope is the same, the odds rise with the same factor $x = (1 + g)$, where g is the rate of growth.

- The old and new situations are clearly consistent.

```
x Odds12 Odds23 == x Odds13
```

```
Odds12 Odds23 x == Odds13 x
```

- This means for the probabilities $\text{Pr}[1, j]$ (in this case for $j = 2, 3$) a rise with factor y_{12} and y_{13} .

```
eqs = {x Odds[Pr[1, 2]] == Odds[y12 Pr[1, 2]],
x Odds[Pr[1, 3]] == Odds[y13 Pr[1, 3]]}
```

$$\left\{ \frac{x \text{Pr}(1, 2)}{1 - \text{Pr}(1, 2)} == \frac{y_{12} \text{Pr}(1, 2)}{1 - y_{12} \text{Pr}(1, 2)}, \frac{x \text{Pr}(1, 3)}{1 - \text{Pr}(1, 3)} == \frac{y_{13} \text{Pr}(1, 3)}{1 - y_{13} \text{Pr}(1, 3)} \right\}$$

- Apparently, the probabilities adjust with a restriction, but it can be done consistently.

```
soly12 = Solve[eqs, {y12}] // Simplify
```

$$\left\{ \left\{ y_{12} \rightarrow \frac{x}{(x-1) \text{Pr}(1, 2) + 1} \right\} \right\}$$

```
soly13 = Solve[eqs, {y13}] // Simplify
```

$$\left\{ \left\{ y_{13} \rightarrow \frac{x}{(x-1) \text{Pr}(1, 3) + 1} \right\} \right\}$$

```
sol = Solve[eqs, {y12}, {x}] // Simplify
```

$$\left\{ \left\{ y_{12} \rightarrow \frac{y_{13} (\text{Pr}(1, 3) - 1)}{-(y_{13} - 1) \text{Pr}(1, 2) + y_{13} \text{Pr}(1, 3) - 1} \right\} \right\}$$

- This shows the consistency.

```
sol /. soly13[[1, 1]] // Simplify
```

$$\left\{ \left\{ y_{12} \rightarrow \frac{x}{(x-1) \text{Pr}(1, 2) + 1} \right\} \right\}$$

Above finding has been used to create another routine that allows us to update the odds in a consistent manner.

- This updates a probability for the general growth (reduction) rate in the odds, deriving from increased (decreased) competence.

```
PrNew[1, 2] == PrIRFromGrowth[Pr[1, 2], g]
```

$$\text{PrNew}(1, 2) == \frac{(g+1) \text{Pr}(1, 2)}{g \text{Pr}(1, 2) + 1}$$

`PrIRFromGrowth[p , g]` gives the new probability of winning $p_{\text{new}} = (1+g) / (1+g p) p$, for probability p when the general growth factor in all odds (determined by the growing competence) grows by rate g (expressed as a perunage)

The growth factor thus is $x = 1 + g$, and x thus applies to the odds, and not to the probability.

7.6.2.4 Multiplicative Odds Matrix

A match probability matrix gives the probabilities of the pairwise duels of n subjects, with the entry $\text{Pr}[i, j]$ the probability that i wins from j . [Note: this thus is similar as the *VoteMatrix*, while, of course, a share of the vote is not the same as the probability of winning. A vote share however can be interpreted as such a probability, when we sample a random voter.]

If we consider a match of n persons, then the Rasch - Elo assumption in fact implies that the *whole* matrix can be constructed from the winning probabilities of only 1 person. Above relationship can namely be applied recursively. Since the odds are multiplicative, we will call the matrix a *Multiplicative Odds Matrix*.

The Rasch - Elo model thus imposes a restriction on the probabilities. Alternatively, when we have data, then it can be tested whether the probabilities satisfy this relationship, and whether the logistic format can be accepted or should be rejected.

- This shows how a whole matrix follows from the winning probabilities of just one person.

`PrIR[{3/4, 3/8, 1/16}]`

$$\begin{pmatrix} \frac{1}{2} & \frac{3}{4} & \frac{3}{8} & \frac{1}{16} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{6} & \frac{1}{46} \\ \frac{5}{8} & \frac{5}{6} & \frac{1}{2} & \frac{1}{10} \\ \frac{15}{16} & \frac{45}{46} & \frac{9}{10} & \frac{1}{2} \end{pmatrix}$$

Let us check the consistency of the growth of the odds above. Let person 1 invest in her or his education, so that the odds of winning against the other opponents rise by 5%. If we apply the growth rule, only the first row (and implied column) changes, but the rest of the matrix remains the same.

- To check the consistency of growth of the odds, suppose that $g = 1 / 20$.

`PrIRFromGrowth[# , 1/20] & /@ {3/4, 3/8, 1/16}`

$$\left\{ \frac{63}{83}, \frac{63}{163}, \frac{7}{107} \right\}$$

- We find that the rest of the matrix remains the same indeed.

`PrIR[%]`

$$\begin{pmatrix} \frac{1}{2} & \frac{63}{83} & \frac{63}{163} & \frac{7}{107} \\ \frac{20}{83} & \frac{1}{2} & \frac{1}{6} & \frac{1}{46} \\ \frac{100}{163} & \frac{5}{6} & \frac{1}{2} & \frac{1}{10} \\ \frac{100}{107} & \frac{45}{46} & \frac{9}{10} & \frac{1}{2} \end{pmatrix}$$

<code>PrIR[i, j, k]</code>	determines $\text{Pr}[j, k]$ from $\text{Pr}[i, j]$ and $\text{Pr}[i, k]$ for the Rasch or Elo model
<code>PrIR[Pr[i, j], Pr[i, k]]</code>	gives $\text{PrIR}[i, j, k]$
<code>PrIR[x, y]</code>	for numbers, assumes that $x = \text{Pr}[1, 2]$ and $y = \text{Pr}[1, 3]$
<code>PrIR[x_List]</code>	creates the matrix from the winning probabilities of just one person.

Note: Remember that Pr is Orderless.

7.6.3 The assumption of independence

The situation of multiplicative odds has a small paradoxical aspect.

- If 1 wins from 2 with probability $1/2$, then we might say that this is like flipping a coin. Thus $\text{Pr}[1, 2] = 1/2$ and $\text{Odds}[1, 2] = 1$.
- If 2 wins from 3 with probability $1/6$, then we might say that this is like throwing a die. Thus $\text{Pr}[2, 3] = 1/6$ and $\text{Odds}[2, 3] = 1/5$.
- We would also think that the match between 1 and 3 would be independent from the other matches. In chess, the players go to another room, and no interference is allowed. Flipping a coin and throwing a die would be independent in time and place.
- Statistical theory instructs us that independence means that we must multiply the probabilities and not the odds. Then we find $\text{Pr}[1, 3] = 1/12$.
- From the multiplicative odds - the Rasch - Elo model - we find that $\text{Odds}[1, 3] = 1 * 1/5 = 1/5$, so that $\text{Pr}[1, 3] = 1/6$.

- But $1/12 \neq 1/6$.
- Clearly, Rasch and Elo would not imply that flipping a coin would be dependent on tossing a die, or that the match between players A and B in room X would depend on the match between C and D in room Y ?

`PrIR[{1/2, 1/6}]`

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{6} \\ \frac{5}{6} & \frac{5}{6} & \frac{1}{2} \end{pmatrix}$$

The answer to this paradox is that *if* the Rasch - Elo model applies, then there *is* a dependence: players A , B , C and D are linked by their levels of competence. We can do some manipulations with the probabilities that look like assuming independence, like flipping a coin and throwing a die, but there is no real independence. If the outcome of the process would really be $1/12$, then the processes are independent indeed, and then the Rasch - Elo model would not apply. The next chapter will discuss this in more detail.

The following is another example that does not correspond with the assumptions of the Rasch - Elo model.

- This is a matrix of winning probabilities that has non-multiplicative odds.

`prs = IRTToMatchPr[{{1, 0, 1, 1}, {0, 0, 1, 0}, {1, 1, 1, 1}}]`

$$\begin{pmatrix} \frac{1}{2} & \frac{3}{4} & \frac{3}{8} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{8} \\ \frac{5}{8} & \frac{7}{8} & \frac{1}{2} \end{pmatrix}$$

- Namely, for `Pr[2, 3]` the matrix contains $1/8$ but multiplicative odds give $1/6$.

`PrIR[3/4, 3/8]`

$$\frac{1}{6}$$

7.6.4 'Observational simplicity'

For the Bernoulli model we derived that the probability that A wins is equal to the probability p of answering correctly iff $q = 1 - p$. For the Rasch - Elo model, we find this observational simplicity for $q = 1/2$, meaning that B and the question would have an equal rating.

- For $m = \text{question}$ and $p = \text{Pr}[A, m]$ and $q = \text{Pr}[B, m]$, the probability that A would win from B is, as discussed:

`WinPr[p, q] // Simplify`

$$\frac{p - p q}{-2 q p + p + q}$$

- For the Rasch - Elo model, the probability of answering correctly is equal to the probability of winning, when:

`Solve[p == WinPr[p, q], q]`

$$\left\{ \left\{ q \rightarrow \frac{1}{2} \right\} \right\}$$

Note that this property of 'observational simplicity' is just something that we used here to highlight the different implications of the two models. It is not a big concept or something, it is just useful to clarify a point.

We can draw three conclusions. (1) The joint Bernoulli model is not the same as Rasch - Elo. (2) One cause is that ties weight more heavily in the Bernoulli model. (3) The route via IR matrices can depict both matches, depending upon how the matrices have been created.

Note for simulation: If we use `RandomIR` for simulation, then for the Rasch - Elo model, we should use `RandomIR[pmat]` with a matrix of probabilities $pmat$. A table of just random probabilities, like taking $pmat = \text{Table}[\text{Random}[], \{i, 4\}, \{j, 10\}]$, seems less appropriate since it does not impose the dependencies. `CreateIR` can be used, with possibly random ratings.

7.7 Cycling in matches

What we can do with the Rasch - Elo probabilities is to multiply them in *subsequent* matches. This looks like independence, but it is a bit different. The probabilities of matches between persons depend on each other, but they do not depend upon the *order* in which the matches take place. A match between A , B and C can be done pairwise, between A & B , then A & C and then B & C . The outcome of this match as a whole (chess tournament) should not be influenced by the order of such sub-matches. Interestingly, this issue brings us also to the topic of the paradoxes of voting.

The Rasch - Elo model assumes that the probability of a tie between two subjects is infinitely small. A person wins or loses. (It is remarkable that the model is used for chess.) This dichotomy has the useful effect that we can create a binomial tree (that only considers two possible outcomes). But in sequential matches cycles can occur, which effectively implies that ties are acknowledged anyway. (Which must be the reason that it is still used for chess.)

- Define the players and make the binomial tree.

```
players = {"A", "B", "C"};

lis = SequentialIR[players];
```

- The tree has only $A > B$ or $B > A$, and similar for the other matches.

```
MatrixForm[lis]
```

$$\begin{pmatrix} A > B \wedge A > C \wedge B > C \\ A > B \wedge A > C \wedge C > B \\ A > B \wedge C > A \wedge B > C \\ A > B \wedge C > A \wedge C > B \\ B > A \wedge A > C \wedge B > C \\ B > A \wedge A > C \wedge C > B \\ B > A \wedge C > A \wedge B > C \\ B > A \wedge C > A \wedge C > B \end{pmatrix}$$

- Thus, the probabilities are dependent on each other, but in sequential draws they can be multiplied since they are independent of order.

```
prs = SequentialIR[Pr, players];
```

`MatrixForm[prs]`

$$\begin{pmatrix} \text{Pr}(1, 2) \text{Pr}(1, 3) \text{Pr}(2, 3) \\ \text{Pr}(1, 2) \text{Pr}(1, 3) (1 - \text{Pr}(2, 3)) \\ \text{Pr}(1, 2) (1 - \text{Pr}(1, 3)) \text{Pr}(2, 3) \\ \text{Pr}(1, 2) (1 - \text{Pr}(1, 3)) (1 - \text{Pr}(2, 3)) \\ (1 - \text{Pr}(1, 2)) \text{Pr}(1, 3) \text{Pr}(2, 3) \\ (1 - \text{Pr}(1, 2)) \text{Pr}(1, 3) (1 - \text{Pr}(2, 3)) \\ (1 - \text{Pr}(1, 2)) (1 - \text{Pr}(1, 3)) \text{Pr}(2, 3) \\ (1 - \text{Pr}(1, 2)) (1 - \text{Pr}(1, 3)) (1 - \text{Pr}(2, 3)) \end{pmatrix}$$

- Simplify the possible results.

`MatrixForm[SequentialIR[Simplify, lis]]`

$$\begin{pmatrix} A > B > C \\ A > C > B \\ A > B > C \wedge C > A \\ C > A > B \\ B > A > C \\ B > A > C \wedge C > B \\ B > C > A \\ C > B > A \end{pmatrix}$$

In the third and sixth row we find 'cycling'. The solution is that there is a deadlock here, and all players should get an equal value. Indeed, what is called 'cycling' for multiperson games, is nothing but the general deadlock or tie between two persons.

In practice one often might not check whether such cycling has occurred. A value "A won over B" is recorded by itself, while this just could be a probabilistic result, and while actually there is equality due to cycling. The consequence is most dramatic when, once $A > B$ and $B > C$ have been recorded, that then the contest of A & C is no longer held.

However, when we make certain that *all* players play against each other, and when match results are recorded in binary 1 or 0, then it appears that we automatically conclude to a deadlock in case of cycling. This approach actually is the Borda approach in voting. We namely get the scores per submatch, shown below.

- If we add {1, 1, 1} to these outcomes then we get the Borda score.

```
score = SequentialIR[lis, players]
```

$$\begin{pmatrix} 2 & 1 & 0 \\ 2 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 2 & 0 \\ 1 & 1 & 1 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

We can determine the expected score for each player from the probability matrix for the duels.

- Assume some competence ratings.

```
comp = {110, 95, 100};
```

```
pmat = CreateIR[comp, comp, Slope -> 0.1]
```

$$\begin{pmatrix} 0.5 & 0.817574 & 0.731059 \\ 0.182426 & 0.5 & 0.377541 \\ 0.268941 & 0.622459 & 0.5 \end{pmatrix}$$

- A symbolic matrix is useful for replacement. Since Pr has the Attribute Orderless, we use another symbol, say P, for the symbolic matrix.

```
PrMat = Table[P[i, j], {i, 3}, {j, 3}];
```

```
repl = Flatten[MapThread[Rule, {PrMat, pmat}, 2]]
```

```
{P(1, 1) -> 0.5, P(1, 2) -> 0.817574, P(1, 3) -> 0.731059, P(2, 1) -> 0.182426,  
P(2, 2) -> 0.5, P(2, 3) -> 0.377541, P(3, 1) -> 0.268941, P(3, 2) -> 0.622459, P(3, 3) -> 0.5}
```

- Replacement now gives the numerical probabilities of the 8 possible outcomes of the tournament.

```
Nprs = prs /. Pr -> P /. repl
```

```
{0.225654, 0.372041, 0.0830135, 0.136866, 0.0503502, 0.0830135, 0.0185228, 0.030539}
```

- The expected scores for the three players thus are:

```
ExpScore = Plus @@ (Nprs score)
```

```
{1.54863, 0.559966, 0.891401}
```


- Which fits the levels of competence that we assigned earlier.

`ListToPref[%]`

`Pref(B, C, A)`

Note that the Expected Score, computed as above, would be a generally acceptable way to calculate the ranking in general. We can do it also for matches that do not satisfy the Rasch - Elo assumptions. In that sense, the Rasch - Elo model has only a limited contribution, which consists of explaining the phenomenon of multiplicative odds by common factors - with the apt explanation of giving a level of competence.

If we want to amend the above for games in which ties can also occur within single matches, then we would get a trinomial model, and a larger tree.

<code>SequentialIR[{A, B, C, ...}]</code>	constructs the sequential binomial tree by pairwise matches with outcomes $A > B$ or $B > A$, etc. Best use Strings, to prevent evaluation of the $>$.
<code>SequentialIR[Simplify, x_List]</code>	makes the outcome $x =$ <code>SequentialIR[c]</code> better readable.
<code>SequentialIR[x_List, c_List]</code>	gives the scores, 1 = win, 0 = lose (recognising only $>$)
<code>SequentialIR[Pr, {A, B, C, ...}]</code>	gives the associated probabilities (using indices)

7.8 The direct approach

7.8.1 Using Pr rather than Logistic[d]

When students are graded then they don't get a rating, but they get a grade on the scale from 0 till 100, which grade effectively gives the percentage of correct answers. Indeed, such a percentages can be more informative than a separate *rating* - for which we would need to estimate the Rasch - Elo model. The discussion above has shown that we can do *without* an explicit rating, for a large set of (implicit) models. We could just give the probability of winning, for known opponents when these are known or against an unspecified average opponent (which would give a 'grade point average').

If the Rasch - Elo model is valid, then it is useful (and required) that there is a well defined statistical model, and it is useful to know that the odds are multiplicative (namely, because of the competencies involved). But, if the model is valid, then we can do without the ratings and unknown parameters, and just work with the probabilities.

A probability is on a ratio scale, with an obvious zero. Competence and difficulty of questions can be on an interval scale, since only the difference matters. The Rasch - Elo model allows us to go from an interval scale to a ratio scale: and that is a paradox, since an interval scale cannot do more than it is. The solution to this paradox is straightforward: Basic are the probabilities that are on a ratio scale, and the Logistic transforms these into differences. Apparently *we ourselves* reduce our information by accepting interval scales for competence and difficulty. If we stick to the probabilities, then we can keep using the ratio scale.

7.8.2 Derivation from the Logistic

Above relationships on the multiplicative odds already showed that we can work with probabilities only. However, for full clarity, we can also use the Logistic relationship to eliminate the unknown parameters. The simple Logistic model gives these three equations, with $\text{Pr}[i, j]$ the probability that i wins from j , and with r_i the ratings.

- These are the IRT equations, with $p = \text{Pr}[1, m]$ and $q = \text{Pr}[2, m]$.

```
eqs = {
  Pr[1, 2] == Logistic[r1 - r2, Slope -> c],
  p == Logistic[r1 - rm, Slope -> c],
  q == Logistic[r2 - rm, Slope -> c]}
```

$$\left\{ \text{Pr}(1, 2) == \frac{1}{1 + e^{-c(r1-r2)}}, p == \frac{1}{1 + e^{-c(r1-rm)}}, q == \frac{1}{1 + e^{-c(r2-rm)}} \right\}$$

- This gives the inverses.

```
sol = Solve[Rest[eqs], {r1, r2}]
```

$$\left\{ \left\{ r1 \rightarrow \frac{c \, rm - \log\left(-\frac{p-1}{p}\right)}{c}, r2 \rightarrow \frac{c \, rm - \log\left(-\frac{q-1}{q}\right)}{c} \right\} \right\}$$

- This eliminates the unknown c and rm .

```
eqs[[1]] /. sol[[1]] // Simplify
```

$$\text{Pr}(1, 2) == \frac{p - p \, q}{-2 \, q \, p + p + q}$$

7.8.3 For estimation

If we have a matrix of winning probabilities, then the question arises whether the Rasch - Elo model applies. We can check whether the odds are multiplicative or not, and accept or reject the model. It is another question whether we can find a set of probabilities that satisfies the model and that is as close to the observations as possible. The latter requires a clear concept of closeness. We can take the error with respect to the probabilities, with respect to the odds, or with respect to the Logit (the logarithm of the odds). Doing the latter actually means that we also estimate the implied competence ratings, and the error on those could be assumed to be normally distributed.

Estimation for the *p-only* approach would run as follows. We have only observations $p_{i,j} = \text{Pr}[i, j]$, and not the games with the 'common' opponent. When each players has an unknown $p_i = \text{Pr}[i, m]$ of winning from this common opponent m , we find errors $e_{i,j} = p_{i,j} - \text{WinPr}[p_i, p_j]$. For 3 players we have 3 unknown parameters and only two independent equations. For more players we can consider taking those parameters that minimize the sum of squared errors.

However, there appears to exist some practice in econometrics to estimate Logit models rather than p 's directly - see Theil (1971). The reason is that the Logit has values from $-\infty$ to $+\infty$, so that extreme values are easier to deal with. It appears that we could use this approach indeed, while the estimated p 's remain independent of the slope of the logistic.

7.8.4 Estimating matches

Let us regard the following match between 3 people (generated from an IR matrix):

```
prs = IRToMatchPr[{ {1, 0, 1, 1}, {0, 0, 1, 0}, {1, 1, 1, 1} }]
```

$$\begin{pmatrix} \frac{1}{2} & \frac{3}{4} & \frac{3}{8} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{8} \\ \frac{5}{8} & \frac{7}{8} & \frac{1}{2} \end{pmatrix}$$

We only have the winning probabilities, and the three ratings are unknown. It appears that we can estimate those ratings. More informative are the probabilities to win against the 'average' opponent

- RatingP indicates the competence ratings, Pr the probability of winning against the average opponent.

```
est1 = MatchPrToRating[prs]
```

```
{SSE → 0.0377379, RatingP → {134.036, -76.2959, 242.26},
 Pr → {0.548826, 0.266034, 0.694005}, Slope →  $\frac{\log(10)}{400}$ }
```

- An alternative consists of just taking the average probabilities (correcting for the diagonal of $1/2$'s). The results of the more complex Logit estimate might not be impressive.

```
AverageMatchPr[prs] // N
```

```
{0.5625, 0.1875, 0.75}
```

<code>MatchPrToRating[<i>pmat_List</i>, <i>opts</i>]</code>	estimates the pairwise match probabilities <i>pmat</i> into ratings and standard probabilities. Given Slope \rightarrow <i>parm_Symbol</i> for estimation of the slope
<code>AverageMatchPr[<i>pmat_List</i>]</code>	gives the average match probabilities, neglecting the diagonal 1/2 s

Uses Options[Heuristic|R] and puts its Slope option into the Logistic function.

- The estimated probabilities are:

`Results[MatchPrToRating]`

$$\begin{pmatrix} 0.5 & 0.770435 & 0.349103 \\ 0.229565 & 0.5 & 0.137792 \\ 0.650897 & 0.862208 & 0.5 \end{pmatrix}$$

- Note that this is consistent with using WinPr on the duels:

`WinPr[Pr[1], Pr[2]] /. Thread[Array[Pr, 3] \rightarrow (Pr /. est1)]`

0.770435

- The errors are:

`prs - Results[MatchPrToRating]`

$$\begin{pmatrix} 0. & -0.0204351 & 0.0258967 \\ 0.0204351 & 0. & -0.0127918 \\ -0.0258967 & 0.0127918 & 0. \end{pmatrix}$$

Above estimation assumed the slope used in Elo rating for Chess games. We can also estimate the slope, by setting it to a non-numerical value. Note that the slope and ratings change a lot while the SSE and probabilities only change slightly. Relevant however are the differences between the ratings, and we find that the implied probabilities of winning against the average opponent have not really changed.

`est2 = MatchPrToRating[prs, Slope \rightarrow q]`

{SSE \rightarrow 0.0377379, RatingP \rightarrow {100.272, 98.5833, 101.142},
Pr \rightarrow {0.548794, 0.266038, 0.694028}, Slope \rightarrow 0.716718}

`errs = prs - Results[MatchPrToRating]`

$$\begin{pmatrix} 0. & -0.0204081 & 0.0259497 \\ 0.0204081 & 0. & -0.0127823 \\ -0.0259497 & 0.0127823 & 0. \end{pmatrix}$$

Technical note: These are the default parameters:

Options[HeuristicIR]

$\left\{ \text{Average} \rightarrow 100, D \rightarrow \text{False}, \text{Factor} \rightarrow \text{Automatic}, \right.$
 $\left. \text{Pr} \rightarrow \text{False}, \text{Slope} \rightarrow \frac{\log(10)}{400}, \text{StartValues} \rightarrow \text{Automatic}, \text{Weight} \rightarrow \text{False} \right\}$

Estimation uses the value of Average for normalisation; its value gives only a psychological reference point, since what matters are the differences in ratings. See Theil (1971:636) on the meaning of the Weight option (including asymptotic standard error weights). The Factor option causes 'conditioning', so that the routine is sufficiently sensitive in the relevant digits. The D option is a remnant of the programming history: It puts the normal equations into Results[MatchPrToRating, D] - and at one occasion it seemed useful to have these.

7.8.5 Application to voting

Consider the voting case that we encountered in section 4.5.5. Let us reproduce it and calculate the vote matrix and estimated ratings.

EqualVotes[]; DefaultItems[];

SetPreferences[{{4, 3, 2, 1},
 {4, 3, 2, 1},
 {4, 3, 2, 1},
 {1, 4, 2, 3},
 {1, 4, 2, 3}}];

v = VoteMatrix[]

$$\begin{pmatrix} 0 & \frac{3}{5} & \frac{3}{5} & \frac{3}{5} \\ \frac{2}{5} & 0 & 1 & 1 \\ \frac{2}{5} & 0 & 0 & \frac{3}{5} \\ \frac{2}{5} & 0 & \frac{2}{5} & 0 \end{pmatrix}$$

estv = MatchPrToRating[v]

$\left\{ \text{SSE} \rightarrow 66.3558, \text{RatingP} \rightarrow \{152.827, 1082.39, -400., -435.218\}, \right.$
 $\left. \text{Pr} \rightarrow \{0.575444, 0.996513, 0.0532403, 0.0438994\}, \text{Slope} \rightarrow \frac{\log(10)}{400} \right\}$

- The estimated probabilities have the same order as the average ones. (The latter now have a systematic error because the diagonal is 0.)

```
AverageMatchPr[v] // N
{0.433333, 0.633333, 0.166667, 0.1}
```

- The implied aggregate preference ordering.

```
ListToPref[RatingP /. estv]
Pref(D, C, A, B)
```

We conclude that the Rasch - Elo rating, as implemented here, falls into the same trap as Borda, by selecting *B* rather than *A*. In fact, we could say that Rasch - Elo is not insensitive to preference reversal as well. If we would disregard the weak players *C* and *D*, then the real match would be between *A* and *B* only, and clearly *A* then has a higher probability of winning and thus a higher rating.

Note, though, from the SSE, that the matrix does not quite satisfy the Rasch - Elo conditions.

- The SSE is high, and these are errors for the probabilities. (Neglect the diagonal again.)

```
v - Results[MatchPrToRating]
( -0.5      0.595279   -0.360164   -0.367234 )
( -0.595279 -0.5      0.000196762  0.000160661 )
( 0.360164 -0.000196762 -0.5      0.0494897 )
( 0.367234 -0.000160661 -0.0494897 -0.5 )
```

The SSE in itself does not form an objection to the point that we want to emphasise now. The estimated matrix satisfies the Rasch - Elo conditions, while it still is sensitive to preference reversal. So the point of such sensitivity has been established, and the SSE of this particular case is not relevant.

7.9 Conclusion for voting

7.9.1 Impression on ranking

The impression is that the Rasch - Elo rating will not quickly overturn the ranking given by the row sums of the matrix of vote shares (though this needs further research). Nevertheless, we find ourselves back on the familiar ground where we already had formed some ideas about the relevance of such row sums.

Consider the example from section 4.7.4.

- Set the items, votes and preferences.

```
DefaultItems[3]; Votes = {.26, .26, .48};
SetPreferences[{{3, 2, 1}, {3, 2, 1}, {1, 3, 2}}];
```

- The binary method gives *A* as the winner, but the count *B*.

```
PairwiseMajority[]
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & 0.04 & 0.04 \\ -0.04 & 0 & 1. \\ -0.04 & -1. & 0 \end{pmatrix} \right\}$$

$1 \rightarrow \{\text{StatusQuo} \rightarrow A, \text{Sum} \rightarrow \{2, 1, 0\}, \text{Max} \rightarrow 2, \text{Condorcet winner} \rightarrow A,$
 $\text{Pref} \rightarrow \text{Pref}(C, B, A), \text{Find} \rightarrow A, \text{LastCycleTest} \rightarrow \text{False}, \text{Select} \rightarrow A\},$
 $N \rightarrow \{\text{Sum} \rightarrow \{0.08, 0.96, -1.04\}, \text{Pref} \rightarrow \text{Pref}(C, A, B), \text{Select} \rightarrow B\}, \text{All} \rightarrow A\}$

- BordaFP finds the Plurality and the Condorcet winner.

```
BordaFP[]
```

BordaFP::chg : Borda gave {B}, Fixed Point is {A}

A

- Rasch - Elo rating follows the count.

```
v = VoteMatrix[]
```

$$\begin{pmatrix} 0 & 0.52 & 0.52 \\ 0.48 & 0 & 1. \\ 0.48 & 0 & 0 \end{pmatrix}$$


```

estv = MatchPrToRating[v]

{SSE → 44.1824, RatingP → {109.27, 762.031, -571.301},
 Pr → {0.513337, 0.978352, 0.0205462}, Slope →  $\frac{\log(10)}{400}$ }

ListToPref[RatingP /. estv]

Pref(C, A, B)

```

7.9.2 Satisfying the assumptions

We have seen some cases now where the vote matrix does not satisfy the Rasch - Elo assumptions of multiplicative odds. The estimating routine finds the Rasch - Elo probabilities that are as close as possible, but the SSE can be large. It would be unsatisfactory to impose something that does not really fit and still draw strong conclusions. This is something to keep in mind.

7.9.3 The paradoxes

Matches appear to have real paradoxes in that an aggregate measure for 'the best player' appears to depend upon who is included in the tournament. If some weak players are included in the tournament, then *B* is selected, and if these are eliminated (for example by a first round against this *B*, then *A* is selected (thanking *B*). Conclusions need not be strong here, since we discuss probabilities, but they are true in terms of expectations.

Matches also have 'paradoxes'. We can put the word between quotes since this category concerns issues that are not even seeming contradictions but obvious trivialities. It is a plain banality that the winner of a soccer cup can have lost one of the matches of the tournament. The world chess champion may have lost a game from some national champion so that the victory is not unblemished. These 'paradoxes' are related to the paradoxes of voting indeed, and not just by analogy but by structure. We tend to accept such 'paradoxes' for matches, and yet, for voting, this suddenly is called a real paradox. The only reason can be that some people at some time took a wrong frame of reference, and started to expect more from voting than it can do.

For matches, we concentrate on having proper rules of the game. We should do the same for voting. The 'paradoxes' of matches clarify that the paradoxes of voting are much less of a problem than often thought. For voting we can reasonably accept that voting results can change when the budget changes. It is

also a rather generally accepted moral rule that we don't neglect realities, so that we should not pursue impossibilities. For matches we already show a great deal of realism. For matches, we don't mind that new people show up every year. So, why would this, suddenly, create 'paradoxes' for voting ? For matches, we don't wait till the end of time, so why should we do so for voting ?

7.9.4 Evaluation

We finally turn to the question whether the Rasch - Elo rating is principally acceptable as a tool to determine the winner of a vote.

It is a serious question whether the Rasch - Elo model applies to chess. It is an even more serious question whether it applies to voting. While the Rasch - Elo rating for candidates indeed is a summary statistic that can be derived from the voting shares (with or without a large SSE), there remains the doubt whether it really reflects something that we can interpret as 'competence'. A notion of 'competence for office' or 'suitability to lead the country' can quickly be rejected, since every voter will have a different idea for each candidate about the level of such competence. However, a measure that might survive our critique is the notion of 'convincing power' or 'electoral appeal'. The reason that this could survive is that this is rather what voting is about.

For this, let us better develop the model. A vote, cast by x , on A vs B seems to be quite something else than a match between A and B . Let us however consider the matches x vs A and x vs B too. There is a way in which we can compare a presidential election with a huge chess tournament. Let us assume that everyone in the country has a private pairwise meeting with everybody else in the country. Each pairwise meeting is a match in which the pair must decide whether they vote each for themselves or for only one of them. Each person has a certain level of convincing power or electoral appeal. Convincing power then is defined as follows: If each has the same level of convincing power, then they are equally likely to get the vote. If one person has more convincing power, then he or she is likelier to get the vote. The Logistic assumption is a good prior hypothesis for the effect on the probabilities. This setup might well be accepted as a serious match in terms of convincing power and electoral appeal. The person of the pair who gets the vote shows a competence in getting the vote. If we aggregate these vote results for the whole population then there appear to be some people who got the confidence of

many other people. Normally we concentrate on those people only, but above pairwise structure could be said to exist *in abstracto* for the whole country.

As a point of theory, we can wonder whether people really adjust the likelihood of their vote according to the differences in 'convincing power'. (1) A person who has not made up his or her mind indeed might toss a coin. Yet it is more likely that many already have a set mind. (2) We can accept that there exists something like 'public opinion', so that people can adjust their views influenced by friends and perceived popularity, but it is a strong step to make 'convincing power' into the *only* thing that matters. However, (ad (1) and (2)), if we define 'convincing' power as nothing else but the phenomenon contained in getting the vote, then the issue becomes tautologous. If a person has a fixed opinion, and seems insensitive to the level of convincing power, then we analyse this as the result of the convincing that already has taken place. To see the matter in this tautologous way seems proper, and it allows the Rasch - Elo model to survive for voting.

We can complete the model by giving a *statistical* interpretation to the match probabilities. The matrix of voting shares then is interpreted as giving the probabilities when we take a sample of the population. A value $1/2$ in the matrix means that if we ask a sample of voters on their views, then half of the sample will vote for one candidate and half for the other. Similarly for the other probabilities. A probability thus does not mean that a voter flips a coin or casts a die. The resulting Rasch - Elo ratings then are proper summary statistics that describe the data. They do not have obvious meanings like the mean or the variance, but the interpretation would be warranted that they reflect the competence of getting votes, since that is what the data are about.

The conclusion would be that the Rasch - Elo model could be as suitable for voting as it is for chess. (Perhaps more so, given the probabilities of having a tie.)

Note that if we have a matrix of vote results, then the Rasch - Elo model implies multiplicative odds, and this is something that we can test. So we are not dependent upon that model to explain voting. As said, it is a serious question whether the Rasch - Elo model applies to chess, because of the ties, yet even then we can still accept its use when the error is not too large. The model is particularly useful for chess since two players who have not met before can forecast their probabilities of winning when they know each other's ratings. This feature might be less relevant for voting, but also this is something to look

into, since candidates for higher office often have run for lower office first, and they thus have shown their competence at winning votes before - and they meet other candidates just like that.

Admittedly, these are preliminary conclusions since the model looked at here is simple and there is little experience with this approach. Given that nations will not quickly adopt Rasch - Elo ratings for presidential candidates, there is enough time however to further investigate the issue.

7.10 Appendix

The statistical packages of *Mathematica* also have the logistic distribution.

- Note that the constant must be entered with an inverse value.

`?LogisticDistribution`

`LogisticDistribution[mu, beta]` represents the logistic distribution with mean μ and scale parameter β .

`PDF[LogisticDistribution[μ , $1/\beta$], x]`

$$\frac{e^{-\beta(x-\mu)} \beta}{(1 + e^{-\beta(x-\mu)})^2}$$

- *Mathematica* makes some properties directly available.

`Mean[LogisticDistribution[μ , $1/\beta$]]`

$$\mu$$

`StandardDeviation[LogisticDistribution[μ , $1/\beta$]]`

$$\frac{\pi}{\sqrt{3} \beta}$$

- We can get an expression with an explicit standard deviation:

`StandardDeviation[LogisticDistribution[μ , $\sigma \text{ Sqrt}[3]/\text{Pi}$]]`

$$\sigma$$

`PDF[LogisticDistribution[μ , $\sigma \text{ Sqrt}[3]/\text{Pi}$], x]`

$$\frac{e^{-\frac{\pi(x-\mu)}{\sqrt{3} \sigma \pi}}}{\sqrt{3} \left(1 + e^{-\frac{\pi(x-\mu)}{\sqrt{3} \sigma \pi}}\right)^2 \sigma}$$

8. Measuring utility

8.1 Introduction

8.1.1 Introduction

The situation thus is as follows:

- Since there are no obvious objective measures for cardinal utility, we have to ask people for their opinion; since people can cheat, we try to limit their options; and then we end up with the paradoxes of voting.
- Above chapters have shown that the paradoxes of voting only *seem* contradictions but are no real contradictions. Which means that we can live with them.

Yet, it could be fruitful to work into the other direction, and to see whether we could develop more acceptable measures for cardinal utility.

The economic literature contains the suggestion, in some important places, that such a measure might be found by experiments in which probability plays a role. This brings us to the subjects of Prospects and Certainty Equivalence. Above, we already looked at games and matches, but we have somewhat neglected the question of the Prize of the match. It was just win or lose, without much utility attached to it. The following deals better with that by including the utilities attached to Profit and Loss.

The discussion below will show, unfortunately, that recovering cardinal utility is still no easy feat. Sometimes people risk their lives to save others, but economic theory still cannot say that there is cardinal comparison in this. So the main conclusion of the following chapters is negative. It does not seem possible, yet, to determine cardinal utility, free of cheating. Above schemes of voting hence cannot be replaced by simply adding (or Nash multiplying) of such utilities. The positive side of this conclusion is that above chapters are not useless.

ResetAll

Economics[Probability, Risk]

Cool`Probability`

Bayes	NumberOfElements
Bordered2DPrSolve	NumberOfFailures
Bordered2DPrToConditionals	Odds
Bordered2DPrToEquations	Options\$SequentialDraws
ConditionalPrForm	OrRule
CPrToFunction	PrDomain
CreateProspect	Prob
Draw	Prospect
FromConditional	ProspectApply
FromConditionalRule	ProspectEV
FromOdds	ProspectInnerSort
FromProb	ProspectQ
IDProspectsPrMatrix	ProspectReList
JoinIDProspects	ProspectUtility
JointProspect	PutIn
JointProspectEV	SequentialDraws
JointProspectPrValue	TakeOut
JointToMarginalProspects	ToConditional
JointToProspects	ToConditionalRule
Laplace	ToExpectedUtility
LevelToPr	ToProb
LevelToSum	ToUtility
NormalisePr	UnitPr

Cool`Risk`

AllaisParadox	RiskAversion
BinaryRiskSimplify	RiskEquations
CertaintyEq	Risket
ConditionalRisk	RiskFactors
IfNegativeToMaxRule	RiskModel
IfNegativeToMinRule	RiskPr
Options\$CertaintyEq	RiskPremium
Prospect2DSimplify	RiskRatio
Prospect3DPrTriangle	RiskRatioSplit
ProspectInverse	RiskRatioSplitPlot
ProspectPlot	Risks
ProspectProjection	RiskyQ
ProspectPrValue	SpreadAversion
ProspectSort	SpreadPremium
ProspectSplit	ToProspect
ProspectSplitPlot	ToRisket
ProspectStatistics	Uncertainty
ProspectTo3D	UncertaintyAversion
RelativeRisk	UncertaintyDefinitionsPlot
RelativeRiskEquations	UncertaintyPremium
Risk	

8.1.2 Structure of the discussion

We have to develop notions of probability and risk before we can deal with certainty equivalence. The definition of risk is mine, and I also develop a new (non-standard) model for certainty equivalence, using this risk measure. The *Mathematica* routines provide an accessible way to verify that these new approaches are sound. We will take examples from Luenberger (1998) and Mas-Colell c.s. (1995:171) so that it is also quickly verified that the routines are sound.

If you are interested in extending on the issues in the following chapters, then you should be aware that Cool (1999, 2001), "The Economics Pack", contains a large number of additional routines on probability and risk that we will not discuss here. You are advised to first look into that Pack before you start programming, since this can save you a lot of work.

8.2 Development of probability

This package implements basic probability theory. The Prospect object allows easy handling of discrete probability situations, and Draw helps you to find a proper playing strategy.

`Economics[Probability]`

8.2.1 The Pr object

With $\text{Pr}[A]$ the probability of event A , we find for two events A and B :

- the joint probability $\text{Pr}[A, B] = \text{Pr}[A \& B] = \text{Pr}[A \cap B]$
- $\text{Pr}[A \text{ or } B] = \text{Pr}[A \cup B] = \text{Pr}[A] + \text{Pr}[B] - \text{Pr}[A \cap B]$
- the conditional probability of A given B : $\text{Pr}[A \mid B] = \text{Pr}[A, B] / \text{Pr}[B]$
- the probability of the ordered event of first A and then B is $\text{Pr}[\{A, B\}]$

Since *Mathematica* has different uses for semicolon ";" and the vertical line "|", we represent conditional probability as $\text{ConditionalPr}[A][B]$. We can use $\text{Prob}[A \mid B]$ as an output printing facility however.

Note that some textbooks interpret "either A or B " as inclusive-or. Here we will use normal English where it is exclusive-or.

<code>Pr[x__]</code>	gives the joint probability of events x
<code>ConditionalPr[x__][y__]</code>	gives the conditional probability of events x given events y
<code>Prob[x__]</code>	is a format for probability that humans can read better, but it is less tractible for <i>Mathematica</i>
<code>ToProb</code>	rules to turn $\text{Pr}[\dots]$ into $\text{Prob}[\dots]$ format
<code>FromProb</code>	rules to turn $\text{Prob}[\dots]$ into $\text{Pr}[\dots]$ format
<code>ConditionalPrForm[x]</code>	control TraditionalForm printing. $x = \text{On, Off or Blank}$

Users may, alternatively, opt to use Prob as the basic function, then use FromProb to get to the structural form that the routines recognise, and then apply ToProb again.

8.2.2 Bayes

`Bayes[A, B] [x___Rule]` is a `SolveFrom` application, using the equations
 $\Pr[A, B] == \Pr[B] \text{ ConditionalPr}[A][B]$
 $\Pr[A, B] == \Pr[A] \text{ ConditionalPr}[B][A]$

```
Bayes[A, B] [Pr[A, B] → .15, Pr[A] → .2, Pr[B] → .3] // Last

{{ConditionalPr[A][B] → 0.5, ConditionalPr[B][A] → 0.75}}

% /. ToProb

{{Prob(A | B) → 0.5, Prob(B | A) → 0.75}}

Bayes[A, B] [ConditionalPr[A][B] → .5, Pr[A] → .2, Pr[B] → .3] // Last

{{Pr(A, B) → 0.15, ConditionalPr[B][A] → 0.75}}

% /. ToProb

{{Prob(A, B) → 0.15, Prob(B | A) → 0.75}}
```

Using matrices is often more instructive. A bordered 2D probability matrix is a $\{n, m\}$ matrix of which the last row (column) is the sum of the preceding rows (columns).

- Give the minimal information, and let *Mathematica* find the rest. Let A be the first column, $\neg A$ the second column, B the first row, $\neg B$ the second row.

```
.15 □ .3
pmat = □ □ □ ;
.2 □ □
```

```
Bordered2DPrSolve[pmat]
```

$$\begin{pmatrix} 0.15 & 0.15 & 0.3 \\ 0.05 & 0.65 & 0.7 \\ 0.2 & 0.8 & 1 \end{pmatrix}$$

```
Bordered2DPrToConditionals[%]
```

$$\left\{ \text{Row} \rightarrow \begin{pmatrix} 0.75 & 0.1875 & 0.3 \\ 0.25 & 0.8125 & 0.7 \\ 1. & 1. & 1 \end{pmatrix}, \text{Column} \rightarrow \begin{pmatrix} 0.5 & 0.5 & 1. \\ 0.0714286 & 0.928571 & 1. \\ 0.2 & 0.8 & 1 \end{pmatrix} \right\}$$

```

Bordered2DPrSolve[x_?MatrixQ, X_Symbol:XYZX]      solves □

Bordered2DPrToConditionals[x_?MatrixQ]

Bordered2DPrToEquations[x_?MatrixQ]

      subroutines, for inner cell conditional probabilities and equations

```

8.2.3 Pr is orderless

Pr has the Attribute Orderless.

- This is as it should be.

```
Pr[A, B] === Pr[B, A]
```

```
True
```

Ordered probabilities are denoted with lists, so that $\text{Pr}[\{A, B\}]$ gives the probability of the ordered sequence $\{A, B\}$. (This is much better than $\text{ClearAttributes}[\text{Pr}, \text{Orderless}]$.)

Be aware of the subtleties of order. Traditional notation is awkward here. The probability of first a black card and then a red card $\text{Pr}[\{B, R\}] = \text{Pr}[B] \text{Pr}[R \mid \{B\}]$ differs from the probability of just a black and red card $\text{Pr}[\{B, R\} \text{ or } \{R, B\}] = \text{Pr}[\{B, R\}] + \text{Pr}[\{R, B\}]$. And the latter differs from the probability $\text{Pr}[B, R]$, since the latter need not have anything to do with sequential draws. By using the $\{\}$ in the conditional part, as in $\text{Pr}[R \mid \{B\}]$, we can express that sequential draws are at hand.

An example can help. Let the universe have n elements, with r elements in R , b elements in B and m elements in the intersection of R and B . Then:

$$\text{Pr}[R] = r/n, \quad \text{Pr}[B] = b/n, \quad \text{Pr}[R, B] = m/n, \quad \text{Pr}[R \mid B] = m/b$$

If the events are independent then $m = 0$ (as would happen with black and red cards):

$$\text{Pr}[\{B, R\}] = \text{Pr}[B] \text{Pr}[R \mid \{B\}] = \frac{b}{n} \frac{r}{n-1}$$

$$\text{Pr}[\{R, B\}] = \text{Pr}[R] \text{Pr}[B \mid \{R\}] = \frac{r}{n} \frac{b}{n-1}$$

The sum $P[\{B, R\}] + \Pr[\{R, B\}]$ clearly may differ from $\Pr[R, B] = 0$.

- In the special case that $n = b + r$, this sum is:

$$\frac{b r}{n (n - 1)} + \frac{r b}{n (n - 1)} \quad /. \quad b \rightarrow n - r \quad // \quad \text{Simplify}$$

$$\frac{2 (n - r) r}{(n - 1) n}$$

- This is actually from the hypergeometric distribution.

$$\text{Binomial}[n - r, 1] \text{Binomial}[r, 1] / \text{Binomial}[n, 2]$$

$$\frac{2 (n - r) r}{(n - 1) n}$$

If the events are dependent then $m \neq 0$ (as would happen with black cards and picture cards):

$$\Pr[\{B, R\}] = \Pr[B] \Pr[R \mid \{B\}] = \frac{b}{n} \left(\frac{m}{b} \frac{r-1}{n-1} + \frac{b-m}{b} \frac{r}{n-1} \right)$$

$$\Pr[\{R, B\}] = \Pr[R] \Pr[B \mid \{R\}] = \frac{r}{n} \left(\frac{m}{r} \frac{b-1}{n-1} + \frac{r-m}{r} \frac{b}{n-1} \right)$$

Taking that example for B = black cards, and R = picture cards indeed, then $n = 52$, $b = 26$, $r = 12$, and $m = 6$ (black picture cards).

$$\Pr[\{B, R\}] = \Pr[B] \Pr[R \mid \{B\}] = \frac{26}{52} \left(\frac{6}{26} \frac{11}{51} + \frac{20}{26} \frac{12}{51} \right)$$

$$\Pr[\{R, B\}] = \Pr[R] \Pr[B \mid \{R\}] = \frac{12}{52} \left(\frac{6}{12} \frac{25}{51} + \frac{6}{12} \frac{26}{51} \right)$$

- Adding these.

$$\frac{b}{n} \left(\frac{m}{b} \frac{r-1}{n-1} + \frac{b-m}{b} \frac{r}{n-1} \right) + \frac{r}{n} \left(\frac{m}{r} \frac{b-1}{n-1} + \frac{r-m}{r} \frac{b}{n-1} \right) \quad // \text{Simplify}$$

$$\frac{2 m - 2 b r}{n - n^2}$$

8.2.4 Prospects

A prospect is an object that collects both the events that can occur and the probabilities that they occur. A binary prospect recognises only two events; if the probability of the first is p , then the probability of the other is $(1 - p)$. The multidimensional prospect generalises from this. Only lists have been implemented here, not continuous variables.

<code>Prospect[event1, event2, Pr[event1]]</code>	a binary prospect
<code>Prospect[x_List, p_List]</code>	is an object with values x and associated probabilities p
<code>ProspectQ[q]</code>	tests whether q is a prospect
<code>ProspectEV[x_Prospect]</code>	gives the expected value of prospect x

The function `Spread` recognises Prospects too.

```
ex1 = Prospect[a, b, p];

ProspectEV[ex1]

$$b(1 - p) + a p$$


ex2 = Prospect[{c, d, e}, {.3, .3, .4}];

ProspectEV[%]

$$0.3 c + 0.3 d + 0.4 e$$

```

8.2.5 Valueing prospects

If we want to be able to compare states of the world or add them, then they must have the same dimensions or be valued to a same dimension. If the dimensions are not the same then we use money or utility.

We use `Utility` to stand for non-stochastic utility that evaluates certain states of the world, and `ProspectUtility` for stochastic utility that for example weighs expected value and spread.

<code>CreateProspect[n_Integer]</code>	creates a prospect with n states and probabilities
<code>CreateProspect[Price, n_Integer]</code>	idem with n priced states and probabilities
<code>CreateProspect[Utility, n_Integer]</code>	idem with n utility states and probabilities
<code>ToUtility[q_Prospect]</code>	makes a Von Neumann – Morgenstern prospect with utilities of the outcomes
<code>ToExpectedUtility[q_Prospect]</code>	combines ToUtility and ProspectEV
<code>ProspectUtility[x_Prospect, crit__]</code>	gives the utility of prospect x based the criteria.

If the Risk package is loaded, then default criteria for ProspectUtility are ProspectEV, Spread and Risk.

- Though states can have different dimensions, evaluation requires that they must be valued by money or utility - or they must have the same dimension (like dimensionless rates of return).

```
sameDims = CreateProspect[2]
```

```
Prospect({State(1), State(2)}, {Pr(1), Pr(2)})
```

```
withMoney = CreateProspect[Price, 2]
```

```
Prospect({Price(1) State(1), Price(2) State(2)}, {Pr(1), Pr(2)})
```

```
withUtility = CreateProspect[Utility, 2]
```

```
Prospect({Utility(State(1)), Utility(State(2))}, {Pr(1), Pr(2)})
```

- The following might be conceptually dangerous since it means that we take the utilities of sums of money. This could be alright, though, if we were to use economic indices like "national income" - since those are constructed as (money) (chain) indices.

```
ToExpectedUtility[withMoney]
```

```
Pr(1) Utility(Price(1) State(1)) + Pr(2) Utility(Price(2) State(2))
```

- This is the Von Neumann - Morgenstern criterion without the latter money complexity.

```
ProspectEV[withUtility]
```

```
Pr(1) Utility(State(1)) + Pr(2) Utility(State(2))
```

- The following would be the utility evaluation of above example ex1 in the $\{\sigma, \mu\}$ space as is common in finance. If we limit our attention to dimensionless rates of return, then the addition is no problem.

```
ProspectUtility[ex1, Spread, ProspectEV] //Simplify
```

$$\text{Utility}\left(\sqrt{-(a-b)^2(p-1)p}, -pb + b + ap\right)$$

8.2.6 Subroutines

Before we continue with the interesting section 8.2.7, we should look at some of the routines that manipulate prospects. We will use these manipulations.

<code>ProspectReList[q]</code>	changes a binary prospect q into a prospect with lists
--------------------------------	--

```
ProspectReList[ex1]
```

```
Prospect({a, b}, {p, 1 - p})
```

<code>ProspectInnerSort[q_Prospect(, p)]</code>

sorts the states and keeps the probabilities right, with p an ordering function

```
ProspectInnerSort[Prospect[{x, q, a}, {.5, .1, .4}]]
```

```
Prospect({a, q, x}, {0.4, 0.1, 0.5})
```

<code>ProspectApply[f, x]</code>	applies f to Prospects in x
----------------------------------	-----------------------------

- Take an example expression.

```
expr = {{g}, func[σ → Prospect[1, 2, .3]]}
```

```
{{g}, func(σ → Prospect(1, 2, 0.3))}
```

- Apply a function f to the prospects in the expression.

```
ProspectApply[f, expr]
```

```
{{g}, func(σ → f(Prospect(1, 2, 0.3)))}
```

■ Take the Spread of prospects.

ProspectApply[Spread, expr]

{{g}}, func($\sigma \rightarrow 0.458258$)}

In later discussions it appears useful to 'put in' values or 'take out' values.

<code>PutIn[x+...+q_Prospect+...+y]</code>	for a single q includes the additions into the prospect
<code>PutIn[a_List, q_Prospect]</code>	takes in the strategy for playing the prospect
<code>TakeOut[q_Prospect, f]</code>	takes out a f[profit, loss]; f = Max takes out profits, f = Min takes out losses.
<code>TakeOut[q, Profit Loss]</code>	for simple prospects: uses positions

Prospect[10] + 56 + test

test + Prospect(10, 0, 1) + 56

PutIn[%]

Prospect(test + 66, test + 56, 1)

TakeOut[% , Profit]

test + Prospect(0, -10, 1) + 66

Prospect[{1, 2, 3, 4}, Laplace[4]]

Prospect({1, 2, 3, 4}, { $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$ })

TakeOut[% , Max]

Prospect({-3, -2, -1, 0}, { $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$ }) + 4

PutIn[%]

Prospect({1, 2, 3, 4}, { $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$ })

8.2.7 Random drawing from Prospects

8.2.7.1 The routine Draw

`Draw[]` is a small but powerful routine to actually draw randomly from a prospect. You can also define a strategy how much to bet at each turn.

- This is a single draw for a "50% win 10, lose 10" proposition. The result is random, and can change with each evaluation.

```
Draw[Prospect[10, -10, 1/2]]
```

```
-10
```

- This draws three times in a row.

```
Draw[3, Prospect[10, -10, 1/2]]
```

```
{-10, 10, 10}
```

The following example is taken from Luenberger (1998). Suppose that there is a *Wheel of Fortune* with areas of size $\{1/2, 1/3, 1/6\}$ that pay out 3, 2, or 6 times the bet on the respective area. A player puts amounts $\{a, b, c\}$ on the respective areas, and thus has a $\text{Prospect}[\{3a, 2b, 6c\}, \{1/2, 1/3, 1/6\}]$.

- Let us set a on area $1/2$, b on area $1/3$ and c on area $1/6$. Let us draw 10 times.

```
prp = Prospect[{3a, 2b, 6c}, {1/2, 1/3, 1/6}];
```

```
Draw[10, prp]
```

```
{3 a, 3 a, 3 a, 3 a, 3 a, 6 c, 3 a, 2 b, 2 b, 2 b}
```

- Here we draw 100 times. We then determine the frequencies (in this case percentages).

```
res = Draw[100, prp]; Frequencies[res]
```

```

$$\begin{pmatrix} 57 & 3a \\ 31 & 2b \\ 12 & 6c \end{pmatrix}$$

```

- For the expected return we must subtract the bet $a + b + c$.

```
ProspectEV[prp] - (a+b+c) // Simplify
```

```

$$\frac{1}{6} (3a - 2b)$$

```

From this, a maximiser of 'expected value' would erroneously conclude that he or she should play the game, and then bet only a on area $1/2$. You bet a , expect to win $3a/2$, and thus your expected winning is $a/2$ per turn. A goldmine ! Or do you see the catch ?

<code>Draw[q_Prospect]</code>	draws from the probability density and returns the drawn outcome.
<code>Draw[a_List, Prospect[v, p]]</code>	draws with strategy a. Part of wealth $(1 - \text{Add}[a])$ is retained, and $a * v$ are the adjusted rewards with probabilities p. Multiply the outcomes to get the wealth after so many tries
<code>Draw[]</code>	draws from the prospect or strategy given earlier
<code>Draw[n_Integer, x_____]</code>	draws n times

8.2.7.2 Strategy

There is something seriously wrong with 'expected value theory' as taught in many less advanced textbooks. Sometimes there is made a distinction between single games and repeated games, but a strong case can be made that a single game should better be regarded (then) as a 'repeated game for one step'. Whatever that be, Luenberger (1998) ch. 15 is essential for your ticket to wealth here. The problem with the 'expected value' approach is that it does not take account of the fact that you may lose all your money, and cannot play the game any more. The House, the owner of the 'wheel of fortune', here has the advantage.

Drawing can be done with a strategy however. Let the player decide on a budget for gambling, and then bet *proportions* $\{a, b, c\}$, while retaining $1 - a - b - c$ on the side. The prospect then becomes $\text{Prospect}[\{3a, 2b, 6c\}, \{1/2, 1/3, 1/6\}] + 1 - a - b - c$. The proportion is on the current wealth. The trick is that retaining a proportion of winnings can cause a bias towards growth. Part of the error in above approach is to assume that there is an unlimited budget such that after any string of losses there still would be $a + b + c$ available. If we define a, b and c as proportions, and take $a + b + c < 1$, then we make sure that something is available indeed. (Though, in practice, runs can go to millionths of cents, and thus the strategy is not always that practical.)

- Let us play the wheel with *proportions* $\{a, b, c\}$. The routine now returns the winnings plus the proportion that was kept on the side. This output can be interpreted as a proportion of the budget again. (Note: each evaluation can give another result.)

```
prop = Prospect[{3, 2, 6}, {1/2, 1/3, 1/6}];
```

```
Draw[{a, b, c}, prop]
```

```
2 a - b - c + 1
```

- Let us play the wheel for 3 times. Each outcome is a proportion of the budget created by the former outcome.

```
Draw[3, {a, b, c}, prop]
```

```
{2 a - b - c + 1, 2 a - b - c + 1, 2 a - b - c + 1}
```

- We can check that only three outcomes are possible.

```
res = Draw[100, {a, b, c}, prop]; Frequencies[res]
```

$$\begin{pmatrix} 51 & 2a - b - c + 1 \\ 30 & -a + b - c + 1 \\ 19 & -a - b + 5c + 1 \end{pmatrix}$$

If we subsequently *multiply* the outcomes, then we get the proportion in terms of the original budget. We then can also make a plot of the evolution of wealth over the turns. To do this, we need numerical values.

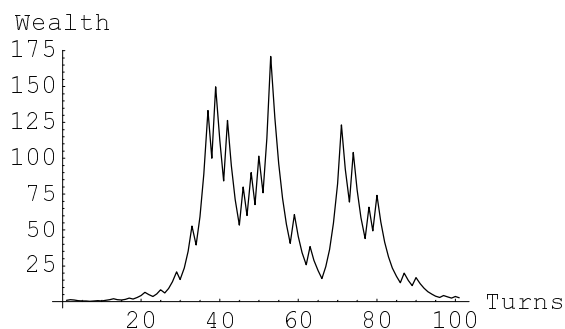
- Let us use a strategy $\{1/4, 0, 0\}$.

```
res = Draw[100, {1/4, 0, 0}, prop];
```

```
wealth = FoldList[Times, 1., res];
```

- This gives the evolution of wealth over time as the wheel turns. A string of wins causes wealth to increase, but a string of losses is possible too. Once the random walk gets in the low wealth range, it may be difficult to get out again because of the implied growth rate of the strategy (see below).

```
PlotLine[wealth, AxesLabel → {"Turns", "Wealth"}];
```



- We can take the $1/n$ root to determine the average growth rate.

```
N[Times @@ res]^(1/100)
```

```
1.01043
```

8.2.7.3 Optimal strategy

Let us find the optimal strategy. (Again I follow Luenberger (1998).)

- First define the formal prospect situation.

```
fp = PutIn[{a, b, c}, prop]
```

```
-a - b - c + Prospect[{3 a, 2 b, 6 c}, {1/2, 1/3, 1/6}] + 1
```

```
properfp = PutIn[fp]
```

```
Prospect[{(2 a - b - c + 1), (-a + b - c + 1), (-a - b + 5 c + 1)}, {1/2, 1/3, 1/6}]
```

- Take logarithmic utility to warrant the highest rate of growth

```
eu = ToExpectedUtility[properfp] /. Utility → Log
```

```
1/2 log(2 a - b - c + 1) + 1/3 log(-a + b - c + 1) + 1/6 log(-a - b + 5 c + 1)
```

- Setting up the first order conditions for a maximum.

```
Economics[Calculus, Print → False]
```

```
foc = Foc[eu, {a, b, c}]
```

$$\left\{ \begin{aligned} &-\frac{1}{3(-a+b-c+1)} - \frac{1}{6(-a-b+5c+1)} + \frac{1}{2a-b-c+1} == 0, \\ &\frac{1}{3(-a+b-c+1)} - \frac{1}{6(-a-b+5c+1)} - \frac{1}{2(2a-b-c+1)} == 0, \\ &-\frac{1}{3(-a+b-c+1)} + \frac{5}{6(-a-b+5c+1)} - \frac{1}{2(2a-b-c+1)} == 0 \end{aligned} \right\}$$

- We can show already that $\{1/4, 0, 0\}$ is not optimal.

```
foc /. Thread[{a, b, c} → {1/4, 0, 0}]
```

```
{True, False, False}
```

- Let us make sure that we are dealing with a probability measure.

```
pm = foc /. c → 1 - a - b;
```

- And this gives the optimal strategy.

```
Solve[pm, {a, b}]
```

$$\left\{ \left\{ a \rightarrow \frac{1}{2}, b \rightarrow \frac{1}{3} \right\} \right\}$$

- We can compare the single period expected values of the prospects. These also give the long run expectation for the average outcome.

```
ProspectEV[properfp] /. Thread[{a, b, c} → {1/4, 0, 0}]/N
```

```
1.125
```

```
ProspectEV[properfp] /. Thread[{a, b, c} → {1/2, 1/3, 1/6}]/N
```

```
1.13889
```

- Let us try this new strategy again. Well, bad luck ! (Growth was not as high as it could have been.)

```
res = Draw[100, {1/2, 1/3, 1/6}, prop];
```

```
N[Times @@ res]^(1/100)
```

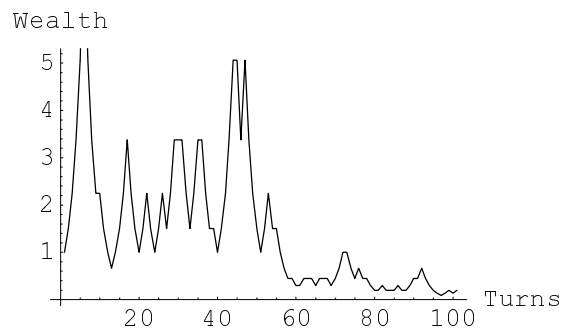
```
0.983912
```

Note that also the optimal strategy may result in long states of low wealth. There is no guarantee that one will have more than the original budget in one's lifetime, or, if it has grown to the sky, that it will remain there.

- Show the evolution of wealth for the above.

```
wealth = FoldList[Times, 1., res];
```

```
PlotLine[wealth, AxesLabel → {"Turns", "Wealth"}];
```



8.3 Risk

8.3.1 Prospects and risk

Prospects are perfect to continue the discussion on risk. The typical binary risky prospect recognises only win or lose situations. Let $profit \geq 0$ stand for the positive return of a prospect, and $-loss \leq 0$ for the negative return of a prospect, where loss is the absolute value of that negative return. The probability of a profit is p , the probability of a loss is $(1 - p)$. The multidimensional prospect generalises from this.

`Prospect[profit, -loss, Pr[Profit]]` a binary Prospect convention for risky situations

Note that loss is an absolute value, and that a real loss must be entered as a negative value.

- An example risky prospect.

```
eg = Prospect[Profit, -Loss, p];
```

```
ProspectEV[eg]
```

$$p \text{ Profit} - \text{Loss} (1 - p)$$

It is important to see that there is nothing in the concept of a Prospect that requires that the second position is reserved for a loss. A binary prospect may well give two profits or two losses. Therefore, the proper definition of Risk requires a formal test on the values of the entries. For formal discussions, however, such a test reads awkward, and we better resort to the positional convention.

- For formal analysis we resort to the convention that the second position is the loss. The Position option works only for binary risk (in non-list-format).

```
SetOptions[Risk, Position → True];
```

- The risk is the probable loss (the loss weighed by its probability).

```
Risk[eg]
```

$$\text{Loss} (1 - p)$$

`RiskyQ[eg]`

True

- The default option setting is `Position → False`, since it is not obvious that you would be using Prospects formally.

`SetOptions[Risk, Position → False];`

<code>Risk[q]</code>	gives the risk, i.e. the expected absolute loss
<code>RiskPr[q]</code>	gives the cumulative probability of a loss in prospect q
<code>RiskyQ[q]</code>	gives False if not risky, True if q is risky (at least one negative possible outcome with nonzero probability)

All defined on prospects q. These routines use an `If[Negative...]` construction since its derivative is defined. For reading, use `IfNegativeToMinRule` or `IfNegativeToMaxRule`. On the binary Prospect, if `Options[Risk]` have `Position → True`, the second position is taken as the loss.

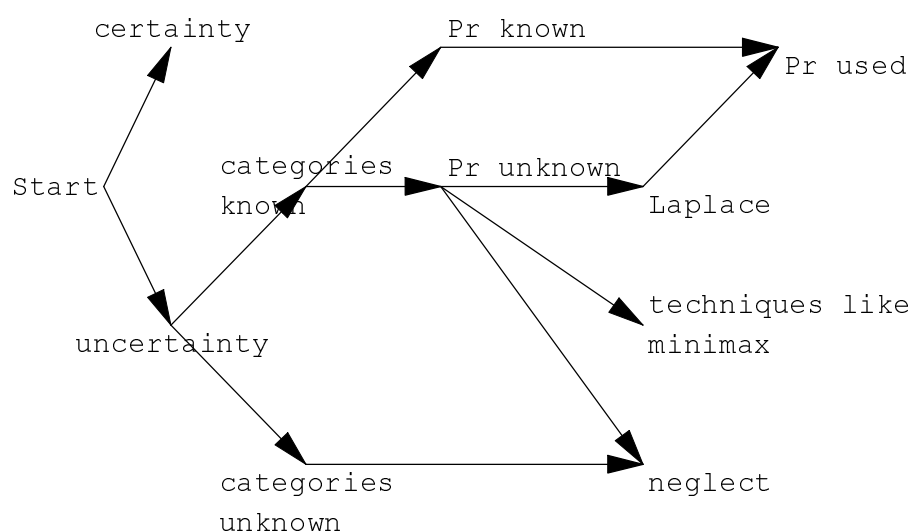
8.3.2 Theoretical definition

We better understand our subject when we first have a foundation for uncertainty and probability:

- (1) First there is the distinction between *certainty* and *uncertainty*.
- (2) Uncertainty forks into *known categories* and *unknown categories*.
- (4) Known categories forks into *known* and *unknown probabilities*.
- (3) Unknown probabilities forks into *assuming a uniform distribution* (Laplace) or use non-probabilistic *techniques like minimax* or *neglect*.

These definitions can be clarified by the following plot.

```
UncertaintyDefinitionsPlot[];
```



What do we mean by 'risk' ? A.S. Hornby's (1985) "Oxford Advanced Learner's Dictionary of Current English" defines 'risk' as: "(instance of) possibility or chance of meeting danger, suffering loss, injury, etc." Also: "at the ~ risk of / at ~ of, with the possibility of (loss etc.)".

Thus, if there are possible outcomes $O = \{o_1, o_2, \dots, o_n\}$, then the situation is risky if at least one of the o 's represents a loss. The risks themselves are the o_i that are those losses. The risks factors are the dimensions or positions of the risky outcomes, the i 's (or the causes that make such positions to be filled).

We will use the term 'valued risk' when a risk is valued with money or utility. When all risks have been made comparable by valuing them, then we can add them, and we will use the term *expected risk value* for the *expected value of the 'valued risks'*. Then, crucially, once these definitions are well understood, then we may also use '*the risk*' for the expected risk value.

With such understanding, risk will be $\rho = -E[x < 0]$. Note that the term 'risk' has not been used in the 4 points above, so that an independent definition is possible.

Relative risk is defined as $\rho(t) = t - E[x < t]$ for some target level t . Risk (or *absolute risk*) takes $t = 0$, and relative risk would allow for a different target level. An interesting application is when x is a stochastic rate of return and r the certain rate (rate of interest), so that there is relative risk $\rho(r) = r - E[x < r]$.

This relative risk answers the question: What is the probable loss with respect to a target return of r ? Here, $r - \rho(r) = E[x < r]$ gives the weight of underperformance in the total target return (which weight has to be compensated by probable profits to achieve the target).

Conditional (relative) risk is defined as $\kappa(t) = t - E[x \mid x < t]$ for some target level t . With respect to rates of return, conditional risk $\kappa(r)$ answers the question: What would one expect to lose with respect to r , if earnings actually underperform and fall below r . Indeed, $r - \kappa(r)$ would give your expected return when actually underperforming.

Conditional risk is related to relative risk by the property that $E[x \mid x < t] = E[x < t] / \Pr[x < t]$. The probable loss thus is corrected for the probability of a loss. Or, the probability measure in the expectation is corrected so that a density is taken that sums to 1.

Note that above definitions are proper in the sense that they conform to every day parlance and the definitions provided by Hornby's dictionary op. cit.. The definitions provided here however differ from other definitions within the economics literature. First there are the definitions of Knight (1921) that have been adopted widely in economics, as for example in The New Palgrave (1998:III:358). Or it has become custom in finance to associate risk with the standard deviation. Cool (1999), "Proper definitions of risk and uncertainty" (available in the Pack and on the internet or as chapter 35 in Cool (2000)) further discusses why such alternatives generate conceptual problems and why the current definitions are preferable.

Below we will develop these notions somewhat further. The Economics Pack (Cool (1999, 2001)) has a much more extensive development that would lead too far here.

8.3.3 Tests in Mathematica

Since prospects need not satisfy the formal convention for risky prospects, *Mathematica* needs a test on what are the negative values.

- This is the proper risk definition.

Risk [eg]

$-(1 - p) \text{If}[\text{Negative}[-\text{Loss}], -\text{Loss}, 0] - p \text{If}[\text{Negative}[\text{Profit}], \text{Profit}, 0]$

```
RiskyQ[eg]
```

```
Negative[-Loss]  $\Rightarrow$  True
```

- For formal analysis, however, we may resort to the convention that the second position is the loss. The Position option works only for binary risk (in non-list-format).

```
SetOptions[Risk, Position  $\rightarrow$  True];
```

```
Risk[eg]
```

```
Loss(1 - p)
```

```
RiskyQ[eg]
```

```
True
```

- If you set the option to True, you can still have formal testing by using the list format. Also, routines like ProspectPrValue internally call ProspectReList, and thus are not affected by the Position option.

```
Risk[ProspectReList[eg]]
```

```
-(1 - p) If[Negative[-Loss], -Loss, 0] - p If[Negative[Profit], Profit, 0]
```

- This gives the risk probabilities.

```
RiskPr[eg]
```

```
1 - p
```

```
eglis = Prospect[{-1, 2, -2, 3}, {0.2, 0.4, 0.3, 0.1}];
```

```
RiskPr[eglis]
```

```
0.5
```

Since risk selects the negative values from the states of the world, and since some of the prospects are algebraic rather than numeric, the Risk function basically uses a conditional statement. Where the choice was between Negative[], Min[] or Max[], the use of Negative[] has been chosen, since the derivative D[] still applies. Since an If[Negative[], ..] statements reads difficult at times, there is the possibility to replace it by the following.

<code>IfNegativeToMinRule</code>	A rule that changes an <code>If[Negative...]</code> statement into a Min condition. The derivative of <code>If[Negative...]</code> is defined, but Min reads better
<code>IfNegativeToMaxRule</code>	A rule that changes an <code>If[Negative...]</code> statement into a Max condition. The derivative of <code>If[Negative...]</code> is defined, but Max reads better

8.3.4 Risk model

This model presumes binary risk, and therefor is a good introduction to the subject.

<code>RiskModel[{rules}]</code>	is a <code>SolveFrom</code> application, default with <code>RiskEquations</code>
<code>RiskEquations</code>	used in <code>RiskModel</code> . Profit, Loss and Risk are nonnegative values, and the expected value is <code>Prob Profit - (1-Prob) Loss</code>

RiskEquations

```
{Risk == Loss (1 - Prob), ExpectedValue == Prob Profit - Risk}
```

```
RiskModel[{Profit → 0.6, Prob → 0.5, Risk → 0.2}] // Last
```

```
{{ExpectedValue → 0.1, Loss → 0.4}}
```

8.3.5 Risket

Given the relevance of expected value, spread, risk and loss probability, we define the `Risket` object.

<code>Risket[μ, σ, ρ, 1 - p]</code>	is an object with expected value μ , spread σ , risk ρ and probability p of profit
<code>ToRisket[q]</code>	turns a Prospect object or <i>pdf</i> or data vector into a Risket object
<code>ToProspect[x]</code>	turns a Risket object or <i>pdf</i> or data vector into a Prospect object

`ToRisket` has default option `Spread → StandardDeviation` (division by $n-1$). Use `Spread → Spread` to divide by n , and `Spread → False` if you want the spread implied by applying `ToProspect` again.

Importantly, the `Risket` object can contain more information than a `Prospect`. Going from a `Risket` to a `Prospect` in generally is a projection, where information about the true spread can be lost.

- This gives a clear formal result since we now work with Position \rightarrow True in Options[Risk].

ToRiskset[eg]

```
Riskset( $p$  Profit - Loss (1 -  $p$ ),  

 $\sqrt{((1 - p)((1 - p) \text{Loss} - \text{Loss} - p \text{Profit})^2 + p(\text{Loss}(1 - p) - p \text{Profit} + \text{Profit})^2)}$ , Loss (1 -  $p$ ), 1 -  $p$ )
```

- This function neglects the spread, since for true binary prospects all information is in the expected value, risk and risk probability. If the original is not truly binary, then information is lost.

ToProspect[%]

```
Prospect(Profit, -Loss,  $p$ )
```

If one wishes to determine a prospect by using the spread instead, the following would be useful.

`ProspectInverse[mean, spread, risk]` gives a binary prospect with these properties

Note that there can be more solutions.

- With these values of mean, standard deviation and risk, there are two binary prospects that satisfy those properties.

```
ProspectInverse[0.5, 0.6, 0.1]
```

```
{Prospect(0.626795, -2.33923, 0.957251), Prospect(0.973205, -0.26077, 0.61652)}
```

8.3.6 Prospect plotting

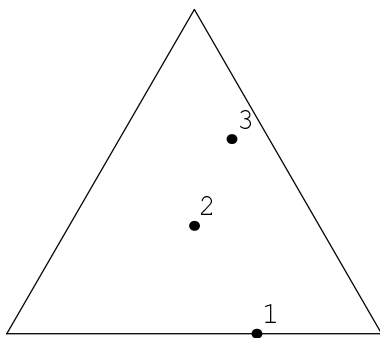
A useful feature is that the probabilities of 3D prospects can be plotted in a 2D triangle, that essentially is a transform of the 3D unit simplex. If the dimension is less than 3 then the 3rd dimension is set to 0. If the dimension is larger than 3 then the higher dimensions can be summed. The triangle has sides $2 / \sqrt{3}$, and the corners are at $c1 = \{0, 0\}$, $c2 = \{2 / \sqrt{3}, 0\}$ and $c3 = \{1 / \sqrt{3}, 1\}$. A point in the triangle has the property that the distances to the sides add up to one. The prospect probabilities $\{p1, p2, p3\}$ are plotted such that the distance from the plotted point to the side opposite to ci gives the probability pi of outcome i .

`Prospect3DPrTriangle[lis_List, opts]` plots probabilities of prospects in *lis* in a triangle

Note: The points are in `Results[ProspectPr3DTriangle]`. Note: Options are: `Point` → `True` (default) plots points, `PointSize` → `size` (default .025) gives the size of these points, `Label` → `Automatic` (default) gives labels. The latter can also be a list, or `None`. If `Point` → `False`, then the labels are printed right on the co-ordinates. Note: Subroutines are: `ProspectPr3DTriangle[]` gives the graphics primitive of the triangle; while `ProspectPr3DTriangle[Point, lis]` gives those of the points.

- Point 1 of the simple prospect plots on the bottom side, since the distance from the bottom side is zero. The distance to the right side is $1/3$ and to the left side $2/3$. Point 2 plots in the middle, and lies on a line through point 1 parallel to the right side. Point 3 then is clear.

```
Prospect3DPrTriangle[{Prospect[3, -2, 1/3],
  Prospect[{a, b, c}, {1, 1, 1}/3],
  Prospect[{a, b, c}, {0.1, 0.3, 0.6}]}];
```



Above plots just the probabilities. The following include the values.

<code>ProspectPlot[x_List]</code>	plots the prospects <i>x</i> in the Expected Value, Risk and Spread space. Subroutines are:
<code>ProspectPlot[Set, x]</code>	sets the data to be plotted (\Leftrightarrow <code>SetOptions[ProspectPlot, Data → ProspectStatistics[x]]</code>)
<code>ProspectPlot[x_Symbol, y_Symbol, opts___Rule]</code>	plots the keys <i>x</i> and <i>y</i> of these
<code>ProspectPlot[All]</code>	plots for the mentioned three keys
<code>ProspectPlot[q_Prospect, a_Symbol]</code>	plots Expected Value, Risk and Spread values of a prospect that is a function of <i>a</i> (in the domain $[0, 1]$)

8.4 Certainty equivalence

This section discusses how one could try to recover an ordinal utility scale from experiments with prospects. The current assumption is that there is no difference between the buying or selling price of a prospect (lottery ticket), although in practice there can be this difference (since people tend to overcharge for what they have, even when they think that the grass of the neighbour is greener).

8.4.1 The price of a lottery ticket

Suppose that there is a lottery with a probability p on a prize, called Profit. We use simple expected values (with the assumption that the proportion to total wealth is very small).

- If you buy a lottery ticket then you expect a positive return. Your selling price of the lottery ticket should generally be larger than your expected return.

$$\text{ProspectEV}[\text{Prospect}[\text{Profit}, 0, p]] - \text{Price}_{\text{Buy}} \geq 0$$

$$p \text{ Profit} - \text{Price}_{\text{Buy}} \geq 0$$

$$\text{Price}_{\text{Sell}} - \text{ProspectEV}[\text{Prospect}[\text{Profit}, 0, p]] \geq 0$$

$$\text{Price}_{\text{Sell}} - p \text{ Profit} \geq 0$$

- In equilibrium, you can sell the ticket only at the price that you pay yourself, and you can only buy it at the price for which you would be willing to sell it.

$$\text{Price} == \text{ProspectEV}[\text{Prospect}[\text{Profit}, 0, p]]$$

$$\text{Price} == p \text{ Profit}$$

The existence of lotteries, where there also are overhead costs, is a mystery that economic theory tries to explain, e.g. with risk preference and risk aversion.

8.4.2 A standard approach

A way to recover utility functions is to precisely use such lotteries.

Luenberger (1998:234): "(...) select two fixed wealth values A and B as reference points. A lottery is then proposed that has outcome A with probability p and outcome B with probability $1 - p$. For various values of p the investor is asked how much certain wealth C he or she would accept in place of the lottery. C will vary as p changes. Note that the values A , B and C are values for total wealth, not just increments based on a bet."

Here we use A for "above" and B for "below". We will reproduce Luenbergers example - but with reversed values for A and B because of this interpretation. Because of the word 'accepting', the certainty equivalent $C = C(p)$ and lottery apparently already are part of the investor's wealth. The investor may win $A - C(p)$ or lose $C(p) - B$ (as absolute loss) compared to the certainty equivalent wealth level $C(p)$ of doing nothing.

- If we look at the budget only (choosing A , B and C as wealth levels).

```
CertaintyEq[Prospect[A, B, p], C, "Budget"] /. Wealth → 0
```

```
C + Prospect(A - C, B - C, p)
```

- However, we should consider utility.

```
cond = CertaintyEq[Prospect[A, B, p], C, None]
```

```
Utility(C) == p Utility(A) + (1 - p) Utility(B)
```

<code>CertaintyEq[q_Prospect, c]</code>	gives the condition so that c is the certainty equivalent of the prospect
<code>CertaintyEq[q_Prospect, c, None]</code>	gives the same but excluding wealth
<code>CertaintyEq[Prospect[A, B, p], C, 'Budget']</code>	
gives the budget for who reasons as follows: certain is $Wealth + C$, and then there is the prospect of winning $A - C$ or losing (absolute) $C - B$.	

- The certainty equivalence condition is invariant for a linear transformation. This means that we can recover only ordinal utility.

```
cond /. Utility[x_] => a U[x] + b // Simplify
```

```
a(-p U(A) + (p - 1) U(B) + U(C)) == 0
```

Because of this property of ordinal utility functions, we can normalise $A = U(A)$ and $B = U(B)$. This means that $U(C)$ collapses to $U(C) = p A + (1 - p) B = EV$. In other words, taking money prospects causes that the utility can be fully represented by the expected value, even though we first argued that we should not take the expected value but should take utility. Next, having a set of points $\{C, EV\}$ allows us to interpolate values (which avoids estimation of parameters), and subsequently to interpret this interpolation as the U .

- We can normalise our findings by setting the utility levels of the extremes of the prospect equal to those extremes, in this case $Utility[Max] = Max = A$ and $Utility[Min] = Min = B$. This uses Luenberger's example with $A = 9$ and $B = 1$ (million dollars).

CertaintyEq[Equations]

$\{c_2 + c_1 \text{Utility}(1) == 1, c_2 + c_1 \text{Utility}(9) == 9\}$

CertaintyEq[Equations] gives the equations for the linear transform of utility

Note that we can also determine the utility as a function of the probabilities, since:

$$U(C(p)) = p A + (1-p) B = U_p(p \mid \{A, B\})$$

It is unclear why we should try to recover values for C . The utility function is wholly determined now, and in practice the investor should concentrate on finding the correct values for A , B and p . Yet, let us see how the standard approach proceeds.

- This example is taken from Luenberger (1998:236). He uses million dollars, for a 'moderately successful venture capitalist'.

below = 1; above = 9;

probpoints = Range[0, 10]/8.

$\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.\}$

- This sets the options for the maximum A and minimum B and the probability points. It calculates the expected values - which might be communicated to the investor, so that he or she is actually informed about his or her utility function as depending upon the probabilities.

```
CertaintyEq[Set, above, below, probpoints]
```

```
{1, 1.8, 2.6, 3.4, 4.2, 5., 5.8, 6.6, 7.4, 8.2, 9.}
```

<code>CertaintyEq[Set, max, min, p_List]</code>	sets the options for the maximum and minimum points, the probabilities p that are being considered, and the implied expected values
---	---

- Luenberger's investor gives these certainty equivalent values $C(p)$ for the various probability points (taking the expected values into account or not).

```
cedata = {1, 1.44, 1.96, 2.56, 3.24, 4, 4.84, 5.76, 6.76, 7.84, 9};
```

- These $C(p)$ data now can be used to derive an interpolated utility function (the goal of the whole exercise).

```
func = CertaintyEq[Data, cedata]
```

```
InterpolatingFunction[{1. 9.}, <>]
```

- For example, at a certainty equivalent value $C = 2$ million dollar (not given in the $C(p)$ data), the utility level would be:

```
func[2]
```

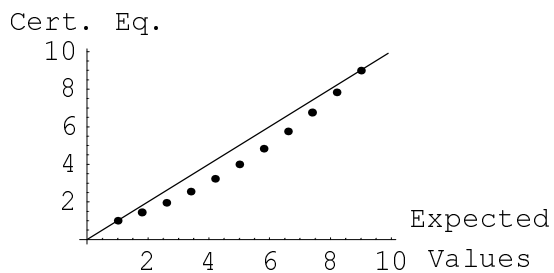
```
2.65701
```

<code>CertaintyEq[Data, CE_List]</code>	sets <code>CertaintyEq[Data]</code> and creates in interpolation function for the certainty equivalent values CE
<code>CertaintyEq[Data]</code>	contains pairs {expected value, certainty equivalent}

Note: The data are recorded in `Options[CertaintyEq]`.

- This plots the {EV, C} points, using `CertaintyEq[Data]`.

```
CertaintyEq[Plot];
```



The reason for the exercise can be that people better understand Utility as something that depends upon income rather than as something that depends upon probability - even though the income depends upon the probability. The Arrow - Pratt measure for 'risk aversion' also requires a normal utility function.

```
ArrowPratt[u[x], x]
```

gives the Arrow-Pratt measure $-u''/u'$ of utility function $u[x]$. This is a measure of concavity that is independent of linear transformations of $u[x]$. Normally $u[x]$ is rising, so that $u' > 0$

```
ArrowPratt[a Utility[x] + b, x]
```

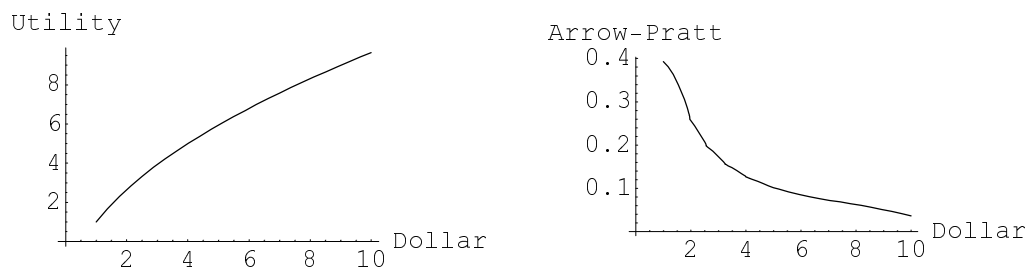
$$-\frac{\text{Utility}''(x)}{\text{Utility}'(x)}$$

- It appears possible to differentiate the interpolated utility function. The Arrow-Pratt measure for 'risk aversion' becomes:

```
ap[x_] = ArrowPratt[func[x], x];
```

- The following plots the estimated utility function as well as the Arrow-Pratt measure.

```
Show[GraphicsArray[{
  Plot[func[x], {x, 1, 10}, AxesLabel → {Dollar, Utility},
    DisplayFunction → Identity],
  Plot[ap[x], {x, 1, 10}, AxesLabel → {Dollar, "Arrow-Pratt"},
    DisplayFunction → Identity]]],
  DisplayFunction → $DisplayFunction];
```



CertaintyEq[ListPlot, opts]	plots CertaintyEq[Data]
CertaintyEq[Plot, opts]	adds a diagonal line for reference

8.4.3 A non-standard approach

The above (standard) approach can be criticised for using nonstochastic utility for a stochastic situation. The curvature of the utility function, that applies for *certain* changes, now is applied to *stochastic* changes. The standard approach also does not explicitly state the risk which is exemplified by the budget line. An alternative (non-standard) approach is to regard a utility function that includes stochastic data and that uses that explicit risk.

- First recall the budget above.

```
CertaintyEq[Prospect[A, B, p], C, "Budget"] /. Wealth → 0
```

$$C + \text{Prospect}(A - C, B - C, p)$$

There is more information in this than is used by above standard approach. By accepting a certainty equivalent, the subject forgoes some expected Income = ΔWealth which is given by ProspectEV. The reason to let this money go is the risk in the prospect. While doing so, the subject remains on the same utility contour *with C just by itself*. We may thus infer, first, that utility depends upon

the wealth level *and* stochastic data as summarised by the expected value and the risk, so that $U(\text{wealth}, \mu, \rho)$. It is not strange to let expected money and risky money into the utility function, where we already had 'certainty equivalent' money in it. And we may infer, secondly, that an indifference contour goes through $U(C, 0, 0) = U(C, \mu, \rho)$, and that this is '*proper* certainty equivalence'.

- There is indifference when the expected value is balanced by the risk.

```
cond2 = Utility[C, 0, 0] ==
  ProspectUtility[Prospect[A - C, B - C, p], C &, ProspectEV, Risk]

Utility(C, 0, 0) == Utility(C, (B - C) (1 - p) + (A - C) p, -(B - C) (1 - p))
```

To tackle this, we need to have access to the prospects for the various probability points.

- These are the prospects per considered probability point. For example, when $p = 0$, then the investor can gain 8 with p and lose 0 with $(1 - p)$. This sets all Options[CertaintyEq].

```
CertaintyEq[Set, Prospect]

{Prospect(8, 0, 0), Prospect(7.56, -0.44, 0.1), Prospect(7.04, -0.96, 0.2),
 Prospect(6.44, -1.56, 0.3), Prospect(5.76, -2.24, 0.4), Prospect(5, -3, 0.5), Prospect(4.16, -3.84, 0.6),
 Prospect(3.24, -4.76, 0.7), Prospect(2.24, -5.76, 0.8), Prospect(1.16, -6.84, 0.9), Prospect(0, -8, 1.)}
```

- The following determines the ProspectEV and Risk implied by the current Options[CertaintyEq], that are to be used for above ProspectUtility.

```
lisev = CertaintyEq[Prospect, ProspectEV]

{0, 0.36, 0.64, 0.84, 0.96, 1., 0.96, 0.84, 0.64, 0.36, 0.}

liskr = CertaintyEq[Prospect, Risk]

{0, 0.396, 0.768, 1.092, 1.344, 1.5, 1.536, 1.428, 1.152, 0.684, 0.}
```

- Above, we called the following *cedata*, but if you have not given a specific name, the data are still is available in a structural manner.

```
lisc = CertaintyEq /. Options[CertaintyEq]

{1, 1.44, 1.96, 2.56, 3.24, 4, 4.84, 5.76, 6.76, 7.84, 9}
```

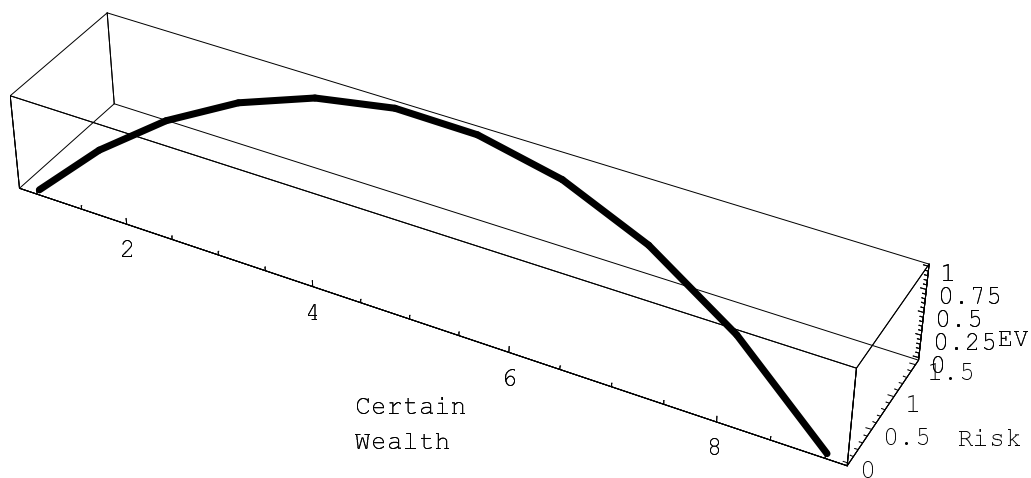

`CertaintyEq[Set, Prospect]` creates the prospects implied by the options

`CertaintyEq[Prospect, f]` applies function f to the prospects in the options

- We can plot the data in 3D space, joining up the observed points by a line. A higher risk needs compensation with higher expected addition to wealth. Alternatively, a higher expected addition to wealth allows taking more risk on it.

```
Needs["Graphics`Graphics3D`"]
```

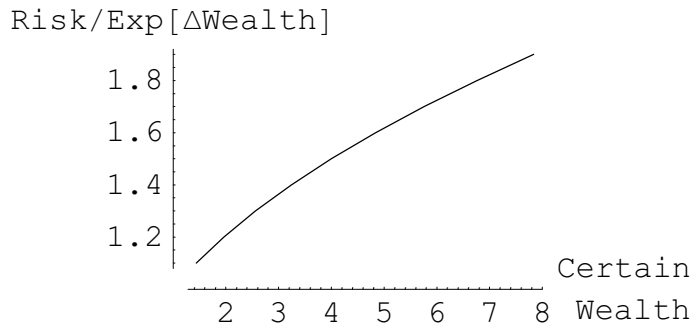
```
ScatterPlot3D[Transpose[{lisc, lisr, lisev}],
  PlotJoined → True, PlotStyle → Thickness[0.008],
  AxesLabel → {"Certain\nWealth", Risk, "EV"}];
```



With rising wealth, this particular experimental person apparently is willing to accept more risk for the same amount of expected income = Δ Wealth. The investor can use this relationship between $C(p)$ and ρ / μ for additional decisions and predictions.

- Plotting the relationship between $C(p)$ and $\rho / \mu = \text{Risk} / \text{ProspectEV}$.

```
PlotLine[lisc, liscr/lisev, AxesLabel →
{"Certain\n Wealth", "Risk/Exp[ΔWealth]"}, AxesOrigin → {1, 1}];
```



PM. Since we have $\rho / \mu = f(C(p))$ then above $U(C(p))$ would imply $U_f(\rho / \mu)$.

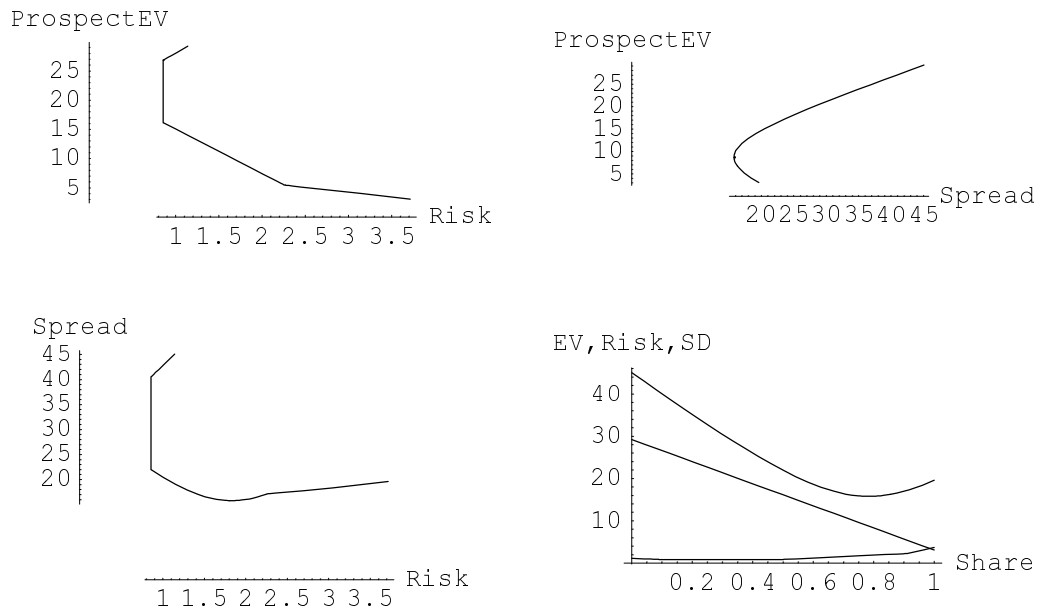
8.4.4 Justification for the non-standard approach

We can better understand the non-standard approach by taking an example from finance. In finance, individual assets (bonds, shares and property) are characterised by their expected values and spreads. For portfolio's of assets, risks can cancel, and then there arises the Markowitz efficiency frontier in the $\{\sigma, \mu\}$ space. It then is assumed that investors have utilities $U(\sigma, \mu)$ such that, with equal spread the higher expected values will be taken, and with equal expected values the lower spreads. Maximising utility then allows the selection of the best mix of assets in the portfolio. For us, it is more appropriate to take $U(\rho, \mu)$ since a high spread is less relevant if it would concern only positive values.

The following example is taken from The Economics Pack. An investor will allocate a budget over two prospects, and will be interested in the optimal mix, allocating share S to one prospect and $1 - S$ to the other. (The calculations are not given here.)

- The following plots for S in $[0,1]$. The finance community is familiar with the upper right hand plot, the other plots are novel.

`ProspectPlot[prs, S];`



8.4.5 Comparing the methods

The final question however is whether we could still accept the oldstyle condition for 'certainty equivalence' as a realistic assumption.

- The 'oldstyle certainty equivalence' condition now becomes:

`cond3 = cond /. Utility[x_] => Utility[x, 0, 0]`

`Utility(C, 0, 0) == p Utility(A, 0, 0) + (1 - p) Utility(B, 0, 0)`

We concluded earlier that $C(p)$, by being 'accepted', apparently was part of wealth already. If a choice is being offered, then the experimental person should place himself or herself into the position, even when it is a thought experiment, that it is a real choice, and hence the value of the choice is part of wealth, in this case the $C(p)$ as the certainty equivalent (properly defined) of the choice. This conclusion allowed us to define proper values for both the expected increase in wealth and the risk, both based upon the Prospect[$A - C(p), B - C(p), p$].

But the conclusion that $U(C, 0, 0) = U(C, \mu, \rho)$ is a quite different conclusion than *cond3*. In principle these are independent conditions, and they do not have to be true at the same time. The alternative notion of 'expected utility' in *cond*, the weighing (forecasted) utilities of prospective different worlds by the probabilities of those worlds, is not self-evident. It is also possible that people balance μ with ρ .

Another way to understand this is to go back to the original experimental setup (the Luenberger quote). If a choice has probability p , then $C = C(p)$ can be derived from the contour of *cond2* by which the subject balances μ with ρ . Then it does not follow yet that at the same time, additionally, *cond3* will hold. The conditions are clearly different, and thus they do not need to hold at the same time. This means that the 'oldstyle certainty equivalent condition' would not be our first hypothesis, but only a secondary possibility.

Hence, it does not seem feasible to estimate the utility function as originally thought. The case is not settled, and real world experiments are required to determine what applies.

Note: The oldstyle notion of certainty equivalence combines with the Arrow - Pratt measure of 'risk aversion', and together they form a strong team. However, the Arrow - Pratt measure is basically related to decreasing marginal utility (events with increases cannot be excluded), and it clearly is not a direct function of probability p (only indirectly, via $C(p)$, which depends upon *cond*), while ρ is not present. Arrow - Pratt runs into problems when marginal utility would behave one way and ρ would behave in the opposite way. My suggestion is that a measure for risk aversion that directly relies on ρ is more convincing. And if the oldstyle notion of certainty equivalence loses its Arrow - Pratt measure, it also becomes deficient for dealing with risk. So a key question, on the point of theory is: can marginal utility of deterministic income serve two purposes ?

8.4.6 Application to voting

There is a suggestion in the literature that prospects can be used to recover utility functions, and sometimes it is suggested that this would be cardinal utility. The first disappointment is that these utility functions are only ordinal, the second disappointment is that the 'utility' basically is expected income, and the third disappointment is that the standard derivation uses utility functions from a deterministic realm for a stochastic realm which does not seem adequate. The approach also neglects cheating, which would be a key issue for voting.

Nevertheless, we have identified some tools that can be used to better deal with voting situations. Indeed, people have to vote on risky prospects, and since the notions of risk are often confusing, this discussion has given some clarity.

8.4.7 A note on independence

Above difference with the standard approach can also be clarified by reference to Mas-Colell c.s. (1995:171) and their theorem 6.B.4 on the "independence" axiom of preferences on prospects. The discussion on prospects there is a bit sterile, since it concentrates on the probabilities (or is in danger of confusion with that) while economically we are rather interested in the commodity space. The authors overstate it when they write that the independence theorem is "at the heart of the theory of choice under uncertainty". Similarly, the discussion there on p 179-180 on the Allais paradox leaves much to be said.

Let us regard three situations: Good, Normal and Bad. Use \supseteq for the preference relation 'at least as good as'. (We do this in this subsection, since we want to use P for a prospect.) The axiom states that the preference between prospects P and P' is independent of any third P'' , or, for $\text{Pref}[P', P]$:

$$P \supseteq P' \Leftrightarrow (\alpha P + (1 - \alpha) P'' \supseteq \alpha P' + (1 - \alpha) P'' \text{ for } \alpha \in (0, 1))$$

This sounds seductively true if we look at probabilities only. But economically, we cannot neglect the wealth effect. Clearly my preference generally is Good \supseteq Normal \supseteq Bad, and when I am in a bad situation then clearly I am willing to gamble on getting better. But when I am in a Normal situation, then I will

hesitate on a gamble with the Bad risk. The independence axiom would force me to gamble though !

In formula's: When at Bad, there can be a β so that

$$\blacksquare \beta \text{ Good} + (1 - \beta) \text{ Bad} \supseteq \text{Normal}$$

Let us take a α and check independence from Normal itself (which the axiom allows). Then:

$$\blacksquare \alpha (\beta \text{ Good} + (1 - \beta) \text{ Bad}) + (1 - \alpha) \text{ Normal} \supseteq \text{Normal}$$

But when I am at Normal, I may not wish to gamble when Bad is a possible outcome !

Let us consider the Allais paradox. If we include the wealth effect and risk aversion (with the proper definition of risk), then we find that the situation need not be irrational.

AllaisParadox[]

contains the four prospects for the Allais paradox,
discussed by Mas-Colell c.s. (1995), p179–180.

People are offered two choices, one between 1 and 2, and one between 3 and 4. They tend to prefer $1 \supseteq 2$ and $4 \supseteq 3$, though this violates the independence axiom. Below shows that their choice is not really irrational.

Prospect 1 appears to have a certain outcome, and Prospect 2 has a risk element. For the choice between 3 and 4, we can use certainty equivalence (properly defined).

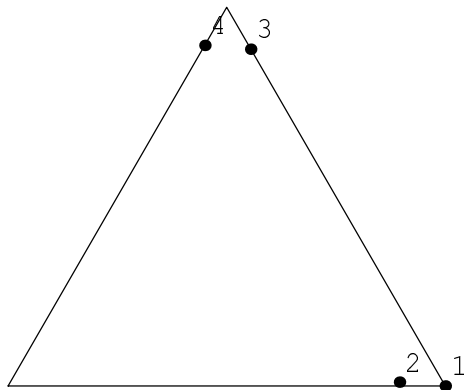
- This is available in the package.

```
alpar = AllaisParadox[]
```

```
{Prospect({2.5 × 106, 500000., 0}, {0, 1, 0}), Prospect({2.5 × 106, 500000., 0}, {0.1, 0.89, 0.01}),  
 Prospect({2.5 × 106, 500000., 0}, {0, 0.11, 0.89}), Prospect({2.5 × 106, 500000., 0}, {0.1, 0, 0.9})}
```

- See above for the explanation of a Prospect Plot.

```
Prospect3DPrTriangle[AllaisParadox[]];
```



- The choice between 1 and 2 is also the choice between certainty and risk. The certain value becomes our default wealth, and we can find the Risket around it.

```
PutIn[alpar[[2]] - 500000]
```

```
Prospect({2. × 106, 0., -500000}, {0.1, 0.89, 0.01})
```

- This gives Risket[$\mu, \sigma, \rho, 1 - p$], with μ the expected value, σ the standard deviation, ρ the risk (properly defined) and $1 - p$ the cumulated probability of a loss.

```
ToRisket[%]
```

```
Risket(195000., 603718., 5000., 0.01)
```

- For the choice between 3 and 4, we can take a reference point in the certainty equivalence (properly defined) of the least attractive option. What this is, depends upon the agent. Let us here take the minimal expected value as the reference point. First determine the expected values.

```
ProspectEV /@ alpar
```

```
{500000., 695000., 55000., 250000.}
```

- It turns out that 3 has the minimum expected value. Taking this as the addition to wealth, we can determine the riskets, and find that option 4 clearly is better. The standard deviation does not tell much, what are important are the ρ and $1 - p$. These are quite comparable, so that the expected value decides.

```
{r3, r4} = ToRisket /@ PutIn /@ (Take[alpar, -2] - 55000)
```

```
{Risket(0., 156445., 48950., 0.89), Risket(195000., 750000., 49500., 0.9)}
```


9. Theoretical base

9.1 Theory overview

9.1.1 Introduction

This part of the book will set out the theory for the practical routines. The theory is rather abstract, so the discussion is intended for advanced readers. However, we try to keep the exposition and use of language as clear as possible, so that less advanced readers could get the gist of the discussion and could decide whether they want to advance their study.

For voting theory, our terminology will differ from these books in particular: (1) Amartya Sen (1970), "Collective Choice and Social Welfare", North Holland, (2) chapter 36 of Cool (2000), "Definition and Reality in the General Theory of Political Economy" (DRGTPE), Samuel van Houten Genootschap, and (3) chapter "1990g" of Cool (1992) "DRGTPE - background publications", Magnana Mu Publishing & Research. The differences are:

- DRGTPE and Sen use the \geq ordering consistently, while here we use the \leq ordering consistent with this book, with Takayama (1974) and with the idea that rising utility means a higher number.
- The culprit axiom now has been baptised the "Axiom of Pairwise Decision Making" (APDM) - what Arrow (1963) before called the "Axiom of Independence of Irrelevant Alternatives" (AIIA). So the axiom remains the same, only the name is different. This new name is much clearer about what the axiom really means in normal English. I have added a separate chapter to clarify this choice of words.
- DRGTPE uses the word 'constitution' for the SWF-GM, but when finishing work on the *Mathematica* packages and writing this book it appeared that it is better to equate constitution = Social Decision Function (SDF), since this is in line with normal English. This also better expresses that the routines implement SDFs. It is merely practical to only determine the winner and not the whole ordering. However, we will show that a SDF has an associated SWF-GM.

In the following we first solve Kenneth Arrow's 'difficulty in social choice', then consider the APDM and then consider some final abstract points.

`Needs["Economics`Pack`"]`

`ResetAll`

`Economics[Voting]`

Cool`Voting`

[Constitutions](#) [ParetoBorda](#) [ParetoPairwise](#) [Vote](#)
[Majority](#) [ParetoMajority](#) [StrategicPref](#) [VoteToPref](#)

`Economics[Voting`Theory]`

Cool`Voting`Theory`

[ChoiceSet](#) [FullyMatched](#) [SubsetConsistency](#)

9.1.2 Notation

Please note that we will have to redefine some symbols for this Part.

Let X be the commodity domain. The normal Social Welfare Function (SWF) or 'Bergson-Samuelson type of social welfare function' is defined directly over X as $SWF: X \rightarrow [0, \infty)$. It is just the group utility function, and works for the group as a utility function works for an individual.

Subsequently, we want that the SWF depends upon the utilities of the individuals in the group. Thus we need a Social Welfare Function Generating Mechanism (SWF-GM). In the literature this is the 'Arrow type of social welfare function'. It arises as follows. An agent is a compound of various properties such as utility, wealth etcetera. Let K be the set of possible compounds on X . With n agents, our interest concerns the function $g: K^n \rightarrow K$, which maps the society into an aggregate compound. Note that strictly speaking $g = g(X)$. An element in K^n is called a profile, and the aggregate compound is $K_T = g(K_1, \dots, K_n)$. Here K_T is associated with aggregate utility or the normal social welfare function SWF, and g is the SWF-GM. Note that an example of a profile is the `Preferences` object, as we have programmed this in *Mathematica*.

Note that a constitution (like the U.S. Constitution), that determines voting rules, generally is not concerned with fully ordering the whole commodity domain. Constitutions satisfy themselves with finding the best element within the available budget set B . In Social Choice Theory such constitutions are also

called Social Decision Functions (SDF) - and a SDF generates a Choice Set $C(B)$ consisting of the best elements. If the choice set depends upon an aggregate compound, then we can also write $C(B, K_T)$. For constitution c we can have: $c: \{B\} \times K^n \rightarrow \{2^B\}$.

Each g obviously implies a c , but it is less obvious whether the converse would hold. We will see however that for subsets $B \subset X$, we can define a compound $g(c \upharpoonright B)$, conditional to the budget.

It suffices to restrict K to preference orderings. These orderings satisfy reflexivity, transitivity and completeness. It is important to add that there is no cheating. Let R denote normal preference, P strict preference, and I indifference. When there is no confusion, we can also use the symbols \leq , $<$ and $=$. A suffix denotes an individual preference, otherwise it is the aggregate, so that $R = g(R_1, \dots, R_n)$.

Finally, we will use the logical operators \vee , $\&$, \Rightarrow and the negation sign \neg .

9.1.3 Implied aggregate ordering

Each voting scheme gives an implied order, namely when we first determine the winning item W , then eliminate it from the list of items, then recalculate the votes again, etcetera. This implied order can be generated by the `VoteToPref[v]` routine that applies voting scheme v repeatedly in this manner. Note that the result of v can depend upon the status quo. If that is the case, then you should make sure that your `StatusQuo[]` function defines a solution for each subset. If your scheme neglects the status quo, then you can set the option to `StatusQuo → False`. If you want to trace the steps while the scheme is collapsing the list of items, then you can set the option `Trace → True`.

- This creates a larger random matrix, and sets the status quo to a specific value.

```
DefaultItems[];
SetRandomPreferences[5, 8];
SetFirstValue[2]
```

$$\begin{pmatrix} 2 & 7 & 1 & 5 & 8 & 4 & 6 & 3 \\ 2 & 1 & 8 & 3 & 4 & 5 & 7 & 6 \\ 2 & 5 & 7 & 1 & 8 & 6 & 4 & 3 \\ 2 & 7 & 8 & 6 & 5 & 3 & 4 & 1 \\ 2 & 7 & 1 & 8 & 6 & 4 & 3 & 5 \end{pmatrix}$$

- This creates the ordering. The default voting scheme is Vote, and this by default is ParetoMajority. This depends upon the status quo. The ordering differs from the status quo until it is eliminated. Thereafter, the first element is the status quo and also selected.

```
ord = VoteToPref[]
```

CheckVote::adj : NumberOfItems adjusted to 4

CheckVote::adj : NumberOfItems adjusted to 8

CheckVote::adj : NumberOfItems adjusted to 7

General::stop : Further output of CheckVote::adj will be suppressed during this calculation.

$$\left\{ \text{StatusQuo} \rightarrow \begin{pmatrix} \text{H} & \text{H} \\ \text{D} & \text{D} \\ \text{C} & \text{C} \\ \text{B} & \text{B} \\ \text{A} & \text{A} \\ \text{A} & \text{F} \\ \text{A} & \text{G} \\ \text{A} & \text{E} \end{pmatrix}, \text{Pref} \rightarrow \text{Pref}(\text{H}, \text{D}, \text{C}, \text{B}, \text{A}, \text{F}, \text{G}, \text{E}) \right\}$$

```
VoteToPref[f:Vote, p:Preferences, v:Votes, i:Items, opts]
```

uses voting function $f[p, v, i]$ to create a group Pref object.

Output of $f[p, v, i]$ should have the format $\{\dots, \text{Select} \rightarrow \text{item} \mid \text{list of items}, \dots\}$. For some f , the StatusQuo[] is important, and then the option must be set to StatusQuo \rightarrow True. It should be redefined to some item in Items whenever Items is reduced. Options Trace \rightarrow True allows you to follow the collapse of the Items.

If we have some elements, say D , E and H , and we want to determine their relative order, then we should clearly distinguish between the following concepts:

1. When we speak about ordering the candidates, then we normally mean above order which depends on the vote that was held on that specific list of items (the budget). When we are interested in a subset, then it suffices to simply take the subset in that ordering.

Pref /. **ord**

Pref(H, D, C, B, A, F, G, E)

PrefSubset[% , {"D", "E", "H"}]

Pref(H, D, E)

2. Another ordering could arise when we would be willing to carry the cost of organising another vote. In that case, we actually have another budget, and taking a subset then means taking another budget. Let us do that, and let us also presume that we also want to neglect the status quo.

Preferences = **TakePref** [{"D", "E", "H"}]

CheckVote::adj : NumberOfItems adjusted to 8

CheckVote::adj : NumberOfItems adjusted to 3

$$\begin{pmatrix} 2 & 3 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 2 \\ 3 & 2 & 1 \\ 3 & 2 & 1 \end{pmatrix}$$

Items = **Results**[**TakePref**];

VoteToPref[**StatusQuo** → **False**]

CheckVote::adj : NumberOfItems adjusted to 2

CheckVote::adj : NumberOfItems adjusted to 1

{StatusQuo → {}, Pref → Pref(H, E, D)}

We find that the ordering was H, D, E , and now is H, E, D . Method 1 gives another order than method 2. Clearly, the two concepts of ordering are different, and we should carefully distinguish them. It is a key property of voting that results are *conditional to the budget* (including the status quo).

Note 1: There thus is a distinction between recalculation based on given votes (under 1) and organising a new vote (under 2). Many texts in Voting Theory literature do not clearly distinguish these.

Note 2: There is also a third type of ordering, generally defined by each scheme itself. There is for example the ordering that is directly generated by Borda (see `BordaAnalysis`), or there is the ordering put out by Plurality. These orderings depend upon the particular schemes, and are less useful for comparisons over schemes. The orderings defined by methods 1 (`VoteToPref`) and 2 (organising a new vote) are the more logical orderings when we want to compare the different schemes. The reason is that the voting schemes tend to focus on finding the winner only, and they tend to spend less attention on the whole ordering. But methods 1 and 2 then find the implied orderings.

Note 3: Sometimes we allow people to refuse to be a candidate. Then they do not participate as candidates in the election. This might influence the result or the ordering that can be derived from the election. For example, if *A* refuses to be a candidate since she sees that *B* will win, then the apparent ordering might show up that $C > D$; but if *A* had participated, though with the expected effect that $B > A$, then it might show up that $C < D$. The proper way to see this kind of phenomenon is - in my considered view - to accept the paradox but not to worry about it. The whole budget set - defined as what reality offers us - actually was *A*, *B*, *C*, *D* & all other existing or qualified people (who we never heard of), and if we allow people like *A* not to participate, then we have implicitly accepted a result that looks paradoxical but that is not really a contradiction (while it is neither clear why we should worry about it).

Note that the implied order of `VoteToPref` is rather robust, since strong candidates are not considered (have already been eliminated) when the lower ranks are being considered. The default `ParetoMajority` is also rather robust. Yet, we must keep in mind that a conditional preference reversal might show up when the budget changes.

9.2 The solution to Arrow's difficulty in social choice

Note: This is slightly adapted from chapter 36 of Cool (2000), "Definition and Reality in the General Theory of Political Economy" (DRGTPE). See the overview chapter for the changes.

Abstract: Arrow's Theorem holds that no SWF-GM can satisfy certain properties, and by implication constitutions. In annex to that theorem, Arrow claims that those properties are reasonable and morally desirable. In Arrow's view there thus is the difficulty that people desire a constitution that cannot exist. While the Theorem stands as a mathematical result, the additional claims concern some other matters, namely the domains of reasonableness and morality. It are these claims that have caused much confusion in the literature. It is shown here that the claims are unwarranted, since inconsistent properties are neither reasonable nor morally desirable. It is shown too that Arrow's axiom of Pairwise Decision Making (formerly known as the Independence of Irrelevant Alternatives) is not realistic, and thus unattractive. We show the existence of some constitutions without that axiom that are consistent and might be optimal to many. The major error made by Arrow and his students is to mix up the context of scientific discovery and learning with the context of application to the real world by educated people.

9.2.1 Introduction

Arrow (1950, 1951, 1963) showed that if certain properties are postulated for a SWF-GM, then such a SWF-GM would not exist. This result has been checked by numerous scholars, is accepted by this author, and thus stands as a mathematical theorem. In fact, we will give a short proof below.

Arrow also claimed, annex to the theorem, and *this* will be at issue here, that those properties would be reasonable and morally desirable. He recently repeated that claim in the Palgrave (1988:125). He writes:

"(...) conditions to be imposed on constitutions (...)"

"(...) there is no social choice mechanism which satisfies a number of reasonable conditions".

For clarity it is useful to introduce the following abbreviations for the theorem and its companion claims, and their conjunction:

AT = the Arrow Theorem

ARC = the Arrow Reasonableness Claim = the properties are *reasonable*

AMC = the Arrow Moral Claim = that they are *to be imposed*

AGV = the Arrow General View = *AT & ARC & AMC*

Note that Arrow's phrasing on ARC and AMC is a bit ambiguous. The "to be imposed" might not be moral but merely logical, in a sense that one needs at least some conditions to make a constitution. However, the topic of collective choice is distinctly a moral one. Secondly, Arrow emphasises what is to be imposed and what is reasonable, but he may not be in a position to impose his views and morals on us. The best interpretation of the situation likely is as follows. Presume that Arrow sees the Founding Fathers at work. He then retreats to his office, and conjectures: 'If I interpret correctly what they want, then it are these properties.' Thus the ARC and AMC are not quite Arrow's personal ideas. Above quotes can best be interpreted as factual statements on what people apparently want and consider reasonable.

Arrow's general view has been accepted in many places in the literature and textbooks, see Luce & Raiffa (1957), Johansen (1969), Sen (1986) or various other entries in that same Palgrave. For example, Tobin (1990):

"We know there is no way to aggregate individual preferences into social rankings (...). As if this were not obvious, Kenneth Arrow proved it rigorously years ago. The impossibility applies to aggregations across contemporaneous cohorts, a fortiori across generations living and unborn."

In a much used book on Cost-Benefit Analysis, A.K. Dasgupta & D.W. Pearce (1980): "(...) no escape route (...) seems yet to be available." Apparently feeling that Arrow's argument destroys the foundations of CBA, they find themselves forced, rather grudgingly, to reduce CBA to something like information gathering.

Jorgenson (1990), once president of the Econometric Society, concludes 'more positively' to dictatorship:

"The classic result of social choice theory is Arrow's (...) impossibility theorem, which states that ordinal noncomparability of individual welfare orderings implies that a consistent social ordering must be dictatorial, corresponding to the preferences of a single individual."

Not everybody falls for dictatorship. The impact of the AGV generally comes from the fact that people find themselves, either from moral obligation or from reasonableness, wanting the impossible. And many simply stay in that fixture.

Note the subtlety in that fixture. The impossibility is logical and not just empirical. An example may help. Let me confide that I want to found a new university on the island of Crete. However, I am not that rich, so I want something impossible. This however does not put me into a fixture, since I am used to the fact that I cannot afford some things that I want. However, the Arrow general view concerns a logical impossibility, which is something quite different.

We can usefully recognise:

reasonable = rational & realistic

Reasonableness is the intersection of rationality and empirical realism. Nonexistence may derive from empirical circumstances or from logical impossibility. Irrationality however is always unrealistic. Inconsistency cannot *exist*, in the true empirical sense. For example a round square cannot exist. The nonexistence of the Arrowian SWF-GM similarly derives not from empirical reality but from logical necessity.

Given the AGV, the question arises what the reasonableness and moral presumptions of Arrow's claims actually are. Are these claims as strong as conjectured?

My position is as follows:

1. As has been said on 'round tables', it is not rational to postulate inconsistent properties. People involved in a learning process may indeed make inconsistent assumptions. However, once the inconsistency is discovered, it is no longer considered to be rational to adopt those assumptions. People may enjoy 'roundness' and 'squareness', but having both simultaneously is seen to be inconsistent, even inconceivable, and hence unreasonable. The Arrowian properties are unreasonable in the exactly same manner. Arrow's pitfall is to confuse the learning process, his context of discovery, with real world applications by educated people.
2. Similarly, one cannot be morally obligated to a logical impossibility. Hence Arrow's properties are morally undesirable.

These points will be clarified below.

Note that people have in practice rejected some of Arrow's properties. Even those scholars who seem to accept the general claim *AGV*, accept, a fortiori, the implied inconsistency, and thus in practice drop some assumptions to cope with the real world. Unfortunately, however, the literature has not converged to some agreement on which properties are best to drop. The position of this chapter will be to forward the proposition that the Arrow axiom of Pairwise Decision Making is the culprit to kill. It is a bad axiom for rational collective decision making, since it appears to be incongruent with that very notion itself.

In the following we develop the concepts, give a short proof and discussion of Arrow's Theorem, construct the argument against the claims, reappraise the literature, and conclude.

9.2.2 Basic concepts

See the overview chapter for the basic notations.

There are the following Arrowian axioms:

<i>AWP</i>	the weak Pareto principle
<i>AU</i>	universal domain (wide ranging preferences)
<i>AD</i>	no dictator
<i>APDM</i>	Pairwise Decision Making
<i>a</i>	<i>AWP & AU & AD & APDM</i> .

The Arrow Theorem can be expressed in various equivalent logical forms:

<i>AT</i>	$a \Rightarrow \text{falsum}$
<i>AT'</i>	$a \Rightarrow \neg a$
<i>AT''</i>	$\neg a$
<i>AT'''</i>	$(AWP \& AU \& APDM) \Rightarrow \neg AD$

with *falsum* a contradiction or falsehood and \neg the negation sign. If something leads to a contradiction, then we conclude to the falsehood of the assumptions themselves.

★

There is a Kantian distinction between technical, pragmatic and moral (categorical) imperatives. Utility, as commonly regarded by economists, likely is of the pragmatic kind. Interestingly, theorists on morality have developed something called 'deontic logic', which appears to give many similar results as economic theory. Deontic logic however applies to propositions and not to commodity domains. It is possible, though, to integrate all these kinds of preferences into an integral utility index, when we replace a point x in the commodity domain by a statement "The state of the world is x ". This integral utility index likely would be lexicographic, in that some moral and constitutional issues might dominate pragmatic results in the commodity domain. Thus, while we would use the same symbols R , P and I , we would need to look into the structure of the index to find the Kantian distinction as made by the particular agent. We conclude that we can usefully introduce and apply some terms from deontic logic. Define:

$Ap \Leftrightarrow (\neg p)Rp$ means that p is allowed (at least as good as $\neg p$)

$Op \Leftrightarrow (\neg p)Pp$ means that p is a moral obligation (one ought to p)

An exemplaric deontic result is:

$Op \Leftrightarrow \neg(A(\neg p))$

Deontic logic allows us to translate:

$AMC = Oa$

The use of deontic logic allows a forceful restatement of Arrow's difficulty in social choice:

$Oa \ \& \ \neg a$

★

Let us consider some more properties of morality and deontic logic.

The gap between Is and Ought (Sein und Sollen) means the rejection of $\forall p p \Rightarrow Op$ ('If something is, then it should be like that') and, in principle, $\forall p Op \Rightarrow p$ ('what ought to be is achieved').

Note what this actually means. A statement p has a truthvalue 1 (true) or 0 (false), depending upon the state of the world. A statement Op has a 'truthvalue' 1 (ought) or 0 (not-ought) depending upon one's preferences. Applying the logical calculus for the propositional operators \Rightarrow , \neg , \vee , & thus is a mental exercise, where empirical and preferential statements are first given the common denominator of 'accepting as valid'. Also, it may be that in one case both p and Op are accepted, but the rejection of $\forall p p \Rightarrow Op$ means that it is rejected as a *rule*.

Moral consistency is reflected in the Deontic Axiom:

$$DA \quad \forall p, q (Op \ \& \ (p \Rightarrow q)) \Rightarrow Oq$$

There is some discussion between moral theorists whether *DA* really holds. It may be felt that the logic is not very compelling for empirical relations of dubious causality. However, if $p \Rightarrow q$ reflects a logical truth, then *DA* is commonly accepted.

On reasonableness, it seems a bit better to attach the properties to the agents rather than to the propositions or commodities. Useful axioms then are:

AF feasibility, X is the budget set (rather than the whole space)

ARe agents are realistic (they only consider feasible options, accept *AF*)

I thus agree with Arrow's 1950 statement: "My own feeling is that tastes for unattainable alternatives should have nothing to do with the decision among the attainable ones; desires in conflict with reality are not entitled to consideration." Thus, also, when one point is (socially) most preferred, it is the one consumed.

The most complex property seems to be good old *rationality*. It appears that we better introduce the information set or knowledge base $I(.)$ and state the condition that it must contain the Arrow Theorem. Then:

ARa agents are rational (they accept logic, have a preference ordering, are morally consistent (*DA*), and are educated on Arrow's Theorem ($I(\neg a)$)).

The $I(\neg a)$ condition is a novel aspect, that, however, should not come as a surprise, given what we said in the introduction. There is a difference between a learning process and a result. In a common classroom or used-car-salesman strategy, people are goaded into buying some axioms as reasonable and attractive, and then burn themselves, which teaches them. This may be called rational from the viewpoint of learning. This paper however concentrates on the after-learning-rationality, the kind of rationality that makes learning so worthwhile.

How does Arrow's original approach relate to the inclusion of $I(\neg a)$? Arrow (1950, 1951, 1963) has no incorporation of learning - though he later has written on 'learning by doing' - so it might be that he assumes standard economic rationality. If that would be perfect foresight, then $I(\neg a)$ is implied. However, it is better to hold that Arrow in that period discussed constitutional choice for agents and not by agents. The choice for people then is made by some algorithm or calculating machine. His axioms do not describe educated people involved in constitutional choice. Alternatively put, another new result in this chapter is the widening of the scopes of utility and rationality to the inclusion of knowledge about the constitutional process itself. In that sense the original Arrowian axioms can be called incomplete. Alternatively, if the idea is that these axioms concern educated people, then there is a hidden inconsistency, in that reasonable agents are assumed to regard inconsistent axioms as reasonable. [Note: If we were to put the question to Arrow, my bet is that he likely prefers incompleteness to inconsistency.]

Hence:

$$ARC = ARe \ \& \ ARa$$

[Note: The reference to 'logic' in *ARa* is not without problem, since there are many logics, such as standard, threevalued, fuzzy, intuitionistic logic, and my own scheme of 'the logic of exceptions' (that I use to solve the liar paradox, and Russells and Gödels problems). However, here it suffices to presume standard logic. Note that the earlier version of this chapter (article) used a 'quantor free logic', where the use of a *variable* indicates the 'for all' quantor, and a *constant* indicates the 'there is' quantor. A subtlety is that this quantifier free logic

distinguishes between “Not ($p \Rightarrow q$)”, that is equivalent to “ $p_0 \ \& \ \neg q_0$ ”, and “ $\neg(p \Rightarrow q)$ ”, that is equivalent to “ $p \ \& \ \neg q$ ”.]

9.2.3 Restatement of Arrow’s Theorem

It appears very useful to discuss an example of the problem that has been discovered by the Marquis de Condorcet 1785. Sen (1970) gives a simple example that appears to be presented first by Nanson 1882. A similar example is reproduced below, and I will refer to it as "the Condorcet case". There are three parties and three topics on ballot, and the numbers of seats and the preferences are such that, with pairwise voting and a majority rule, a cycle results: $A < B < C < A$.

Table 1: Condorcet 1785

Party	Seats	Low	Mid	High		A	B	B	C	C	A
Red	25	A	B	C			25		25	25	
Green	35	C	A	B			35	35			35
Blue	40	B	C	A		40			40		40
Total	100					40	60	35	65	25	75
					Win	B			C		A

It is, in all clarity, not that easy to aggregate votes on more than two topics. [Note: That there should be at least 3 topics is actually an axiom that we have taken for granted.] For two topics one can indeed ask for pro and contra, and find a majority (and occasional ties). For more topics, votes will scatter across the topics, and there will often be no clear majority. Therefor, pairwise voting seems to be a good strategy to get the required information on the preferences. However, pairwise voting apparently also causes problems. So, basically, the search is for a strategy without such problems. And that is, basically, also the suggested value of Arrow’s Theorem: that it states that there would be no such good strategy.

However, in the Condorcet case, we may clearly conclude that the cycle primarily means that there is a tie. The situation is in a deadlock, and the group, as a collectivity, is *indifferent*. That there are indifferences or ties, is nothing special. Standard economic analysis allows agents to be indifferent (we even draw indifference curves), so groups should be allowed to be indifferent too. In Condorcet’s case, indifference is even a logical choice, since when we assume something else, then we quickly run into difficulties.

There is the famous case of Buridan's Ass (AD 1358). A donkey stands between two equal stacks of hay, at equal distances. He cannot decide which stack to take, and dies of starvation. The upshot of this parable is that rational beings can devise a decision. Constitutions generally state what happens when there are ties. Commonly the Status Quo persists. (This may happen even if it was one of the topics under ballot, and apparently was rejected at that stage.) Alternatives are that the chairman decides, or points are (re-) negotiated, and one can use dice. It is important to see the difference between *voting* and *deciding*. In two stages, the chairperson first lists the votes, and then only secondly gives the decision with a tick of the hammer. Table 1 essentially gives a voting field, and no decision yet. There are additional rules that translate the field into a unique decision.

We can use Condorcet's case to give a short proof of Arrow's Theorem, restricting our attention to majority voting.

Proof: The group decision in the Condorcet case is indifference, so that $B I C$. Under the axiom of universality we can look at various preference *profiles*, of which Condorcet's case is only one. Now regard the adjusted profile such that the preferences on B and C remain the same, but the preference on A drops to the lowest position. The profile thus is $\{\{A, B, C\}, \{A, C, B\}, \{A, B, C\}\}$. Since the preferences on B and C have not changed, the APDM outcome on B and C should be the same. Majority voting now however results into $B P C$ which differs from $B I C$. Contradiction. Thus there is a counterexample to the axioms. So the axioms are inconsistent. **Q.E.D.**

The merit of this short proof is that it clearly shows the awkwardness of the APDM. In Condorcet's case the conclusion $B I C$ is a sound decision, and in the case of the adjusted example the conclusion $B P C$ is sound too. That preferences outside of the pair B and C have changed is *vital* to the group decision, since the shift helps a change from clear indifference to clear preference. The preferences on other topics are quite relevant, and not 'irrelevant'. APDM excludes vital information about the preferences - to be precise: it destroys information that exists - and it should come as no surprise that paradoxes and inconsistencies arise. The APDM is incongruent with the notion of group decision making. Perhaps an individual can exclude information about other topics, but this can be doubted, and a group certainly cannot. (Or an individual brain that works as a group cannot.) It is a surprise that APDM has not been killed right in 1951.

The following sections use formal logic.

Note: Considering this proof in *Mathematica*:

- Take Condorcet's case and find the pairwise majority decision.

```
Condorcet[]; PairwiseMajority[]
```

```
VoteMarginToPref::cyc : Cycle {C, A, B, C}
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -0.2 & 0.5 \\ 0.2 & 0 & -0.3 \\ -0.5 & 0.3 & 0 \end{pmatrix}, \right.$$

```
1 → {StatusQuo → A, Sum → {1, 1, 1}, Max → 1, No Condorcet winner → {A, B, C},
  Pref → Pref({A, B, C}), Find → {A, B, C}, LastCycleTest → True, Select → A},
N → {Sum → {0.3, -0.1, -0.2}, Pref → Pref(C, B, A), Select → A}, All → A}
```

- Rework the preferences, so that "A" drops.

```
p = ListToPref /@ Preferences
```

```
{Pref(A, B, C), Pref(C, A, B), Pref(B, C, A)}
```

```
pnew = p /. Pref[x____, "A", y____] :> Pref["A", x, y]
```

```
{Pref(A, B, C), Pref(A, C, B), Pref(A, B, C)}
```

- Set the new preferences, and find the pairwise majority decision.

```
pnew // SetPreferences;
```

```
PairwiseMajority[]
```

$$\left\{ \text{VoteMargin} \rightarrow \text{VoteMargin} \begin{pmatrix} 0 & -1. & -1. \\ 1. & 0 & -0.3 \\ 1. & 0.3 & 0 \end{pmatrix}, \right.$$

```
1 → {StatusQuo → A, Sum → {0, 1, 2}, Max → 2, Condorcet winner → C,
  Pref → Pref(A, B, C), Find → C, LastCycleTest → False, Select → C},
N → {Sum → {-2., 0.7, 1.3}, Pref → Pref(A, B, C), Select → C}, All → C}
```

9.2.4 A lemma

Lemma I: *AF* implies that a SWF-GM p satisfies the property $Op \Rightarrow p$.

First proof: *AF* means that desires (Op) in conflict with reality ($\neg p$) are not entitled to consideration. But $\forall p \neg(Op \ \& \ (\neg p))$ is equivalent to $\forall p Op \Rightarrow p$. Q.E.D.

Second proof: We already concluded that the most preferred point (Op) would also be the chosen point (p). Thus $\forall p Op \Rightarrow p$. (If the point is not preferred, then the implication is true ex vacuo.) **Q.E.D.**

Discussion: We have enlarged the domain with SWF-GMs, and hence the axiom of feasibility becomes a bit stronger. The extension itself is rather weak, since we only extend on consistency (and not empirical validity). Our criterion is that a reasonable society would stick to its rules. The gap between Is and Ought still exists in principle, but can in practice be bridged by the human effort to attain one's ends.

9.2.5 Rejection of the Arrow Moral Claim (AMC)

Theorem A.1: For a reasonable society, the AMC is invalid.

First proof by rationality & moral consistency (DA): Assume Oa . But $a \Rightarrow \neg a$, and with DA we get $O\neg a$. But this gives a preference inconsistency $Oa \& O\neg a$. Hence $\neg Oa$. **Q.E.D.**

Second proof by rationality & moral consistency (DA): Assume Oa . Since $a \Rightarrow \text{falsum}$ we find $O\text{falsum}$. Thus for some p_0 we have $O(p_0 \& \neg p_0)$. But this means $Op_0 \& O\neg p_0$, and that is a preference inconsistency. Hence $\neg Oa$. **Q.E.D.**

First proof by realism (AF): Assume Oa . By the lemma $\forall p Op \Rightarrow p$ we find a . But then we have $\neg a \& a$, which is an inconsistency. Hence $\neg Oa$. **Q.E.D.**

Second proof by realism (AF): Since $\neg a$ and above lemma $\neg a \Rightarrow \neg Oa$, hence $\neg Oa$. Thus the axioms are not morally desirable either. **Q.E.D. Note:** $q \Rightarrow p$ is equivalent to $\neg p \Rightarrow \neg q$, and we may take $q = Op$.

9.2.6 Rejection of the Arrow Reasonableness Claim (ARC)

Theorem A.2: For a reasonable society, the ARC is invalid.

Proof: Given AF, infeasible choices are not considered. Since $\neg a$, apparently a is not feasible, and the Arrow SWF-GM is not reasonable. So it is invalid that the axioms would be reasonable. **Q.E.D.**

Discussion: As we stated above, we have enlarged the domain with SWF-GMs, and hence the axiom of feasibility becomes a bit stronger. The extension itself is

rather weak, since we only extend on consistency (and not empirical validity). But the conclusion is strong. No reasonable society in its right mind would want to accept Arrow's axioms as its SWF-GM. Supposedly at a chaotic Boston Tea Party a SWF-GM $g = a$ might be tried, but pretty soon rational people would see that they should make another constitution, for otherwise the situation will remain chaotic, and the Tea Party will not go down into history as a notable event.

Note that Arrow adopts feasibility, but also wants to impose infeasible conditions.

9.2.7 Selection of the culprit axiom

The selection of the culprit axiom is straightforward. We order the axioms by preference, for example $AU > AWP > AD > APDM$. From $\neg a$, we conclude that we have to drop one of the axioms. We drop the least preferred one.

Lemma I: If all agents have $[AU, AWP, AD]_i > APDM$ then, with AWP , society has $[AU, AWP, AD] > APDM$. Note: here $[x, y, z]$ means the unordered set.

Proof: *obvious*.

Discussion: When all people put AU , AWP and AD in any individual order, but all would have $APDM$ below these, then society can reject $APDM$ unanimously. In fact, the condition AU might as well be regarded as part of the *definition* of a SWF-GM, and similarly, AWP could as well be regarded as part of the *definition* of the notion of collective preference. So the real choice concerns AD and $APDM$. Here only a selfish dictator and his associates would have $\neg AD > APDM > AD$, so the real choice is between dictatorship or not. The Jorgenson quote points to his preference for a benevolent and non-selfish dictatorship, but, also since such dictatorships tend to turn sour, my impression is that he would be an associate of a real dictator. Most likely, he did not understand the situation when the quote was printed.

My discussion on Condorcet's example should generate support for the rejection of $APDM$. Basically though, scientists can only advise on preferences, and the proper decision is up to the body politic.

Note that ordering the axioms means that the deontic predicate O is not homogeneous. This means that deontic logic may be more related to preference theory than deontic theorists think.

9.2.8 Examples of consistent constitutions

Consistent constitutions violate one of the axioms of Arrow's Theorem. Violating one of these axioms is to be considered useful for reasonableness and morality, rather than the reverse. (That is what we proved above.)

One general feature is a Status Quo that persists when there are ties.

One example already has been mentioned in the discussion of the Condorcet problem. With majority voting, a cycle means indifference, and there are various ways to solve ties. One possible solution is the persistence of the Status Quo.

Another example constitution is the "Pareto-Majority" rule. One first selects all Paretian improvements from the Status Quo. That is, those points where some advance while nobody loses. There may be more Paretian points, such as $B > A$ and $C > A$, with A the Status Quo. When there is no Paretian order between B and C , then it suffices to decide on these points by simple majority. Of course, with more than two points, majority voting can result into cycling, but that again means indifference, which could be settled by dice, by the chairperson, or by other creative ways.

Examples of such consistent constitutions have been implemented here in *Mathematica*.

Note that these implementations, as there are now, are Social Decision Functions (which selects the top choice) rather than SWF-GMs (which gives the whole ordering). Theoretically, the one can be transformed into the other, however - see below.

9.2.9 A reappraisal of the literature

Our discussion arrives at a conclusion that differs from the literature, and thus warrants a reappraisal of that literature. This reappraisal is not the topic of this chapter, but some examples are useful.

(1) Note that the Tobin quote above was misleading. The problem with 'unborn generations' should not be mixed up with the Arrow difficulty. The Tobin problem actually can have a rather simple solution. It are the preferences of the currently living that matter, and what they prefer for the future unborn (which can also be based on a forecast of such preferences). These future preferences cannot logically be included, since they don't exist yet.

(2) Arrow 1951 also stated:

"If consumers' values can be represented by a wide range of individual orderings, the doctrine of voters' sovereignty is incompatible with that of collective rationality."

This is clearly inaccurate. The statement suggests that we have to adopt Arrow's axioms, while the sensible thing is to reject these axioms and to adopt both voters' sovereignty and collective rationality.

(3) One of the more interesting points made here is the distinction between the learning process and the end result. How should Arrow's result be presented in the future ? Is it possible to maintain the teaching strategy to call the axioms 'reasonable', then have the students get into a fixtured, and then let them find a way out ? It is good teaching practice ! However, in a Palgrave meant for a wider audience (or a general encyclopedia that even might be read by dictators), it might be improper to call Arrow's axioms 'reasonable'. It should be 'seemingly reasonable' at the least.

Note that the phrase then becomes less enchanting:

'there is no social choice mechanism which satisfies a number of seemingly reasonable conditions'.

(4) I am a bit shocked by Mueller's (1989, p406-407) discussion of Arrow's general view. One would expect a more critical attitude, but finds instead:

"The Arrow and Sen theorems (...) raise fundamental questions about the possibility of establishing collective choice procedures satisfying minimally appealing normative properties (...) But the negative side should not be overemphasized. We have suggested that both sorts of paradoxes might be avoided with the use of cardinal, interpersonally comparable utility information. Arrow explicitly eschewed the use of such information, and the independence of irrelevant alternatives [thus Pairwise Decision Making / TC] axiom was imposed to rule out voting procedures that might make use of such information (... But it) is possible that the citizens may be trusted to make these comparisons in an ethically acceptable way."

Well, interpersonal comparison of course occurs, minimally, when we assign votes to people, assign rights to put topics on ballot, and the like. So interpersonal comparison is not as bad as many economists seem to think. But my solution to Arrow's difficulty does not rely on cardinality and cardinal comparison. So, disappointingly, Mueller both accepts the idea that Arrow would cause 'questions' about the possibility of social choice, and he comes with a wildly wrong conclusion. This is supposed to be a modern textbook !

(5) What is important, is that the development of economic theory and the development of real economies have been hindered by the confusion generated by the standard explanation. Where decision makers were divided, some interested in social welfare and others not, the latter group was provided with decisive gunpowder - and beware of people who have an ideology and even wield a mathematical theorem to prove their lunacy. Generations of students have been taught by Nobel Prize laureats that research into social welfare would be subject to impossibilities. Creative energy has been directed to enlarging the impossibilities rather than to devising structures that might improve practical situations. Practical research into social choice functions and parameters has been aborted, all with reference to a misunderstood theorem !

Economic research also leads to a suggestion of a constitutional amendment, see Cool (1996b, 2000). I hope that this present chapter helps to clarify that this kind of research is a useful type of economics.

(6) This analysis also clarifies a confusion about the relation of the SWF-GM and constitutions to the SWF. While many economists argued that constitutions could not be reasonable or morally acceptable, they did accept the Bergson-Samuelson SWF, even though the latter was derived from the SWF-GM - and nobody seems to care about this inconsistency. Which is now removed, since the properties of the SWF-GM are projected into the SWF.

(7) It is relevant to note that I gave this analysis earlier, in Cool (1990c, 1992a). This chapter is almost 99% the same as 1997b, and a rephrasing of the main principles. I have had no success so far in getting a publication, neither at the CPB nor in a journal.

[Note: Discussion (evaluation and thus eventual publication) of (1990c) was blocked by the CPB directorate with the comment 'this issue exceeds the CPB

intelligence' - which was inconsistent since I worked there. The EER referee reports of (1997b) are nonsense too.]

9.2.10 Conclusion

Arrow's Theorem has given some problems in the literature, see the quotes above. We have achieved the following solution:

- There is more clarity now, by the distinction between the theorem proper ($a \Rightarrow \text{falsum}$), the moral claim (Oa) and the claim on reasonableness (AF and $I(\neg a)$).
- From a mathematical point of view, the Arrow axioms are incomplete for decision making in a reasonable society.
- It has been shown that the *APDM* is undesirable. Dropping *APDM* is not a sad state of affairs, as is sometimes suggested in the literature, but a sign of understanding group decision making.
- The Arrow axiomatisation does not capture the truly desirable properties required for neither constitution nor SWF-GM, both by incompleteness and *APDM*.
- There are detail results, such as the distinction between voting and deciding, the integration of preference theory and deontic logic, and a proof of Arrow's Theorem that shows clearly the abuse by *APDM*.
- We have given examples of consistent constitutions that many might regard as optimal.

9.3 Constitution and SWF-GM

9.3.1 Ordering vs Choice Set

Below, we will define a generator $g(c \mid B)$ based on the SDF c for budget B . The difference between generator $g(c \mid B)$ and generator $g(X)$ (SWF-GM) arises from the point that the first depends upon the budget set, while the second is supposed to be valid for the whole commodity domain. We could consider $g = \text{Limit}[g(c \mid B), B \rightarrow X]$. In some sense this is a theoretical point, since it is not specified that we could not take $c(X)$ in the first place.

For a SWF-GM we find:

- A SWF-GM implies an constitution, since the constitution can take the top of the ordered list, so that $c(B, R_1, \dots, R_n) = C(B, g(R_1, \dots, R_n)) = C(B, R)$.
- SWF-GMs can give orderings $g(S)$ on $S \subset X$. A question however is whether they are consistent with one another.

A constitution (Social Decision Function (SDF)) generates basically only the choice set $C(B)$ of $B \subset X$. We could write S instead of B , but the idea is that a constitution takes the budget very serious. A constitution (SDF) seems weaker than a SWF-GM. However, we can find that a SDF also creates an ordering:

- It need not be obvious how a constitution can generate an ordering. Reasonable proposals still can create paradoxes. However, a constitution c with a budget set B gives an implied order. This chapter shows how it can be constructed, and derives the implications.

?ChoiceSet

ChoiceSet is a symbol only. ChoiceSet[S] denotes the set of best elements in $S \subset X$. If a binary relation R is used:
 An element x is best in S , iff for all $y \in S$: $R(y, x)$.
 Read " $y \leq x$ ". In our notation: " $\text{Or}[\text{Pref}[y, x], \text{Pref}[\{y, x\}]]$ "

The question about the consistency of SWF-GM orderings is the mirror question of the question how orderings can be created by SDF's conditional to the budget.

9.3.2 Implied order

Consider the following generator $g(c, B)$ that depends on some constitution c :

1. Top in the list is $C_1 = C(B)$.
2. Next in the list is $C_2 = C(B \setminus C_1)$, which is the best element when we neglect above winner.
3. Next in the list is $C_3 = C(B \setminus \{C_1, C_2\})$, which is the best element when we neglect the results of above.
4. Etcetera.

This thus is the ordering of method 1 of section 9.1.3. Note that tie-breaking rules make sure that always 1 element is selected.

Let us take the example $B = [D, E, F, G]$, which is the unordered set. Suppose that this gives the ordering $c(B) = \text{Pref}[D, E, F, G]$, which is ordered - thus with G the best and D the least. However, if we now take a subset $S = [D, F] \subset B$, then we do not have the guarantee that still $F > D$. It may well be that the constitution generates $F < D$. However, taking this subset should not be confused with taking a new budget. We thus must be more specific that the subset remains conditional to the budget.

In that case we can find a generator that creates consistent orderings for every possible subset of the budget set. In formulas:

$$g(c \mid B) \Leftrightarrow \{g(S \mid B, c) = S \cap g(c, B), \text{ for all } S \subset B\}$$

The meaning of this is that we take subsets now of $c(B)$ and we do not recompute the constitution $c(S)$ (organise a new vote) for $S \subset B$. Hence we have an ordering for every constitution (SDF) and decision situation.

9.3.3 Conditional generator

If we use the whole commodity domain, and construct 'voting fields' e.g. full of cycles, then we find for the group decision:

- that cycles mean indecision or indifference,
- the decision on what is the budget B can affect where cycles arise.

We thus solve the problem by the distinction between *voting* and *deciding*. The budget set determines how the ordering looks like. The budget thus is important for group decision making, so that:

- X should rather be interpreted as the budget set, $X = B$. We are not just interested in the social ordering over X , but rather on the orderings over the set of subsets of X (i.e. 2^X).
- the ordering created by a constitution (SDF) is conditional to the budget set.

Note that $g(c \mid B)$ 'imposes' an ordering on subsets that could give different results if the budget would be different. This however is entirely logical, and not necessarily undemocratic.

PM. There is some circularity in that a group decision on what actually constitutes the budget set can affect the result as well. Like the chicken & egg problem, this requires either dynamics or fixed points.

9.3.4 Ratio of taking subsets

If we take subsets, then there are two ways of doing this:

- If we are not considering an alternative budget set, then the order remains conditional to the budget.
- If we consider an alternative budget, then we should take into account that the ordering will change, since the joint preference depends upon the budget.

The selection of subsets other than B is something which a mathematician might propose, but it is not by itself relevant for the decision making process. In the decision making process we do not necessarily take subsets, and it is questionable as well whether we would do this using the original whole constitution again - while treating that subset as if it were the new budget.

If one item turns out to be unavailable, or if one candidate for office drops out of the race, then the budget set becomes smaller, and then the group decision can be affected. However, if this is during the race, then the vote has not been held yet, and thus it is of little consequence. If it happens after the vote has been held, then the proper view is that the vote was on the given budget, and not on the new budget (without that disappearing item). In both cases there is no problem - though the result might seem paradoxical.

- In the Condorcet case, item *A* would be chosen as the best fixed point.

```
Condorcet[] ; BordaFP[]
```

A

- The day after the elections, *C* gets a heart attack. *C* lost, so one would think that *A* still is the winner. However, if elections would be held now, *B* would win.

```
SelectPreferences[{"A", "B"}];
```

```
BordaFP[]
```

B

The latter might be called paradoxical, but it is part of the game. *C* was a fixed point, a strong contender who drew votes. Perhaps he or she could be replaced by a similar contender, but that depends upon the situation.

If people are interested in what the ordering would be (conditional to a budget), then we reduce the number of paradoxes by adopting voting schemes that are less sensitive to preference reversals. Voting schemes like BordaFP, that are more robust, can be used to check whether an item, that threatens to drop out of the race, is a fixed point. However, we would do all this only if the ordering would be important. Normally, it is not.

9.3.5 Conditional generator and SWF-GM

The distinction between a constitutional generator $g(c \mid B)$ and a commodity space generator $g(X)$ is rather academic. For practical purposes we never use the $g(X)$ and we normally use the constitution (SDF) with a budget.

It is true that this discussion makes us more aware of the possibility of preference reversals when the budget would change. Parliaments would be wise to consider alternative budgets. But we arrive at this insight not because of Arrow's Theorem, but because of the discussion above. Arrow's Theorem and its interpretation have actually hindered us in achieving this clarity.

9.4 Renaming and rejecting APDM

9.4.1 Introduction

Arrow (1963) introduced an axiom the "Independence of Irrelevant Alternatives" (*IIA*) that has caused much misunderstanding. That axiom now has been baptised the "Axiom of Pairwise Decision Making" (*APDM*). Thus the axiom remains the same, only the name is different. The new name is much clearer about what the axiom really means in normal English.

Since the name "IIA" is so entrenched in the literature, this change of name requires some explanation. The explanation is along the lines:

- There is the distinction between voting and deciding.
- Items that cause cycles cannot be called 'irrelevant' for decision making.
- The criterion to separate the relevant items from the irrelevant ones is rather the budget and is not necessarily found in pairwise voting for all items.

9.4.2 The axiom

Sen's (1970:41) gives us the following definition for what he calls, after Arrow, "Independence of Irrelevant Alternatives", but what we will call the "Axiom of Pairwise Decision Making" (*APDM*). We use $g(X)$ where Sen uses the symbol f for the collective generator, which also clarifies that the commodity space is used.

APDM: Let R and R' be the social binary relations determined by function $g(X)$ corresponding respectively to two sets of individual preferences, $\{R_1, \dots, R_n\}$ and $\{R'_1, \dots, R'_n\}$ on the commodity space X . If for all pairs of alternatives x, y in a subset S of X , $x R_i y \Leftrightarrow x R'_i y$, for all i , then $C(S, R) = C(S, R')$.

We note immediately that this axiom implies:

- $APDM \Rightarrow$ there is deciding (and pairwise comparison is not used for the construction of a voting field only).
- $APDM \Rightarrow$ there is no recognition of conditional dependence on $X = B$, but the axiom takes $X = \text{commodity domain}$.

Note that the 'pairwise comparison' in itself follows from the definition of a binary relation R . Two relations R_i and R'_i are the same on S if the choices on all the pairs are the same. If people make the same choice for a subset of issues, then their preferences are the same. This seems innocent enough. But if preferences are the same on a subset of issues, then it is dangerous to conclude that preferences would agree also on a larger set. And if the preferences are not the same for the larger set, then we should be aware of the possibilities of cycles and preference reversals.

The problem arises from a way of aggregation that neglects the budget set. We should take $X = B$, or, we could take $S = B$ and then look at $S' \subset B$. And in both cases the choice sets would remain conditional on B . If the profiles are the same for B , then $g_1(B) = g_2(B)$ and then the conditionals orderings for all possible subsets are the same. If the profiles would be different for B , then they might generate different orderings - and then we know that the cause is that they are different over the whole of B . (See section 9.4.3 below.)

Pairwise comparison under these conditions (such that it neglects the condition of the budget set) is sufficient to cause paradoxes. Note that it is not pairwise-ness per se. We likely might also create paradoxes by comparison of three items, or another number, limited by the budget size itself. So, strictly speaking, it is not pairwise comparison by itself that is the root cause. The root causes are the confusion of voting and deciding and the confusion of the commodity domain and the budget set. But the *APDM* can be seen as a stand-in for that general problem. In itself, however, the *APDM* name is quite apt. If this axiom were valid, then one could construct the whole aggregate ordering from pairwise comparisons. This justifies its name *APDM*.

9.4.3 Why we can reject APDM

Let us suppose that we have a budget set B and two profiles $\{R_1, \dots, R_n\}$ and $\{R'_1, \dots, R'_n\}$ to the effect that we have two orderings $g_1(B) \neq g_2(B)$. Let us take a subset $S \subset B$ for which the pairwise preferences of the voters are the same, so that the preferences are the same for the subset, so that $g_1(S) = g_2(S)$ if S would be the budget. Then we have:

$$g_1(S \mid B) = S \cap g_1(B) \neq g_1(S) \quad (\text{in principle})$$

$$g_2(S \mid B) = S \cap g_2(B) \neq g_2(S) \quad (\text{in principle})$$

The conditional orderings need not be the same either, since the overall orderings for the whole budget set would be different. Hence *APDM* conflicts with a reasonable way of ordering the items.

9.4.4 On an example by Sen

Sen (1970:37) tries to clarify *APDM*. He writes: (a) "To give an analogy, in an election involving Mr. *A* and Mr. *B*, the choice should depend on the voter's orderings of *A* vis-a-vis *B*, and not on how the voters rank Mr. *A* vis-a-vis Lincoln, or Lincoln vis-a-vis Lenin." and (b) "Views on Lincoln or Lenin could enter the picture (indeed *must* do so) if and only if the voters' orderings of *A* vis-a-vis *B* should themselves change as a consequence of a revision of opinion on Lenin or Lincoln."

Well, Lincoln and Lenin are not in the budget set any more. I think that this is a perfectly acceptable reason to call these items 'irrelevant'.

It is also a point of consideration that we should not confuse this issue with the phenomenon of preference reversal that can occur when we consider a highly relevant Mrs. *C*. The BordaFP scheme has been developed to deal with preference reversals on less relevant items.

9.4.5 Discussion

In some respects the discussion can become very confusing at this point. It is reasonable that people neglect farfetched possibilities. Arrow's axioms result into something like that. Does this make his axioms reasonable?

Arrow's axioms on using the whole commodity domain and universal preferences introduce the possibility that we might also be obligated to consider farfetched items. Arrow introduced the *APDM* to limit this effect again, since it allows that a decision on our current issues can be taken independently from other farfetched possibilities. Thus Arrow on one hand opens the door wide for such farfetched possibilities, and on the other hand introduces a strict condition that kills the relevance of this. The whole looks

reasonable, since people in fact neglect farfetched possibilities. (Farfetched would e.g. be the future when chickens have evolved and have developed a preference for what they consider to be humanoid worms. These axioms require that our Parliaments should take this into consideration when they now want to decide on your water bill. *APDM* prevents this - which might make it seem 'reasonable'.)

Yet, the whole does not conform with the practical situations in Parliaments, where the problem is defined for existing voters and where the issues on table are given by the budget set.

A defence of Arrow's approach is to say that it is just a logical exercise, to show that some axioms result into a contradiction. I am quite happy with that point of view. But my problem was the claim that the axioms would be reasonable and morally desirable.

If we want to deal with possibly farfetched preferences of some citizens, which is the moral meaning of the axiom of universal preferences, then I think that we should work towards practical procedures that work. Assuming inconsistent axioms is not a good way to deal with that moral question.

9.4.6 Survival

Since Arrow's Theorem is so convoluted, there is the question how it should survive in Social Choice Theory. I think that the best version would be a theorem that pairwise decisions can be taken iff orderings would be independent from the budget. This of course would be a condition on the preferences. In some cases the preference profiles allow this to happen.

I would like to warn against versions of Arrow's Theorem that would replace *APDM* with something like 'assume independence of the budget set' - and then deduce some impossibility. Such versions would still express respect for the brilliance of the achieved impossibility. I would prefer the true story of accurate mathematics - as far as that goes - and bad economics.

9.5 Subset Consistency and Fully-Matched-ness

9.5.1 Introduction

Sen (1970:17) introduces properties α and β , which we will give specific names:

- Condition $\alpha = \text{SubsetConsistency}$. For example: If the world champion is a Pakistani, he must also be the champion in Pakistan.
- Condition $\beta = \text{FullyMatched}$. For example: If one champion of Pakistan is a world champion, then all champions of Pakistan must be world champion as well.

The implied ordering for a constitution (SDF), that it is conditional on the budget set, satisfies the conditions of subset consistency and fully matched-ness, since it is an ordering.

The currently defined constitutions (SDFs) like Borda or ParetoMajority however are unconditional. Thus we should be able to show that they do not satisfy these properties. The routines `SubsetConsistency` and `FullyMatched` have been written to show this. These routines test single situations, and they do not search for general properties. They thus are useful to test counterexamples. These routines thus also are paradox-prone.

9.5.2 Condition Alpha: Subset Consistency

9.5.2.1 Definition

Sen p17 calls property α a basic requirement of rational choice):

$$(x \in S_1 \subset S_2) \Rightarrow (x \in C(S_2) \Rightarrow x \in C(S_1))$$

A better formulation is:

$$(x \in C(S_2)) \Rightarrow (x \in S_1 \subset S_2 \Rightarrow x \in C(S_1))$$

Sen also shows that if we have an ordering, then it is subset consistent. Since we showed that $g(c \upharpoonright B)$ is an ordering on B , it satisfies subset consistency.

However, the constitutions (SDFs) that we have defined lack the property. We can use this routine to show this:

?SubsetConsistency

SubsetConsistency[S1_List, S2_List, SDF] for lists
 of items $S1 \subset S2$, tests whether $(S1 \cap C[S2]) \subset C[S1]$.
 Here $C[S]$ is the choice set over S . The routine now uses $SDF[S]$ but
 this is the unconditional choice set, and properly speaking the
 choice set should be used that is conditional to the budget set.
 SubsetConsistency is condition α in Sen (1970:17): If the world
 champion is a Pakistani, he must also be the champion in Pakistan

9.5.2.2 Counterexample: Borda is not Subset-Consistent.

We can check that Borda does not satisfy this property of subset-consistency - because of preference reversal. Namely, let x be the Borda winner, let FP be the fixed point selected by BordaFP, and let y be the irrelevant item that causes the difference between Borda and BordaFP. For the Borda constitution (SDF), $x \in C(S_2)$. Let $S_1 = \{x, FP\}$ and then $\{FP\} = C(S_1)$. But then $(x \in S_1 \subset S_2) \ \& \ \text{Not}[x \in C(S_1)]$.

Consider this example of preference reversal.

```
EqualVotes[]; DefaultItems[];
{{3, 2, 1}, {3, 2, 1}, {1, 3, 2}} // SetPreferences;

b = Borda[]

{A, B}

BordaFP[]

BordaFP::chg : Borda gave {A, B}, Fixed Point is {A}

A

SubsetConsistency[{"A", "B"}, Items, Borda]

CheckVote::adj : NumberOfItems adjusted to 2

CheckVote::adj : NumberOfItems adjusted to 3

{ChoiceSet({A, B}) → {A}, ChoiceSet({A, B, C}) → {A, B},
 Intersection[S1, C[S2]] → {A, B}, MemberQ → {True, False}, Condition → False}
```

9.5.2.3 Counterexample: current BordaFP is not Subset-Consistent

The Condorcet example gives a counterexample for BordaFP. The reason of course is the tie-breaking rule, that cannot take into consideration that the budget might be changed in all kinds of directions. Let $S_2 = \{A, B, C\}$ and $S_1 = \{A, B\}$. $C(S_2) = \{A\}$. Hence $\{A\} = C(S_2) \cap S_1$. The question is whether $A \in C(S_1)$. It appears however that $C(S_1) = \{B\}$.

```
Condorcet[]; SubsetConsistency[{"A", "B"}, Items, BordaFP]
```

```
CheckVote::adj : NumberOfItems adjusted to 2
```

```
CheckVote::adj : NumberOfItems adjusted to 3
```

```
BordaFP::set : Local set found: {A, B, C}
```

```
BordaFP::chg : Borda gave {A}, Fixed Point is A
```

```
{ChoiceSet({A, B}) → {B}, ChoiceSet({A, B, C}) → {A},
```

```
Intersection[S1, C[S2]] → {A}, MemberQ → {False}, Condition → False}
```

Note that the concept of Fixed-Point-ness uses the 'alternative match when the winner does not participate' i.e. $S_1 = S_2 \setminus C(S_2)$. For this subset S_1 , condition α is satisfied ex vacuosi. In a sense, this is an interesting relationship between FP-ness and condition α . But this is limited to just this specific subset, and it is not general.

9.5.2.4 Budget-Conditional-BordaFP would be Subset-Consistent

The reason why the Condorcet case is a counterexample for BordaFP, is that selecting just part of a cycle causes the 'indifference by indecision' to disappear. However, once we have established that A and B are a cycle at the aggregate level, but that A is the overall winner, then we can declare this also for all subsets, and then the ChoiceSet of $\{A, B\}$, conditional on the whole ordering, would be $\{A\}$ again.

This means, that if the budget was $\{A, B, C\}$, then there should be a memory for the aggregate result over the budget set, and a true comparison is conditional to the aggregate result. The true ordering is defined to be consistent over subsets, rather than that this would be an independent condition that needs to be tested.

In this case, the BordaFP routine has been written to consider only the items under review, and it neglects the larger budget. The fact that this routine just

works so, should not cause us to think that we could not write a different routine, one that remembers the aggregate result. Indeed, precisely the condition of Subset Consistency could be used to create overall rationality.

`VoteToPref` allows us to create an order that is conditional to the budget. BordaFP cannot be used here directly, since its output is not a list containing the Select key, such as {..., Select \rightarrow A, ...}. It would be simple to write a routine BordaFP2 that has this output format. However, `VoteToPref` uses ParetoMajority, and this relies on BordaFP, so basically we have shown that it is possible to construct an ordering conditional to the budget.

9.5.2.5 Relation to *APDM*

Sen (1970:39) gives an example how a preference change on an 'non-essential' item causes a change in the collective choice. This is basically the preference reversal situation discussed above.

An example of preference reversal is not necessarily a proof that *APDM* is required. Of course, it shows that *APDM* is violated, but, it does not show that *APDM* is the only condition that can prevent a preference reversal.

Thus, preference reversal and *APDM* are independent concepts, since we can solve such preference reversal for a given budget set by BordaPF, while BordaFP violates pairwiseness. So the problem of preference reversal is solved, and we do not have to require *APDM*. BordaPF cannot solve the phenomenon, however, that the aggregate ordering could change if the budget changes.

The fixed point method uses a sieve to find the relevant items - and eliminates the non-essential ones. In that sense, it is a re-interpretation of "independence of irrelevant alternatives". The fixed point method filters in a similar way as Pareto: by using some criterion of 'dominance'.

There is with Sen too much the suggestion that *APDM* would be reasonable and that it would catch the notion of 'irrelevance'.

Sen (1970:17, footnote 9) notes that condition α has been called "Independence of Irrelevant Alternatives" by Nash 1950, Luce & Raiffa 1957 and others. The idea would be that adding irrelevants should not affect the decision. This would point to another definition of "independence of irrelevant items". If we

call items irrelevant when they are dominated by a BordaPF winner of the budget set, then increasing the budget set with more of those items would not matter. (But items that are fixed points themselves, would matter.)

9.5.3 Condition Beta: Fully Matched

Let us use $C_i = C(S_i)$. Sen p17 has, for $S_1 \subset S_2$: $(x, y \in C_1) \Rightarrow ((x \in C_2) \Leftrightarrow (y \in C_2))$

This is equivalent to: $S_1 \subset S_2 \Rightarrow (C_1 \cap C_2 \neq \{\} \Rightarrow C_1 \subset C_2)$

This condition is quickly fulfilled if there is only 1 winner in each C_i .

?FullyMatched

FullyMatched[S1_List, S2_List, SDF] for lists of items $S_1 \subset S_2$, tests whether, when $(C[S_1] \cap C[S_2]) \neq \{\}$, then $C[S_1] \subset C[S_2]$. Here $C[S]$ is the choice set over S . The routine now uses $SDF[S]$ but this is the unconditional choice set, and properly speaking the choice set should be used that is conditional to the budget set. FullyMatched is condition β in Sen (1970:17): If one champion of Pakistan is a world champion, then all champions of Pakistan must be world champion as well

DefaultItems[]; Condorcet[];

Borda[]

A

FullyMatched[{"A", "B"}, Items, Borda]

CheckVote::adj : NumberOfItems adjusted to 2

$\{ChoiceSet(\{A, B\}) \rightarrow \{B\}, N \rightarrow 1, Condition \rightarrow True\}$

I have not spent the time yet to find a good (counter-) example.

But we can keep the same principle in mind as for Subset Consistency: the conditional constitutional ordering can be defined such that this holds.

9.5.4 Reproduction of Sen (1970:39) on APDM

The following can usefully be noted about the traditional view on APDM. We compare two situation, in which there is only a change for an 'irrelevant' item. Below we will develop the situations for preferences of 3 persons on 3 items, x , y , and z . Then Sen (1970:39) argues:

"While everyone's ordering of x and z are still the same, the social choice between x and z is not the same, and this of course violates condition I." (Which is axiom *APDM*.)

We can solve this situation by BordaFP. This violates *APDM*, but still solves the situation. Thus *APDM* is not required to prevent preference reversals by themselves.

■ Situation 1

```
Clear[x, y, z]; EqualVotes[]; Items = {x, y, z};
```

```
lis = {Pref[z, y, x], a = Pref[y, x, z], a}
```

```
{Pref(z, y, x), Pref(y, x, z), Pref(y, x, z)}
```

```
PrefToList[lis] // SetPreferences;
```

Preferences

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 1 & 3 \\ 2 & 1 & 3 \end{pmatrix}$$

Borda[]

```
{x, z}
```

BordaFP[]

BordaFP::chg : Borda gave {x, z}, Fixed Point is {z}

z

StrategicPref[Borda, 1]

StrategicPref::non : Strategy useless, iter 1: item x is the best result, also honestly

```
{Borda → {x, z}, Out → {3, 2, 1}}
```

■ Situation 2

```
lis2 = {Pref[y, z, x], a, a}
```

```
{Pref(y, z, x), Pref(y, x, z), Pref(y, x, z)}
```

```
PrefToList[lis2] // SetPreferences;
```

Preferences

$$\begin{pmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 2 & 1 & 3 \end{pmatrix}$$

Borda[]

z

BordaFP[]

z

StrategicPref[Borda, 1]

StrategicPref::str : Iter 1: A strategic vote will give item x in the solution

$\{\text{Borda} \rightarrow \{z\}, (3 \ 2 \ 1) \rightarrow (x \ z)\}$

- Note that BordaFP now works against cheating.

StrategicPref[BordaFP, 1]

BordaFP::chg : Borda gave {x, z}, Fixed Point is {z}

StrategicPref::non : Strategy useless, iter 2: item z is the best result, also honestly

$\{\text{BordaFP} \rightarrow \{z\}, \text{Out} \rightarrow \{3, 1, 2\}\}$

9.6 The possibility of a Paretian Liberal

9.6.1 Introduction

This book's moral position on the Pareto principle is that you yourself should decide whether you adopt it for some situation or not. It can however be observed that many people appreciate it on various occasions, and hence it is useful to discuss its properties (and to design computer programs for it).

It frequently occurs that society limits Paretian liberty, and imposes norms and values. This can take the form of a law, e.g. when it is forbidden to sell alcohol to people younger than 16. It also happens more informally when in one society person *A* is allowed to take offence of choices made by person *B* - e.g. on the length of his hair. What one society would consider a purely personal affair, another society might have a norm on.

Amartya Sen (1970) came up with an argument that created doubt on the 'possibility' of Paretian liberty. The implication of his argument could be that society always has to impose some norms - or there could be more disturbing logical complexities. It turns out that Sen's argument suffers from the same problem as Arrow's argument, i.e. there is a correct mathematical deduction that premisses imply a consequence, but the verbal conclusions are wildly off. The best way to summarise the situation is that Sen gives a wrong implementation of Paretian liberty.

Since the literature contains many restatements of his argument, you are likely to come across a version that confuses the situation in one way or another. For this book, it would be a wrong conclusion to think that there would be something wrong with the application of the Pareto principle or Pareto routines. Hence it appears useful to consider Sen's argument, and to clarify the situation.

9.6.2 The moral situation

Economics [Logic`Deontic]

Cool`Logic`Deontic`

Allowed	MoralConclude	Ought	ToNotAllowed
AllowedQ	MoralSelect	OughtQ	ToOught
DeonticAxiom	NotAllowed	SetDeontic	
Freedom	NotAllowedQ	ToAllowed	
FreedomQ	NotOught	ToFreedom	

Let us assume that society always imposes some r and does not allow some s . Define also some p and q on which a moral stand might be taken.

p = "Mr. A reads a copy of *Lady Chatterly's Lover*"

q = "Mr. B reads a copy of *Lady Chatterly's Lover*"

The possible combinations are:

$$\begin{array}{ccc} q & \neg q \\ p & d & a \\ \neg p & b & c \end{array}$$

Sen considers the combinations $a = p \ \& \ \neg q$, $b = \neg p \ \& \ q$ and $c = \neg (p \vee q)$. In his example there is only one copy of the book and it can be read only once, so that $d = p \ \& \ q$ cannot be considered. It is no deviation from the notion of liberty, however, to allow for a multiple read, and we will do so - until we get to Sen's specific example and temporarily conform with it for the sake of the argument. Note that if for example (A : $\neg p > p$) and (B : $q > \neg q$), and no other preferences, then b would be a solution.

We can consider three different cases.

(1) In a liberal society, people are free to decide whether they want to read the book. This society looks like this.

- In a liberal society, people are free on p or q .

SetDeontic[$\{p, q, r, s\}, \{r\}, \{s\}$]

$\{\{p, \neg p, q, \neg q, r, \neg r, s, \neg s\}, \text{Ought}(\{r, \neg s\}), \text{NotAllowed}(\{\neg r, s\}),$
 $\text{Allowed}(\{p, q, r, \neg p, \neg q, \neg s\}), \text{Freedom}(\{p, q, \neg p, \neg q\})\}$

Ought[**Universe**]

$$\begin{pmatrix} p & q & r & \neg s \\ p & \neg q & r & \neg s \\ \neg p & q & r & \neg s \\ \neg p & \neg q & r & \neg s \end{pmatrix}$$

There are two equivalent ways to describe the situation:

- One view is that issues on p and q are not seen as belonging in the budget set. If society decides on something, then it are the issues of r and s , and not p and q . The latter only become interesting for a group vote, if the group decides that it should have a moral opinion on them.
- An alternative view is to hold that p and q are in the commodity space, and to argue that apparently choices are made in reality. It is felt as a strong desire to include such choices in the SWF-GM. This can only be done consistently by imposing restrictions on that SWF-GM. This will be that society adopts the same view as the persons in their individual domains, making them 'local masters of the (their) universe'.

(2) In a less liberal society, a norm can be imposed that nobody should read the book ("It is depraved") or that all have to read the book ("It is good material for a compulsory course in English literature"). Perhaps the example of reading a book is a bad choice for an example. However, a liberal society requires norms and values, such as for example the creation of a system of justice and property rights, and Sen's example will have to do. Let us consider the choice between "all: $p \& q \& \dots$ " or "neither p nor q nor ..." as such a possible norm.

- If all have to read it

SetDeontic[$\{p, q, r, s\}, \{r, p, q\}, \{s\}$]

$\{\{p, \neg p, q, \neg q, r, \neg r, s, \neg s\}, \text{Ought}(\{p, q, r, \neg s\}),$
 $\text{NotAllowed}(\{\neg p, \neg q, \neg r, s\}), \text{Allowed}(\{p, q, r, \neg s\}), \text{Freedom}(\{\})\}$

Ought[**Universe**]

$(p \ q \ r \ \neg s)$

- If nobody may read it.

SetDeontic[{p, q, r, s}, {r}, {s, p, q}]

{ {p, ¬p, q, ¬q, r, ¬r, s, ¬s}, Ought({r, ¬p, ¬q, ¬s}),
NotAllowed({¬r, p, q, s}), Allowed({r, ¬p, ¬q, ¬s}), Freedom({}) }

Ought[Universe]

(¬p ¬q r ¬s)

(3) Sen now considers the situation that Mr. *A* and *B* have opinions that cause them to meddle with each other indeed. Note that such meddling may be the beginning of the whole society starting to impose a norm. As we will see, the problem with Sen's example is that it gets stuck in the middle, between having meddlesome opinions and honest imposition. This amounts to inconsistent assumptions, since personal freedom is only possible if meddling is not effective. Or alternatively, in imposing norms, meddling becomes the major issue, but then the personal freedom disappears (particularly for who loses the vote).

Note that preferences are *independent* iff $(i: p \vee \neg p \mid q) = (i: p \vee \neg p \mid \neg q)$, where *p* is private to *i* and where *q* is private to *j*. It will be useful to distinguish two kinds of meddling:

- There is direct meddling if the preferences of agent *i* concern some *q* of some *j*. If the SWF-GM reflects such direct meddling, then there is a clearcut case of imposing norms - and thus there is no private liberty.
- There is an indirect form of meddling when the preferences of agent *i* on its private *p* are dependent on some *q* of some *j*. There is no problem with dependence if we do not require the issues to enter the budget set, and simply exclude them if we do not wish to impose norms. People are free to follow fads and fashions, or to go against them. However, if we introduce those items in the budget set, and if we do not define the SWF-GM with care, then this dependence can cause an indirect form of meddling - causing imposition anyway.

If there is meddling and dependence, and if we still want the issues to be reflected in the SWF-GM, *then we must clearly state which agent gets priority on which issue*. As said there thus are the situations if we do not impose norms: (1) We exclude the issues from the budget, so that agents decide for themselves. Since the agents decide for themselves, there is no meddling (otherwise than

shifts towards Pareto improvements). (2) We do not exclude the topics from the budget, so that the SWF-GM has to express what is being chosen. The individual choice now must be mimicked as a 'social' choice, and this must be modeled so that the individuals have *priority* on their choices - which means that there cannot be meddling.

Hence, if we include the topics in the budget *but still allow meddling*, then there is a confusion between the two proper approaches. We shall see that this is what Sen does.

Let us first consider Sen's example before we continue with the formal proof.

9.6.3 Restatement of the example

9.6.3.1 The argument

Sen's (1970:80) example is:

"Let the social choice be between three alternatives involving Mr. *A* reading a copy of *Lady Chatterly's Lover*, Mr. *B* reading it, or no one reading it. We name these alternatives *a*, *b*, and *c*, respectively. Mr. *A*, the prude, prefers most that no one reads it, next that he reads it, and last that "impressionable" Mr. *B* be exposed to it, i.e., he prefers *c* to *a*, and *a* to *b*. Mr. *B*, the lascivious, prefers that either of them should read it rather than neither, but further prefers that Mr. *A* should read it rather than he himself, for he wants Mr. *A* to be exposed to Lawrence's prose. Hence he prefers *a* to *b*, and *b* to *c*."

We can usefully construct the following table.

	<i>a</i>	<i>b</i>	<i>c</i>
Mr. <i>A</i>	2	1	3
Mr. <i>B</i>	3	2	1
Society for <i>A</i>	1	–	2
Society for <i>B</i>	–	2	1
Society	1	3	2
Paretian	2	1	–

If there would be a multiple read, then these preferences now create a "prisoners' dilemma", where, if one cell would be chosen, one of the two always has a motive to defect (and in this case in either direction). With $\{x, y\}$ giving the profit of *A* respectively *B*:

	q	$\neg q$
p	{0, 2}	{1, 1}
$\neg p$	{1, 1}	{2, 0}

Let us adopt Sen's example of a single read book now, so that we have $\neg (p \& q)$. Both A and B agree on a preference for " $q \Rightarrow p$ " which can only be logically realised by $\neg q$. Note that this still allows $p \vee \neg p$.

```
(q => p) && ¬ (p && q) // LogicalExpand
```

```
¬ q
```

Sen's argument then is:

1. "A liberal argument can be made for the case that given the choice between Mr. A reading it and no one reading it, his own preference should be reflected by social preference. So that society should prefer that no one reads it, rather than having Mr. A read what he plainly regards as a dreadful book. Hence c is socially preferred to a ."

We can translate this as: Comparing $a = p \& \neg q$ and $c = \neg (p \vee q)$, we consider $\{p \vee \neg p \mid \neg q\}$, and find $(A: \neg p > p \mid \neg q)$, or that $(A: c > a \mid \neg q)$, conditional to the assumption that B does not read the book.

2. "Similarly, a liberal argument exists in favor of reflecting Mr. B 's preference in the social choice between Mr. B 's reading it and no one reading it. Thus b is preferred to c . Hence society would prefer Mr. B reading it to no one reading it, and the latter to Mr. A reading it."

We can translate this so: Comparing $b = \neg p \& q$ and $c = \neg (p \vee q)$, we consider $\{q \vee \neg q \mid \neg p\}$, and find $(B: q > \neg q \mid \neg p)$, or that $(B: b > c \mid \neg p)$, conditional to that A does not read the book.

3. "However, Mr. B reading it is Pareto-Worse than Mr. A reading it, even in terms of the weak Pareto criterion, and if social preference honors that ranking, then a is preferred to b ."

Ergo. "Hence every alternative can be seen to be worse than some other. And there is thus no best alternative in this set and there is no optimal choice."

Sen's reasoning is reflected in the table above.

This reasoning is absurd on *three* angles: (1) general norms, (2) specific norms, (3) purely private issues. Sen's argument properly concentrates on (3), but it is useful to consider the other two first.

9.6.3.2 General norms

It is obvious that $(A: a > d > b)$ since A does not want that B reads the book, or if B would read it then A would not want so himself. Similarly $(B: d > a)$ since B wants that A reads the book, so that all should read it. We can now construct the general table. We find that a norm is imposed in d and c , while a and b reflect freedom. It now becomes obvious that we should not confuse the imposition of a *norm* - that *all* should read it or *nobody* - with a circumstantial *happenstance* that might arise in a case of freedom.

	$d = p \ \& \ q$	$a = p \ \& \ \neg q$	$b = \neg p \ \& \ q$	$c = \neg (p \vee q)$
Mr. A	2	3	1	4
Mr. B	4	3	2	1
Freedom, for A	—	1	—	2
Freedom, for B	—	—	2	1
Society	?	?	?	?
Paretian		2	1	

We find:

1. If a norm is imposed - cases d or c - then the private considerations are no input for the SWF-GM. People will not necessarily agree to turn a happenstance into a norm. If B happens to choose not to read the book, then this does not mean that he agrees to forbid reading it.
2. If no norm is imposed - cases a or b - then:
 - 2a. If B does not read the book, then A is better off not reading the book either, so that nobody reads it. Society can have sympathy for A on this, but since no norm is imposed, there it ends. Similarly for B 's perspective.
 - 2b. There indeed is a Paretian improvement from b to a , so that if b is the case, then A would willingly offer B that that he, A , read the book if B promises not ever to read it himself or herself. (This has problems of controlling it, but that is another issue.)

9.6.3.3 Specific norms

Society often imposes "specific rules". Thinking up general rules is a humanly impossible task, and social systems generally allow that specific cases are settled by a specific ruling. We can assume that there is some committee, that follows Sen's step 1 and 2 again.

Ad Sen 1. Compare $a = p \ \& \ \neg q$ and $c = \neg (p \vee q)$ again. Then indeed $(A: a < c)$.

However, once we are considering norms, then there is no obvious reason to follow A. Sen suggests so, but does not mention all factors. Indeed, $(B: c < a)$, and thus there are at least two opposing views.

Ad Sen 2. A and B again have opposing views here as well.

In both cases, a jury or committee, considering the issue of making a specific norm finds that it is no settled issue. Some of the considerations are valid, but there are also other considerations.

9.6.3.4 Purely private issues

Sen's argument properly concerns 'private issues'. If these are not excluded from the budget set, then these could be reflected in the SWF-GM. Such reflection however is only useful for purely private issues that are not subject to some meddling by others. Once such meddling would be imposed on the SWF-GM, then we have the discussion of imposing norms on others - and that is another issue than purely reflecting private issues.

Sen thus found a case that clearly shows that we should not confuse these issues. Let us follow him on the assumption that there is only one single read book.

Ad Sen 1. For A there is direct meddling and dependence. (I) We find $(A: \neg q > q \mid \neg p)$ which means that A has a preference on what B does. Thus there is a meddling that is inconsistent with the assumption of having a 'private issue'. (II) A's preference is not independent from B. If we are in a then a move to b would be rejected. (We found $(A: \neg p > p \mid \neg q)$. We also find $(A: p > \neg p \mid \mid q)$, with ' $\mid \mid$ ' the counterfactual. The fact that there is only a one read book - so that d is not allowed - makes it perhaps difficult to see this dependence.)

Ad Sen 2. For B there is direct meddling and dependence as well. (I) There is meddling, since $(B: p > \neg p \mid \mid q)$ and $(B: p > \neg p \mid \neg q)$. (II) We found $(B: q > \neg q \mid \neg p)$, and can find $(B: \neg q > q \mid a$ (which implies p)), which is a preference reversal because of the fact that the book can only be read once.

Hence, if we include the topics in the budget but still allow meddling, then there is a confusion between the two proper approaches (of either *imposition* or *freedom*).

9.6.3.5 Conclusion

Sen's 'paradox' thus is caused by combining *meddling* with *personal freedom*. This amounts to inconsistent assumptions, since personal freedom is only possible if there is no meddling.

We have to reject this approach. The issue can be reformulated as the one of imposing norms (general or specific) or not, and this gives a consistent formulation. In imposing norms, meddling becomes the major issue, but then the personal freedom disappears (particularly for who loses the vote).

Consider for example the situation that a committee has to decide on Sen's example. The issue then directly becomes one of imposing norms, and can only become an issue of personal freedom again if the norm is adopted that it is or remains a matter of freedom. If the matter just concerns *A* and *B*, then a specific decision can be required. If the committee wants to respect *A*'s and *B*'s meddling views, then it can tell *A* and *B* that logically only the conclusion $\neg q$ is possible. Then *A* should be aware of the danger that the committee still might decide *p*, and *B* should be aware that it is also possible that $\neg p$ is chosen. Hence, if *A* & *B* continue to press for $\neg q$, then the committee still has to decide for $p \vee \neg p$, and it is not a clearcut case that either prevails. The committee might feel the book objectionable for *B*, or it might feel that it is excellent compulsory material for a course in English literature for *A*. In some cases the committee might also decide that, since norms are being imposed, that the Pareto improving $\neg q$ is not allowed - as sometimes two robbers are not allowed to practice their agreement on robbing.

Hence, since both the Pareto principle and the other considerations are all part of the general realm of morals, there is hardly any reason to speak about the 'impossibility of a Paretian liberal'. Classical liberals have been very much aware that there are more moral principles, and that choices often are difficult.

9.6.4 Restatement of theorem and proof

9.6.4.1 Restatement

Sen's theorem does not exactly match his example, and the mathematics do not fit the verbal explanations anyway. It is a good exercise, now that we have discussed the example, that you read this restatement of theorem and proof, and then try to find the critique yourself.

Sen (1970:87): "Liberal values seem to require that there are choices that are personal and the relevant person should be free to do what he likes. It would be socially better, in these cases, to permit him to do what he wants, everything else remaining the same."

Sen then defines the condition of liberalism in a very weak form:

Condition L^* (*minimal liberalism*): There are at least two persons k and j and two pairs of distinct alternatives $\{x, y\}$ and $\{z, w\}$ such that k and j are decisive over $\{x, y\}$ and $\{z, w\}$, respectively, each pair taken in either order.

An individual i is decisive on a pair $\{x, y\}$ if $(i: x > y) \Rightarrow (x > y) \ \& \ (i: x < y) \Rightarrow (x < y)$

Presuming acyclicity, he then proceeds to prove (Sen, "Theorem 6.1"): there is no SDF satisfying conditions AU , AWP and L^* . Sen's proof is as follows:

1. If the pairs are the same, then the condition obviously cannot hold.
2. If the pairs have one common element, say $x = z$, then take $(k: x > y)$, $(j: w > x)$ and $(all \ i: y > w)$. From L^* , $(w > x)$, $(x > y)$ hence $(w > y)$ while from AWP $(y > w)$. Sen: "This violates acyclicity and there is no best alternative."
3. Let all items be different. Then taken $(k: x > y)$, $(j: z > w)$ and $(all \ i: w > x \ \& \ y > z)$. From L^* , $(z > w)$, $(x > y)$ and from AWP $(w > x \ \& \ y > z)$. Sen: "But this too violates acyclicity."

For 4 different items, see this table. Here "-" means that an opinion is not considered, and we print in bold the decisive private decisions.

Sen	w	x	y	z	w
1		$>$	$>$	$>$	$-$
2		$>$	$-$	$>$	$>$
Others		$>$	$-$	$>$	$-$
Hence All		$>$	$>$	$>$	$>$

9.6.4.2 Discussion

The problem with Sen's approach is that it does not take into account that private issues should be decided upon and chosen *simultaneously* with the group choice.

What we should expect to see: 1's private freedom allows it to choose x , 2's private freedom allows it to choose z , and *simultaneously* society can choose some additional u , so that the state of the world is $\{x, z, u\}$.

Sen's approach suggests that that *only one item should be chosen*, but then the notion of personal freedom loses its meaning since everything becomes subordinated to the fact that the group as a whole should decide upon only one item. The whole purpose of the exercise was that the group should respect individual freedom. If only one item can be chosen from the budget set, then clearly only in exceptional cases both the private choices will co-incide. Similarly, what was supposed to be a private choice for the two individuals suddenly becomes imposed on the whole group, and that is hardly 'private'.

Thus, on close inspection, we find that theorem does not fit the intended problem area. It is incomplete with regards to the intended application.

We can clarify this point also as follows. Since the choices on $\{x, y\}$ and $\{z, w\}$ are private, *all items* $\{x, y, z, w\}$ *are private items*, and other people have nothing to do with them. Those other people may have private opinions like $w > x$ or $y > z$, but those opinions are irrelevant. Thus the true table is as follows.

True	w	x	y	z	w
1		$-$	$>$	$-$	$-$
2		$-$	$-$	$-$	$>$
Others		$-$	$-$	$-$	$-$
Hence All		$-$	$>$	$-$	$>$

Strikingly, Sen's very example on *Lady Chatterly's Lover* (LCL) is not an example of his theorem but provides a *counterexample*.

Since we now consider the general theorem, we can assume that A and B might both read a copy of the book, also simultaneously. We can usefully substitute the p and q items in the Sen table, while keeping the ' $>$ ' signs to stay with the 'proof' though we should rather write " $-$ ". Note that using $\{p, \neg p, q, \neg q\}$ is equivalent to using $\{a, b, c, d\}$, but avoids the complex discussion on logical interdependence.

LCL	$\neg q$	$\neg p$	p	q	$\neg q$
A	$>$	$>$	$>$	$-$	
B	$>$	$-$	$>$	$>$	
Others	$>$	$-$	$>$	$-$	
Hence All	$>$	$>$	$>$	$>$	

With this example we see more clearly that Sen's proof is deficient in that it suggested that it would be sufficient to consider at most 4 items. For, taking (these) 4 items provides a counterexample to his implementation. Clearly p and $\neg p$ are logically dependent, and q and $\neg q$ as well, and, hence, when society would hold that A can decide on p or $\neg p$ as a private issue and that B can decide on q or $\neg q$ as a private issue, then $\{p, \neg p, q, \neg q\}$ are all private issues, and then everyone else forfeits any right to make decisions like $(p > q)$ and $(\neg q > \neg p)$. Opinions on private matters of others of course can be allowed, but these are not allowed for any aggregate *decision*, and should be left out of consideration.

Note that use of this counterexample uses p and q rather than $\{a, b, c, d\}$ (in their appropriate order). This is done only for illustration purposes, since it emphasises the logical connections between private freedoms, and avoids the more complicated discussion about independence and meddling. The general line of argument has been given by clarifying that $\{x, y, z, w\}$ are all private, and not open to decisions of others.

For which cases does Sen's theorem apply ? Using the counterexample while returning to the committee on A and B , and following Sen by taking the committee = Others: Then we see that the committee apparently has made some moral choices about the private issues of A and B , so that apparently it sees it fit to impose its norms. The committee finds that the book is better read by A than B , while it also finds that the book is better not read by B than not

read by A . (This is just the opposite, and consistent.) Clearly, the committee makes an interpersonal utility comparison about matters which a liberal person would rather declare private issues. There is a lot of meddling around and imposition of norms and values. At the same time, the committee has not really done its homework, for it has not clearly said what should happen. It is blank on $p \vee \neg p$, or it is blank on who shall have priority on his personal domain. In other words its preference profile is incomplete. Sen's axioms depict the situation that it tries to leave the decision to A and B : but that clearly does not help, as there arises a cycle of aggregate indecision or indifference, with the need for additional rules on breaking ties. Thus, Sen's axioms describe a committee that does not do its homework, that hesitates between imposing values and not imposing values, and that tries to shift the hot potato around.

There is an argument that Sen's axioms are very weak, and thus are a subset of anything that demands more. If the subset already causes problems, then anything larger will cause problems as well. This argument however does not work if the weak set is the wrong model. The logical calculus may remain intact and hold, but it does not cause a problem. Indeed, the proper axiomatisation describes how a committee imposes a norm or does not impose a norm. The axiomatisation should be so rich, that we also can describe what happens when the committee flunks its job. In that sense, Sen's axioms can get a place. But that does not imply that all committees will flunk their job.

9.6.5 Evaluation

We find that Sen's theorem does not fit the intended application, while the words do not fit the math. There is an accurate logical calculus that some premisses result in some consequence, but that, alas, is not a sufficient condition for accuracy or relevance.

There are some other quotes that show that Sen had difficulty grasping the issue. He writes (1970:79):

"A still weaker requirement than condition L is given by condition L^* , which demands that at least *two* individuals should have their personal preferences reflected in social preference over one pair of alternatives each. This condition is extremely mild and may be called the condition of "minimal liberalism", since cutting down any further the number of individuals with such freedom (i.e., cutting it down to one individual) would permit even a complete dictatorship, which is not very liberal."

Well, a dictator would be someone who imposes *all* his preferences, and not just a preference on two items of private consideration. It is incomprehensible why Sen would think that if *one* person would want to drink coffee without sugar, as a private decision, that this would mean that he or she would want that everyone does this, that in fact everyone should obey him or her in everything, and that social decision should reflect this.

My impression is that Sen somewhere in his mathematical exercises lost track of what he was doing. Typically, he does not discuss the budget set.

On p82/83:

"Public policy is often aimed at imposing on individuals the will of others even on matters that may directly concern only those individuals. However, condition L^* is really extremely weak and a rejection of it is to deny such liberal considerations *altogether*. My guess is that condition L^* , in that very weak form, and even the somewhat stronger condition L , will find many champions. [note] To deny condition L^* is not merely to violate liberalism, as usually understood, but to deny even the most limited expressions of individual freedom. And also to deny privacy, since the choice between x and y may be that between being forced to confess on one's personal affairs (x) and not being so forced (y). Thus support for L or L^* may come even from people who are not "liberals" in the usual sense."

Answer: a liberal would reject L^* as much too weak. All persons should have items for private liberty. These can be neglected in the budget set, since they would not be relevant for group decision making. Or, if they are included, then preferences of others on them are not relevant.

On p84:

"If the Pareto principle is rejected, the consequences of that for collective choice in general and for welfare economics in particular must be immense. Most of the usual political choice mechanisms are Pareto-inclusive. (...) What seems to follow from the problem under discussion is that Pareto-optimality may not even be a desirable objective in the presence of externalities in the shape of "nosiness". [note] The consequences of all this are far-reaching."

Well, Paretian liberals have in the past effortlessly accepted that society imposes certain norms, notably those that foster a free society with a good system of property rights and so on. Clearly, crooks and criminals have been put into jails even though they might have tried to veto the idea. The Pareto principle has always been applied with some limitation. Sen must have been off-track when he wrote the above.

In general, Sen gave a wrong axiomatisation of how classical liberals balance the Pareto principle with their other values. He gave a caricature of that position, suggesting that they adhere to and should reject principles - while they never adhered to such principles. The banality that chaos results if a committee does not do its work properly, has been abused to draw wildly wrong conclusions.

It is a serious problem that the literature on the issue is not straightforward in the same conclusion. Let us consider Mueller (1989) as an important reference and example. He writes: "In the presence of such a long-run liberalism, books like *Lady Chatterly's Lover* and individuals like *A* and *B* may from time to time come along and lead to a short-run conflict between this liberalism condition and the Pareto principle." (p405). Mueller thus takes the conflict seriously, and does not see that the problem is caused by a bad formulation. Similarly: "The solution to Sen's paradox as with Arrow's paradox rests ultimately on the use of cardinal, interpersonally comparable utility information (...)." (p406). Which is not true for either of them.

Thus, you are warned now about the literature on Sen's Theorem of the Impossibility of the Paretian Liberal.

10. Evaluation of Arrow's Theorem

10.1 Introduction

10.1.1 Introduction

Overviewing the Arrow complex, one feels inclined to cut the whole knot, rather than disentangling the separate strands. However, for historical and clarification purposes, the following (repetitious) remarks might be needed.

Cool (1990g) already contains the analysis of this book. You will recognize that the first chapters above concern a general introduction into Voting Theory while they also provide an introduction into the programs done in *Mathematica*. The parts in (1990g) on reasonableness and moral desirability were polished up in 1992, which gave rise to Cool (1992f) and subsequently chapter 36 in DRGTPE - and this has become chapter 9.2 above. The other pages of (1990g) are now copied into this part of the book. My work here has remained limited to reordering it, cleaning up some dust, and adding some of the niceties of the *Mathematica* routines.

DRGTPE has a chapter on the "Definition & Reality methodology". DRGTPE in fact drew on the insights of (1990g). The text in chapter 10.2 below, taken from Cool (1990g), namely uses that method. If you would have further questions on the method, see DRGTPE.

10.1.2 Looking back at (1990g)

Reading (1990g) again, and seeing how its argument has stood up the test of time and the application in *Mathematica*, I feel annoyed again that that paper was so improperly blocked from discussion and eventual publication by the directorate of the Dutch Central Planning Bureau. This is a word of continued protest against this abuse of power.

You should compare the Abstract of this book with the Abstract of (1990g), which reads:

"A distinction is made between voting and deciding, so that an individual vote is a decision too, but an aggregate vote result does not necessarily render an aggregate decision. From this distinction it follows that Arrow's (1951) impossibility theorem is rather irrelevant. Moreover, Arrow's verbal explanations of the theorem appear not to match its mathematics, and deontic logic shows those verbal statements to be incorrect. It appears that social choice

is rational by definition, and from this follows the need for the design of proper procedures. The paper gives also a short history and indicates some consequences."

10.1.3 Evaluation

This is not the place to fully evaluate the whole Arrow contribution in its historical context. Yet a suggestion would be correct. I increasingly feel that "Arrow" is only interesting for the history and methodology of science rather than for Social Choice Theory itself. It is said that Arrow applied the axiomatic method to social choice as the first person in history to do so. This is a dubious argument. The axiomatic method existed before, and others have laid down rules for social choice, so that we enter into a discussion whether there is a difference between rules and axioms - and then we find out that actually there is not. Neither did Arrow discover the voting paradoxes. We do not need *APDM* to generate voting paradoxes (though it causes many). Sen (1970) claims that Arrow's axioms are very economical in that only their combination creates the inconsistency - but that is an unwarranted statement. Finally, having the axioms does not solve anything, it only needlessly complicates the issues. Arrow's axioms are a wrong axiomatisation of a rational social decision process. Thus I would opt that Arrow's contribution was a gradual improvement in the mathematical accuracy of voting analysis.

10.2 Definition & Reality methodology

10.2.1 Mankind as the SWF-GM

Starting from the notion of social choice, which happens all around us, ... Then secondly, the general position defended here is, that collective choice *has to be rational*, in just the same manner as economists assume that individuals are rational (as a collection of brain parts); and that we need such an assumption in order to scientifically describe the world we live in. Social choice is rational, *by definition*, and it is possible, *exactly on this ground*. This rationality does not only hold for a SWF but also for a SWF-GM. Not quite metaphorically: mankind itself may be regarded as the machine SWF-GM calculating its SWF.

Our position can be given more nuance, by distinguishing scientists and society. The scientists observe social choice in actuality, and they look for a rational explanation of what is happening - like Darwin, they try to recover the rules which Nature has chosen. In a society, people have the more moral problem of which rules to choose. But there are some rules governing this process, like scientists observe, like the whole objective of social choice theory is concerned with, and like society has some inclination to listen to scientists.

Additionally, there has been some experience now with democracy and consumer sovereignty. Indeed, the 'theory of economics' started with Adam Smith's observations on individual freedom. Having this experience, then for many cases for which *AWP & AU & AD* was valid, we still would maintain the validity of collective rationality. It turns out that the analysis which Arrow proposed as being reasonable, appears not to be reasonable at all. Precisely since it allows this contradiction with aggregate rationality, it must be rejected, either the proposed axioms or the deduction.

Indeed, many have followed this logic (cf. Luce & Raiffa). Many have checked Arrow's mathematical deduction. After Blau's correction, the mathematics however appears to be sound, and, after checking this ourselves tediously, we accept that it is so. But then, applying the same logic, (a) we reject the proposed axioms, (b) we reject the logical validity of the *verbal* Arrowian statements, and, after having turned those, above, into strictly defined propositions, we reject the validity of the claim on reasonableness and moral desirability.

10.2.2 The world is given

We sometimes expand the commodity domain X and regard the consequences for an individual. In fact, the world is given for present and future, so we cannot simply change X . In the same vein, the budget set must be chosen with care, to prevent semantic problems. Any 'change' then must be attributed to myopia and the like; likely it has been demonstrated sufficiently that people are feeble beings; and then the 'change' is reasonable, since this occurs in another problem setting. This can be extended to the SWF-GM. So examples of the 'devastating effect of the sudden introduction of new alternatives', are not reasonable for the problem at hand. When we change the problem, then we get a dynamic SWF-GM field of inquest.

The former *static/dynamic* fallacy of composition has been a major source of confusion in the social choice analysis. E.g. Arrow (1950) takes a lot of recommendable effort of finding paradoxical voting situations, but this understandable passion appears to muffle the perception of the underlying rationality of events.

10.2.3 Rational reconstruction of the paradoxes

It has been the assumption that votes or conjectures would be rational in the sense of giving a neat preference ordering - but the voting paradoxes show that they are only rational in terms of other definitions.

Since it cannot be denied that voting is paradoxical in some specific instances, it must be shown that the issue can be solved by a proper choice of definitions, so that the paradox is no contradiction. Hence the paradoxes of voting should be used, not to kill voting, but rather to refine the process, and to show that some first intuitive ideas need to be refined, in order to maintain the defined rationality of the process of collective choice.

The notion of rationality has been a source of confusion in economic theory. Apparently there is a family of notions, with the general property of implying the absence of contradiction. Explicit definitions depend upon circumstance, here e.g. the assumption of honesty or non-cheating for the Arrow Theorem. Then in particular: *deciding* is rational when it reflects a proper preference ordering, *voting* is rational when it is not assumed to render a proper preference ordering. So, also, we can say that a paradox (a seeming

contradiction which arises from some bad definitions) is rational (in terms of proper definitions). Though, it does sometimes not help people's understanding, to say that paradoxes are rational.

10.2.4 Axiomatic method

It holds for the Condorcet voting paradox in the same manner as for the Liar sentence paradox or the Russell set paradox: after a proper axiomatisation there further is no issue for the practical and analytical mind. The paradoxes are shown to be caused by a premature and foggy set of ideas and definitions, and they evaporate under the sunlight of reason. And here, contrary to Tarski and Zermelo, Arrow has failed to provide us with the proper axioms, failed to provide us with that sunlight of reason.

10.2.5 Axiomatic method & empirical claim

It may be suggested, incidently, that the Russell set and the Liar paradoxes are primarily logical problems, while Arrow's Theorem has an empirical claim. This distinction might be granted, but of course the paradoxical element in Arrow's Theorem remains logical by definition. In the same field, but conversely, when I say that a SWF-GM *exists*, I fundamentally take the empiricist position: social choice happens around us all the time.

As a source of confusion ever since Lobachevsky, mathematicians tend to identify 'existence' with consistency. The 'existence of competitive equilibrium' for example merely means that some axioms result into some properties (but remain consistent); and this does not mean that our actual world satisfies these axioms. Here I prefer the word 'consistency'; and valid is only that existence implies consistency, or conversely that inconsistency implies non-existence.

Thus, one may safely conclude that there must be a set of axioms which copies our reality. Even stronger, one can experience various group decision settings at different times and places. A conjecture is then that there also has been some occurrence of consumer sovereignty and collective rationality in terms of reasonable definitions. We only face the problem of locating those cases and determining their conditions.

Note: For the axiom of universal domain (*AU*) it might be useful to enlarge our domain of individuals with chickens and potplants etcetera, so that we may truly imagine a wide range of possible utility functions. Eating chickens,

showing individuals to be dividual, only violates their sovereignty if they protest that it does not make them better off.

If power exists, then, logically, it has its way. From empirical observation we thus can deduce that the whole idea of ordinal incomparability is fundamentally inadequate. It may derive from some liberal philosophy, but without empirical foundation. For example, we eat chicken. Perhaps these evolve, or laws are amended, so that they pass voting registration, and angrily make us stop munching their brothers' and sisters' meat and bodies. Perhaps on one planet there are aliens who regard us as their potential natural resource.

10.2.6 Does APDM belong to the definition of rationality ?

Arrow, in the Palgrave, might be interpreted as referring to *APDM* as a 'consistency condition' which hence relates to 'rationality'.

Thus, our definition of 'rationality' could be challenged.

But our definition of rationality is quite basic, and it shows that *APDM* should be rejected. So it is *petitio principii* (*begging the question*) to want to include *APDM* in a redefinition of rationality and then argue that rational social choice would be impossible.

10.2.7 Other elements for empirical work

Important for our empiricist position (as Luce & Raiffa note) it will be useful to add an axiom SQ.

Empirical work should also include altruism and preference drift and reference drift.

10.3 Arrow 1950

10.3.1 The Forsythe-Borda paradox

Arrow (1950) presented the The Forsythe-Borda paradox. Let (1: $x > y > z > w$), (2: $x > y > z > w$) and (3: $z > w > x > y$) and let there be a Borda ranking.

Arrow (1950): "Under the given electoral system, x is chosen. Then, certainly, if y is deleted from the ranks of the candidates, the system applied to the remaining candidates should yield the same result, especially since, in this case, y is inferior to x according to the tastes of every individual; but, if y is in fact deleted, the indicated electoral system would yield a tie between x and z ."

To this example and quote, Cool (1990g:33) reacts: "Well, granted that this is paradoxical, it nevertheless does not stand up to a moment's scrutiny. Firstly, deleting candidates is dynamic and not at issue; secondly, the electoral system would allow a new ranking of tied candidates, with x remaining the clear winner."

- Define the Forsythe-Borda case.

```
EqualVotes[3]; Clear[x, y, z, w]; Items = {x, y, z, w};

{a = Pref[w, z, y, x], a, Pref[y, x, w, z]} // SetPreferences;

Preferences


$$\begin{pmatrix} 4 & 3 & 2 & 1 \\ 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 \end{pmatrix}$$

```

- Borda and BordaFP give the same result.

```
Borda[]

x

BordaFP[]

x
```

BordaAnalysis[]

$$\left\{ \text{Select} \rightarrow x, \text{BordaFPQ} \rightarrow \{\text{True}\}, \text{WeightTotal} \rightarrow \left\{ \frac{10}{3}, \frac{7}{3}, \frac{8}{3}, \frac{5}{3} \right\}, \text{Position} \rightarrow (1), \text{Ordering} \rightarrow \begin{pmatrix} \frac{5}{3} & w \\ \frac{7}{3} & y \\ \frac{8}{3} & z \\ \frac{10}{3} & x \end{pmatrix} \right\}$$

■ Deleting y .

SelectPreferences[{x, z, w}];

CheckVote::adj : NumberOfItems adjusted to 3

■ Borda and BordaFP now give a different result: item z is not a fixed point.

Borda[]

$\{x, z\}$

BordaFP[]

BordaFP::chg : Borda gave $\{x, z\}$, Fixed Point is $\{x\}$

x

BordaAnalysis[]

$$\left\{ \text{Select} \rightarrow \{x, z\}, \text{BordaFPQ} \rightarrow \{\text{True}, \text{False}\}, \text{WeightTotal} \rightarrow \left\{ \frac{7}{3}, \frac{7}{3}, \frac{4}{3} \right\}, \text{Position} \rightarrow \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{4}{3} & w \\ \frac{7}{3} & x \\ \frac{7}{3} & z \end{pmatrix} \right\}$$

10.3.2 Arrow 1950 paradox

Arrow (1950) presented the paradox where (1: $x > y > z$) and (2: $z > x > y$), with simple majority vote and without rational reflection, give rise to $(x > y = z = x)$, which would be irrational.

10.3.3 Cool 1990 (a)

Regard Arrow's 1950 paradox. Apart from the SQ axiom, society might solve the paradox with the unanimously accepted rule $((1: x > y > z) \ \& \ (2: z > x > y)) \Rightarrow (x = z > y)$ (violating *APDM*). Then the resulting SWF $(x = z > y)$ is wholly rational. Let y be the status quo: then there is a status quo dominated by a deadlock on alternatives: and again the SQ comes to the rescue, so that y is chosen. (It appears that SQ is needed to reduce the choice to a single point. It is not clear why Arrow in 1951 did not use the SQ axiom to solve voting paradoxes. But perhaps a solution was not his objective.)

The notion of a status quo dominated by a deadlock can enlighten one's understanding of an Edgeworth box. Regard the core, as the collection of Pareto Optimal points (not necessarily satisfying *APDM*); chose a status quo on that core; then as a result of death (with uncertain inheritance), one player is replaced, and the core shifts, so that the status quo is no longer on the core. One gets a lense-shaped region of better options, and within that, a section of the core dominating that lense. However, our axioms are not rich enough to select any particular point on that section. The status quo dominates in reality, and the core only as a counterfactual.

10.3.4 Cool 1990 (b) using the Pareto SWF-GM

Suppose that the Pareto optimality principle is the SWF-GM. The rule is: "(there is unanimity voting) & SQ & (in absence of unanimity and SQ there is indifference)". One can certainly imagine a (Lionel Robbins) society where this SWF-GM *ought* to hold.

But then regard the Arrow 1950 paradox. Obviously, there is no possibility that an item gets unanimous approval, and hence the status quo prevails. This is quite consistent. It appears that the paradox only arises since pairwise majority voting requires the assumption of *APDM*. When we drop that, apply SQ and Pareto, then rationality is retained.

In 1950, Arrow had not fully presented his Theorem yet. But his 1950 paradox shows his struggle with the Pareto principle.

Of course, if the Pareto SWF-GM were extended to the whole commodity and preference domains, then there would be no simple proof of consistency, since

mathematics has few tools for this, primarily only mathematical induction, which might be tried.

A good understanding is required. Arrow (Palgrave p124): "If every individual prefers one policy to another, it is reasonable to postulate, as is always done by economists, that the first policy is to be preferred. The problem arises in making social choices (...) when some individual criteria prefer one policy and some another." Well, if the latter is the fundamental problem in social choice theory, then it would seem to be simple, that the notion of Pareto optimality precisely has been introduced to cover the situation, that there will be action *only* when preferences agree, and otherwise not (the status quo). It would not be correct to ask for a proof, where everything is *defined* as this. (And it would be even stranger, to do as Arrow, to add *APDM*, and then derive the "impossibility of deciding" altogether.)

Note that Arrow's axioms use only the weak Pareto principle, which should be a source for concern. The Pareto SWF-GM restores strong Pareto optimality, so that its obviousness is enhanced.

10.3.5 Cool 1990: note of caution on Pareto

A further complexity is that the normal idea of Pareto-optimality may not suffice. Suppose that you and me are offered 10 dollars. But why would I be happy to let you have 10 dollars, when you might as well give 9 to me (you would still gain 1 !) ?

Indeed, further on this line, one should try once in a community or in a family, to continuously better one person, and keep the others in a constant position: and see how people respond to that in the end. Dating back at least to Adam Smith, but also noted by Tinbergen (1956:186): "people's happiness is not only determined by the absolute level of their own physical situation, but also by the relative level in comparison to other individuals, a fact gradually recognized by economists but not yet given its full place in welfare economics."

Indeed, it is for this kind of function that one can see the usefulness of proportional or balanced growth.

10.3.6 Cool 2000

Well, in 1990 I did not have this reaction:

- Define the case.

```
DefaultItems[]; EqualVotes[];
Clear[x, y, z]; Items = {x, y, z};

{Pref[z, y, x], Pref[y, x, z]} // SetPreferences;

Preferences


$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix}$$

```

- Pairwise majority gives a cycle, but x would be the Condorcet winner with the highest margin.

```
PairwiseMajority[]

VoteMarginToPref::cyc : Cycle {z, x, z}

VoteMarginToBinary::dif : Selection x differs from Condorcet winning {x, z}

{VoteMargin → VoteMargin $\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ },
1 → {StatusQuo → x, Sum → {2, 1, 2}, Max → 2, Condorcet winner → {x, z},
  Pref → Pref({x, y, z}), Find → {x, y, z}, LastCycleTest → True, Select → x},
N → {Sum → {1, -1, 0}, Pref → Pref(y, z, x), Select → x}, All → x}
```

- BordaFP shows that x and z form a fixed point set.

```
lis = BordaFP[]

BordaFP::set : Local set found: {x, z}

BordaFP::chg : Borda gave {x}, Fixed Point is {x, z}

{x, z}
```

- Of course, Arrow did not specify what the status quo was.

```
ParetoMajority[]

{StatusQuo → x, Pareto → {x}, Select → x}
```

10.4 Frerejohn and Grether paradox

10.4.1 Proposal vs alternatives

Cool (1990g) suggested the "proposal versus alternatives" approach, of comparing x versus $X \setminus \{x\}$. This has been implemented here in *Mathematica* as the Fixed Point for Borda. It is interesting to look back on the issue.

10.4.2 The Frerejohn and Grether paradox

The Frerejohn & Grether, henceforth F&G, paradox (Sen (1986:1103)): three preference orderings (1: $x > y > z > w$), (2: $y > z > w > x$) and (3: $z > w > x > y$), would, with simple majority vote without reflection, give $(x > y > z > w > x)$: then, if we were to declare Dahl-like indifference, then this (Sen p1103) would violate the unreflected 'weak Pareto principle' (Sen p1075) [(For all i : ($i: x > y$)) $\Rightarrow (x > y)$], where in this case, if w were chosen, everybody would benefit from a move to z . (So that indifference combines with preference, which is irrational.)

[Note 2001: We have discussed this paradox above.]

10.4.3 Cool 1990

The Condorcet, Arrow 1950 and F&G paradoxes show that the problem arises from the desire to fold the voting process into a *logical* mold. If voting would be processed by (Borda 1781) numerical rank ordering, then there is little issue or problem.

As an example, we find the following possible procedural amendments for majority deciding.

- For Condorcet, allow for indifference. This shows also from voting on a proposal versus all its alternatives; hence x versus $\{y, z\}$ (only one supporter), etc.
- In the Arrow paradox, a binary *proposal / alternatives* vote shows that nobody would propose y , so that it might be dropped from the discussion space.

- In the F&G paradox, the latter would hold for w , creating indifference for the remainder. *Caveat*: 'indifference, as far as this voting procedure shows for these choices' !!

These amendments are just indications of possibilities, and there is no claim for universality. For example, if there is a status quo dominated by a deadlock of alternatives, it may not be feasible to drop this status quo.

10.4.4 Cool 1990 on pairwise-ism

It may well be that the 'pairwise' aspect in the *APDM* axiom has subconscious connotations, which make that people want to accept it, even though its formulation leads to moral chaos.

- As said, pairwise-ism holds for voting.
- In some cases pairwise approaches work.
- Perhaps people like to see such pairwise statements, as they are trained by economic textbooks to want to see those. (Though see Hicks (1981), "Wealth and welfare", on the British Parliament.)
- Much pairwise-ism might be retained under the *proposal versus alternative* comparisons, i.e. x versus $X \setminus \{x\}$.

10.5 Proof particulars

10.5.1 Voting vs deciding

Arrow-type impossibility approaches are generally misguided. They suggest by choice of words that they distinguish voting from deciding, but in effect they don't do that. They mix up the paradoxes of voting with complexer kinds of 'proof'. In effect, they don't *solve* those paradoxes. And they add moral chaos. We will however resolve the paradoxes of voting, by a proper distinction between voting and deciding, so that an individual vote is a decision too, but an aggregate vote result does not necessarily render an aggregate decision.

Note that economics already has the notion of a *core* or *choice correspondence*, which concerns the set of possible decisions depending upon a profile of individual preferences (e.g. the core of an Edgeworth-Bowley box). This differs from the distinction between voting and deciding, and I think that the latter is better for general communication. Mixing up voting and deciding is a prime cause for voting paradoxes, and, in particular, Arrow's *APDM* mixes the concepts up too.

10.5.2 Theorem that voting is not deciding

Let us distinguish the vote $v(.)$ (conjecture, belief) (...) which concerns a possible decision, from the actual decision $d(.)$ (...). This distinction creates a protective hull against Arrow-type arguments, as anybody experienced in predicate logic would readily grant. One will note that the range of deciding $d(.)$ is R (rational preference), but that the range of voting $v(.)$ is "SPR" [see Cool (1990) for its definition] so that rationality and consistency are retained by dropping the requirement that voting renders a preference ordering; and thus the *APDM* axiom may still apply to voting. As a consequence, there is a function $h(.)$ such that $d = h.v$ which turns any (paradoxical) voting result into a neat decision.

We find that *APDM* needs a protective hull for a proper SWF-GM. Thus, exactly from the paradoxes of voting, it follows that a choice xRy can never be limited to a pairwise vote: it would be foolish to do so after the given counterexamples. A good chairman writes down the *decision* only after checking that all possibilities are accounted for.

Theorem: Voting v is not the same as deciding d .

Proof: Suppose that it is the same, then $v = d$. Since this would hold for any circumstance, regard majority voting, and then the Condorcet 1785 case. In this case, there are three contradictory majorities. Thus voting \neq deciding. Q.E.D.

Corollary: Since d in our definition represents rational deciding, its range is $\{R\}$, and this cannot hold for v .

Corollary: Since our definition is that v satisfies *APDM*, this cannot hold for d .

Note: This by-passes Arrow's rather complex deduction.

Note: Arrow (Palgrave 1988:124): "Voting procedures have one very important property which will (play) a key role in the conditions required of social choice mechanisms: only individual voter's preferences about the alternatives under consideration affect the choice, not preferences about unavailable alternatives." I think this substantiates (a) that our definition of voting is acceptable, (b) that Arrow confuses voting and deciding (social choice mechanisms).

Note: The 'group contraction & expansion lemmas' use *APDM*. But our analysis has shown that this axiom is only valid for voting. Hence the proposed and proper voting/deciding distinction kills elements of the 'proof': and in that sense, there appears to be a *hidden assumption*, that voting is equivalent to deciding.

10.5.3 Axioms for voting (as differing from deciding)

PM. Cool (1990) conjectures some axioms for voting, as differing from deciding. I have not developed this issue here, since it was more useful to develop the *Mathematica* programs.

10.5.4 Solution approach

Accepting axioms seems to be an area reserved to intuition rather than to logic. But that is not wholly true. Some advance in logic can be made.

- Firstly, we have applied deontic logic above.
- Secondly, we have shown that morals are part of the preferences which are aggregated.

- Thirdly, having solved the voting paradoxes, the natural question is: would such adapted procedures not belong to the SWF-GM ?

Taken together: the Arrow axiomatisation does not capture the true properties required for a SWF-GM (notably: feasibility !), and in a wider sense, his logic can be rejected.

10.5.5 Reaction to Sen on APDM

Sen (in the 'social choice' entry in the Palgrave) writes:

"(...) it is not clear why it would be thought perfectly okay that social preferences might change over a given pair when there is a change of individual preferences over some pair of alternatives quite unconnected with this particular one. The need for some interprofile consistency is hard to deny altogether."

Well, the reason why, is quite simple: since we are aggregating preferences, then, by the very act of aggregation, *all pairs are connected*. Hence there is no 'quite unconnected'; and interprofile consistency is only achieved by taking this into account. Given the dependence of the SWF upon the individual preferences, it would not be wise to adopt a SWF-GM which tries to construct a SWF on a pair $\{z, y\}$ without taking into account the preferences of the x in the background.

APDM implies the expansion lemma, which holds that decisiveness on a particular issue implies this on any issue, or, that if a group has its way in one instance, then this will happen with any group preference. A popular example is the majority voting rule, which in popular thought implies that a majority party can have its way in everything it agrees on itself. However, closer inspection reveals the following. With (1: $x > y > z$), (2: $x > y > z$) and (3: $z > x > y$) (whith aggregate ($x > y > z$)), it may be that (1 & 2) have their way on $y > z$, but that does not mean that this still holds when 2 changes its mind into (2: $y > z > x$) (whith aggregate ($x = y = z$)). We find that 2's preference change has a consequence, since it activates the preferences of somebody not in the group; or it means in other words that 2 effectively leaves the group. (That is, if one rationally applies not majority 'voting' but a majority *deciding* rule.)

10.5.6 Ex falso sequitur quodlibet

It is a property of logic that you can deduce anything from a falsehood. "If the moon is made of green cheese, you should give me all your money." In Latin, this property has the name *Ex Falso Sequitur Quodlibet* (from a falsehood follows whatever you want).

Arrow's proof has the *reductio ad absurdum* structure. On this, Arrow draws a specific conclusion. But rather, once a contradiction is found, then everything is under attack. The selection of the cure is quite a different matter.

A warning is in place when you study Arrow's Theorem. Given the wild conclusions aired about it, it is only natural that you set out with the assumption that there must be some logical error. It is dangerous however to say this aloud, since some people will be so unkind to start doubting your mathematical competence. Subsequently, it does not help that the Theorem has a *reductio ad absurdum* format, since, due to the *ex falso sequitur quodlibet* property, a falsehood, once reached, can result into any conclusion. In particular, a conclusion that Arrow apparently made an error somewhere belongs to the possible impressions that you might arrive at. Subsequently, you are subjected to a logical struggle, from which you will survive only by rejecting the axioms.

Cool (1990g) gives three examples how you might get the impression that there is something wrong with the proof of the Theorem. It is useful to write 'proof', since we can clearly indicate logical problems.

1. One possible 'proof' (presented by Luce and Raiffa (1957) or Feldman op. cit.) starts with a one person decisive group, tries to expand the group, then arrives at a contradiction, and then concludes that this person must be a dictator. There the 'proof' stops. **But:** it would only be logical to continue as follows: start with *every* person, and show that *everyone* is a dictator ! Hence the idea that one has identified a dictator must be rejected.
2. One kind of impossibility 'proof' starts with assuming that the whole community is decisive. **But:** under the Condorcet situation we have ($x = y = z$), and hence there is no strict preference, and hence there is no decisive group.
3. Similar, but a bit more complicated: Regard Sen (1986:1073-1080). Note that the definition of *decisiveness* p1078 depends upon a *specific* preference

profile $\{R_i\}$. The proof of the *group contraction lemma* p1080 uses this notion, for an arbitrary profile. The '*proof of the GPT*' p1080 uses this. Hence what the 'proof' would show is that for an arbitrary though specific choice one might identify a supporter. **But:** *arbitrary* is not *all*. That is, this *does not show* that for *any* $\{R_i\}$ this would be the *same* person. Thus the definition of a *dictator* (p1078) is not fulfilled. Hence the 'proof' is invalid. (Though nowadays I doubt whether this deduction is fully Arrow-ian.)

10.6 Deontic logic vs preference

10.6.1 Domain of the preference orderings

Since our position is more abstract than the theorem itself, it should not come as a surprise that we gain new insights. One insight is that the domain of preference orderings can be extended from the commodity domain to the domain of constitutional rules. Using deontic logic we then can prove that Arrow's view is unsound.

[Addendum 2001.] Note the following possible source of confusion. (1) In the base situation, we have the logical calculus of Arrow's Theorem (AT) and the environment of the discussion about it (E). We can usefully depict this as $AT \subset E$, so that E has a hard core. (2) Now I say that I formalise part of E , and arrive at a new construct F . (3) This apparently can be misunderstood as giving only $AT \subset F \subset E$. It is thought that AT is entirely logical, so that it has a general validity, and so that no F can affect it. In this view, the conclusion is forwarded that I tell nothing new. This view is in itself deficient, since $F \subset E'$, and a part $E \cap E'$ disappears, some of the confusion disappears. (4) The better explanation however is this. Both AT and F are general, so it does not help when we interpret these sets such that we get two universals. It is better to define the elements in the sets as "empirical applications". In that case $AT = \emptyset$. Then $F \cap AT^c \neq \emptyset$. I think that this is a major step. AT is not suitable for rational agents who are also in control of their own constitution. (And some fuzzy confusion $E \cap E'$ still disappears.)

The above may be clarified by considering whom the theorem would apply to. We can distinguish (1) irrational beings - who would throw dice for their decisions, (2) dumb robots - comparable to ants who just run a program, (3) intelligent beings - who try to solve a problem. Since group 1 does not have consistent preferences, it drops out. Group 3 would be my target, but we have shown that Arrow's axioms cannot apply to those (since they would reject them). It follows that only group 2 is a candidate for application of Arrow's theorem. But then, when we try to apply the axioms, it is seen again that they are inconsistent, so that even the ants cannot apply them.

10.6.2 Constitutional process

It would seem that the very social process of making a constitution makes people as a whole more conscious of the fact that the rules must be feasible for the constitution to exist. This will create some pressure to settle on one of the feasible rules. It is possible that individual frustration continues with $Oa \ \& \ \neg a$, but such cannot happen for the aggregate. (I refrain from developing a theorem on that.)

10.6.3 Deontic logic vs preference

A reduction of deontic logic to economic preference is not quite in the spirit of both subjects, vide the Kantian argument; but it might nevertheless apply for utility functions with a clear top (*satisfied*), provided that such a top can be called the *summum bonum*.

For example, deontic theorists discuss the rationality of the Deontic Axiom (DA): $Op \ \& \ (p \Rightarrow q) \Rightarrow Oq$. A pragmatic example is: you like sugar, sugar makes you fat, thus you like to grow fat. (Note that $(p \Rightarrow q)$ is technical.) In the pragmatic realm, people compromise on an indifference curve between sugar and growing fat. In ethics, there would be no such compromise. You should not kill, if you press the button many will die, hence you should not press this button.

Perhaps it is feasible to have ethical ideas about buttons; perhaps it is insane thinking of it as a button and rejecting all personal responsibility for its electrical connections... Most likely, such complexities are solved by introducing distinctions between primary ethical ideas and their practical consequences.

10.6.4 Degrees of moral obligation

Considering the axioms, we have found that Ought can have some structure, in that some Oq are more important than some other Op . The operator O does not indicate this, and it thus is not really an ordering, but mainly the separation of the Oughts from the Alloweds. It indicates a cutoff point in the whole R , where mere preference turns into moral obligation. Thus, what Op and Ap do, is that they clarify for a p to what class it belongs, while neglecting the intensity of the preference or degree of moral obligation.

We might use $(Op \ R \ Oq)$ to indicate that $O(p \ \& \ q) \ \& \ (p \ R \ q)$.

But indeed, it is not always necessary to express the degree of obligation. Having such a class of Oughts, we would, at one time, only wish to communicate that only that class is obligated, and that all other q are allowed.

We can define "Strong Ought" as "Only p ought, and there is not some other q that ought as well" and denote this as $O!p \Leftrightarrow ((Y \setminus \{p\}) \ P \ p)$. Since *Mathematica* uses "!" for " \neg ", this can be confusing though. If we would use the diameter symbol \varnothing , then this might be misunderstood as "Ought does not apply, you may". Clearest then is $(OO)p$, taking "only ought" as a single symbol.

In that case we might say $(OO)(AU \ \& \ AWP \ \& \ AD)$.

But note that we can still be flexible in the implementation of AU and X . For practical constitutions, we would not mind excluding future generations, and we would concentrate on the current budget. So the clarity of OO is not too large.

10.7 Some literature on Arrow's Theorem

10.7.1 Duncan Black

Arrow & Scitovsky (1969) bundle various classic papers, e.g. dating from 1943. These papers show a small tragedy. It was Duncan Black (1948) who gave an impetus to the study of rationality of the voting process, and who called attention to the paradoxes of voting. Arrow (1950) & (1951) concentrated primarily on the latter. But, Black (1948) also wrote: "the committee adopts as its decision ("resolution") that motion, if any, which is able to get a simple majority over every other". From Black's elaborate description of procedures too it is evident that he was conscious of the distinction between the simple act of casting a vote, and the complex effort of arriving at a decision. But somehow this important distinction got lost in the subsequent onslaught.

10.7.2 Robert Dahl 1956

Dahl (1956) (sensitive to *deciding*) calls the Condorcet situation a 'deadlock'. In other words, there indeed could be a case for social indifference; and to further resolve it: allow for a chairperson, or power, or dice, or more bargaining, or a stricter budget - or whatever gets Buridan's ass going. But, while indifference indeed is the solution, Dahl at the same time (sensitive to *voting*) prefers the solution of intransitivity of social choice (p43 fnnt 12).

10.7.3 Jan Tinbergen 1956

Tinbergen (1956:14) is the most detached: "the author doubts the relevance of the question whether social welfare functions can or cannot be derived from individual ones (...). For the time being the margins of inaccuracy (...) would seem so large, and our exact knowledge of individual welfare functions so limited, that the theory of economic policy would be better to take the policy-maker's welfare function as its starting point. But, no doubt, this has to be a temporary attitude only."

10.7.4 Leif Johansen 1969

Schrijver (1987) and Ancot & Hughes Hallet (1984) have empirical methods; and the latter write: "a collective preference function is not necessarily the same as a social welfare function, and Arrow's 'Impossibility Theorem' often does not apply (Johansen 1969). For example, bounded rationality, restricted choice sets, or a degree of cooperative bargaining (...)"

10.7.5 Amartya Sen 1986

With understandable enthusiasm, expositions on the Impossibility Theorem try to impress the reader with the Need of Accepting the Arrow axioms. For example, see Sen's "Big Bang", or see Feldman (in the 'welfare economics' entry in the Palgrave) who calls Arrow's Theorem the 'Third Fundamental Theorem of Welfare Economics'.

10.7.6 Representative agent 1989

One approach 'around the problems' has been, like Blanchard & Fischer (1989:567) indicate, among other approaches: "If the economy has identical (...) individuals, the social welfare function naturally coincides with the utility function of these individuals." But the position of not assuming a SWF but only $SWF/n \cdot n$ is not only limited but might also be inadequate anyhow (apart from wrong conclusions on poll taxes).

10.7.7 Dale Jorgenson 1990

Let us repeat the quote by Jorgenson (1990), once president of the Econometric Society, who concludes 'more positively' to dictatorship:

"The classic result of social choice theory is Arrow's (...) impossibility theorem, which states that ordinal noncomparability of individual welfare orderings implies that a consistent social ordering must be dictatorial, corresponding to the preferences of a single individual."

(1) Arrow's result hinges on *APDM* and this is not the same as noncomparability.

(2) It seems to me that there can only be noncomparability when the only rule is Pareto-optimality, which gives universal veto powers within a system of well

established property rights. And here *APDM* is redundant (even inconsistent, for consistency requires dropping at least one axiom).

(3) Accepting dictatorship does not by itself imply consistency. Arrow proves that a set of axioms renders inconsistency, but he does not prove that dropping one axiom and choosing an *actual* dictator achieves consistency.

Regard dictator The Great Cornelia, who happens to be a chicken. According to the viewpoint of The Great Cornelia, all human history has only taken place just to culminate into the following laboratory setup, groomed to warrant and monitor her supreme well-being. The Great Cornelia finds bliss by having at most 120 picks a day, at either a red button (grain) or a yellow button (water). On sunny days she has 10 units of grain and 2 units of water, giving $x = \{\text{sun}, 10, 2\}$, but when it rains then the humid air allows her to drink less, so that she accepts $y = \{\text{rain}, 11, 1\}$. Thus the preference order between $a = \{10, 2\}$ and $b = \{11, 1\}$ depends upon a third option, violating *APDM*. Some readers might object to this setup, so we put the weather in her budget, with a door to the left leading to an artificial sun and a door to the right leading to a shower. Now it appears that after some sunny days she longs for a shower, and after a shower she longs for some sun. Thus the choice between x and y depends upon a third phenomenon, the history of events. Of course, the setting for Arrow's Theorem is statics, and not dynamics, yet it would seem that there is sufficient cause to start wondering.

10.7.8 Researchers Anonymous 1990

It appears that some researchers are conscious of the fact that Arrow's and others's words do not match the mathematics. To these researchers, the value of Arrow's deduction appears to consist of the fact that it shows that axiomatisation of social choice is difficult: so that one should be very careful and hesitant in expecting too much from any one particular effort.

It should be noted that this reading is very different from the one expounded by Arrow and adopted in many places in the literature. So, this difference in opinion in itself already provides a justification of this book. Also, the distinction between voting and deciding, in itself and through its consequences, might clarify that the basic problem in axiomatisation is *not* too difficult. It are only the ramifications that create difficulty in the design of proper procedures for each situation.

10.7.9 Mas-colell, Whinston and Green 1995

Addendum 2000: Andreu Mas-colell, Michael Whinston and Jerry Green 's 1995 "Microeconomic Theory" is just wonderful. A great, magnificent book. Generally speaking, though, since they erroneously write: "Either we must give up the hope that social preferences could be rational in the sense introduced in Chapter 1 (i.e. that society behaves as an individual would) or we must accept dictatorship." (p780). And the subsequent discussion indeed leads the student in the bogs and misdirections so typical of 20th century 'social choice theory'. The math is OK, but concerns something like the question of how many angels can dance on a pin's head - and the whole induces the student to become wary of social decision making. (To be sure: I appreciate the other qualities, and have used the book for important sections of my Economics Pack.)

10.7.10 Conclusion

See also the quotes provided in chapter 9.2.

The Arrow proposition has caused an amazing tension within economic theory and the profession. The manner in which the proposition is accepted ranges from enthusiasm, to detachment, to grudge. To the enthusiast, the Ramsey/Tinbergen approach would not be rational, even though some still adopt it, albeit perhaps implicitly. In this, also, the voting aspect is seen as more important than the trivial clarity of additive cardinality (power processes).

10.8 Sen

[Addendum 2001.] In 1990 I relied more on Sen's discussion in the Handbook, but now I have also used Sen (1970). I can usefully react to that. The following discussion repeats a bit what has been said in the main body of this book, since Sen has been important for the whole subject. Yet, this is a good place to discuss Sen's general approach and to consider the question whether Sen really understood the issue at that time.

10.8.1 APDM

Sen (1970:38) calls Arrow's result "rather stunning" and gives a rather inaccurate description of the role of APDM.

Sen (1970:36):

"(...) it is logically perfectly alright to postulate the following SWF: If person A ("that well-known drunkard") prefers x to y , then society should prefer y to x , and if A is indifferent, then so should be society. As a SWF this can be best described by a non-technical term, viz., wild, and in serious discussions it may be useful to restrict the class of SWFs (...) by eliminating possibilities like this. One way of doing it is to require that the SWF (...) must satisfy certain conditions of "reasonableness". Since reasonableness is a matter of opinion, it is useful to impose only very mild conditions, and one might wonder whether one could really restrict the class of SWF's very much by such a set of mild conditions. Well, surprisingly, the problem comes from the other end. In his "General Possibility Theorem" Arrow proved that a set of very mild looking conditions are altogether so restrictive that they rule out not some but *every possible* SWF."

In reaction to this:

- It is a wrong to suggest that matters of opinion on reasonableness can be settled by the strategy of imposing 'mild' conditions. If opinions differ wildly, then 'mild' conditions certainly will not help. A proper way to settle matters of opinion is to develop the consequences of the different assumptions, and to see whether there can be some common ground for their consequences.
- It is wrong to suggest that such wild SWFs are the reason to look into the conditions for the SWFs. The reason to look into the conditions are the problems of cheating.

- Arrow's axioms cannot be called 'mild' conditions, and neither are they reasonable. Witness their consequences. (Perhaps there is a confusion between 'mild' and 'simple'. *APDM* looks rather simple. But a statement as "kill all mosquitos" looks simple as well. We need only 3 words to convey a complex message. But we would not call that message 'mild'.)

Sen (1970:39) uses the following example to show that Borda's method does not satisfy *APDM*. We have discussed this above, but can usefully repeat it.

- Define the problem of (1: $x > y > z$) and (2 & 3: $z > x > y$):

```
EqualVotes[3]; Clear[x, y, z]; Items = {x, y, z};

{Pref[z, y, x], a = Pref[y, x, z], a} // SetPreferences;

Preferences
```

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 1 & 3 \\ 2 & 1 & 3 \end{pmatrix}$$

- Borda would select both x and z , while only z is a fixed point.

```
Borda[]
```

```
{x, z}
```

```
BordaFP[]
```

```
BordaFP::chg : Borda gave {x, z}, Fixed Point is {z}
```

```
z
```

```
BordaAnalysis[]
```

$$\left\{ \text{Select} \rightarrow \{x, z\}, \text{BordaFPQ} \rightarrow \{\text{False}, \text{True}\}, \text{WeightTotal} \rightarrow \left\{ \frac{7}{3}, \frac{4}{3}, \frac{7}{3} \right\}, \text{Position} \rightarrow \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \text{Ordering} \rightarrow \begin{pmatrix} \frac{4}{3} & y \\ \frac{7}{3} & x \\ \frac{7}{3} & z \end{pmatrix} \right\}$$

Sen proceeds to show that Borda does not satisfy *APDM*. When individual 1 changes his mind on y (which Sen calls "irrelevant") such that y now becomes less than both x and z , then z gets one point more, and Borda now selects z while it does not mention x any more. Note that everyone has kept the same preference order on x and z . *APDM* then requires that the group order would remain the same - but it has changed. Sen's exposition on this example is 'matter of fact', and technically sound.

But Sen presents this example to show something. He presents this discussion within the framework of suggesting (a) that Arrows axioms are mild and reasonable, and (b) that we can nihilistically conclude that it is useless to search for something else to satisfy those, and by implication *such*, conditions. His whole use of language and frame of mind is off-track for the problem of group decision making. He does not see:

- A change in a preference on y is not irrelevant, since we are discussing group decision making, and everything then depends upon everything. (In this case, the change directly affects z for individual 1, which clearly shows the effect. The proper reaction is not to say that it is just a simple example, which would suggest that the dependence would not exist in complexer cases - but the proper reaction is to say that this effect is exemplary, so that in complexer examples the same effect is less obvious.)
- The problem with Borda is rather the influence of budget changes. We would get a different result if y is included in the budget set or not, even though y does not have a chance of winning. BordaFP gives us a better criterion to judge on this interpretation of relevance. Having y in the budget or not has no effect on BordaFP, and in that respect y could be judged irrelevant for the group decision.
- Note that BordaFP retains a bit of pairwiseness, since it compares the Fixed Point winner with the winner of the alternative match. But a bit of pairwiseness does not make a whole APDM reasonable. If your skin has a nice small green spot, then it does not mean that you want to be wholly green.
- The real problem is that we should judge Borda (BordaFP) in terms of how it contributes to controlling cheating.

Sen (1970:40):

"The importance of the General Possibility Theorem lies in the fact that we can predict the result in each case, viz., that the specific example considered will not pass the four conditions, even without examining it. The theorem is completely general in its nihilism, and saves a long (and perhaps endless) search."

In reaction to this:

- It is curious that Sen combines the words "Possibility" and "nihilism".

- If "nihilism" is intended neutrally, then we might accept this quote as a fair conclusion. Arrow's axioms don't lead anywhere.
- But the axioms are also called mild, reasonable and morally desirable, and then the "nihilism" becomes, by implication, the philosophical world view, and then the conclusion is wholly absurd and unwarranted.
- And there is still the suggestion that the axioms would be proper for decision making by rational agents, while we have shown that those agents are concerned with other matters than these axioms describe. Sen still suggests that a search for this issue would be useless, which it is not. The title of his book is "Collective choice and social welfare", and thus one expects a discussion of agents who are sensitive to the paradoxes around the budget. But now it seems that this is declared a useless search.

Hence Sen (1970) does not understand it. This is not to say that I do not value much of his work. Sen's work has been an important for the development of my own analysis. As DRGTPE repeated: On the shoulders of giants, we can look further.

10.8.2 Pareto and the status quo

Sen rejects the Pareto principle because of the redistribution aspects (exploitation) and uses his 'impossibility of the Paretian liberal' as munition.

Sen (1970:198) takes a modest position: "It is not being argued here that no general principles exist that would secure total adherence of a person, but that the simple principles usually recommended are not of that type."

Thus, ideals still may exist. Yet, modest as this position is, it still is misleading of course, in that the 'simple principles' are rejected by a wrong argumentation.

P197: "If one takes the view that Pareto optimality is the only goal, and as long as that is achieved, we need not worry further (an approach that is implicitly taken in much of modern welfare economics, but rarely explicitly), then (...) we must declare all Pareto-optimal points as indifferent (...) This result gives an axiomatization of an approach that is implicit in a substantial part of modern welfare economics. (This result ...) is quite disturbing. All the imposed conditions are superficially appealing, but the conclusion that Pareto-optimal points are indifferent, irrespective of distributional considerations, is very unattractive. In fact, it is this aspect of modern welfare economics that is most often separated out for special attack. (...) We found difficulties with even a very limited use of Pareto optimality (...) with a very weak condition of individual liberty, which gives individuals the freedom to do certain personal things (e.g. choosing what one should read).

Even if only two individuals are given such freedom and over one pair each, the Pareto relation may still have to be violated to ensure acyclicity (...). Hence, Pareto optimality even as a necessary but not sufficient condition is open to some question. (...What) emerges (...) is a serious doubt about its *merit* as a goal (...)"

- Pareto optimality is not the only goal. It is one of the goals.
- Pareto optimality is only accepted as a necessary condition with the understanding that there are other conditions in the background as well. It is a condition e.g. for voting rules while assuming a civilised setting. Perhaps one person once suggested that it would be universally necessary, without compromise, but classical liberals generally were wiser. For example, allowing criminals the right to veto their imprisonment, is not a feature of classical models of society.
- Sen's discussion of his 'impossibility of the Paretian liberal' has been shown to be inadequate.
- Please explain why it is unattractive that society is indifferent about all kinds of possibilities ? This is just posed, not shown. (See below, for the tie-breaker of the status quo.)

Sen (1970:118): "There is a certain social state \hat{x} (the "status quo") which will be the outcome if the two persons fail to strike a bargain. (...) Nash (...) proposes a solution that is given by maximising the product of the differences between the utility from a cooperative outcome x (Pareto-superior to \hat{x}) and the status quo outcome \hat{x} for the two, i.e., maximising $[U_1(x) - U_1(\hat{x})][U_2(x) - U_2(\hat{x})]$." P120: "In (...) the Nash approach no such comparability is introduced, but the origins are knocked out through the use of the status quo and the units are rendered irrelevant through the multiplicative form. (...) In splitting the gains from an agreement, state \hat{x} is clearly relevant."

P121: "This does not, however, mean that the Nash solution is an ethically attractive outcome and that we should recommend a collective choice mechanism that incorporates it. A best prediction is not necessarily a fair, or a just, outcome. In a labor market with unemployment, workers may be agreeable to accept subhuman wages and poor terms of employment, since in the absence of a contract they may starve (\hat{x}), but this does not make that solution a desirable outcome in any sense. [sic] Indeed, compared with \hat{x} , while a particular solution may be symmetric in distributing utility gains from the bargain between workers and capitalists, we could still maintain that the workers were exploited because their bargaining power was poor. (...) Whether or not the Nash solution is predictive (...), its ethical relevance does seem to be very little."

P123: "The special importance attached to the status quo point and to threat advantages, and the complete avoidance of interpersonal comparisons, seem to rule out a whole class of ethical judgments that are relevant to collective choice."

- We introduce ordinality to reduce the possibility of cheating. It is true that this reduces the possibilities for interpersonal comparisons. It is a hefty price, yet, it would be the required price when we think that cheating is a serious problem. Sen now neglects this point, and uses interpersonal comparisons to single out the status quo. This does not seem balanced. We should be aware that people could cheat, saying that the status quo is really terrible for them.
- It is true that social choice theory should try to develop mechanisms that use interpersonal comparisons, since that is the basic issue. But the fact that this appears to be difficult, cannot be held against intermediate solutions, such as the Pareto principle.
- The Pareto principle also protects workers. In general, systems of justice basically protect the weak - since the strong do not need official protection.
- If people really would starve, then they would be wise, and it would be ethically just, to accept work that is an improvement. From that position onwards, they can try to better their position. Thus there *is* a sense in which the solution is acceptable.
- The word 'exploitation' applies if some minority uses inappropriate means to improve their own position. Sen would have to show more about the situation before we could conclude this.

10.8.3 In “Development as freedom”

[This is taken from DRGTPE.]

Sen (1999a:250-253) contains a short summary discussion on his current view on the Theorem. First I quote him and then give my comment. Sen states:

“The Arrow Theorem does not in fact show what the popular interpretation frequently takes it to show. It establishes, in effect, not the impossibility of rational choice, but the impossibility that arises when we try to base social choice on a limited class of information.”

This is not correct. Using the information provided by pairwise voting results, we can decide to a deadlock (indifference) when such might arise. It is the adoption of the *APDM* axiom that, wickedly, turns this indifference into an inconsistency. The *APDM* does not mean lack of information, it only corrupts, eliminates, the information that exists.

“At the risk of oversimplification, let me briefly consider one way of seeing the Arrow theorem. Take the old example of the “voting paradox,” with which eighteenth-century French mathematicians such as Condorcet and Jean-Charles de Borda were much concerned. If person 1 prefers option x to option y and y to z , while person 2 prefers y to z and z to x , and person 3 prefers z to x and x to y , then we do know that the majority rule would lead to inconsistencies. In particular, x has a majority over y , which has a majority over z , which in turn enjoys a majority over x . Arrow’s theorem shows, among other insights it offers, that not just the majority rule, but *all* mechanisms of decision making that rely on the same informational base (to wit, only individual orderings of the relevant alternatives) would lead to some inconsistency or infelicity, unless we simply go for the dictatorial solution of making one person’s preference ranking rule the roost.”

Locating the problem in the informational base is erroneous. Clearly, majority decision does *not* lead to inconsistencies, for it is the use of the *APDM* axiom that does so - and we don’t need it for majority decisions. The Arrow Theorem does *not* show that there are inconsistencies for *all* mechanisms - we namely can use mechanisms without *APDM*.

“This is an extraordinarily impressive and elegant theorem — one of the most beautiful analytical results in the field of social science. But it does not at all rule out decision mechanisms that use more — or different — informational bases than voting rules do. In taking a social decision on economic matters, it would be natural for us to consider other types of information.”

I don’t know about “extraordinarily impressive and elegant”. Condorcet came up with his paradox, as earlier people came up with paradoxes when dividing by zero, as Bertrand Russell had his set-paradox, and as the Cretian Epimenides said “All Cretians are liars.” Arrow’s Theorem solves the Condorcet paradox by showing that we must not use *APDM* - though Arrow apparently did not realise that. The theorem is basic, and we must be glad that we have it, as *APDM* apparently can cause a lot of confusion, as the last 50 years have shown.

“Indeed, a majority rule — whether or not consistent — would be a nonstarter as a mechanism for resolving economic disputes. Consider the case of dividing a cake among three persons, called (not very imaginatively) 1, 2, and 3, with the assumption that each person votes to maximize only her own share of the cake. (This assumption simplifies the example, but nothing fundamental depends on it, and it can be replaced by other types of preferences.) Take any division of the cake among the three. We can always bring about a “majority improvement” by taking a part of any one person’s share (let us say, person 1’s share), and then dividing it between the other two (*viz.*, 2 and 3). This way of “improving” the social outcome would work — given that the social judgment is by majority rule — even

if the person thus victimized (viz., 1) happens to be the poorest of the three. Indeed, we can continue taking away more and more of the share of the poorest person and dividing the loot between the richer two—all the time making a majority improvement. This process of “improvement” can go on until the poorest has no cake left to be taken away. What a wonderful chain, *in the majoritarian perspective*, of social betterment!”

Remember that Sen writes this book for a general audience of economists who will not have gone deeper in social choice theory. Though Sen now relates basic truisms, his reasoning nevertheless is a bit off. Indeed, Western democracies tend to have property rights and a “status quo” rule, and a Madisonian philosophy that democracy actually exists to protect the minorities. We use all kinds of additional information, in order to settle problems of fairness and equity. Thus the majority rule is not suggested for the raw form that Sen uses as an example. Then, crucially, when Sen suggests that this example clarifies that we must use more information to solve the Arrow paradox, then this is a non-sequitur. His argument becomes seductive, since the reader is seduced into thinking that, indeed, we use more information. But the truth is that we use this additional information to solve equity matters, and not to solve the Arrow inconsistency.

Besides, when we have the Pareto-Majority constitution, then we can rely on the poor for blocking proposals that reduce their part of the cake even more. So we use the same informational base, the preference orderings, to solve Sen's example problem.

“Rules of this kind build on an informational base consisting only of the preference rankings of the persons, without any notice being taken of who is poorer than whom, or who gains (and who loses) how much from shifts in income, or any other information (such as how the respective persons happened to earn the particular shares they have). The informational base for this class of rules, of which the majority decision procedure is a prominent example, is thus extremely limited, and it is clearly quite inadequate for making informed judgments about welfare economic problems. This is not primarily because it leads to inconsistency (as generalized in the Arrow theorem), but because we cannot really make social judgments with so little information.

“Acceptable social rules would tend to take notice of a variety of other relevant facts in judging the division of the cake: who is poorer than whom, who gains how much in terms of welfare or of the basic ingredients of living, how is the cake being “earned” or “looted” and so on. The insistence that no other information is needed (and that other information, if available, could not influence the decisions to be taken) makes these rules not very interesting for economic decision making. Given this recognition, the fact that there is *also* a problem of inconsistency—in dividing a cake through votes — may well be seen not so

much as a problem, but as a welcome relief from the unswerving consistency of brutal and informationally obtuse procedures."

Sen is aware that his reasoning is not strict (vide his use of the word 'primarily' and "*also*") but, still, he makes the suggestion, which is erroneous.

"Indeed, the spirit of "impossibility" is not, I believe, the right way of seeing Arrow's "impossibility theorem." [footnote] Arrow provides a general approach to thinking about social decisions based on individual conditions, and his theorem—and a class of other results established after his pioneering work — show that what is possible and what is not may turn crucially on what information is taken into effective account in making social decisions. Indeed, through informational *broadening*, it is possible to have coherent and consistent criteria for social and economic assessment. The "social choice" literature (as this field of analytical exploration is called), which has resulted from Arrow's pioneering move, is as much a world of possibility as of conditional impossibilities. [footnote]"

This quote just repeats the error - and adds a string of perceptions to sweeten the cake. The footnotes are references to his "Collective choice and social welfare", his Handbook contribution and the Nobel lecture, Sen (1999b), and add no news, for us, to the essence discussed here. Indeed, the obviously relevant Nobel lecture just repeats the error.

Hence, Sen basically does not understand the problem. I do value his work on social choice since it was a useful guide to me in making Arrow's result accessible, and in seeing the various perspectives of it. As Newton is reported to have said: "Standing on the shoulders of giants, we can look further."

10.9 Miscellaneous sources for confusion

10.9.1 Introduction

Cool (1990g) finally contains various comments on other possible sources for confusion. These are of miscellaneous character.

10.9.2 Arrow and "Arrow"

We will refer frequently to "Arrow", forced by regrettable convention - which convention however is also adopted by Arrow, see his entry "Arrow's theorem" in the Palgrave. Such frequent reference to Arrow and "Arrow" might give the impression of polemics, but this is not intended and derives only from that regrettable circumstance. Thus it should be understood that we are concerned only with abstract thought among a large body of authors and readers.

10.9.3 Keeping the logic straight

Some people seem to reason as follows: (a) There are axioms for rational social choice. (b) From this we deduce a contradiction. (c) Hence rational social choice is impossible. Now, replace 'rational social choice' with Z. Then we get the structure: (a) axioms for Z, (b) contradiction, (c) ergo, non-Z. You can verify the illogic by choosing your own Z. The proper conclusion is (c') ergo, not these axioms for Z.

10.9.4 Axiomatic analogy

There is a nice small analogy that accurately copies the situation. Mr. X poses the following axioms: (a) To be able to ride, a bicycle must have round wheels. (b) To be able to stand still, a bicycle must have square wheels. Mr. X then finds that wheels cannot be round and square at the same time. Hence Mr. X concludes that bicycles cannot exist. And Mr. X argues that the axioms are reasonable and morally desirable. This analogy is not simplistic, and it accurately has the shape of the argument. (1) There is a convoluted concept like "square wheel" that asks too much in a wrong way - which is Arrow's Axiom of Pairwise Decision Making (here *APDM*, Arrow *AIIA*). (2) The set of axioms is incomplete for the problem that it tries to capture - there is a mixing up of voting and deciding. (3) Solutions like using a bike-support can be reasonable and morally desirable. What Mr. X does is quite curious, since people have been riding around and parking bikes for ages. The only way to understand the situation around Mr. X is that his result has come about in a specific historical setting, and not everything has been happening with full rationality.

10.9.5 Major paradox of voting

One of the major paradoxes of voting is that many of its theorists try hard to avoid interpersonal comparison of utility, while voting in itself is an explicit way of doing just that. Why vote at all - when it is not for settling something together? (See also Dahl & Lindblom (1953/76:423).)

10.9.6 Refutation of the verbal claims

In our clarification we take a 'meta' point of view. We accept the logical calculus of Arrow's Theorem, and do not reproduce it. It suffices if you have a 'Palgrave level of understanding' of Arrow's Theorem. But we will look at the interpretation of it, and at its place within the field of inquest and its possible impact. It must be noted that the claims on rationality and moral desirability are verbal only, i.e. non-mathematical. They are words only, no formulas. We will put these words into formulas, and then the problem disappears.

Formalising the loose words of the common interpretation of Arrow's Theorem introduces an element of arbitrariness, and it thus becomes a possible source of criticism in that we would not have made the right translation, and that we might create our own straw-man to tear down. However, formalising loose

words always causes this problem. One may try alternative forms, but the clarity created by the present formalisation remains.

Our analysis here is that Arrow's and other's words do not match the mathematics. Specifically, what Arrow proposes to embed in his axioms, is not what people generally regard as rational or moral. Thus we accept the logic of the Arrow & Blau mathematical deduction that certain axioms result into an impossibility, but we reject the view that these axioms reflect collective rationality, and we reject the view that this deduction would be of interest anyhow, or that this impossibility has any importance or relevance.

10.9.7 It is not just rejection of the axioms

One may interpret our position here as a rejection of one of Arrow's axioms. But actually things are not so simple. Arrow's Theorem is that his axioms are inconsistent, so Arrow himself also has to reject at least one of his axioms. The more valuable interpretation of this book thus concerns our arguments about which axiom. Rather than being happy with loose comments on rationality and moral desirability we will give a reformulation of the problem.

10.9.8 Mental blocks

The present author does not claim to be a specialist in the field of Social Choice Theory. Since I restrict myself to this topic of Direct Single Seat Elections and the questions of interpretation, all at a textbook level, I do not feel that one can object to my lack of knowledge of other specifics of Social Choice Theory. Yet, specialists in Social Choice Theory tend to object to outsiders refuting one of their core views.

For the non-economic professions, it is useful to add that Kenneth Arrow is a well-respected economist, and for example winner of the Nobel Prize in 1972. Pikkemaat (1990) quotes Donald George:

"Every profession needs its heroes. Physicists have Einstein, guitarists have Eric Clapton, dogooders have Batman and economists have Kenneth Arrow."

Arrow's standing has had the - in itself curious - consequence that many people have accepted his words uncritically or with more tolerance for error than would be wise. A professor in mathematical economics, while acknowledging the truth of my conjecture that Arrow's words do not match the mathematics, countered tolerantly with "Quod licet Jovi, non licet bovi." Another researcher

acknowledged that Arrow's calling his axioms "reasonable" and "morally desirable" is only a loose use of words, but he declined interest in what would happen if we would formalise these words.

The directorate of the Dutch Central Planning Bureau (CPB), an institute that professionally should be interested in social welfare functions, caused some problems about making the internal note Cool (1990g) and blocked internal discussion and the possibility of eventual publication of it - without decent argumentation.

Since 1990 I have submitted two articles to an economic journal, but got only insulting comments for reply. NB. Some readers may feel that if journals come to the same rejection as the CPB directorate, that then the likelihood grows that there was something wrong with the analysis. This however would be too quick a conclusion. Being partners in crime does not provide a justification, and when there is system in the madness, it still remains madness.

It is a happy conclusion that the analysis now has grown into this book, but both the unscientific treatment and the ten years delay should give everyone cause for concern.

10.9.9 Evaluating Saari's approach

Though this Part basically just reprints my texts from 1990, I can usefully add the following. In 2000, I put below introduction on the internet. Writing this book, I can add another subsection.

10.9.9.1 Introduction

See section 4.8.6 again. Note that Saari claims that this approach solves the paradoxes of voting, and that, if we want "fairness", that we must implement "symmetry".

I think that Saari is right for 90% but still dangerously off-track. I am afraid that his approach creates a new illusion in the off-direction. Please note that I have the impression that Saari gives nice work in general, and that he should be recommended for his critical approach to the conventional views on Arrow's Theorem. Please note as well that my reaction does not come from 'priority' considerations. I have no doubt about it that my own analysis simply stands. My worry is on content, i.e. the policy errors and the failing academia.

Serious errors are being made in real life politics, based upon improper understanding of the issue. The academia make their errors with regards to logic - and thus don't provide proper clarification. Suppose, indeed, that you think that Saari is right: then you would agree with me that there have been serious policy errors ! Then you should worry as I do ! And then you might acknowledge that Cool (1990) was right on a point. This should not be without consequences.

Having said this, there are the following objections to Saari's approach.

(1) The suggestion might be (I don't say that Saari says so, however): "Arrow posed a deep problem, that took 50 years to solve, and it needed deep mathematical insights in 'symmetry' to solve it." If this is really suggested, then the proper comment is: Arrow did not pose a deep problem, his view should have been killed from the start. Economists and other theorists alike should be deeply ashamed for dropping common sense in the face of some math. And it just repeats the error to think that, for voting, in this manner, you need *symmetry* to be able to judge on *fairness*.

(2) Let us look closer into the use of the word "symmetry". I do agree that *symmetry* is a powerful concept and has a general attraction. People are psychologically very sensitive to symmetry - and math persons (like me) likely even more. Yet for that reason we should be careful in employing the term. The VoteMargin matrix is negatively symmetric, and an extension beyond 2 items is a serious option. But are we then discussing *indecision* or *symmetry* ?

Saari poses that symmetry is 'natural'. This however would relieve human beings from moral questions - and that is something that cannot be done.

Saari proposes *one* particular alternative to Arrow's "Axiom of Independence of Irrelevant Alternatives" (*AIIA*, *APDM*). Arrow says that *APDM* is reasonable and morally desirable. Saari apparently suggests that symmetry is reasonable and morally desirable. Now, how are we going to decide that ? A person in the street might say: "Well I like symmetry - a nice short word - and I never heard about 'independence of irrelevant alternatives' and it sounds like something really bad". As scientists we should be wary about such reactions, and provide proper schemes for reasoning. Note that this book gives a mathematical proof that *APDM* is neither reasonable nor morally desirable, which is much stronger

than only suggesting an alternative. Subsequently, Saari's "symmetry" could become a candidate. Yet, then, the argument should be on morals.

(3) Let us assume "symmetry" \Leftrightarrow Borda. Then it is up to human discourse on morality whether we want symmetry (Borda) or something else, depending upon location, time and purpose. Sometimes people don't vote explicitly, but assign 'wisdom' or 'experience' to selected individuals, and accept their decisions. These are situations that could be morally acceptable but they are not covered by Saari's 'natural' symmetry. And symmetry does not help either in the case of a deadlock, for example.

In his internet papers Saari suggests that the real problem is that *APDM* doesn't discriminate between rational and irrational voters, and argues that it is not surprising that unsophisticated procedures do not lead to sophisticated results. I do not think that this argument holds. It has been the strength of the Arrow argument that very *weak technical* assumptions already generated a contradiction. My argument is, on the other hand, that *APDM* blocks the flow of information that is required for group decision making. This blocking occurs whether voters are rational or irrational. Blocking the flow of information makes that the weak technical assumption becomes a *huge immoral* one. (While of course the conventional axioms presume rational voters anyway.)

Thus, I think that Saari provides useful insights in the matter, but with a limitation that is very much the same as the general limitation to the literature on Voting Theory. Saari might well be excused for thinking that he has found the resolution to Arrow's Theorem, but he should also study my analysis - and then he might be the first to see that this is the real solution.

10.9.9.2 Saari on rationality and transitivity

On the example discussed in 4.8.6, Saari (S&C:7) writes: "The (...) plurality vote and the Borda Count, ignore the Condorcet portion so they retain their ($B > A > C$) ranking but the destruction of transitivity now crowns (A) as the Condorcet winner! Rather than reflecting (A's) merits, this outcome underscores a serious flaw of the pairwise vote and Condorcet's approach. Incidentally, this example is essentially the one used by Condorcet to discredit the Borda Count and all other weighted voting systems. (To recover Condorcet's example, add another voter for each of the six possible types.) But, by destroying the assumption of transitive preferences, rather than the example supporting Condorcet's approach, it underscores problems with the Condorcet winner!"

I consider this reasoning very curious. Saari accepts that the Borda preferences of the individual voters are transitive - clearly, since he also supports the Borda approach. But suddenly, when these very same preferences are processed through the mill of pairwise voting, they suddenly are called irrational ...

It is interesting to look into this a bit deeper. Suppose that there is pairwise voting and that someone is indifferent between A , B and C . If the vote is about $\{A, B\}$, then this person, rather than voting $\{1/2, 1/2\}$ might rather pass, causing $\{0, 0\}$. However, in some cases, it is not allowed to vote $\{1/2, 1/2\}$, and neither to pass. The person might be 'forced' to make a decision. In that case, the rational voting strategy is to vote in the cycle, in the vote on $\{A, B\}$ to vote for A , in the vote on $\{B, C\}$, to vote for B , and in the vote on $\{A, C\}$ to vote for C . In that case each item gets 1 vote, and the person has expressed indifference. We have seen the same situation for chess games.

Saari argues that the pairwise voting scheme does not distinguish between rational voters and irrational voters. "Cycles, then, manifest the deplorable property that the pairwise vote vitiates the critical assumption of rational voters with transitive preferences." (S&C:6). He also argues: "(...) notice that the Borda Count does monitor whether a voter is rational".

This is not only strange in the light of the particular example, but also since at another point he shows himself aware that Borda and Condorcet are quite related, as we showed this too. The Borda count arises from adding the pairwise votes. This relationship is extended, in that voting 'irrational' is the same as voting 'indifference'. Of course, if C is not present, then, in this case, it

is not possible to express indifference by voting in a cycle, and so it is better to have systems that allow voters to pass or to express their indifference. Allowing for this, we can restrict attention to rational voters, so that the cause of the problem really must be sought in another direction. Saari argues that imposing the condition that voters are rational, causes the contradiction for Arrow's theorem. In my impression this is turning the world upside down. I would rather say that *APDM* is only valid for voting fields and does not fit the problem of deciding. As has been argued above.

Incidentally, I also reject his discussion of Sen's theorem. Saari's suggestion that there is a connection via 'rationality' is not correct. It is fine, though, to see that Saari agrees that personal liberty should be properly implemented, and that Sen's version of it is incorrect.

11. Conclusion

This book has discussed Voting Theory from the bottom up. The basic insight has been that people can cheat on their preferences, and that this problem is rather equivalent to the problem that preferences are difficult to measure and compare anyhow. The different voting schemes that have been proposed, try to solve this problem, but, by doing so, they also create the paradoxes of voting. These paradoxes are just seeming contradictions, but no real contradictions. The major conclusion is that group decisions can depend upon the budget of items that they vote about. If the budget changes, then there can occur swings in the decisions that could be surprising. If we understand what is happening, however, then we can deal with the situation, and in general we can find solutions that are reasonable and morally desirable - where of course each group has to decide itself what it considers reasonable and desirable.

One mechanism is suggested that many might see as useful for common occasions. This is the Pareto Majority scheme, where first all Pareto points are selected, and where these are subjected to a fixed point Borda scheme. Ties between such fixed points could be settled by the vote margin over the entire budget. Final ties would be up to particular preferences, such as dice, a chairperson decision, etcetera. It may be noted that accepting a tie-breaking rule that depends upon the budget increases the sensitivity to the budget, where Pareto and the fixed point Borda reduced that, but it would seem that a tie breaking rule is necessary, and since all points already are Pareto, a small additional sensitivity to the budget does seem proper.

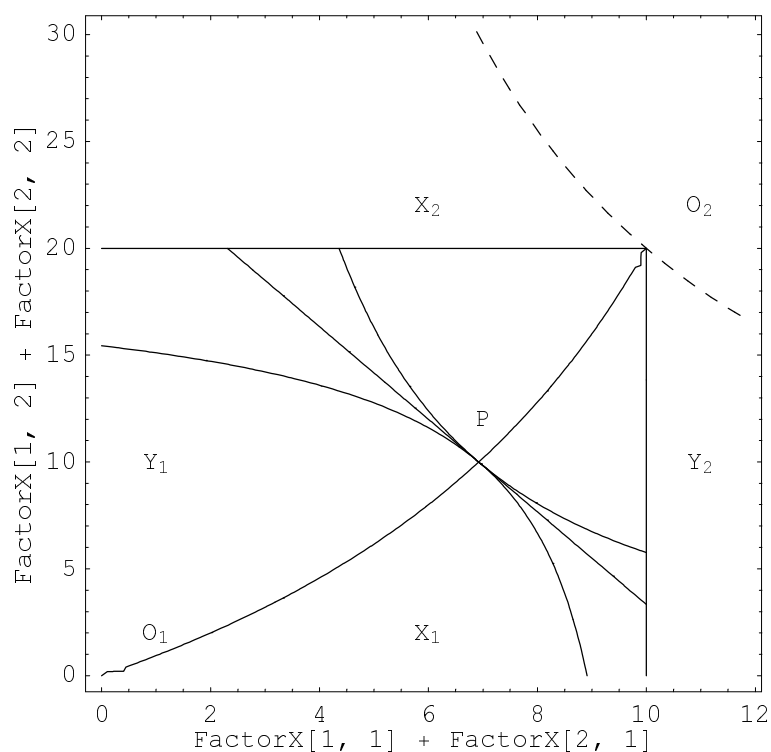
Looking at the existing literature on voting theory, we find that it is grossly inadequate and highly misleading on these key issues. In particular, Arrow's theorem on the 'impossibility of a social welfare mechanism' and Sen's theorem on the 'impossibility of a Paretian liberal' appear to be mathematical constructs that have a different meaning than suggested by these authors and they do not support their conclusions - which are generally adopted in the literature. By discussing these errors and showing where the claims were unwarranted, we have in fact restated what we did in the beginning: namely showing how voting can be a reasonable and desirable process.

Our discussion has also highlighted some areas where more research in Voting Theory would be fruitful. This book has forwarded the new criterion of the fixed point Borda scheme. From its principle and from the experience with these programs and examples, it seems better than the Condorcet scheme of pairwise voting, and its use can be advised with a large degree of confidence. If there is doubt on particular occasions, the provided routines allow one to check, of course. Yet, there is no full mathematical clarity, and research here would be useful. Since voting theory is beriddled with paradoxes and misunderstandings, more certainty here is advisable. Another point is that the Borda ranking requires voters to order all items. Perhaps that is asking too much, and perhaps it would be sufficient to allow voters to rank only the first five, with the rest neglected (if they want to). This could create its own 'paradoxes' at times, but it might work better than asking voters to order all 26 candidates in a U.S. Presidential election. There are, in other words, still practical problems as well.

Appendix: Hicks 1981

It is sometimes suggested that a Social Welfare Function (SWF) would not exist, and would not be derivable from the ordinal utility functions of the individuals. Hicks (1981:170) however gives an illuminating explanation how it can be derived from utility functions that satisfy standard assumptions. It suffices to quote him:

"That collective indifference curves (or hyper-surfaces) (...) can in general be constructed is made occularly evident by that adaptation of the celebrated box diagram which we owe to Kaldor. It is sufficient to take the case of two persons and two commodities. If individual *I* is at position *P*, with respect to axes $O_1 X_1$ and $O_1 Y_1$, while *II* is at position *P* with respect to axes $O_2 X_2$ and $O_2 Y_2$, the superposition of the two *P*'s (with axes reversed) as shown (...), enables us to read off the total quantities at the disposal of the pair by considering the coordinates of the second origin (O_2) with respect to the *I* axes. (...) If we insist that the tangency condition is to be maintained throughout, we can move the *II* curve round on the *I* curve, keeping contact; the locus of O_2 , with respect to the *I* axes, will then be the collective indifference curve. It is evident that it has the same shape as an ordinary indifference curve. And clearly we may extend the same method of compounding to any number of persons and any number of commodities."



Above plot has been generated using the Applied General Equilibrium package (AGE[®]) of The Economics Pack, Cool (2001:144) (which relies on an earlier package by Asahi Noguchi and Silvio Levy). The EdgeworthBowley plot has been extended here with Kaldor's scheme. The resources are $\{x, y\} = \{X_1 + X_2, Y_1 + Y_2\} = \{10, 20\}$, while $P = \{6.9, 10\}$.

Some notes are: (1) You may have noted that this SWF contour depends upon the distribution of income: each different allocation P on the contract curve gives another contour. (2) Once a distribution of income has been chosen, changes along this SWF contour imply that the individuals are aware that the distribution of utility remains the same even though the income distribution in market prices changes. This is only valid if people do not suffer from money illusion, while people in practice suffer from that mildly. (3) If the resources change, then the individuals need no longer be on the contract curve, and it is a difficult issue how they get back to it. This is likely the same question as how the distribution of income came about in the first place - i.e. how P was chosen. The AGE[®] package requires you to enter a fixed SWF that directly implies the distribution. (4) The scheme requires a market situation. For public goods, both individuals would get the same consumption points, and it is not valid to assume that one could get less than the other. The individual indifference

contours also would rather cross than just touch. The allocation is namely determined by voting and Public Choice complications (like free riders and usurpating bureaucrats). The scheme hence does not apply. The claim that a SWF could not be derived hence would be limited to the case of public goods. This is a much milder claim than it might have been originally thought.

Note that we could complicate the issue. Frequently, the distribution of income is affected by voting and political decision. Also, voters indirectly affect macro-economic and monetary policy, which interacts with mild forms of money illusion. We can complicate the issue by all kinds of angles. Doing this would however distract. It is best to settle the claim on the existence of the SWF within the realm where the idea has been formulated. In this realm we can reduce the question to public goods and thus the national budget. It is another question how we continue with practical questions in real life. Here all kinds of complexities arise and have to be dealt with pragmatically. There certainly is an impact of voting on the distribution of income, for example, but we should not confuse that question with the fundamental question whether the SWF can be derived, in acceptable fashion, from individual preferences or not.

Literature

EWP references are to the Economics Working Papers Archive at the Washington University at St. Louis: <http://econwpa.wustl.edu>. See also <http://www.dataweb.nl/~cool>.

Ancot & Hughes Hallet (1984), "Establishing public preferences for coalition government: an empirical study in economic planning behavior", *De Economist*, p153-170

Aronson, E. (1972, 1992a), "The social animal", Freeman, sixth edition 1992

Aronson, E. (ed) (1973, 1992b), *Readings about The Social Animal*, Freeman, sixth edition 1992

Arrow, K. & T. Scitovsky eds. (1969), "Readings in welfare economics", Allen & Unwin

Arrow, K. & F. Hahn (1971), "General competitive analysis", North Holland

Arrow, K. (1950), "A difficulty in the concept of social welfare", *J. of political economy*, pp328-346, reprinted in Arrow & Scitovsky (1969) pp147-168

Arrow, K. (1951, 1963), "Social choice and individual values", J. Wiley

Barrow, J.D. (1998), "Impossibility. The limits of science and the science of limits", Vintage

Bernstein (1992), "Capital ideas", The Free Press

Bernstein (1996), "Against the gods", Wiley

Blanchard, O. & S. Fischer (1989), "Lectures on macroeconomics", MIT

Bochenski (1956, 1970), "A history of formal logic", 2nd edition, Chelsea

Csikszentmihalyi (1997), "Finding flow", Basic Books

Cool (1990c), "Why a social welfare (meta) function *does* exit: The Arrow Impossibility Theorem for Social Choice resolved, A better analysis suggested," internal note Central Planning Bureau, The Hague, reprinted in Cool (1992b) pp57-104

Cool (1992a), "On the reasonableness and the moral desirability of the conditions of Arrow's Theorem", published in Cool (1992b) pp273-285

Cool (1992b), "Definition and Reality in the general theory of political economy; Some background papers 1989-1992", ISBN 905518-207-9, Magnana Mu Publishing and Research, Rotterdam

Cool (1996b), "A constitutional amendment for an Economic Supreme Court", ewp-get/9604003

Cool (1997b) "The solution to Arrow's difficulty in social welfare", ewp-get/9707001

Cool (1999, 2001), "The Economics Pack, Applications for *Mathematica*", Scheveningen, JEL-99-0820, ISBN 90-804774-1-9

Cool (1999a), "Proper definitions for risk and uncertainty", EconWPA ewp-get/9902002

Cool (2000), "Definition and Reality in the General Theory of Political Economy", Samuel van Houten Genootschap, ISBN 90-802263-2-7, JEL 2000-1325

- Cool (2000a), "Aspects of testing (matching, ranking and rating)", ewp-get/0004004
- Cool (2000b), "The choice on sustainability: information or the meta-SWF approach to a shift of preferences", ewp-othr/0004004
- Cool (2000c), "The seminal contribution of Roefie Hueting to economic science: Theory and measurement of Sustainable National Income", November 16 2000, see the internet
- Cullis, J. & P. Jones (1992), "Public finance & public choice", McGraw-Hill
- Dahl & Lindblom (1976), "Politics, economics and welfare", Chicago
- Dahl, R (1956), "A preface to democratic theory", Chicago
- Dasgupta, A.K. & D.W. Pearce (1980), "Cost-benefit analysis", Macmillan
- DeLong, H. (1971), "A profile of mathematical logic", Addison Wesley
- Doel, J. van den, and B.C.J. van Velthoven (1990), "Democratie en welvaartstheorie", Samson
- Dopp (1969), "Formale Logik", Benziger Verlag, Einsiedeln
- Eatwell c.s. (1998), "The New Palgrave", Macmillan (entries risk, risk aversion, Arrow' Theorem, Social Welfare)
- Elo, A.E. (1978), "The Rating of Chess Players, Past and Present", Arco Publishing, Inc., New York.
- Ferguson, Th. (1967), "Mathematical statistics", Academic Press
- Freeman, J. (1994), "Simulating neural networks with Mathematica", Addison-Wesley
- Good, A. & F. Graybill (1950, 1963), "Introduction to the theory of statistics", McGraw-Hill
- Hart (1961, 1997), "The concept of law", Oxford
- Hicks (1981), "Collected essays on economic theory, volume I; Wealth and welfare", Blackwell
- Hornby (1985), "Oxford advanced learner's dictionary of current English", Oxford
- Huang, C. & P. Crooke (1997), "Mathematics and Mathematica for economists", Blackwell
- Johansen (1969), "An examination of the relevance of Kenneth Arrow's general possibility theorem for economic planning", Economics of planning 9, pp5-41
- Jorgenson, D. (1990), "Aggregate consumer behavior and the measurement of social welfare", Econometrica 58 no 5 pp1007-1040
- Luce & Raiffa (1957), "Games & decisions", J. Wiley
- Luenberger, D. (1998), "Investment science", Oxford
- Maeder, R. (1991), "Programming in Mathematica", Addison-Wesley
- Mas-colell, A., M. Whinston and J. Green (1995), "Microeconomic theory", Oxford
- Mueller (1989), "Public Choice II", Cambridge
- Noguchi, Asahi (1993), "General Equilibrium models", in Varian (ed), "Economic and financial modeling with Mathematica", Springer Telos 1993 pp 104-123

- Pikkemaat, A. (1990), "Arrow's significance for economic science in the Netherlands", *De Economist* 138 No 3 p235-255
- Rafiei, D. & A. Mendelzon (2000), "What is this Page Known for? Computing Web Page Reputations", <http://www9.org/w9cdrom/index.html>
- Rijken van Olst, H. (1969), "Inleiding tot de statistiek", Part I-II-III, Van Gorcum
- Saari, D. (internet), "The symmetry and complexity of elections" (S&C)
- Saari, D. (internet), "Explaining all three alternative voting outcomes" (E3)
- Saari, D. (internet), "Connecting and resolving Sen's and Arrow's theorems" (C&R)
- Schrijver (1987), "Analyse van macro-economische beleidspreferenties met behulp van optimale besturingstheorie", Research memorandum 35, CPB
- Sen, A. (1970), "Collective choice and social welfare", North Holland
- Sen, A. (1986), "Social choice theory," p1073-1181 in Arrow & Intrilligator eds. (1986), "Handbook of mathematical economics, Volume III", North Holland
- Sen, A. (1999a), "Development as freedom", Knopf
- Sen, A. (1999b), "The possibility of social choice", *AER* p349-366 (excluding literature)
- Shone, R. (1998), "Economic dynamics", Cambridge (uses *Mathematica*)
- Sudgen, R. (1999), "Book review: On voting. A public choice approach. By Gordon Tullock", *JEL* 1715-1716 (December)
- Swank, O. (1989). "The objectives of Democrats and Republicans In the United States, 1956—1984", Rotterdam
- Takayama, A. (1974), "Mathematical economics", The Dryden Press
- The Economist (2000),
- The New Palgrave, Macmillan 1988 - see Eatwell
- Theil, H. (1971), "Principles of econometrics", John Wiley
- Tinbergen, J. (1956), "Economic policy: principles and design", North-Holland
- Tinbergen, J. (1985), "Production, income and welfare: the search for an optimal social order", Weatsheaf books, Harvester press
- Tobin, J. (1990), "On the theory of macroeconomic policy", Tinbergen Lecture, *De economist* 138 pp1-14
- Varian, H. (ed) (1993), "Economic and financial modeling with Mathematica", Springer Telos
- Varian, H. (ed) (1996), "Computational economics and finance", Springer Telos
- Wilson, J.Q. (1993), "The moral sense", The Free Press
- Wolfram, S. (1992), "Mathematica", Addison-Wesley
- Wolfram, S. (1996), "Mathematica 3.0", Cambridge

Routine Definition Index

A

AllaisParadox, 286
 Allowed, 72
 AllowedQ, 72
 Approval, 108
 ArrowPratt, 278
 AverageMatchPr, 237

B

BalancedPareto, 189
 Bayes, 253
 BinomialPass, 210
 Borda, 99
 BordaAnalysis, 99
 BordaComplement, 104
 BordaField, 99
 BordaFP, 102
 BordaFPLocalSet, 104
 BordaFPQ, 104
 BordaXvsXCom, 104
 Bordered2DPrSolve, 254
 Bordered2DPr –
 ToConditionals, 254
 Bordered2DPrToEquations,
 254

C

CertaintyEq, 275 – 281
 ConditionalPrForm, 252
 Constitutions, 146
 CreateIR, 221
 CreateNames, 55
 CreateProspect, 257

D

DefaultItems, 54
 DefineFast, 64

DeonticAxiom, 74
 Draw, 261
 DuncanBlackPlot, 84

E

Economics, 48
 EconomicsPack, 48
 EqualVotes, 57
 ExamplePrefs, 64

F

Freedom, 72
 FreedomQ, 72
 FromOdds, 203
 FromProb, 252
 FromVoteMargin, 120

G

GraphToPref, 135
 GroupOrderQ, 121

I

IfNegativeToMaxRule, 714
 IfNegativeToMinRule, 271
 InvertGraph, 135
 IRToMatch, 206
 IRToMatchPr, 205
 IRToMatchValue, 206
 Items, 54

J

Jealous, 188

L

ListToGraph, 140
 ListToOrderedPairs, 122
 ListToPref, 66

ListToPrefOrderQ, 67
 ListToVoteMargin, 83
 ListToVoteMatrix, 79
 Logistic, 215
 Logit, 214

M

Majority, 149
 MatchPrToRating, 237
 MatchToPr, 206
 MoralConclude, 74
 MoralSelect, 74

N

NOrderPair, 122
 NotAllowed, 72
 NotAllowedQ, 72
 NotOught, 73
 NPairs, 121
 NPrefOrderQ, 122
 NumberOfItems, 54
 NumberOfVoters, 57

O

Odds, 203
 Ought, 72
 OughtQ, 72

P

Pairs, 121
 PairsToPaths, 122
 Pairwise, 121
 PairwiseField, 121
 PairwiseMajority, 114
 PairwiseToMatrix, 82
 PairwiseToVoteMargin, 82
 PairwiseVoter, 121

Pareto, 94
 ParetoApproval, 163
 ParetoBorda, 152
 ParetoMajority, 152
 ParetoPairwise, 158
 ParetoPlurality, 161
 Plurality, 96
 PPath, 122
 Pr, 252
 PrDistance, 219
 Pref, 59
 Preferences, 63
 PreferencesPlot, 84
 PrefOrderQ, 67
 PrefToList, 66
 PrIR, 227
 PrIRFromGrowth, 226
 Prob, 252
 ProperPrefsQ, 62
 Proportional, 187
 Prospect, 256, 266, 250
 Prospect3DPrTriangle, 273
 ProspectApply, 258
 ProspectEV, 256
 ProspectInnerSort, 258
 ProspectInverse, 272
 ProspectPlot, 273
 ProspectQ, 256
 ProspectReList, 258
 ProspectUtility, 257
 PutIn, 259

R

RandomGet, 187
 RandomIR, 201
 RandomWant, 187
 RatingP, 203
 RatingQ, 203
 RelativeJealous, 189
 ResetAll, 48
 Risk, 257
 RiskEquations, 271

Risket, 271
 RiskModel, 271
 RiskPr, 257
 RiskyQ, 257

S

SatisfiedQ, 189
 SelectPreferences, 66
 SequentialIR, 233
 SetDeontic, 71
 SetPreferences, 63
 SetRandomPreferences, 64
 SetRandomVoteMargin, 83
 ShowListGraph, 140
 ShowPrefGraph, 137
 SortIR, 202
 StatusQuo, 54
 StrategicPref, 173
 StrictRisingPrefsQ, 62

T

TakeOut, 259
 TakePref, 66
 ToAllowed, 70
 ToApproval, 108
 ToExpectedUtility, 257
 ToFreedom, 70
 ToNotAllowed, 70
 ToOught, 70
 ToPref, 60
 ToProb, 252
 ToProspect, 271
 ToRisket, 271
 ToUtility, 257
 ToVoteMargin, 120
 ToVoteMatrix, 81

U

Universe, 72
 UseFindCycle, 140

V

VoteMargin, 80
 VoteMarginPlot, 84
 VoteMarginToBinary, 115
 VoteMarginToCount, 115
 VoteMarginToGraph, 138
 VoteMarginToOrderQ, 83
 VoteMarginToPref, 81
 VoteMatrix, 79, 120
 VoteMatrixToPref, 79
 Votes, 57
 VoteToPref, 276

W

WinPr, 223