NICTA

PRICAI 2014 Tutorial Program: Computational Social Choice

Haris Aziz and Nicholas Mattei
NICTA and UNSW

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

Schedule for the Day

Time	Plan
8:30 – 8:45	Intro. to Computational Social Choice
8:45-10:00	Preferences and Voting
10:00 – 10:30	Coffee Break
10:30 – 12:00	Matching, Resource Allocation, and Fair Division
12:00 – 1:30	Lunch
1:30 – 2:15	Advanced Topics in Preference Aggregation
2:15 – 2:45	Matching, Resource Allocation, and Fair Division II
2:45 – 3:00	Closing Remarks and Survey

Broad Overview

- Part 1: Overview of ComSoc and Related Areas
- Part 2: A Primer on Preferences
  - Formalisms and Languages
  - Constraints v. Preferences
- Part 3: Voting
  - Classic Paradoxes and Results
  - Computational Aspects of Voting
  - Game Theoretic Aspects of Voting

Social Choice

- Given a collection of agents with preferences over a set of things (houses, cakes, meals, plans, etc.) we must…
  - 1. Pick one or more of them as winner(s) for the entire group

    OR....
  - 2. Assign the items to each of the agents in the group.

Subject to a number of exogenous goals, axioms, metrics, and/or constraints.

2 Threads of ComSoc

Analyze Results

Analyze computational aspects of Social Choice. Many classic results in Social Choice Theory ignore the computational aspects of the theory.

↔

Import Ideas to AI

Implement ideas from Social Choice Theory in designing, implementing, and deploying systems across computer science including AI and multi-agent systems.

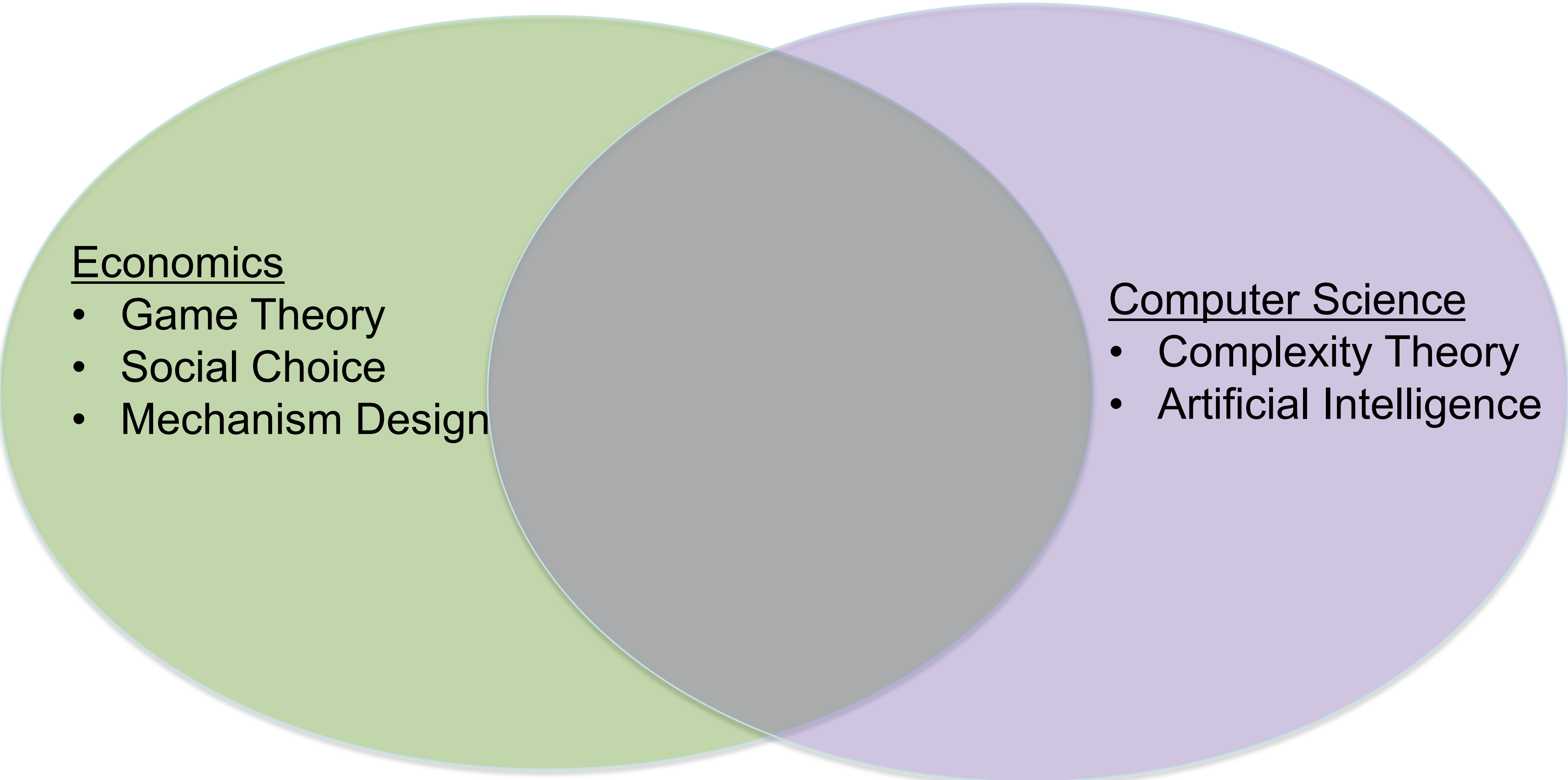
AGT and ComSoc

Economics

- Game Theory
- Social Choice
- Mechanism Design

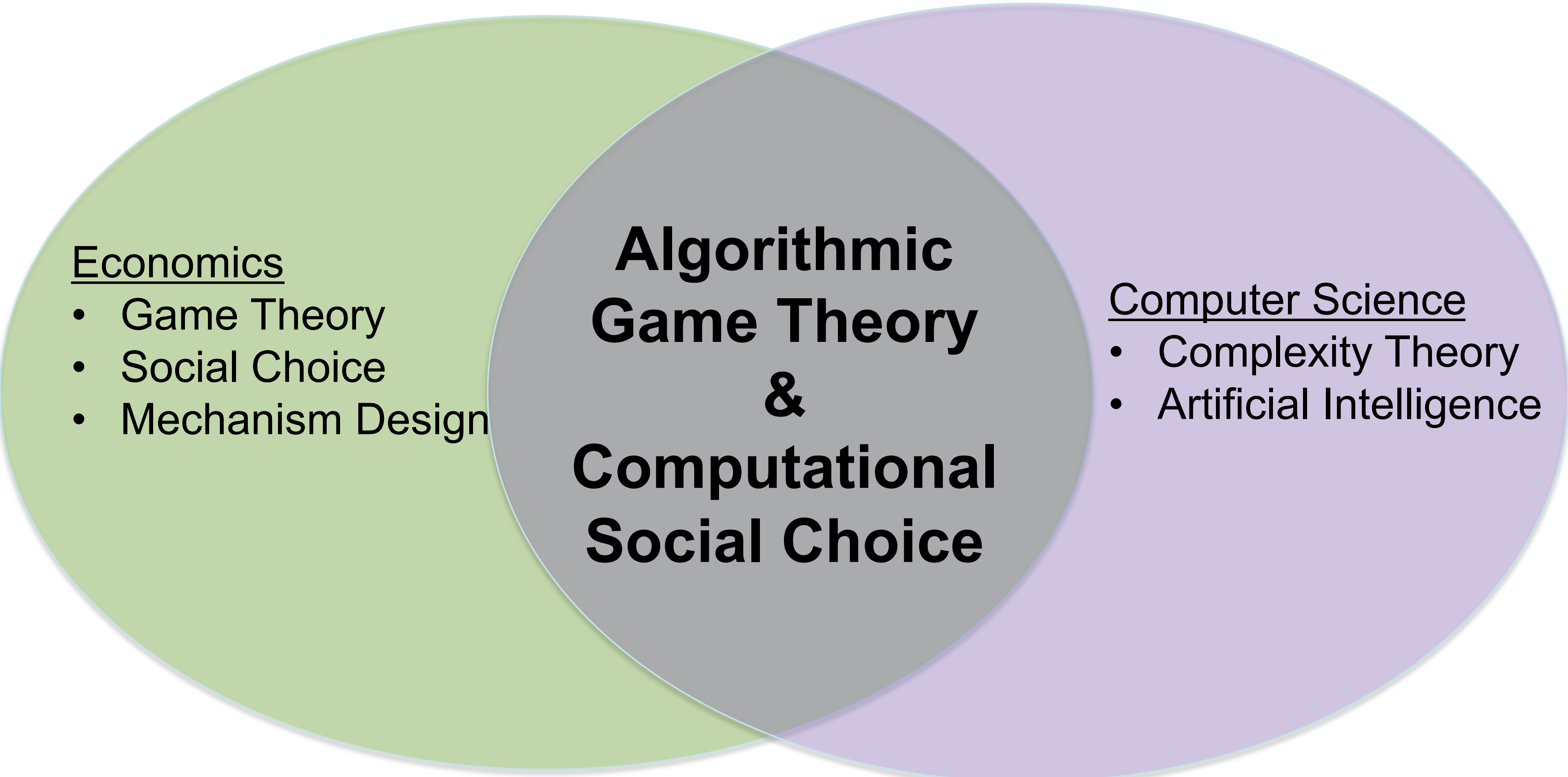
Overview Article: Vincent Conitzer. Making Decisions Based on the Preferences of Multiple Agents. Communications of the ACM (CACM), 2010

AGT and ComSoc



Overview Article: Vincent Conitzer. Making Decisions Based on the Preferences of Multiple Agents. Communications of the ACM (CACM), 2010

AGT and ComSoc



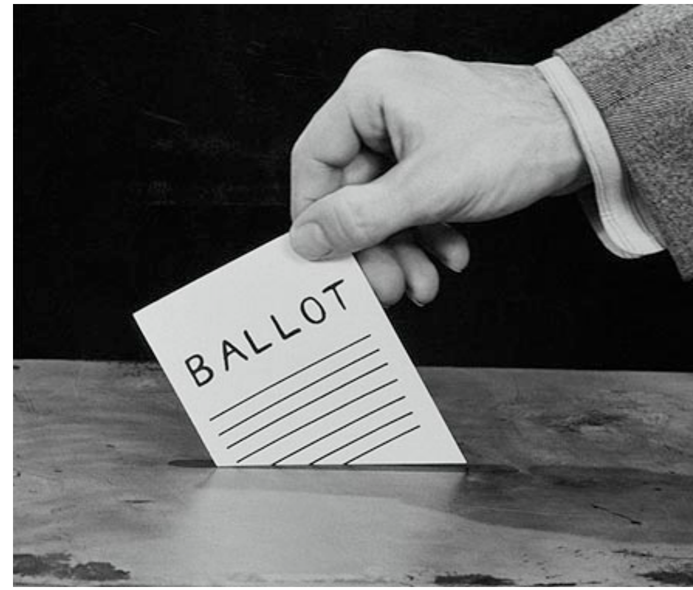
Overview Article: Vincent Conitzer. Making Decisions Based on the Preferences of Multiple Agents. Communications of the ACM (CACM), 2010

Group Decisions

- Problems arise when groups of agents (humans and/or computers) need to make a collective decision.
- How do we aggregate individual (possibly conflicting) preferences and constraints into a collective decision?

Voting and Ranking Systems

- Voting has been used for thousands of years - many different elections systems which have been devloped.
  - Used to select one or more alternatives that a group must share.



- Ranking systems are the social choice setting where the set of agents and the set of choices is the same.

Markets and Mechanisms

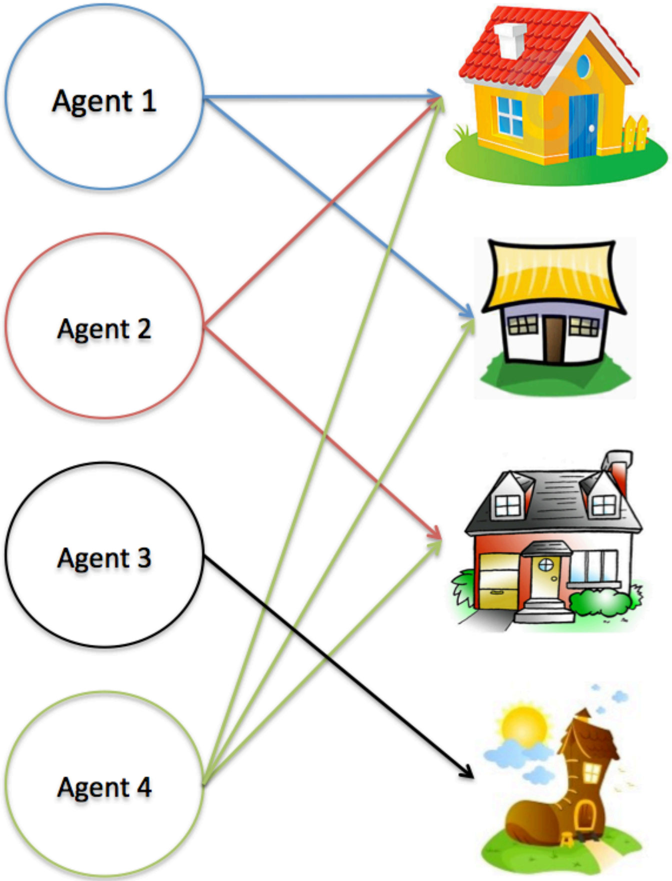
- Bidding, Auctions and Markets are other mechanisms used to aggregate the preferences of a collection of agents for an item or sets of items.

- All these mechanisms usually require a central agent to collect the bids, announce a winner, collect the final price and in some cases, return value to the losing agents.

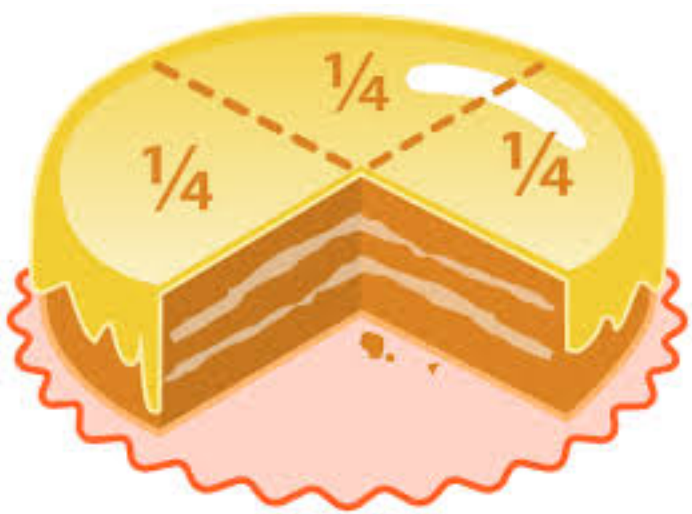
Nicholas Mattei
NICTA and UNSW

Matching and Assignment

- Assign items from a finite set to the members of another set.
  - Useful in many applications including allocating seats in schools, kidneys for transplant, runways to airplanes.
- Many axes to consider.
  - Divisible v. Indivisible Goods
  - Centralized v. Decentralized
  - Deterministic v. Random
  - Efficiency v. Fairness



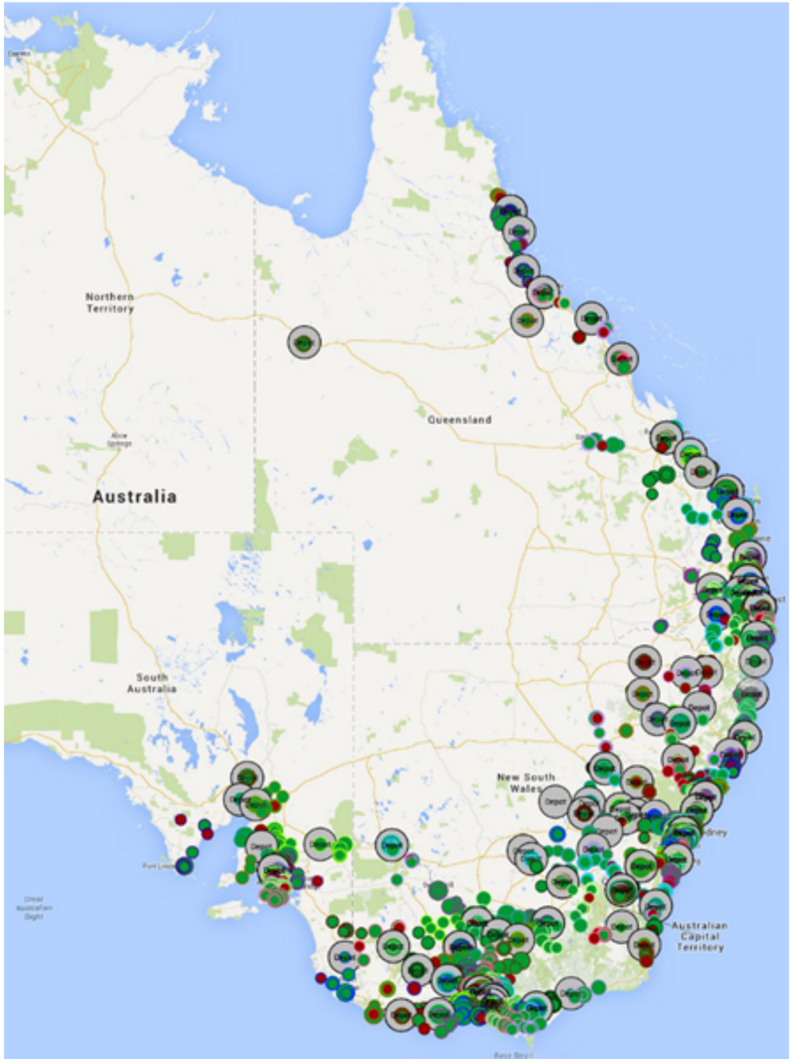
Resource Allocation and Fair Division



- Given a divisible, heterogeneous resource (such as a cake) how do we divide it among agents who may have different constraints, preferences, or complementarities over the portions?
  - Use to allocate land, spectra, water access…
- Many similar considerations:
  - Proportionality, fairness, no disposal, no crumbs…

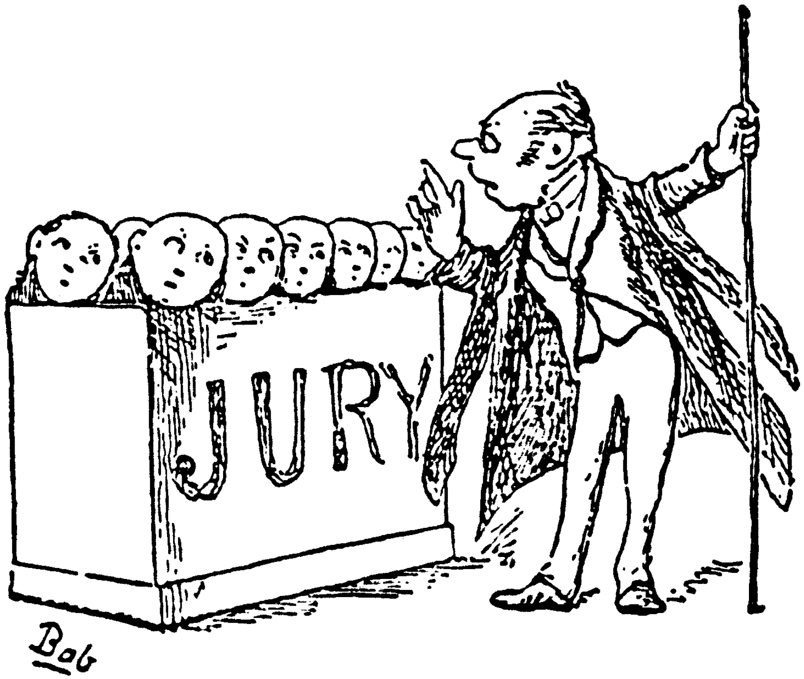
Coalition Formation

- Agents form teams or groups which improve utility.
  - How and when will these groups form?
  - How do we allocate costs or revenues for these groups?
  - How stable are these groups?
- Part of cooperative game theory and studied in many areas.



Judgment Aggregation and Belief Merging

- Judgment Aggregation: Groups may need to aggregate judgments on interconnected propositions into a collective judgment.
- Belief Merging: Groups may need to merging a set of individual beliefs or observations into a collective one.
  - Extensively studied in logics and other areas.



Why?

- For collecting and ranking search results, movies, pizzas, etc...
- For selecting leaders in distributed network structures.
- To find optimal allocations of resources.
- To coordinate and control distributed systems.
- To make group judgments, decisions, views of reality…

Broad Overview

- Part 1: Overview of ComSoc and Related Areas
- Part 2: A Primer on Preferences
  - Constraints v. Preferences
  - Formalisms and Languages
- Part 3: Voting
  - Classic Paradoxes and Results
  - Computational Aspects of Voting
  - Game Theoretic Aspects of Voting

Preferences v. Constraints

- In common usage we often conflate constraints and preferences.

- A *constraint* is a requirement.
  - The car must be blue.
  - I cannot eat peanuts.

- A *preference* is a soft ("nicer") constraint.
  - I *prefer* pizza to pasta.
  - I want a red car.

So What Are Constraints?

- A constraint is a requirement. A set of constraints limits the feasible space to a set of points such that all constraints are satisfied.
- Basic Computational Paradigm:
  – Given a set of Variables $\{X_1 \ldots X_n\}$ and their domains $\{D_1 \ldots D_n\}$.
  – Given a set of Constraints $C(X_1, X_2)$ is a relation over D_1 X D_2.
  – Find an assignment to $\{X_1 \ldots X_n\}$ that is consistent.
- Common in many applications:
  – Scheduling, time-tabling, routing, manufacturing…

So What Are Preferences?

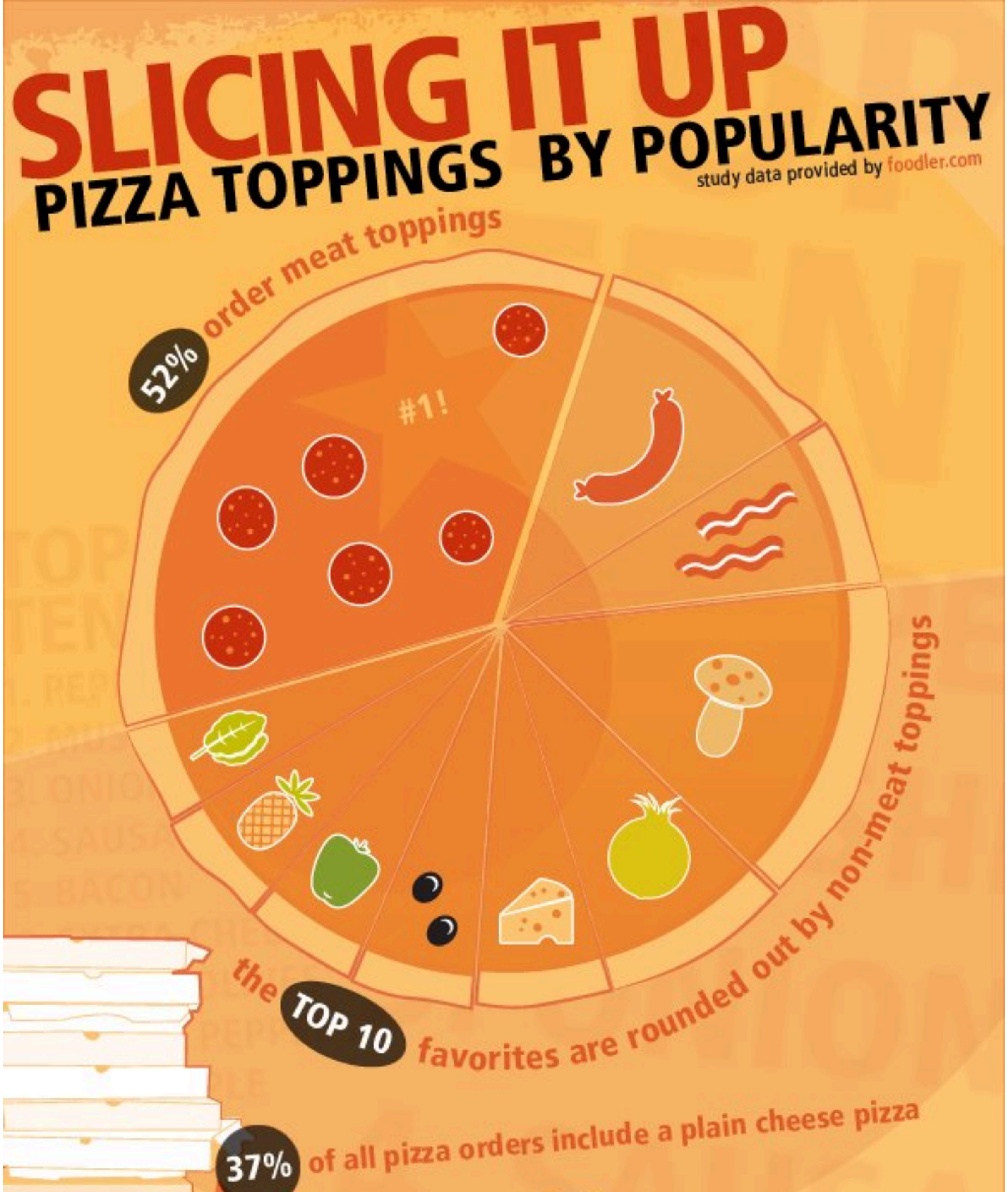
- Preferences kick in when we have an under constrained problem that admits many solutions.
- Often we have constraints mixed in with preferences:
  - The new cars for the fleet have to meet EPA standards of 100km/l fuel consumption, but for color the boss prefers yellow to green.
- Preferences are flexible and represent which of two alternative assignment are more acceptable.

So What Are Preferences?

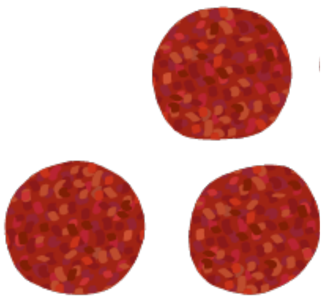
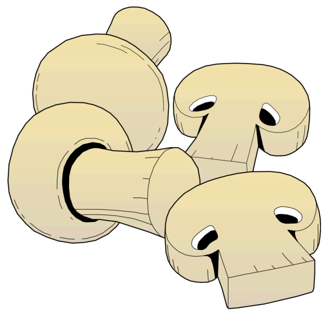
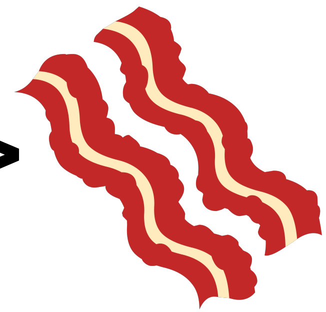
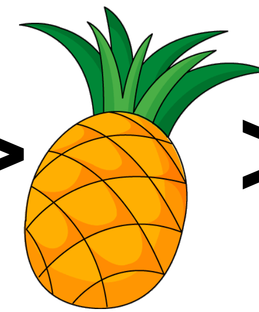
- Positive
  - I like peperoni on my pizza.
- Negative
  - I don’t like anchovies.
- Unconditional
  - I prefer extra cheese on my pizza.
- Conditional
  - If we have two pizzas, I prefer a sausage and a bacon pizza, otherwise I prefer an extra cheese pizza.
- Quantitative v. Qualitative
  - My preference is 0.4 for sausage and 0.5 for bacon.
  - Sausage pizzas are better than bacon pizzas.

Preferences, Everywhere…

- Preferences appear in many formalisms:
  - Soft constraints.
  - Weighted CSP problems.
  - Problems with objective values.
  - Weighted logics.
  - Possibilistic logics.
- Preferences are usually formalized a relation over the domain of the variables of a constraint problem.
  - Given a set of Variables $\{X_1 \dots X_n\}$ and their domains $\{D_1 \dots D_n\}$.
  - A preference is a relationship over the elements of $D_{i.}$
    - However there are more complicated definitions too!

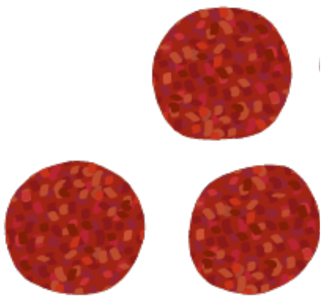
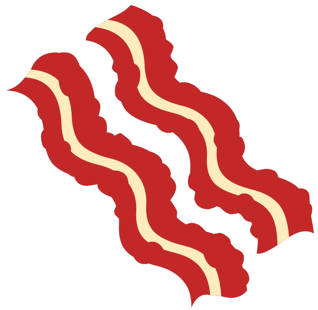
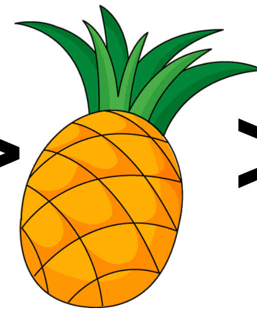

SLICING IT UP
PIZZA TOPPINGS BY POPULARITY
study data provided by foodler.com
52% order meat toppings
#1!
the TOP 10 favorites are rounded out by non-meat toppings
37% of all pizza orders include a plain cheese pizza

Complete Strict Orders

 >  >  >  > 

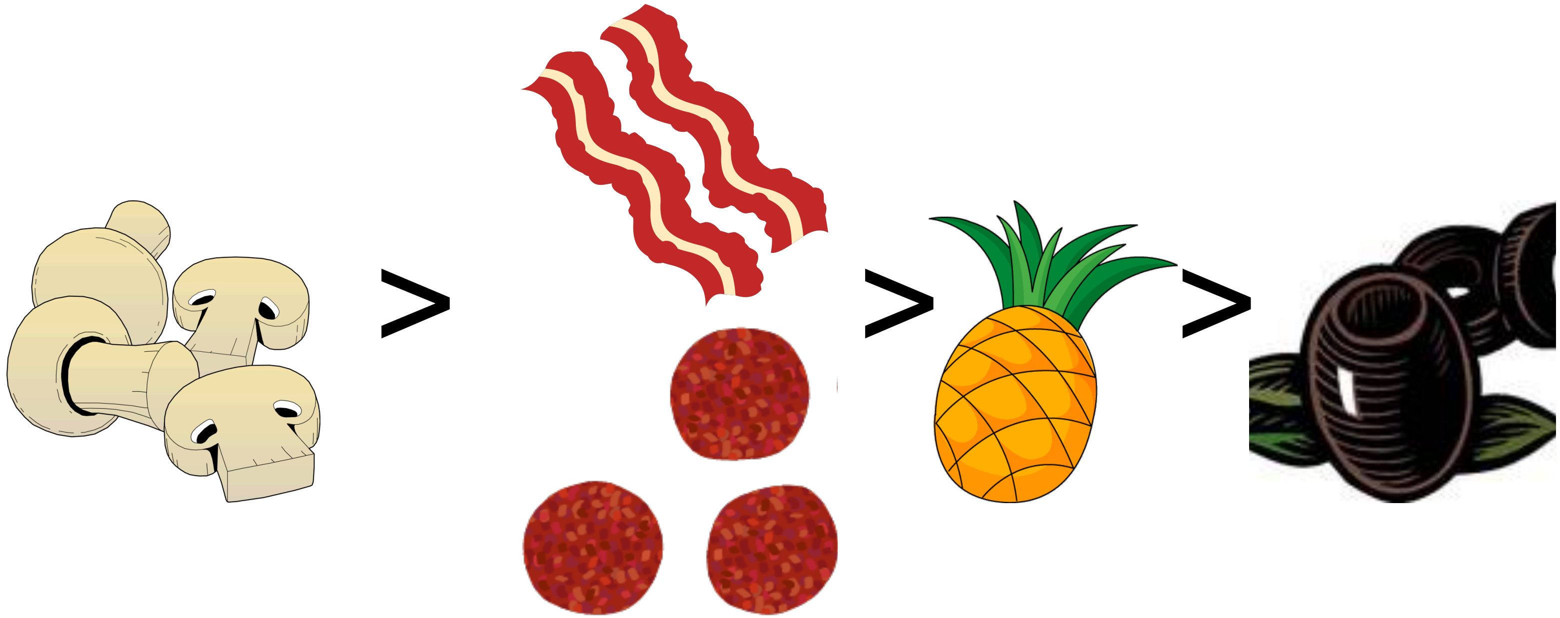
- Every item appears once in the preference list.
- All pairwise relations are complete, strict, and transitive.

Incomplete Strict Orders

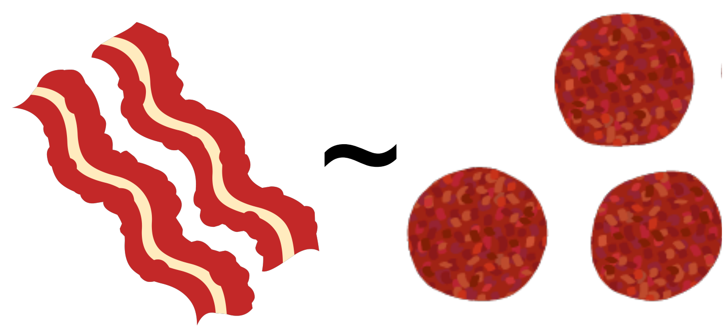
 >  >  > 

- Not every item appears once in the preference list.
- The pairwise relations that ***are*** present are complete, strict, and transitive.

Complete Orders with Indifference

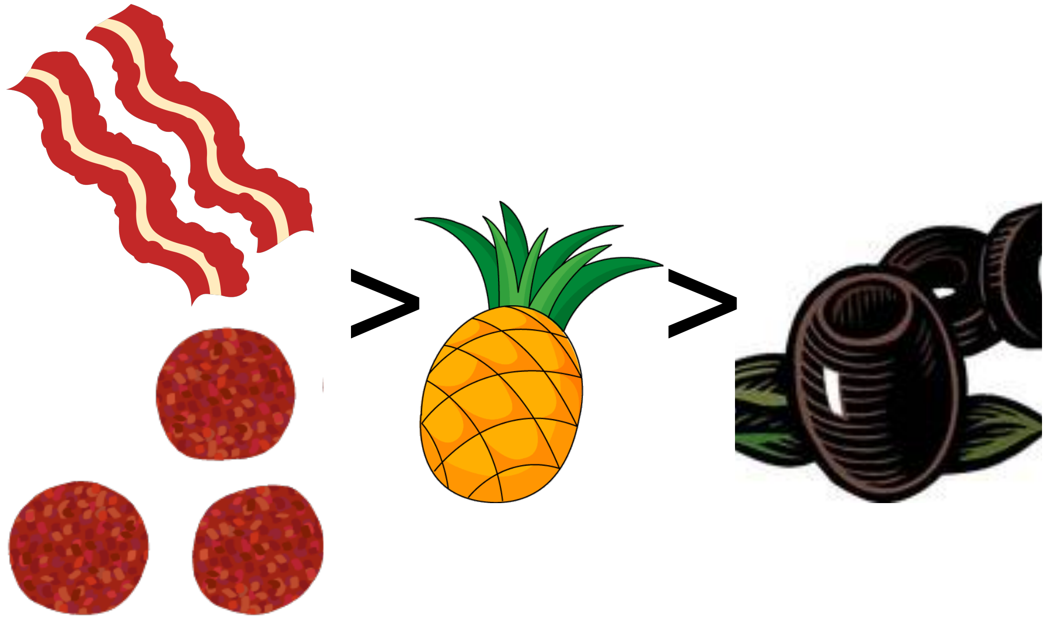


- Every item appears once in the preference list.
- Pairwise ***ties*** are present.
- We denote indifference with the ~ operator.



Nicholas Mattei
NICTA and UNSW

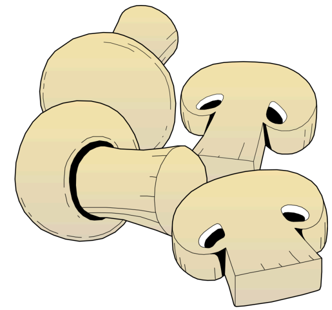
Incomplete Orders with Indifference



- Not every item appears in the preference list.
- Pairwise ties are present.

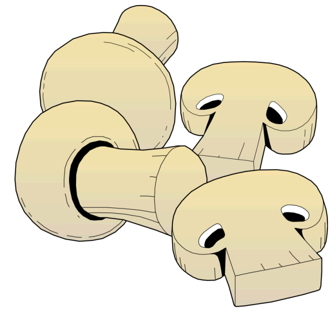
Complex Topping Options

Veg

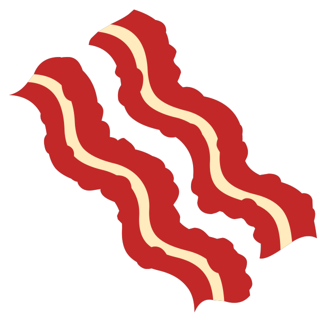


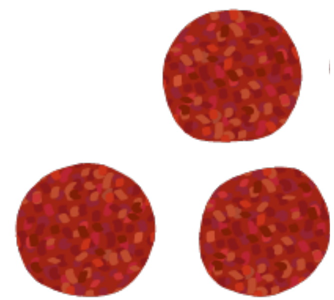
Complex Topping Options

Veg



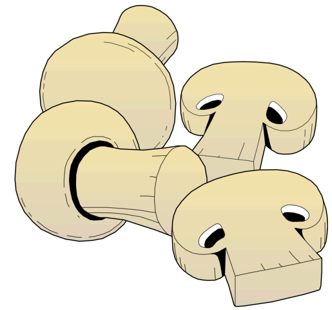
Meat



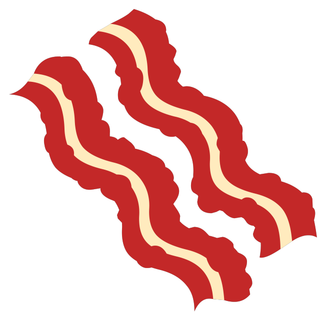


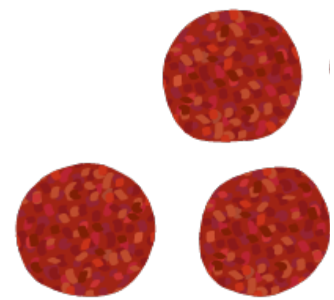
Complex Topping Options

Veg

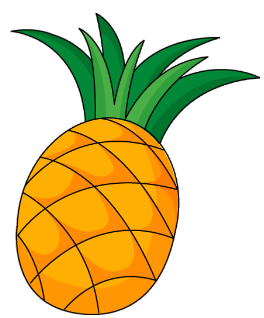


Meat

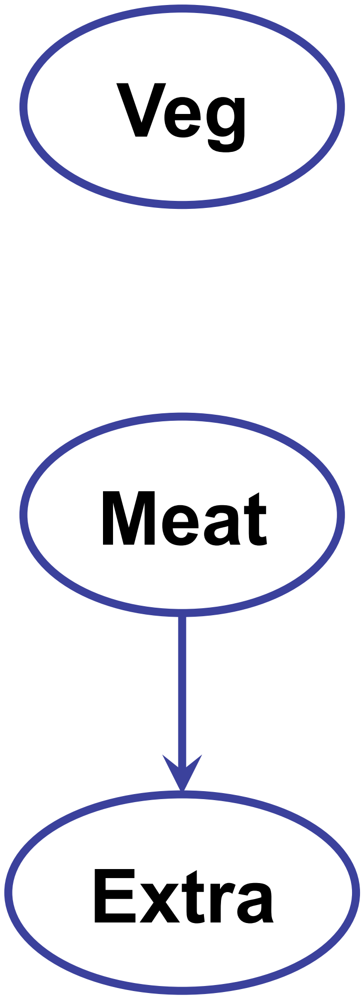


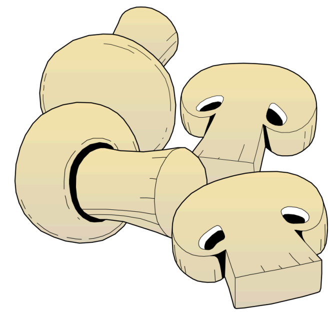


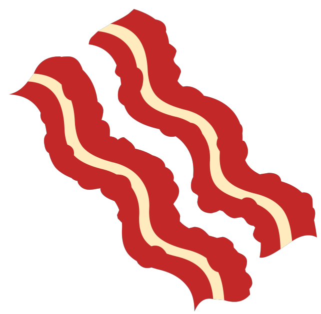
Extra

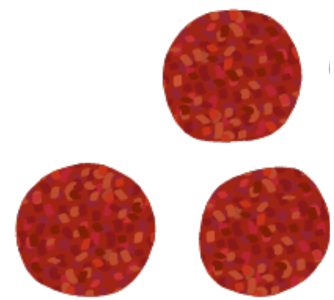


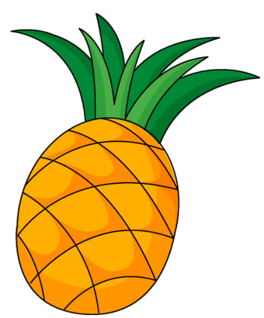
Complex Topping Options











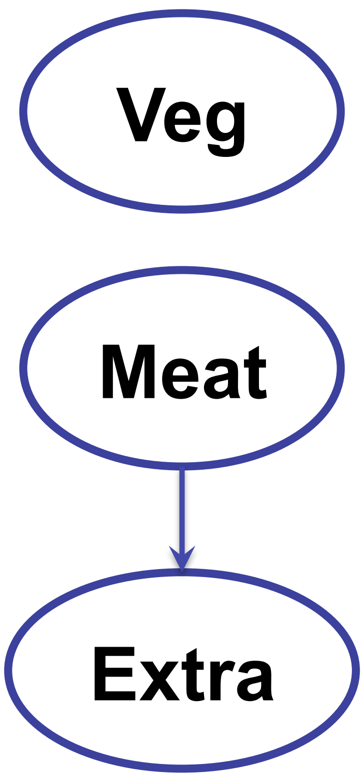
CP-Nets

- CP-nets are a graphical model for representing conditional preference relations – sets of cp-statements.
  - "All else being equal, I prefer pineapple to olives if we have bacon pizza."

- Formally we have:
  - A set of **issues** or **variables** $F = \{X_1, \ldots, X_n\}$ each with finite domain $D_1, \ldots D_n$.
  - A (empty) set of **parents** for each issue $Pa(X_i)$.
  - A preference order over each **complete assignment** to the parents for each issue.

CP-Nets

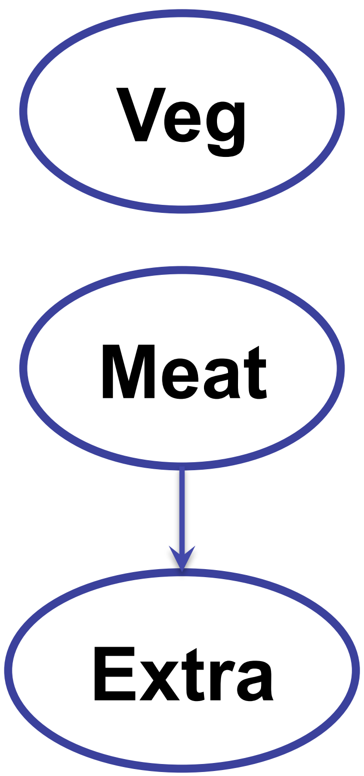
Veg	Spinach > Mushroom
Meat	Pepperoni > Bacon
Extra	Pepperoni: Olives > Pineapple Bacon: Pineapple > Olives

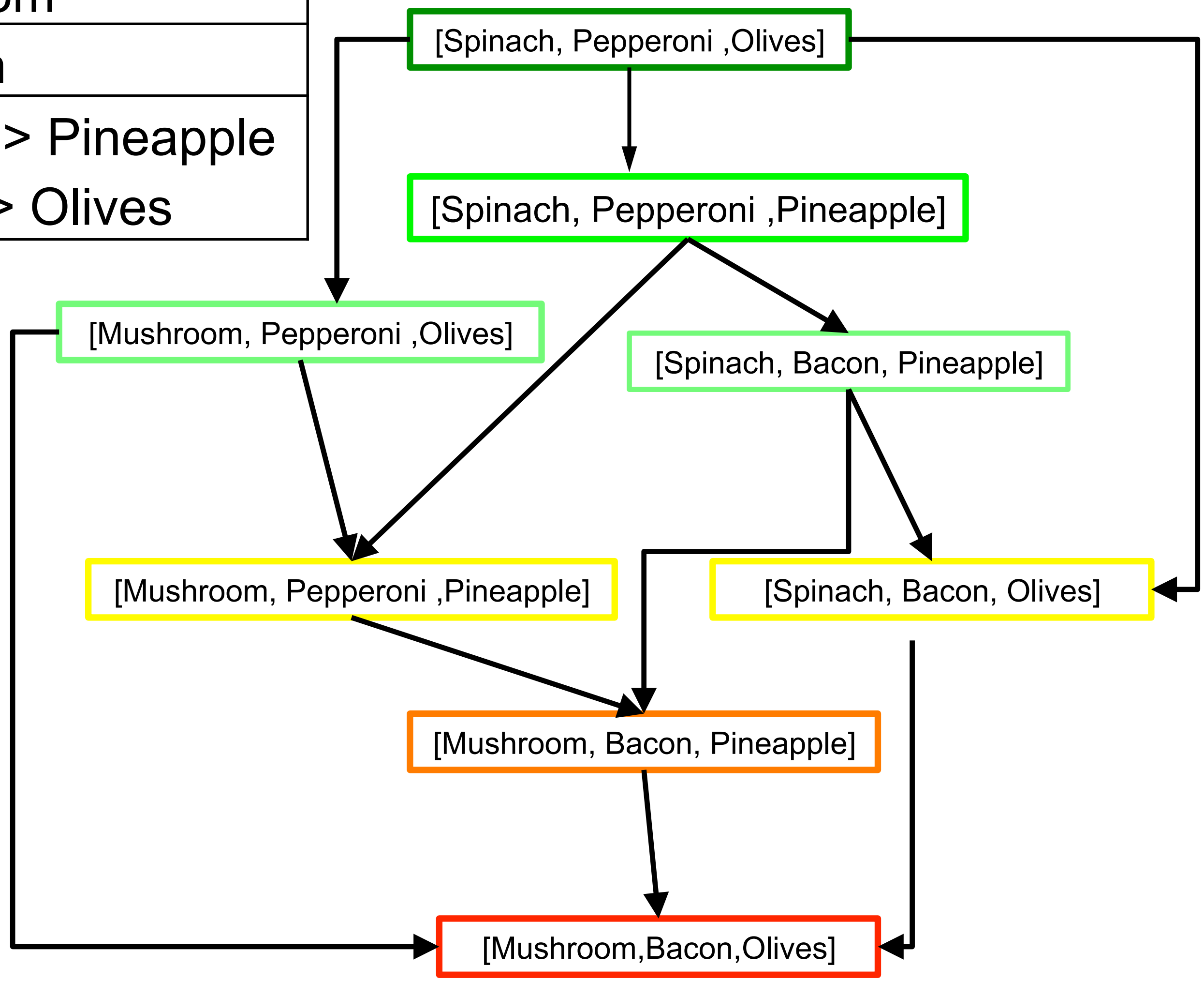


- **Acyclic** CP-nets have no cycles in the dependency graph.
- A CP-net is **compact** if the number of parents is bounded.

CP-Nets

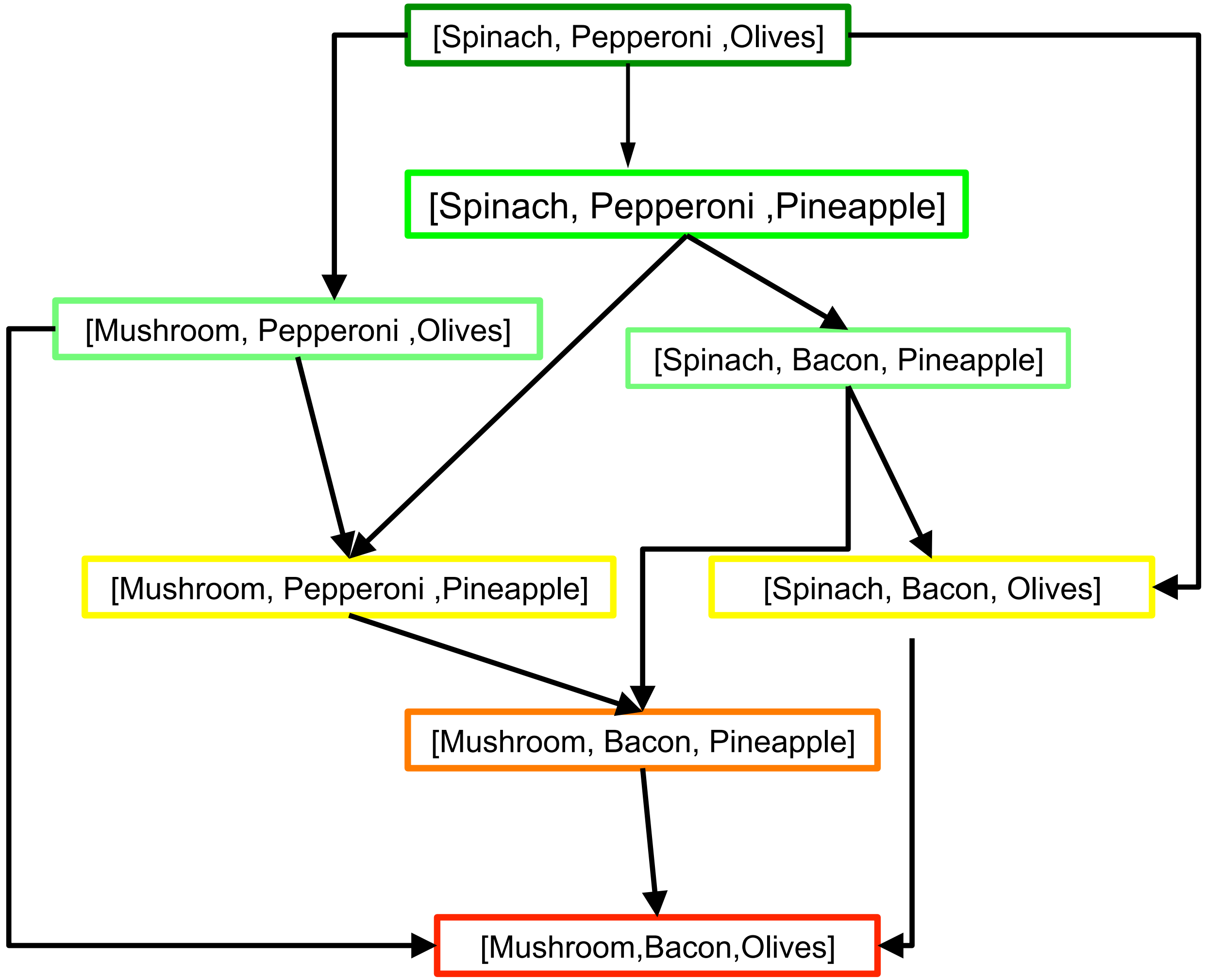
Veg	Spinach > Mushroom
Meat	Pepperoni > Bacon
Extra	Pepperoni: Olives > Pineapple Bacon: Pineapple > Olives





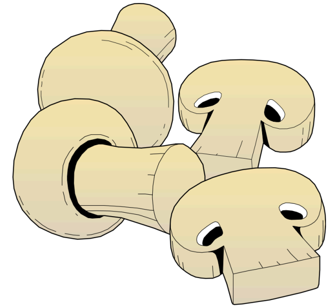
CP-Nets

- A **worsening flip** is change in the value of a variable to a less preferred value.
- One outcome is preferred to another if there is a chain of worsening flips between them.

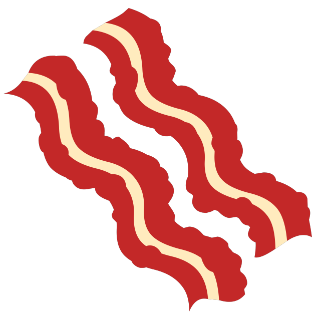


Numerical Preferences (Utility)

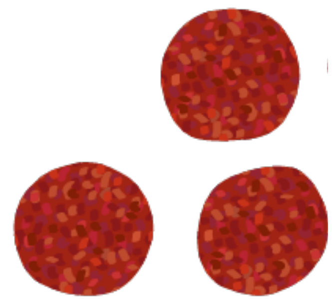
= 5.0



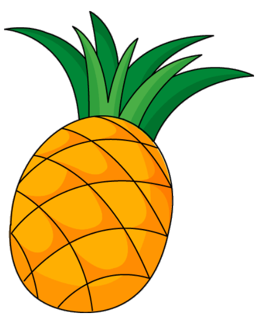
= 0.1



= 0.001



= 0.0



= 10.0

= 22.5764

Nicholas Mattei
NICTA and UNSW

Numerical Preferences (Utility)

- Utilities can indicate a degree of preference for an object
- Can be from a ranked list of options
  - 1 to 5 stars for movies.
  - +1 and -1 for Like and Dislike.
- Decreases complexity often – but also decreases expressiveness.
- Many issues with combining utilities, scaling, formatting etc. which Haris will touch on later!

PrefLib: A Library for Preferences

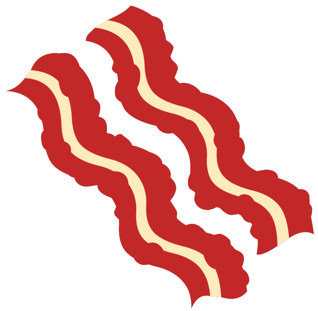
- Many research communities have libraries, datasets, and tool chains that are standard and widely used.
- Preference handling and computational social choice have largely centered around theoretical results.
- We have collected datasets and tools to establish PrefLib, a library of preference data, as a service to the wider community.

Not Uncommon….

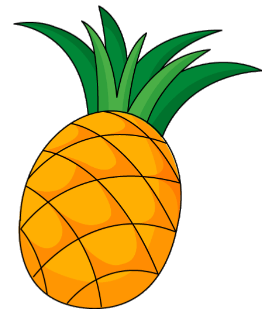
- Irish Election Data:
  - 5% submitted complete ballots for Dublin North.
  - 12% for Dublin West.
- APA Election Data
  - Chamberlin's original data had over 65% incomplete ballots over 5 candidates.
- ANES Thermometer Rankings
  - Takes ratings and turns them into rankings, breaking ties randomly.
- Sushi Dataset
  - Incomplete survey's are discarded (sample bias).

Models - Agnostic

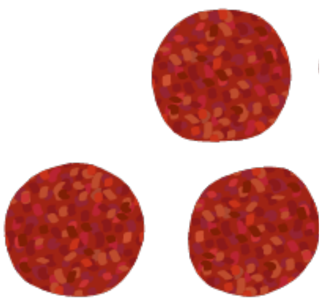
>



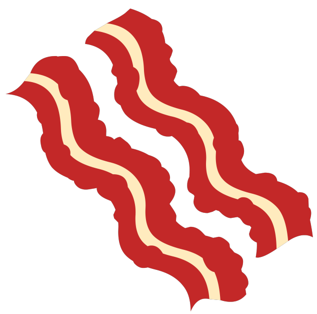
>



Models - Pessimistic



>

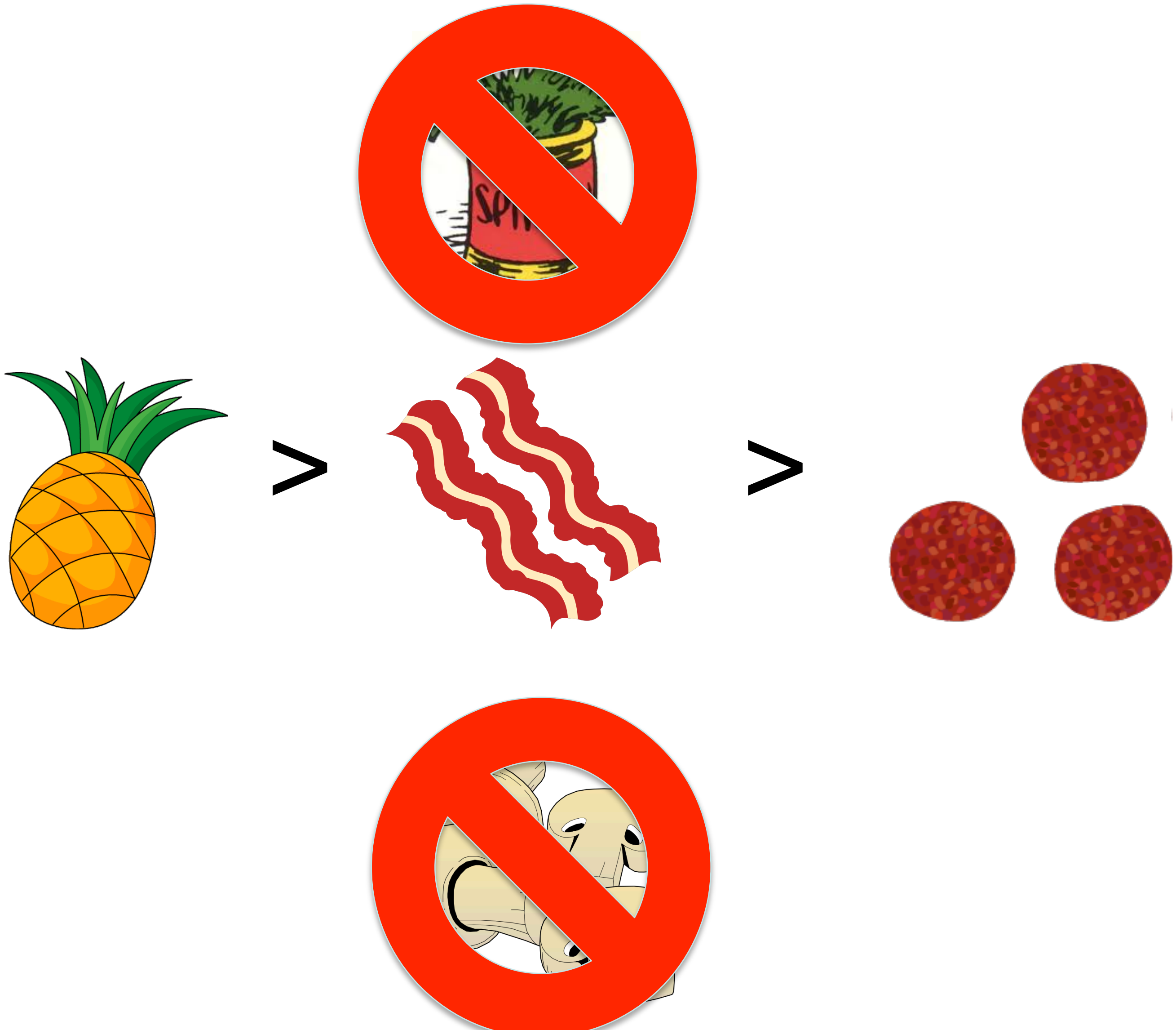


>

>

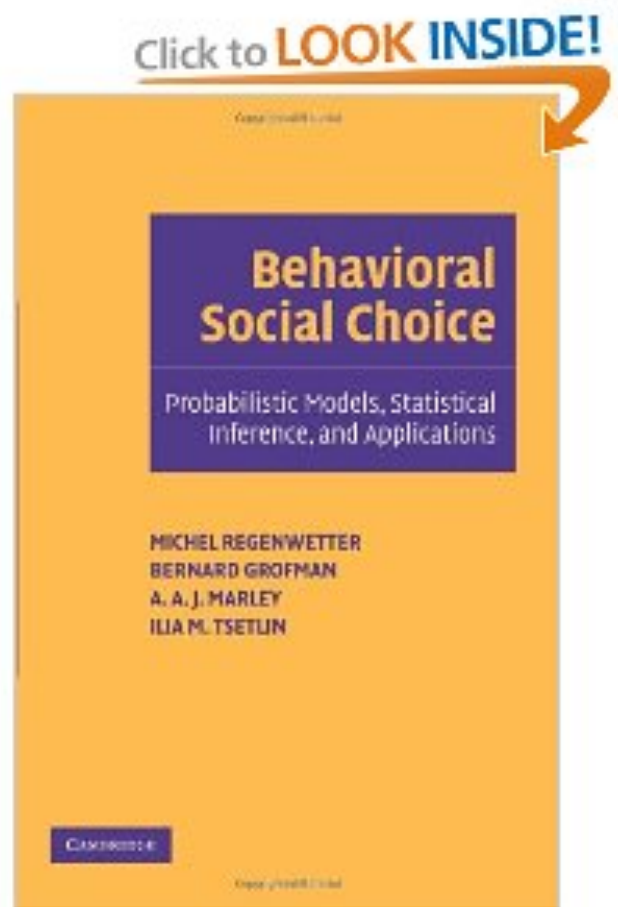
Nicholas Mattei
NICTA and UNSW

Models – Anchor and Adjust



Bootstrap Reliability – Use Statistics

- For an election with *n* votes:
  - Draw *n* votes, with replacement.
  - Evaluate election
  - Do it 10,000 times
- Gives an indication of how robust the sample (election) is to small amounts of random noise.
- *Inference* instead of sampling.



Beware of Model Dependent Effects!

- Behavioral aspects of individuals can have substantial impacts on the resulting computational problems
  - Youtube's dropping of the star ratings system…
  - Dropping non-responders…
- Pay particular attention to the domain in which we wish to deploy our results.
- Move towards preference reasoning as an *inference* problem rather than a sampling problem and be acutely aware of model dependence.

{PrefLib}: A Library for Preferences

Main | About | Papers | Data Formats | Data By Domain | Data By Type | Tools

A reference library of preference data and links assembled by Nicholas Mattei and Toby Walsh. We currently house over 3,000 datasets for use by the community.

We want to provide a comprehensive resource for the multuple research communities that deal with preferences, including computational social choice, recommender systems, data mining, machine learning, and combinatorial optimization, to name just a few.

Please see the about page for information about the site, contacting us, and our citation policy. We rely on the support of the community in order to grow the usefulness of this site. To contribute, please contact Nicholas Mattei at: nicholas**{dot}**mattei@nicta.com.au

$a > b > c > d$

$a > b, c, d > e$

$\frac{1}{2} : a > b > c$

$\frac{1}{4} : c > b > a$

$\frac{1}{4} : b > c > a$

Supported By:

NICTA

Sept. 3, 2013:
A big update today brings us over 3000 datasets hosted on the site with a full data archive over 7 GB!

We have also added a Thanks!! section to recognize those individuals who have helped make PrefLib possible.

July 1, 2013:
Our paper has been accepted to 2013 Conference on Algorithmic Decision Theory. We have also had several new donated datasets which have been parsed and posted.

We have added a new Papers section to the site with a list of papers that have used PrefLib!

Links

- UC Irvine Machine Learning Repository
- University of Minnesota GroupLens Data Sets
- CSPLib: A Problem Library for Constraints
- Microsoft Learning to Rank Datasets
- SATLib: The Satisfiability Library
- Preference-Learning.org
- Toshihiro Kamishima's Sushi Preference Dataset
- MAX-SAT Evaluations and Datasets

{PrefLib}: A Library for Preferences

Main **About** **Papers** **Data Formats** **Data By Domain** **Data By Type** **Tools**

A reference library of preference data and links assembled by Nicholas Mattei and Toby Walsh. We currently house over 3,000 datasets for use by the community.

We want to provide a comprehensive resource for the multuple research communities that deal with preferences, including computational social choice, recommender systems, data mining, machine learning, and combinatorial optimization, to name just a few.

Please see the about page for information about the site, contacting us, and our citation policy. We rely on the support of the community in order to grow the usefulness of this site. To contribute, please contact Nicholas Mattei at: nicholas**{dot}**mattei@nicta.com.au

$$a > b > c > d$$

$$a > b, c, d > e$$

$$\frac{1}{2} : a > b > c$$
$$\frac{1}{4} : c > b > a$$
$$\frac{1}{4} : b > c > a$$

Supported By:

NICTA

www.preflib.org

Sept. 3, 2013:
A big update today brings us over 3000 datasets hosted on the site with a full data archive over 7 GB!

We have also added a Thanks!! section to recognize those individuals who have helped make PrefLib possible.

July 1, 2013:
Our paper has been accepted to 2013 Conference on Algorithmic Decision Theory. We have also had several new donated datasets which have been parsed and posted.

We have added a new Papers section to the site with a list of papers that have used PrefLib!

Links

- UC Irvine Machine Learning Repository
- University of Minnesota GroupLens Data Sets
- CSPLib: A Problem Library for Constraints
- Microsoft Learning to Rank Datasets
- SATLib: The Satisfiability Library
- Preference-Learning.org
- Toshihiro Kamishima's Sushi Preference Dataset
- MAX-SAT Evaluations and Datasets

Broad Overview

- Part 1: Overview of ComSoc and Related Areas
- Part 2: A Primer on Preferences
  - Constraints v. Preferences
  - Formalisms and Languages
- Part 3: Voting
  - Classic Paradoxes and Results
  - Computational Aspects of Voting
  - Game Theoretic Aspects of Voting

Social Choice

- Given a collection of agents with preferences over a set of things (houses, cakes, meals, plans, etc.) we must…
  - 1. Pick one or more of them as winners for the entire group

    OR....
  - 2. Assign the items to each of the agents in the group.

Subject to a number of exogenous goals, axioms, metrics, and/or constraints.

So What Do We *DO* With Preferences?

- We take a multi-agent viewpoint: each preference comes from a different agent and we need to make a group decision.
- We want to select the most preferred alternative(s) according to the preferences of all the agents.

View 1: Vote to compromise among subjective preferences.

View 2: Vote to reconcile noisy observations to determine truth.

amazon

Nicholas Mattei
NICTA and UNSW

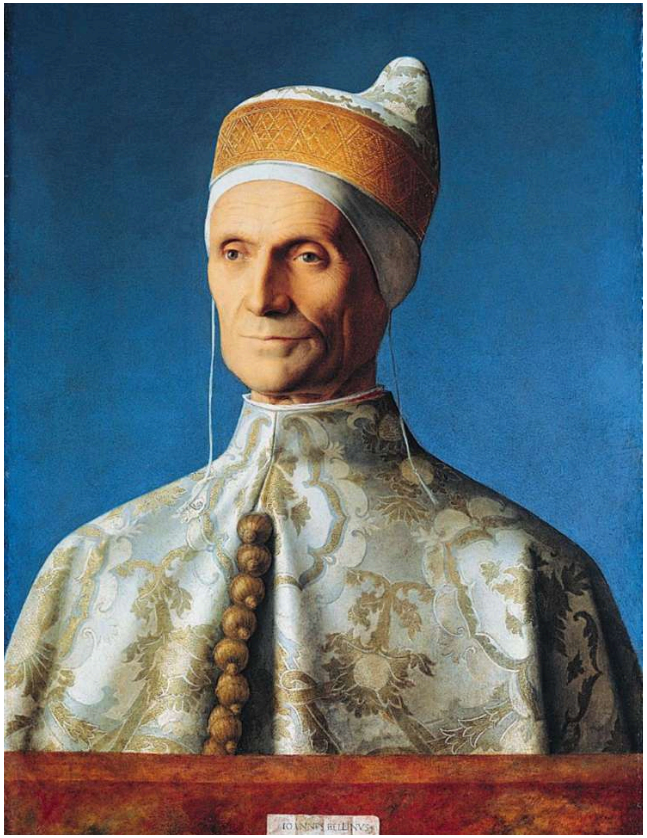
Elections

- In general, we define an election as:
  - A set of alternatives, or candidates *C* of size *m.*
  - A set of voters *V* of size *n*.
  - All together, called a profile*, P.*
  - A **resolute voting rule** selects a single winner from *C.*
  - A **voting correspondence** selects a set of winners from *C.*
  - A **social welfare function** returns an ordering (ranking) over C.

Elections

- In general, we define an election as:
  - A set of alternatives, or candidates *C* of size *m.*
  - A set of voters *V* of size *n*.
  - All together, called a profile*, P.*
  - A resolute voting rule, voting correspondence or social welfare function, *R.*
- Aggregate the set of votes from *V* over the set of candidates *C* and return the result according to *R.*

Unreasonable Voting Rules?

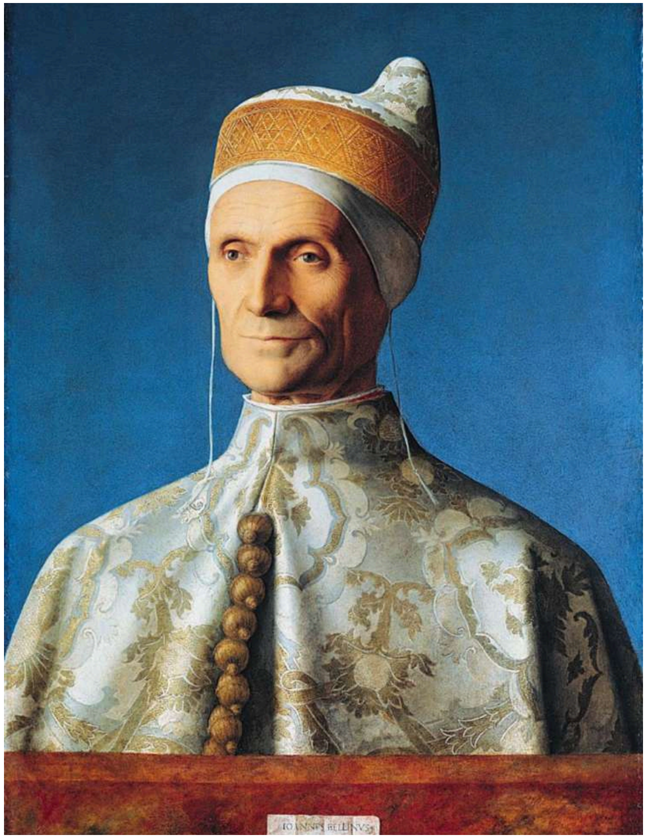


- Select random boy off the street to draw lotteries.

Unreasonable Voting Rules?

- Select random boy off the street to draw lotteries.
- Round 1: Every member of the Great Council is narrowed to 30 via lottery.
- Round 2: Narrow this to 9 out of 30 by lottery.

Unreasonable Voting Rules?



- Select random boy off the street to draw lotteries.
- Round 1: Every member of the Great Council is narrowed to 30 via lottery.
- Round 2: Narrow this to 9 out of 30 by lottery.
- Round 3: By a minimum vote of 7/9, select 40 representatives from the Great Council.

Unreasonable Voting Rules?

- Select random boy off the street to draw lotteries.
- Round 1: Every member of the Great Council is narrowed to 30 via lottery.
- Round 2: Narrow this to 9 out of 30 by lottery.
- Round 3: By a minimum vote of 7/9, select 40 representatives from the Great Council.
- Round 4: Select 12 out of 40 by lottery.
- Round 5: The 12 elect 25 each requiring 9/12 votes.

Unreasonable Voting Rules?

- Select random boy off the street to draw lotteries.
- Round 1: Every member of the Great Council is narrowed to 30 via lottery.
- Round 2: Narrow this to 9 out of 30 by lottery.
- Round 3: By a minimum vote of 7/9, select 40 representatives from the Great Council.
- Round 4: Select 12 out of 40 by lottery.
- Round 5: The 12 elect 25 each requiring 9/12 votes.
- Round 6: Reduce the 25 to 9 again by lottery.
- Round 7: The 9 elect a college of 45 requiring 7/9 votes.

Unreasonable Voting Rules?

- Select random boy off the street to draw lotteries.
- Round 1: Every member of the Great Council is narrowed to 30 via lottery.
- Round 2: Narrow this to 9 out of 30 by lottery.
- Round 3: By a minimum vote of 7/9, select 40 representatives from the Great Council.
- Round 4: Select 12 out of 40 by lottery.
- Round 5: The 12 elect 25 each requiring 9/12 votes.
- Round 6: Reduce the 25 to 9 again by lottery.
- Round 7: The 9 elect a college of 45 requiring 7/9 votes.
- Round 3: The 45 were again reduced to 11 by lottery.
- Round 3: The 11 elect a college of 41 by 9/11 majorities.
- Round 10: The 41, with a majority vote of at least 25/41, elect the Doge of Venice.

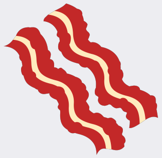
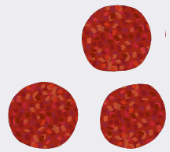
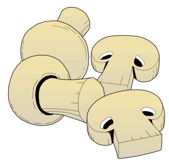
Really?

- 75 Doges were elected over 600 years (between 1172 and 1797).
- Only stopped because Napoleon took over.
- Many interesting and useful properties.

Selecting a Voting Rule

- We can start from first principles or ***axioms*** that we want voting rules to satisfy.
- **Anonymity:** the names of the voters do not matter.
- **Non-dictatorship:** there is no voter who always selects the winner.
- **Neutrality:** the names of the alternative do not matter.
- **Condorcet Consistency:** If one alternative is preferred by a majority in ***all*** pairwise comparisons, this alternative should win.
    - And many many more..

Simple Majority Rule

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

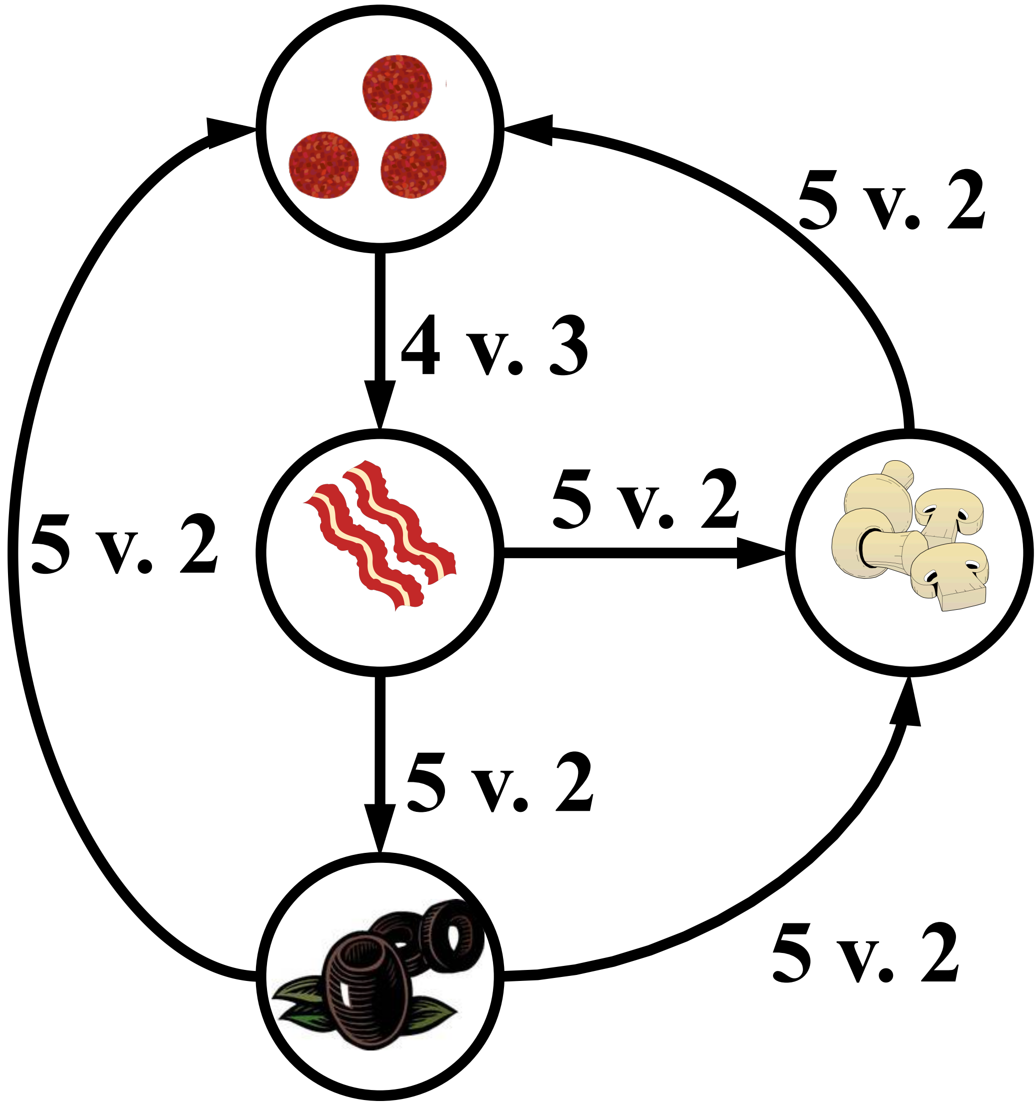
Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Nicholas Mattei
NICTA and UNSW

Condorcet's Paradox!

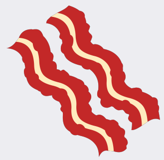
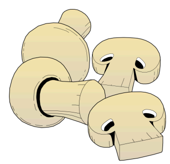
Candidates	
Bacon	Pepperoni
Olives	Mushroom

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B



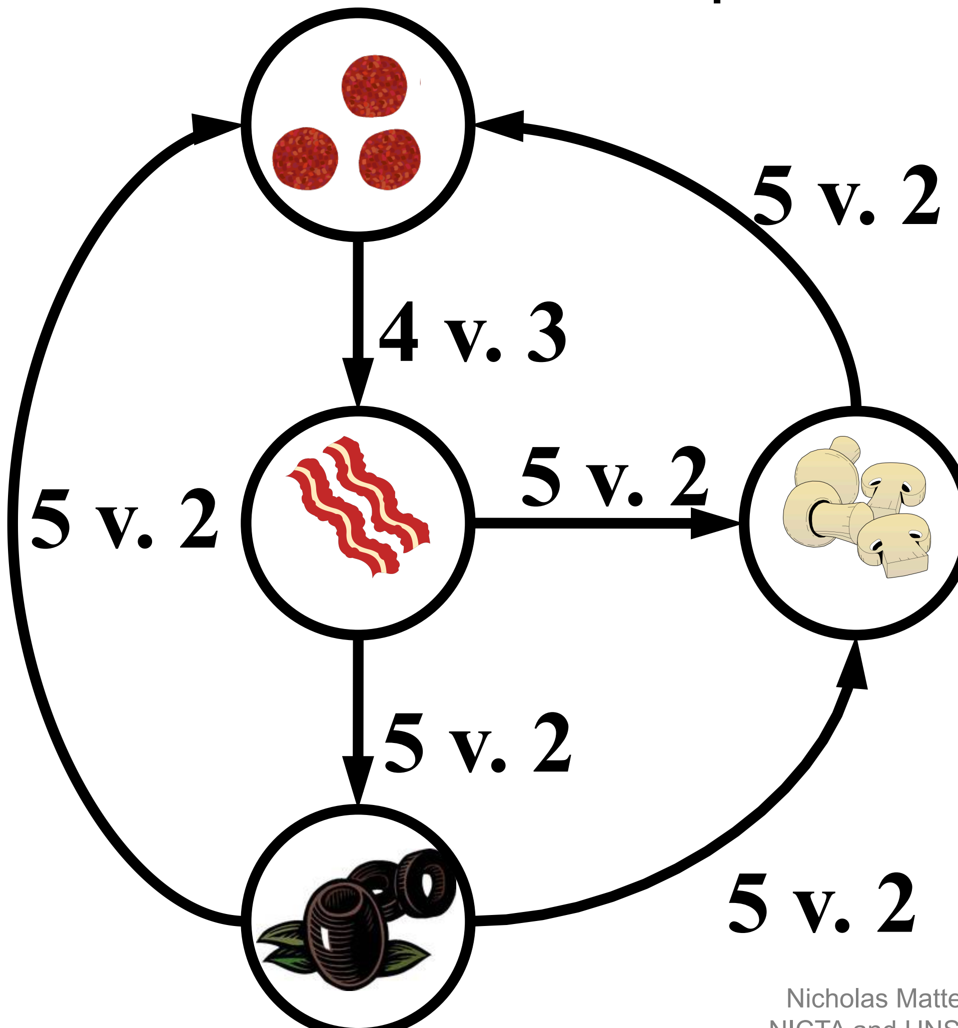
Nicholas Mattei
NICTA and UNSW

Copeland Scoring

Candidates	
 Bacon	 Pepperoni
 Olives	 Mushroom

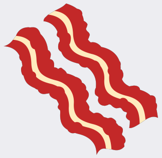
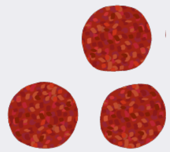
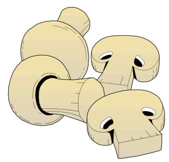
Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

- In all pairwise contests, the winner receives a point.



Nicholas Mattei
NICTA and UNSW

Copeland Scoring

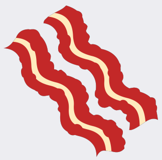
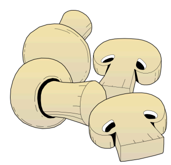
Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

- In all pairwise contests, the winner receives a point.

Pair	Result	Winner
P v. B	4 to 3	P
P v. O	2 to 5	O
P v. M	2 to 5	M
B v. O	5 to 2	B
B v. M	5 to 2	B
O v. M	5 to 2	O

Copeland Scoring

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

- In all pairwise contests, the winner receives a point.

Pair	Result	Winner
P v. B	4 to 3	P
P v. O	2 to 5	O
P v. M	2 to 5	M
B v. O	5 to 2	B
B v. M	5 to 2	B
O v. M	5 to 2	O

Result
O and B tie with 2 each.

Nicholas Mattei
NICTA and UNSW

Scoring Rules

- A **family** of voting rules where we award points for placement in the preference list
- **Plurality:** First place gets a point (S = [1, 0, 0 … 0]).
- **Veto:** All but last gets a point (S = [1, 1, 1, …, 0]).

Plurality	
B	3
O	2
P	2
M	0

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Scoring Rules

- A **family** of voting rules where we award points for placement in the preference list
- **Plurality:** First place gets a point (S = [1, 0, 0 … 0]).
- **Veto:** All but last gets a point (S = [1, 1, 1, …, 0]).

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Plurality	
B	3
O	2
P	2
M	0

Veto	
O	7
B	5
M	5
P	4

Scoring Rules

- **Borda:** A candidate receives more points for being placed higher in the preference list ($S = [m - 1, m - 2, \ldots 0]$).

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Scoring Rules

- **Borda:** A candidate receives more points for being placed higher in the preference list ($S = [m - 1, m - 2, \ldots 0]$).

Borda	
O	2*1 + 3*2 + 2*3 = 14
B	2*2 + 3*3 + 0 = 13
P	2*3 + 0 + 2*1 = 8
M	0 + 3*1 + 2*2 = 7

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Plurality with Runoff

- **Round 1:** Plurality Score.

Count	Vote
10	P > B > O > M
7	M > P > B > O
6	O > M > P > B
3	B > O > M > P

Plurality	
P	10
M	7
O	6
B	3

Plurality with Runoff

- **Round 1:** Plurality Score.

Count	Vote
10	P > B > O > M
7	M > P > B > O
6	O > M > P > B
3	B > O > M > P

Plurality	
P	10
M	7
O	6
B	3

- **Round 2:** Select the most preferred remaining.

Count	Vote
10	P > M
7	M > P
6	M > P
3	M > P

Run-Off	
M	16
P	10

Single Transferable Vote (STV)

- Also known as Instant Run-off Voting and used in Australia, Ireland, and places in the US.
- We have *m-1* rounds where we eliminate the alternative with lowest plurality score.
- Winner is the last one left.

Count	Vote
10	P > B > O > M
7	M > P > B > O
6	O > M > P > B
3	B > O > M > P

Round 1	
P	10
M	7
O	6
B	3

Single Transferable Vote (STV)

- Also known as Instant Run-off Voting and used in Australia, Ireland, and places in the US.
- We have *m-1* rounds where we eliminate the alternative with lowest plurality score.
- Winner is the last one left.

Count	Vote
10	P > O > M
7	M > P > O
6	O > M > P
3	O > M > P

Round 2	
P	10
O	9
M	7
B	--

Single Transferable Vote (STV)

- Also known as Instant Run-off Voting and used in Australia, Ireland, and places in the US.
- We have *m-1* rounds where we eliminate the alternative with lowest plurality score.
- Winner is the last one left.

Count	Vote
10	P > O
7	P > O
6	O > P
3	O > P

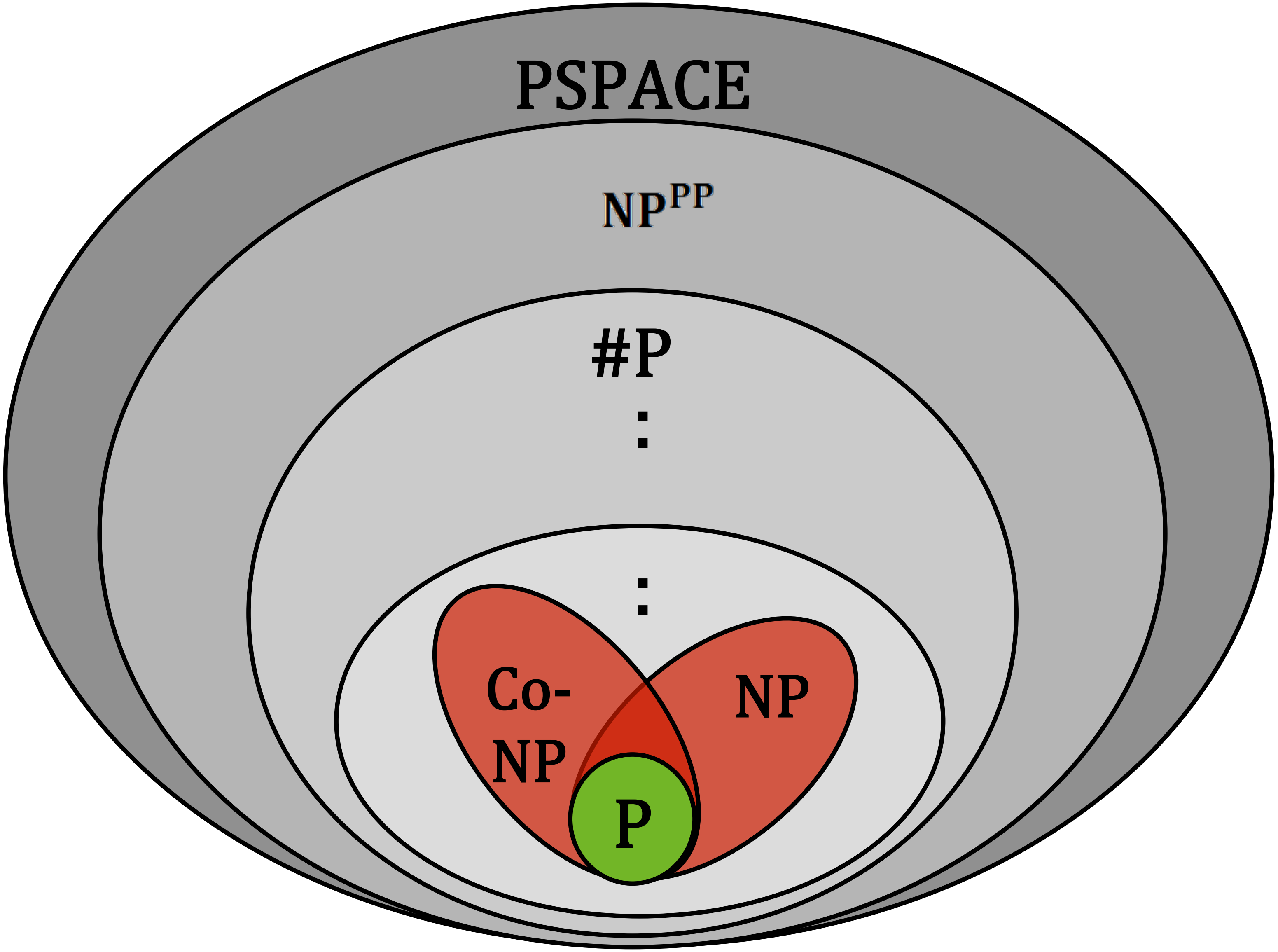
Round 3	
P	17
O	9
M	--
B	--

More Complicated Rules

- **Dodgson's Voting:** Select the winner which has the closest swap distance to being a Condorcet Winner.

- **Kemeny-Young:** Select the ordering which minimizes the sum of Kendall-Tau (Bubble Sort) distances to the input profile.
- However, these rules are intractable!!

Finding Winners Should Be EASY!



2 Threads of ComSoc

Analyze Results

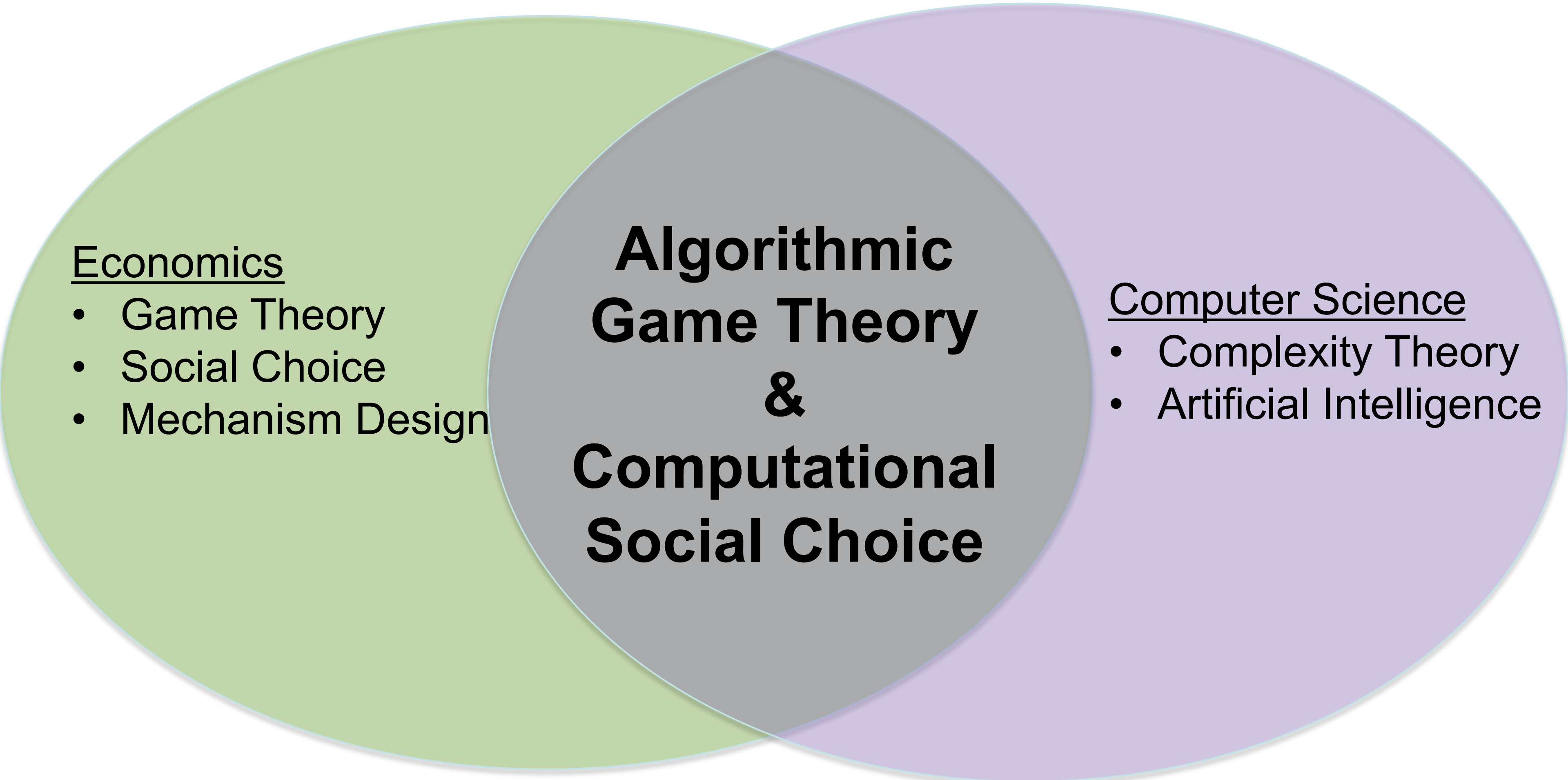
Analyze computational aspects of Social Choice. Many classic results in Social Choice Theory ignore the computational aspects of the theory.

↔

Import Ideas to AI

Implement ideas from Social Choice Theory in designing, implementing, and deploying systems across computer science including AI and multi-agent systems.

AGT and ComSoc



Even More Axioms…

- We’ve seen: anonymity, neutrality, non-dictatorship, and the Condorcet Criteria.

- What other good axioms can you come up with?

Even More Axioms…

- We've seen: anonymity, neutrality, non-dictatorship, and the Condorcet Criteria.
- What other good axioms can you come up with?
- **Non-Imposition** or **Universal Domain**: each alternative is the unique winner under at least one profile.
- All neutral, resolute voting procedures satisfy non-imposition (surjective (onto) functions).

The No-Show Paradox

- With Plurality with Run-off it can be better to abstain..

Count	Vote
25	P > M > O
46	O > P > M
24	M > O > P

Plurality	
O	46
P	25
Winner	
Olives!	

The No-Show Paradox

- With Plurality with Run-off it can be better to abstain..

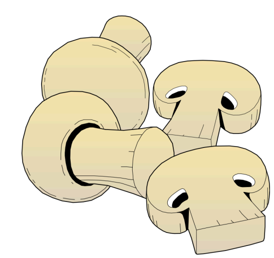
Count	Vote
25	P > M > O
46	O > P > M
24	M > O > P

Plurality	
O	46
P	25
Winner	
Olives!	

- Removing 2 voters....

Count	Vote
23	P > M > O
46	O > P > M
24	M > O > P

Plurality	
O	46
M	24
Winner	
Mushroom!	



Nicholas Mattei
NICTA and UNSW

Participation and Reinforcement

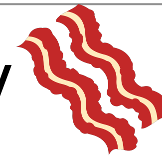
- **Participation:** Given a voter, his addition to a profile *P* results in the same or a more preferred result.
  - **We never have an incentive to abstain.**

- **Reinforcement (Consistency):** Given 2 profiles P_1 and P_2 over the same set of candidates C and rule R if we have $R(P_1) \cap R(P_2) \neq \varnothing$ then $R(P_1 \cup P_2) = R(P_1) \cap R(P_2)$.
  - **If**  **is elected in two disjoint profiles..**

  Bacon

  combining them together shouldn't change this.

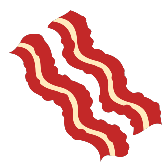
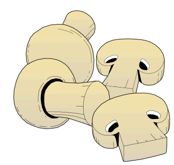
Axioms about Strong Preferences

- **Unanimous:** If *all* voters say  is the best

  Bacon

  then we select 

  Bacon

- **(Weak) Pareto Condition:** If all voters in the profile prefer  to  then we never

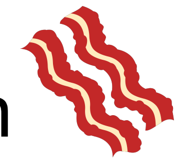
  Bacon Mush.

  select 

  Mush.

Monotonicity

- A current *winner* should not be made a *loser* by increasing support.

- If  Bacon is a winner given a vote *v*, then  Bacon

  must **remain a winner** in all other votes

  v' obtained from *v* where  Bacon is ranked higher.

Mmmmm….  Bacon

Nicholas Mattei
NICTA and UNSW

Picking on Plurality with Run-off..

Count	Vote
27	P > M > O
42	O > P > M
24	M > O > P

Plurality	
O	42
P	27
Winner	
Olives!	

It's Non-Monotonic!

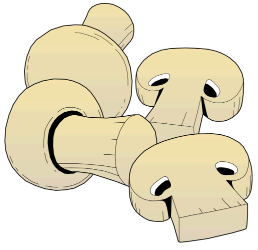
Count	Vote
27	P > M > O
42	O > P > M
24	M > O > P

Plurality	
O	42
P	27
Winner	
Olives!	

- By switching 4 votes **TO** olives..

Count	Vote
23	**P > M > O**
46	**O > P > M**
24	M > O > P

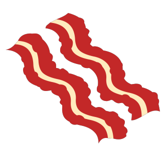
Plurality	
O	50
M	24
Winner	
Mushroom!	



Independence of Irrelevant Alternatives

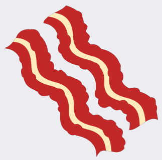
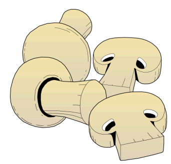
- Another (very strong) axiom about how preferences can change when adding new votes.

- **IIA:** whenever **B** is a winner and **M** is not and we modify *P* such that the relative ranking of **B and M** does not change in *P* then **M** cannot win.

  – 

  Bacon ***remains a winner despite any possible changes to irrelevant alternatives.***

Using Axioms…

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

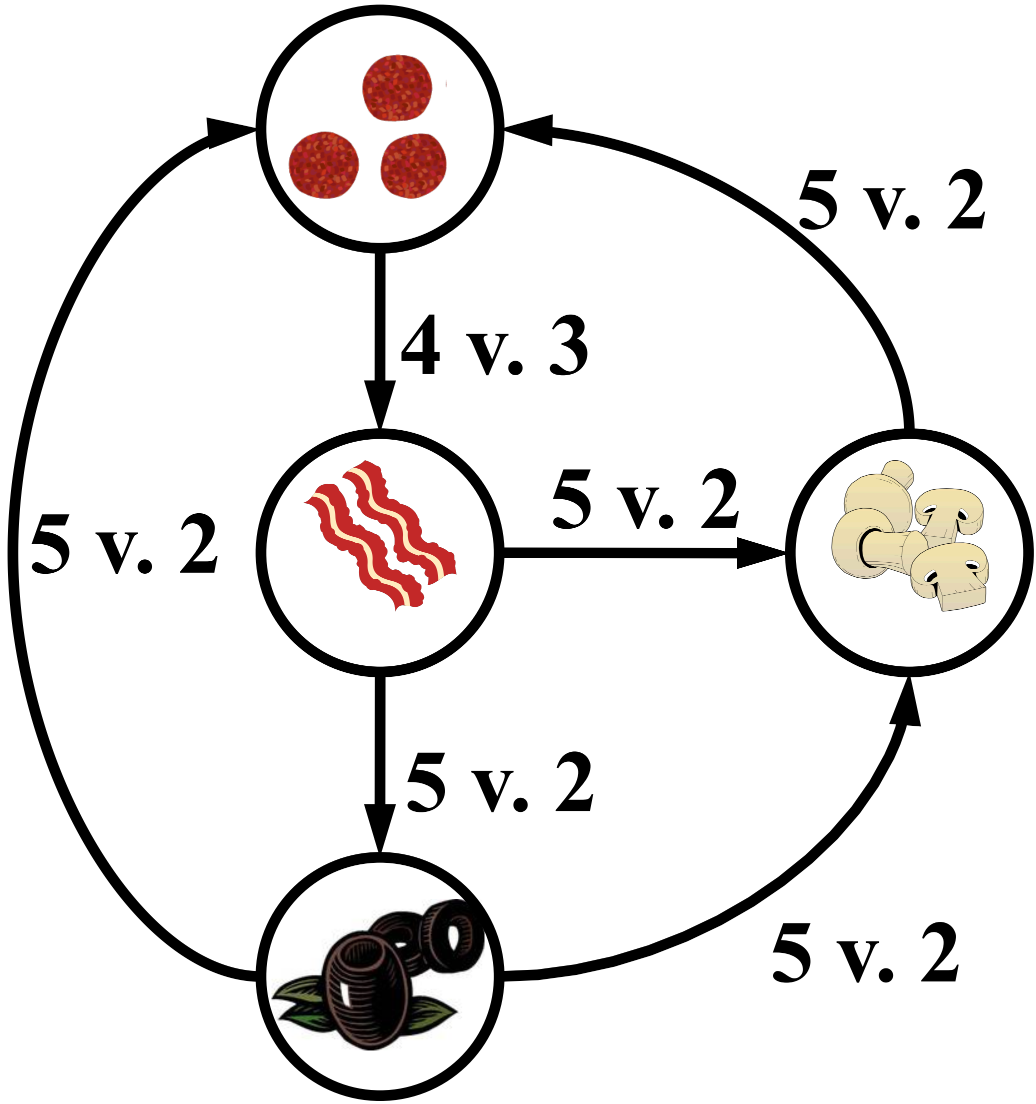
Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B

Nicholas Mattei
NICTA and UNSW

Condorcet's Paradox!

Candidates	
Bacon	Pepperoni
Olives	Mushroom

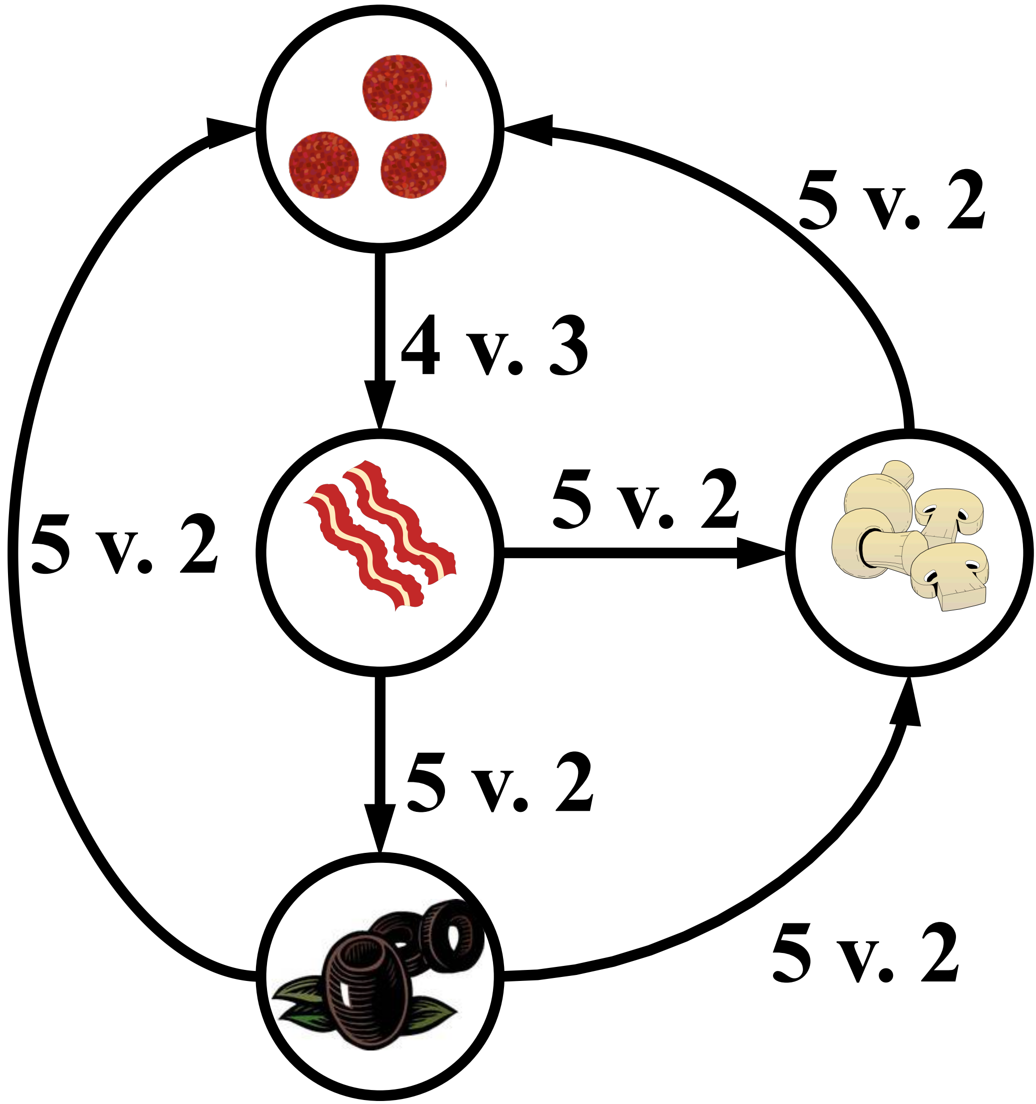
Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B



Condorcet's Paradox!

- This result can be expanded to prove the following fact…
- [**Fishburn 74**]: There exists no positional scoring rule that is Condorcet Consistent!

Count	Vote
2	P > B > O > M
3	B > O > M > P
2	O > M > P > B



Nicholas Mattei
NICTA and UNSW

Positive Facts...

- Using the axioms we have discussed we can come up with some positive results!

AP

- [**May 52**] If a voting rule is decisive, anonymous, neutral, monotone (positively responsive) and has only two candidates, then it must be the **majority rule**!

So maybe we got that right…

Mostly Bad News Though…

- **Arrow's Theorem [Arrow 51]:** If there are more than three alternatives then we cannot devise a voting rule that satisfies weak Pareto optimality, non-dictatorship, and independence of irrelevant alternatives (IIA)!

K. J. Arrow 1951. *Social Choice and Individual Values*. John Wiley and Sons.

And Worse!

- **[Muller and Satterthwaite 77]:** If there are at least 3 candidates then no voting rule simultaneously satisfies non-imposition, monotonicity, and is non-dictatorial!

Other Pitfalls of Voting Systems

- **Gibbard – Satterthwaite:** Any resolute voting procedure for at least 3 candidates that is surjective and strategy-proof is dictatorial.

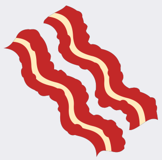
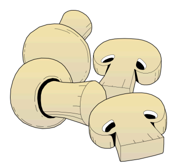
Dictatorships are starting to look good….

- A. Gibbard 1973. Manipulation of voting schemes. *Econometrica 41*.
- M. Satterthwaite 1975. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *J. Econ. Theory 10*.

Manipulation and Voting

- 3 primary ways to look at affecting an aggregation procedure:
  - Manipulation
  - Bribery
  - Control
- Given a preferred candidate, can we make it a winner?

Coalitional Manipulation

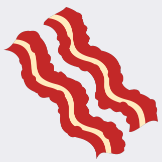
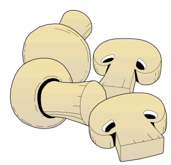
Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	P > O > B > M

- Can an agent or group of agents misrepresent their preferences in such a ways as to obtain a better result?

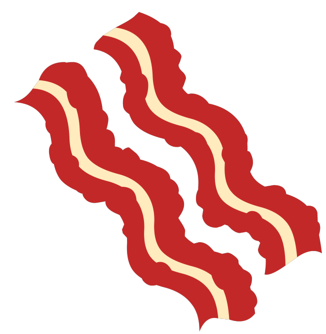
- We generally make worst case assumptions:
  - Manipulator(s) know all.
  - Tie-breaking favors them…

Coalitional Manipulation

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	P > O > B > M

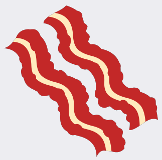
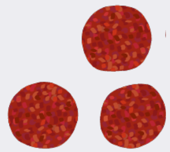
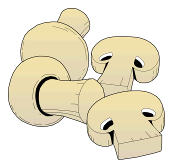
- Can an agent or group of agents misrepresent their preferences in such a ways as to obtain a better result?



Bacon

Nicholas Mattei
NICTA and UNSW

Coalitional Manipulation

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	**O > P > B > M**

- Can an agent or group of agents misrepresent their preferences in such a ways as to obtain a better result?

Olive!

Nicholas Mattei
NICTA and UNSW

Computer Science To The Rescue!

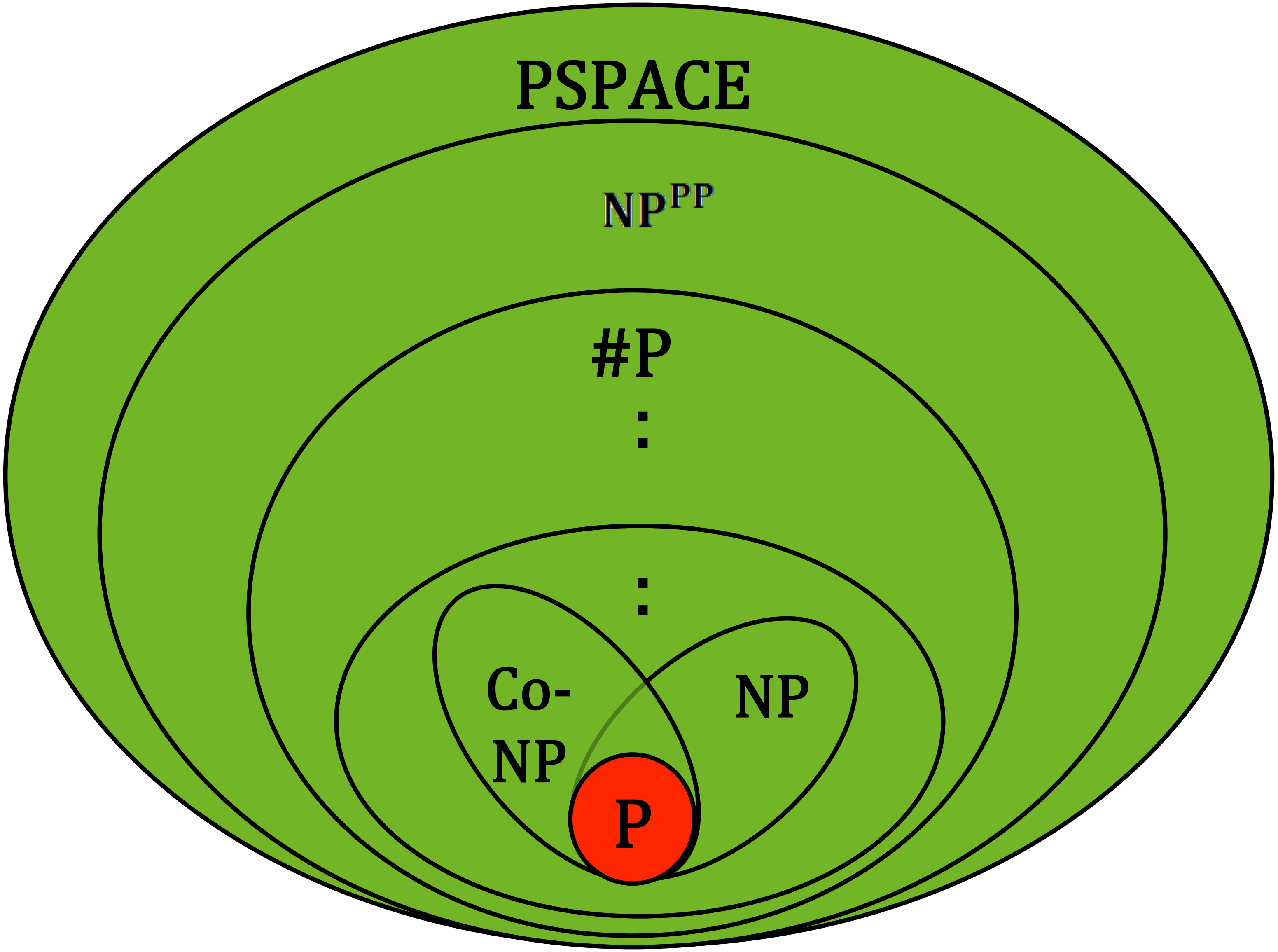
- An idea by Bartholdi, Tovey, and Trick on how to protect elections: **COMPLEXITY!**

- Like cryptography, if a manipulation is NP-hard to compute then maybe elections will not be manipulated.

- Founded a line of research that is still highly active in the ComSoc community.

- J. Bartholdi, III, C. Tovey, and M. Trick 1989. The computational difficulty of manipulating an election. *Social Choice and Welfare, 6(3)*.

Good is Bad!



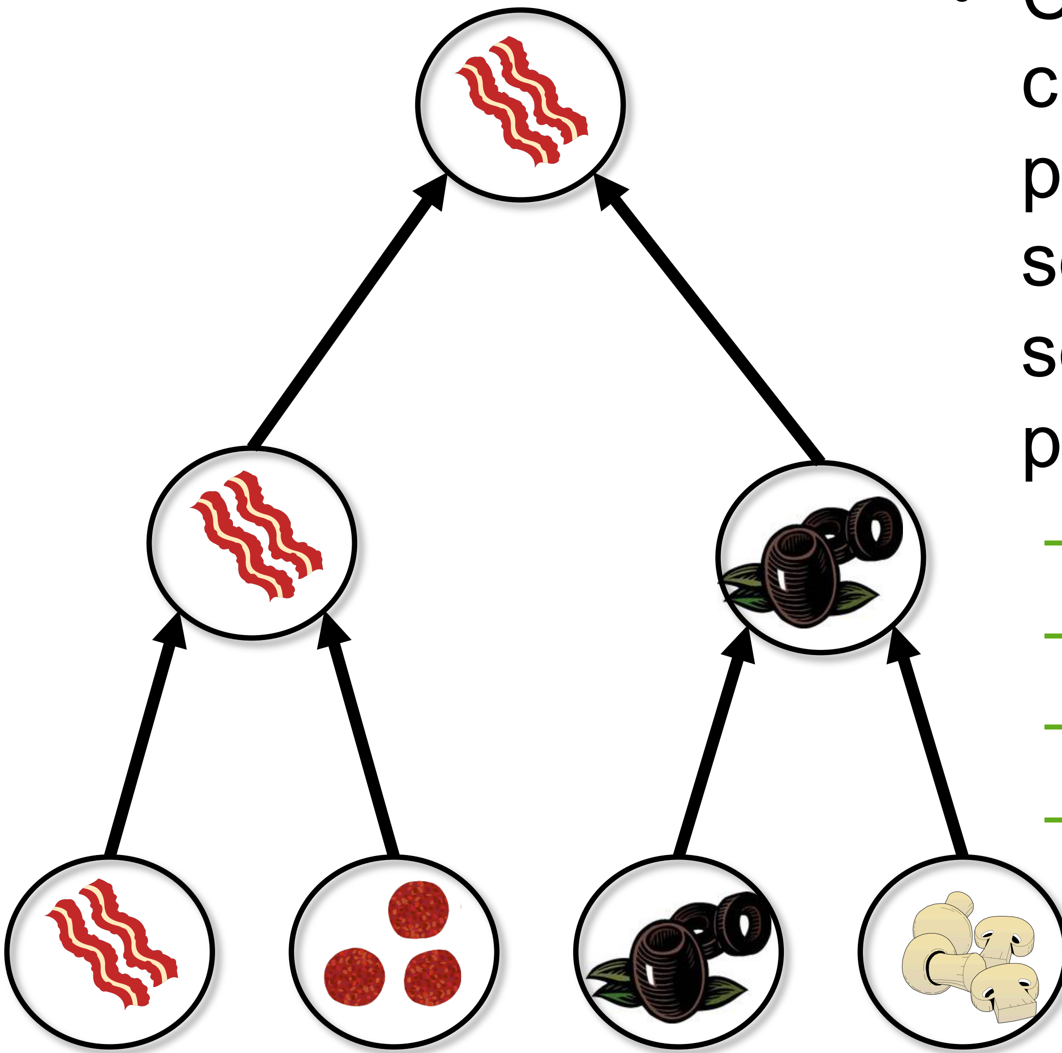
Nicholas Mattei
NICTA and UNSW

Coalitional Manipulation Results!

Voting Rule	One Manipulator	At Least 2
Copeland	Polynomial	NP-Complete
STV	Polynomial	NP-Complete
Veto	Polynomial	Polynomial
Plurality with Runoff	Polynomial	Polynomial
Cup	Polynomial	Polynomial
Borda	Polynomial	NP-Complete
Maximin	Polynomial	NP-Complete
Ranked Pairs	NP-Complete	NP-Complete
Bucklin	Polynomial	Polynomial
Nanson's Rule	NP-Complete	NP-Complete
Baldwin's Rule	NP-Complete	NP-Complete

– Many of these appeared in top AI (AAAI, IJCAI, etc.)
– Thanks to Lirong Xia for the table!

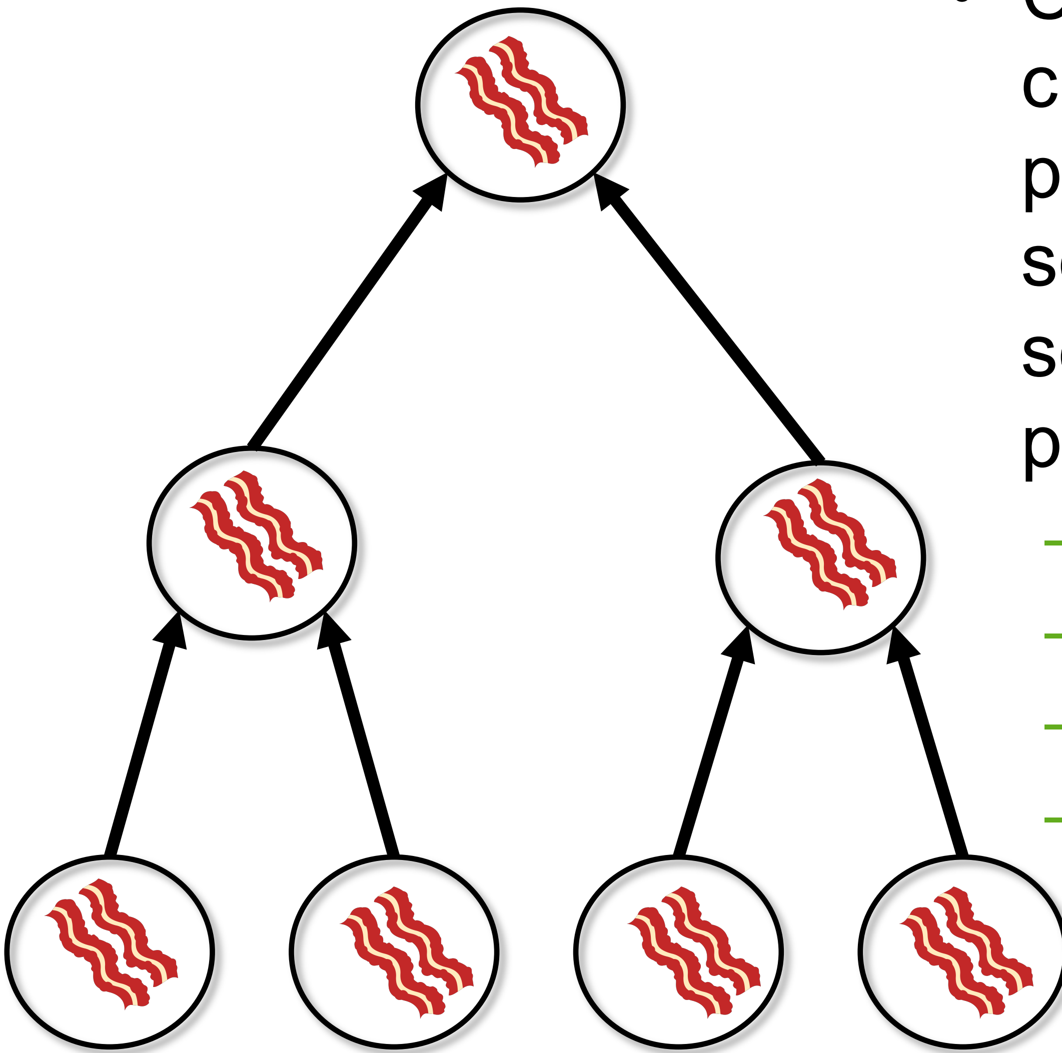
Control Problems



- Control involves changing some parameter of the setting in order to select a more preferred candidate.
  - Change the voting tree
  - Add candidates
  - Replace candidates
  - Add/Delete/Replace voters…

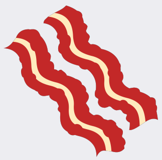
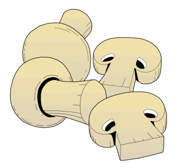
Nicholas Mattei
NICTA and UNSW

Control Problems (with constraints!)



- Control involves changing some parameter of the setting in order to select a more preferred candidate.
  - Change the voting tree
  - Add candidates
  - Replace candidates
  - Add/Delete/Replace voters…

Bribery

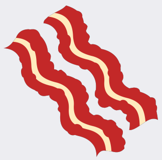
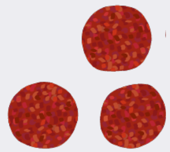
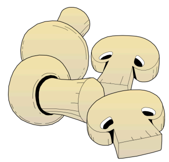
Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	P > O > B > M

- Can we expend some resource in order to make a particular candidate a winner.
  - Money
  - Time
  - Pollsters
- Usually subject to hard constraints or can only affect probability of changing someone’s mind..

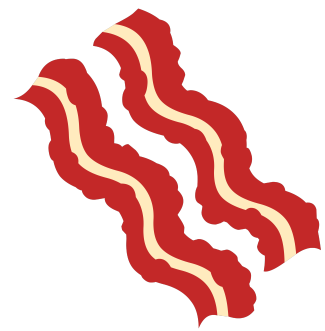
Nicholas Mattei
NICTA and UNSW

Bribery

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	P > O > B > M

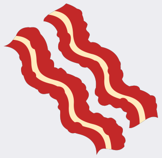
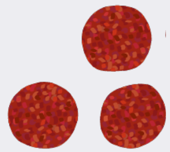
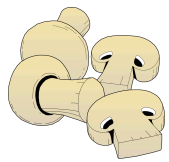
- Can we expend some resource in order to make a particular candidate a winner.



Bacon

Nicholas Mattei
NICTA and UNSW

Bribery

Candidates	
 **B**acon	 **P**epperoni
 **O**lives	 **M**ushroom

Count	Vote
49	B > O > P > M
20	O > P > B > M
20	O > B > P > M
11	**O > P > B > M**

- Can we expend some resource in order to make a particular candidate a winner.

Olive!

Broad Overview

- Part 1: Overview of ComSoc and Related Areas
- Part 2: A Primer on Preferences
  - Constraints v. Preferences
  - Formalisms and Languages
- Part 3: Voting
  - Classic Paradoxes and Results
  - Computational Aspects of Voting
  - Game Theoretic Aspects of Voting

2 Threads of ComSoc

Analyze Results

Analyze computational aspects of Social Choice. Many classic results in Social Choice Theory ignore the computational aspects of the theory.

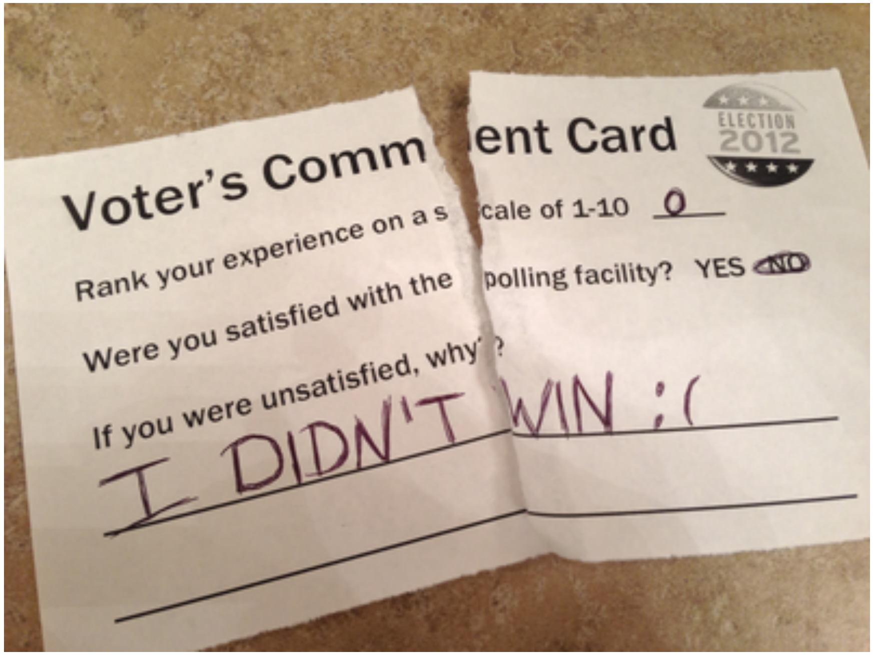
Import Ideas to AI

Implement ideas from Social Choice Theory in designing, implementing, and deploying systems across computer science including AI and multi-agent systems.

- Many interesting and exciting avenues of research both theoretical and practical in just **Part One**!!

Thanks!

- Questions



- Comments

Nicholas Mattei
NICTA and UNSW

Part II: Tie-Breaking and Multi-Winner Rules

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

Schedule for the Day

Time	Plan
8:30 – 8:45	Intro. to Computational Social Choice
8:45-10:00	Preferences and Voting
10:00 – 10:30	Coffee Break
10:30 – 12:00	Matching, Resource Allocation, and Fair Division
12:00 – 1:30	Lunch
1:30 – 2:15	Advanced Topics in Preference Aggregation
2:15 – 2:45	Matching, Resource Allocation, and Fair Division II
2:45 – 3:00	Closing Remarks and Survey

Social Choice

- Given a collection of agents with preferences over a set of things (houses, cakes, meals, plans, etc.) we must…
  - 1. Pick one or more of them as winners for the entire group

    OR....

  - 2. Assign the items to each of the agents in the group.

Subject to a number of exogenous goals, axioms, metrics, and/or constraints.

2 Threads of ComSoc

Analyze Results

Analyze computational aspects of Social Choice. Many classic results in Social Choice Theory ignore the computational aspects of the theory.

↔

Import Ideas to AI

Implement ideas from Social Choice Theory in designing, implementing, and deploying systems across computer science including AI and multi-agent systems.

Elections

- In general, we define an election as:
  - A set of alternatives, or candidates *C* of size *m.*
  - A set of voters *V* of size *n*.
  - All together, called a profile*, P.*
  - A **resolute voting rule** selects a single winner from *C.*
  - A **voting correspondence** selects a set of winners from *C.*
  - A **social welfare function** returns an ordering (ranking) over C.

Elections

- In general, we define an election as:
  - A set of alternatives, or candidates C of size m.
  - A set of voters V of size n.
  - All together, called a profile, P.
  - A resolute voting rule, voting correspondence or social welfare function, R.
- Aggregate the set of votes from V over the set of candidates C and return the result according to R.

What About Ties?

- Up till now we've only talked about "breaking ties in favor of the manipulator"
- This transitions us from a **voting correspondence** to a **voting rule**.
- Does it have an affect on the complexity of manipulation?

Co-Manipulators

Nina Narodytska

Toby Walsh

Haris Aziz

Serge Gaspers

Tie-breaking Rules

- An arbitrary pre-set order (lexicographic, random, who's tallest)…

$$a > b > c > \ldots$$

Tie-breaking Rules

- Select a candidate uniformly at random from the set of co-winners…
- New Mexico, "any reasonable game of chance such as poker or craps."
  - Random candidate manipulation:
    - S. Obraztsova and E. Elkind. "On the complexity of voting manipulation under randomized tie-breaking." Proc. IJCAI 2011
    - S. Obraztsova, E. Elkind, and N. Hazon. "Ties matter: Complexity of voting manipulation revisited." AAMAS 2011.

Tie-breaking Rules

- Select a random vote from the set of votes and select the most preferred co-winner in the vote…

Tie-breaking Rules

- Select a random vote from the set of votes and select the most preferred co-winner in the vote…

- It's immune to clones and has an obvious disincentive for misreporting.

Random Vote != Random Candidate

$C = \{$ 

, 

, 

, 

$\}$

Nicholas Mattei
NICTA and UNSW

Random Vote != Random Candidate

$C = \{$ 

, 

, 

, 

$\}$

$Nick =$ 

Random Vote != Random Candidate

$C = \{$ 

, 

, 

, 

$\}$

$Nick =$ 

$Liz =$ 

Nicholas Mattei
NICTA and UNSW

Random Vote != Random Candidate

$Nick =$ 

$Liz =$ 

Borda Score
0
4
4
4

Nicholas Mattei
NICTA and UNSW

Random Vote != Random Candidate

$Nick =$ 

$Liz =$ 

Borda Score	Random Candidate
0	0
4	1/3
4	1/3
4	1/3

Nicholas Mattei
NICTA and UNSW

Random Vote != Random Candidate

$Nick =$ 

$Liz =$ 

	Borda Score	Random Candidate	Random Vote
	0	0	0
	4	1/3	0
	4	1/3	1/2
	4	1/3	1/2

Nicholas Mattei
NICTA and UNSW

Random Vote != Random Candidate

- There are voting correspondences for which the manipulation problem for tie-breaking with a random vote is NP-complete while random candidate is polynomial and vice-versa.
- Borda is NP-complete for random vote but polynomial for random candidate.
- We can demonstrate a voting rule to show the opposite direction.

k-approval

- For an unbounded k, it is NP-complete to compute the manipulator's vote.
- We show a reduction from *Hall Set:* given a bipartite graph G = (X, Y, E) and an integer z, is there a subset $S \subseteq X, |S| = z,$ and $|N(S)| < |S|$?
- Intuitively, construct an instances such that *p* can win with Pr > t if and only if there is a *Hall Set* in the profile.
  - A different reduction from Hall Set works for Bucklin, a reduction from 1in3 SAT for Borda.

Comparison of Voting Rules

Tie-Breaking Rule	P	NP-Hard	Open
Random Vote	plurality/veto k-approval (fixed) plurality with run-off	k-approval (unbounded) Borda ranked pairs STV simplified-Bucklin	Copeland maximin
Random Candidate	plurality/veto k-approval Borda plurality with run-off simplified Bucklin	Copeland (general) maximin STV ranked pairs	

Results for random candidate manipulation from S. Obraztsova and E. Elkind. "On the complexity of voting manipulation under randomized tie-breaking." Proc. IJCAI 2011 and S. Obraztsova, E. Elkind, and N. Hazon. "Ties matter: Complexity of voting manipulation revisited." AAMAS 2011.

Comparison of Voting Rules

Tie-Breaking Rule	P	NP-Hard	Open
Random Vote	plurality/veto k-approval (fixed) plurality with run-off	**k-approval (unbounded)** **Borda** ranked pairs STV **(simplified)-Bucklin**	Copeland maximin
Random Candidate	plurality/veto **k-approval** **Borda** plurality with run-off **(simplified)-Bucklin**	Copeland (general) maximin STV ranked pairs	

Results for random candidate manipulation from S. Obraztsova and E. Elkind. "On the complexity of voting manipulation under randomized tie-breaking." Proc. IJCAI 2011 and S. Obraztsova, E. Elkind, and N. Hazon. "Ties matter: Complexity of voting manipulation revisited." AAMAS 2011.

Social Choice

- Given a collection of agents with preferences over a set of things (houses, cakes, meals, plans, etc.) we must…
  - 1. Pick one or more of them as winners for the entire group

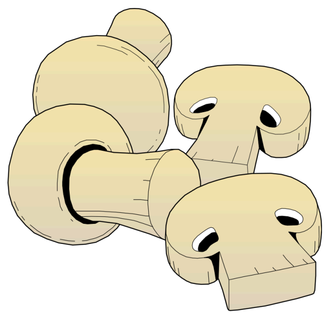
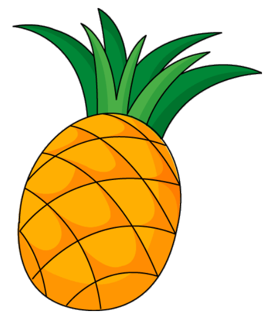
    OR....

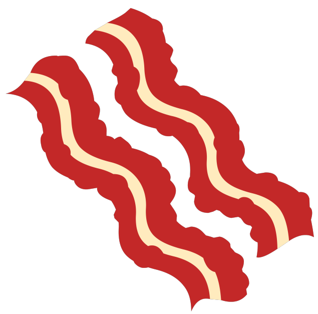
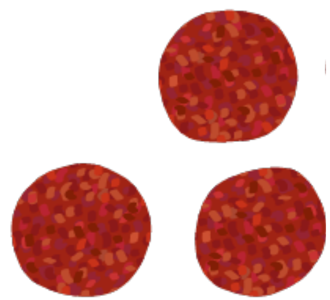
  - 2. Assign the items to each of the agents in the group.

Subject to a number of exogenous goals, axioms, metrics, and/or constraints.

Nicholas Mattei
NICTA and UNSW

Approval Ballots

Yes:   

No:  

- Every item appears once as either approved or not approved.
- We can approve as many or as few items as we like

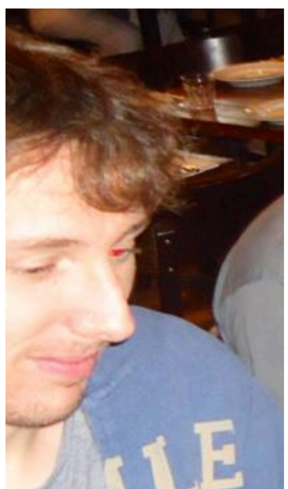
Nicholas Mattei
NICTA and UNSW

Approved Co-Workers

Haris Aziz

Serge Gaspers

Joachim Gudmundsson

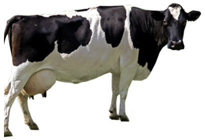


Simon Mackenzie

Toby Walsh

Ordering Lunch

Ordering Lunch



Ordering Lunch

X	X	X	X	X		
X		X	X	X		
X			X	X		
				X	X	X
				X	X	

Nicholas Mattei
NICTA and UNSW

Formally…

Given:

- An approval based voting rule R
- A set C of candidates with size m.
- A profile of n approval ballots $A = (A_1, \ldots A_n)$.
- A committee size k.

Formally…

Given:

- An approval based voting rule R
- A set C of candidates with size m.
- A profile of n approval ballots $A = (A_1, \dots A_n)$.
- A committee size k.

Question:

- ***Winner Determination:*** What is the winning set $W \subseteq C$?

Approval Voting

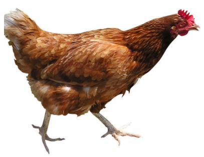
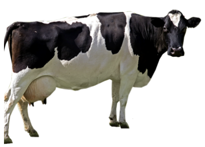
Given:

- An approval based voting rule R
- A set C of candidates with size m.
- A profile of n approval ballots $A = (A_1, \ldots A_n)$.
- A committee size k.

Question:

- ***Winner Determination:*** What is the winning set $W \subseteq C$?
- ***Approval Voting:*** The winner(s) are the k candidates receiving the most approvals across all submitted ballots.

Ordering Lunch

	X	X	X	X	X		
	X		X	X	X		
	X			X	X		
					X	X	X
					X	X	

Nicholas Mattei
NICTA and UNSW

Approval Voting Result

								AV
	X	X	X	X	X			5
	X		X	X	X			4
	X			X	X			3
					X	X	X	3
					X	X		2

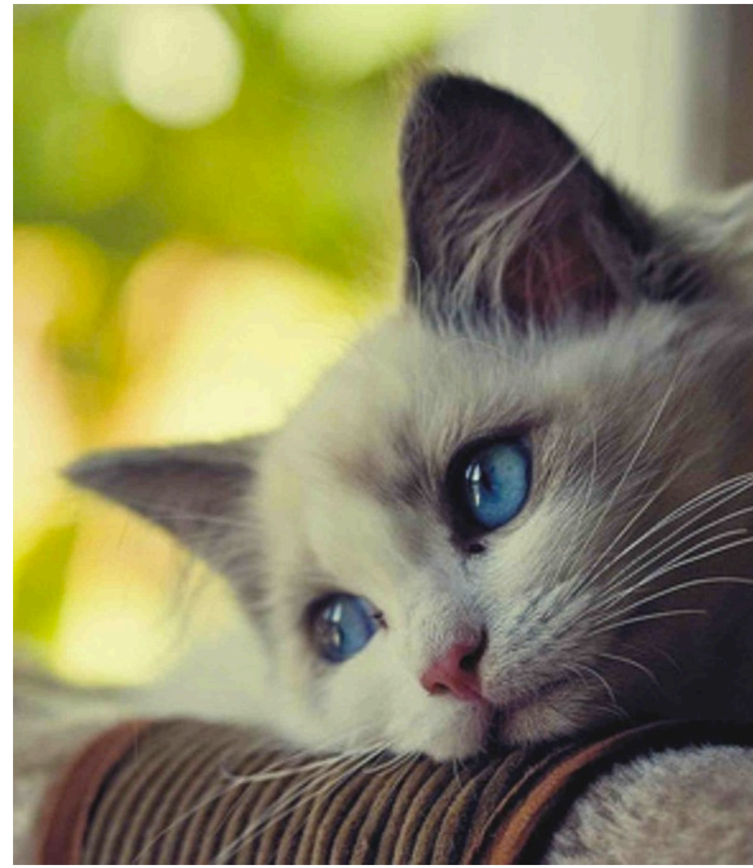
AV For Single Winner

- In the single winner setting AV is an excellent choice of voting rule!
- It satisfies many nice properties including its “simplicity, propensity to elect Condorcet winners (when they exist), its robustness to manipulation, and its monotonicity” – J.F. Laslier and M. R. Sanver eds., *Handbook on Approval Voting*.
- For dichotomous preferences AV is strategy proof and will elect the Condorcet winner.

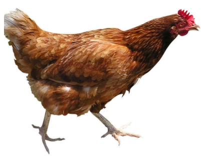
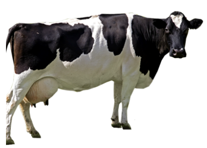
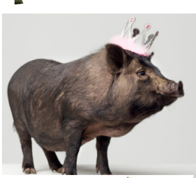
AV For **Multi-Winner**

- However, in the multi-winner case the merits of approval voting are “much less clear” – J.F. Laslier and M. R. Sanver eds., *Handbook on Approval Voting*.

- **None** of the nice properties are preserved for the multi-winner case and AV is prone to manipulation.



Approval Voting Result

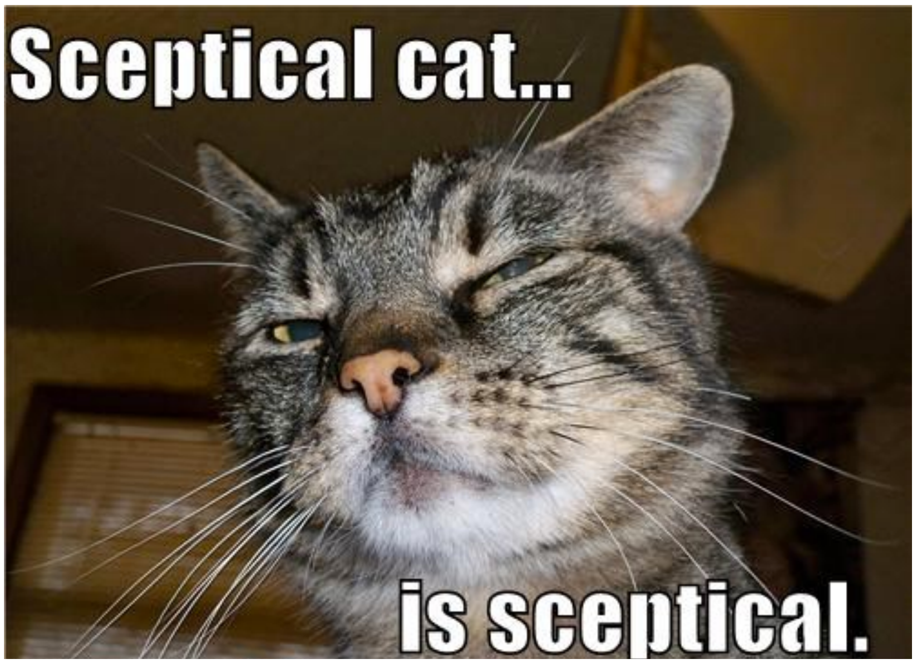
								AV
	X	X	X	X	X			5
	X		X	X	X			4
	X			X	X			3
					X	X	X	3
					X	X		2

Nicholas Mattei
NICTA and UNSW

Approval Voting Result, *k* = 2

								AV
	X	X	X	X	X			5
	X		X	X	X			4
	X			X	X			3
					X	X	X	3
					X	X		2

AV in Multi-Winner Elections



- Not only do we not have our nice properties – AV is majoritarian and does not represent the preferences of many of the agents.
- Other AV variants have more egalitarian objectives without the hard constraints imposed by rules like Chamberlin-Courant or Monroe.

Other Key Questions

Given:

- An approval based voting rule *R*
- A set *C* of candidates with size *m*.
- A profile of *n* approval ballots $A = (A_1, \ldots A_n)$.
- A committee size *k.*
- A number of agents *j* still to vote.
- A preferred candidate *p.*

Question:

- ***Winner Manipulation:*** Are there *j* additional approval ballots that make *p* a member of the winning set under *R* ?

Other Key Questions

Given:

- An approval based voting rule R
- A set C of candidates with size m.
- A profile of n approval ballots $A = (A_1, \dots A_n)$.
- A committee size k.
- A number of agents j still to vote.
- A set of preferred candidates $P \subseteq C$.

Question:

- ***Winning Set Manipulation:*** Are there j additional approval ballots that make P the winning set of candidates under R ?

Satisfaction Approval Voting

Given:

- An approval based voting rule *R*
- A set *C* of candidates with size *m*.
- A profile of *n* approval ballots $A = (A_1, \ldots A_n)$.
- A committee size *k.*

Satisfaction Approval Voting:

- A voter's **satisfaction score** is the fraction of approved candidates selected $(|W \cap Ai|) / |Ai|$.

- SAV selects the winning set of size *k* maximizing the sum of these scores for all voters.

Ordering Lunch

	X	X	X	X	X		
	X		X	X	X		
	X			X	X		
					X	X	X
					X	X	

Nicholas Mattei
NICTA and UNSW

SAV Scores

CB
CP
CF
CM
BP
BF
BM
PF
PM
FM

SAV Result

CB	2/3	1	1	2/3	2/5	0	0
CP	2/3	1	1/2	2/3	2/5	0	0
CF	1/3	1	1/2	1/3	2/5	1/2	1
CM	1/3	1	1/2	1/3	2/5	1/2	0
BP	2/3	0	1/2	2/3	2/5	0	0
BF	1/3	0	1/2	1/3	2/5	1/2	1
BM	1/3	0	1/2	1/3	2/5	1/2	0
PF	1/3	0	0	1/3	2/5	1/2	1
PM	1/3	0	0	1/3	2/5	1/2	0
FM	0	0	0	0	2/5	1	1

Nicholas Mattei
NICTA and UNSW

SAV Result

								SAV Score
CB	2/3	1	1	2/3	2/5	0	0	3.7333
CP	2/3	1	1/2	2/3	2/5	0	0	3.2333
CF	1/3	1	1/2	1/3	2/5	1/2	1	4.0666
CM	1/3	1	1/2	1/3	2/5	1/2	0	3.0666
BP	2/3	0	1/2	2/3	2/5	0	0	2.2333
BF	1/3	0	1/2	1/3	2/5	1/2	1	3.0666
BM	1/3	0	1/2	1/3	2/5	1/2	0	2.0666
PF	1/3	0	0	1/3	2/5	1/2	1	2.5666
PM	1/3	0	0	1/3	2/5	1/2	0	1.5666
FM	0	0	0	0	2/5	1	1	2.4

Nicholas Mattei
NICTA and UNSW

Satisfaction Approval Voting

Satisfaction Approval Voting:

- A voter's **satisfaction score** is the fraction of approved candidates selected $(|W \cap A_i|) / |A_i|$.

- SAV selects the winning set of size *k* maximizing the sum of these scores for all voters.

Key Results:

- ***Winner Determination*** is easy.
- ***Winner Manipulation*** is easy for dichotomous preferences.
- ***Winning Set Manipulation*** is NP-hard.

Proportional Approval Voting

Given:

- An approval based voting rule *R*
- A set *C* of candidates with size *m*.
- A profile of *n* approval ballots $A = (A_1, \ldots A_n)$.
- A committee size *k*.

Proportional Approval Voting:

- Let *j* be the number of approved candidates selected, a voter's satisfaction score is:

$$1 + 1/2 + 1/3 + \ldots + 1/j.$$

- *PAV* selects the set of size *k* with maximum score.

Ordering Lunch

X	X	X	X	X		
X		X	X	X		
X			X	X		
				X	X	X
				X	X	

Nicholas Mattei
NICTA and UNSW

PAV Scores

CB	1+1/2	1	1+1/2	1+1/2	1+1/2	0	0
CP	1+1/2	1	1	1+1/2	1+1/2	0	0
CF	1	1	1	1	1+1/2	1	1
CM	1	1	1	1	1+1/2	1	0
BP	1+1/2	0	1	1+1/2	1+1/2	0	0
BF	1	0	1	1	1+1/2	1	1
BM	1	0	1	1	1+1/2	1	0
PF	1	0	0	1	1+1/2	1	1
PM	1	0	0	1	1+1/2	1	0
FM	0	0	0	0	1+1/2	1+1/2	1

PAV Result

								PAV Score
CB	1+1/2	1	1+1/2	1+1/2	1+1/2	0	0	7
CP	1+1/2	1	1	1+1/2	1+1/2	0	0	6.5
CF	1	1	1	1	1+1/2	1	1	7.5
CM	1	1	1	1	1+1/2	1	0	6.5
BP	1+1/2	0	1	1+1/2	1+1/2	0	0	5.5
BF	1	0	1	1	1+1/2	1	1	6.5
BM	1	0	1	1	1+1/2	1	0	5.5
PF	1	0	0	1	1+1/2	1	1	5.5
PM	1	0	0	1	1+1/2	1	0	4.5
FM	0	0	0	0	1+1/2	1+1/2	1	4

Proportional Approval Voting

Proportional Approval Voting:

- Let j be the number of approved candidates selected, a voter's satisfaction score is:

$$1 + 1/2 + 1/3 + \ldots + 1/j.$$

- *PAV* selects the set of size k with maximum score.

Key Result:

- Winner determination for PAV is NP-hard (W[1]-hard).
- Holds for any decreasing score function.

Reweighted Approval Voting

Given:

- An approval based voting rule R
- A set C of candidates with size m.
- A profile of n approval ballots $A = (A_1, \ldots A_n)$.
- A committee size k.

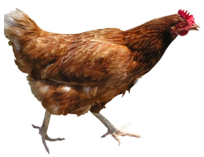
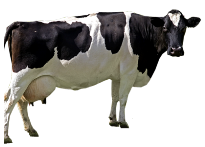
Reweighted Approval Voting:

- Multi-round: elect the candidate with the most approvals in each round.
- Reweight the voters by a fraction of the winners selected so far, $1 / (1 + |W \cap Ai|)$.

Ordering Lunch

X	X	X	X	X		
X		X	X	X		
X			X	X		
				X	X	X
				X	X	

RAV: Round 1

								RAV RD. 1
	1	1	1	1	1			5
	1		1	1	1			4
	1			1	1			3
					1	1	1	3
					1	1		2

Nicholas Mattei
NICTA and UNSW

RAV: Round 2

							RAV RD. 2
1	1	1	1	1			5
½		½	½	½			2
½			½	½			1.5
				½	1	1	2.5
				½	1		1.5

Reweighted Approval Voting

Reweighted Approval Voting:

- Multi-round: elect the candidate with the most approvals in each round.
- Reweight the voters by a fraction of the winners selected so far, $1 / (1 + |W \cap A_i|)$.

Key Results:

- ***Winner Determination*** is easy.
- ***Winner Manipulation*** is NP-hard!!
- ***Winning Set Manipulation*** NP-hard!!

Main Results

	Winner Determination	Winner Manipulation	Winning Set Manipulation
Approval Voting	in P	in P	in P
Satisfaction Approval Voting	in P	in P	NP-hard
Proportional Approval Voting	NP-Hard	coNP-hard	coNP-hard
Reweighted Approval Voting	in P	NP-hard	NP-hard

Nicholas Mattei
NICTA and UNSW

Conclusions

- Approval voting is a widely used and quite natural way for agents to express preference for a given set of items.
- When selecting multiple winners with approval ballots there are a number of rules with more attractive properties than traditional AV.
- We studied these rules and show RAV specifically is egalitarian, can compute winners efficiently, and is resistant to manipulation.

2 Threads of ComSoc

Analyze Results

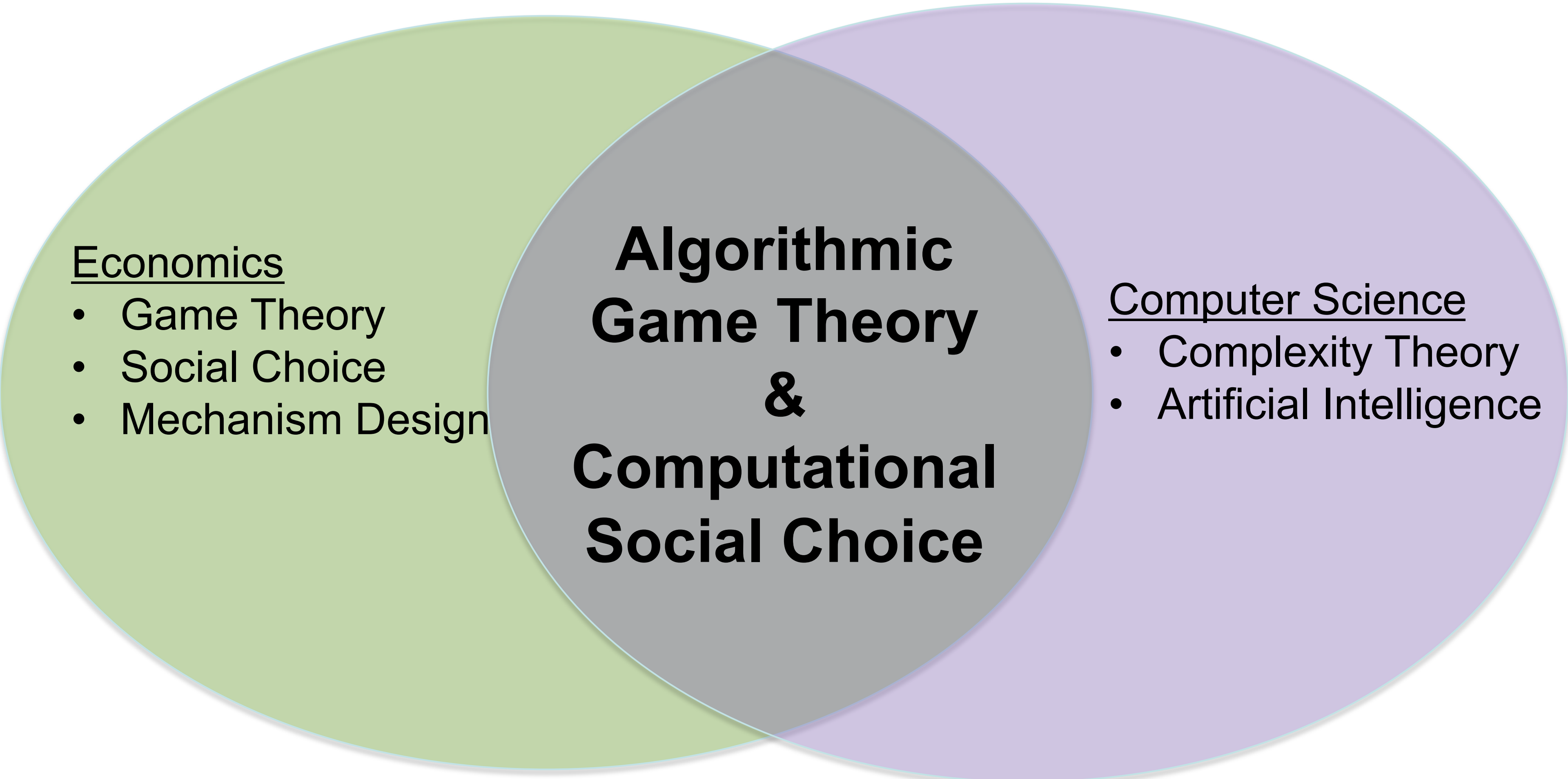
Analyze computational aspects of Social Choice. Many classic results in Social Choice Theory ignore the computational aspects of the theory.

↔

Import Ideas to AI

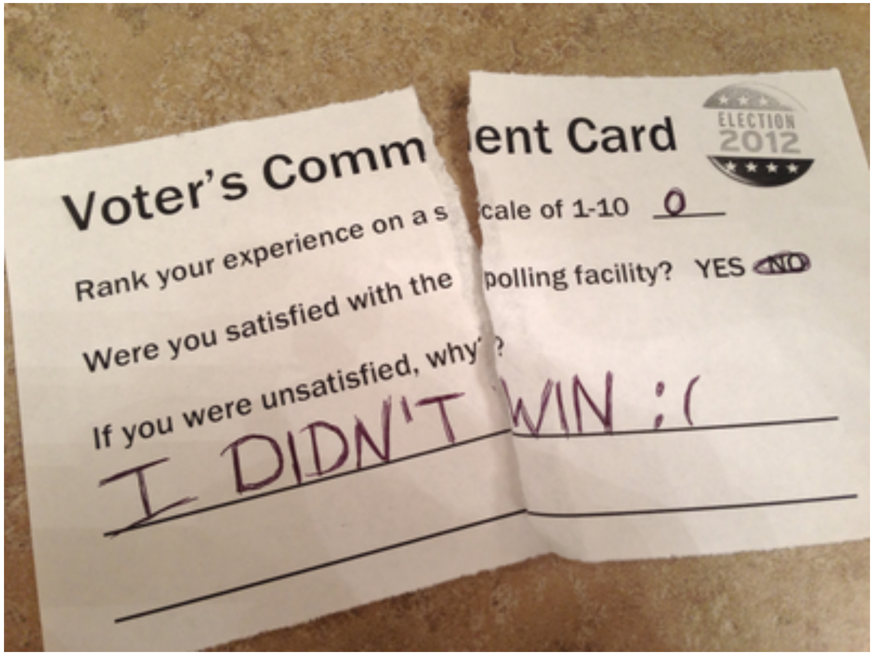
Implement ideas from Social Choice Theory in designing, implementing, and deploying systems across computer science including AI and multi-agent systems.

AGT and ComSoc



Thanks!

- Questions



- Comments

Nicholas Mattei
NICTA and UNSW