CPS 296.1
Voting and social choice

Vincent Conitzer
conitzer@cs.duke.edu
Voting over alternatives

- Voting rule (mechanism) determines winner based on votes
- Can vote over other things too
  - Where to go for dinner tonight, other joint plans, …
Voting (rank aggregation)

• Set of m candidates (aka. alternatives, outcomes)
• n voters; each voter ranks all the candidates
  – E.g., for set of candidates {a, b, c, d}, one possible vote is b > a > d > c
  – Submitted ranking is called a vote
• A voting rule takes as input a vector of votes (submitted by the voters), and as output produces either:
  – the winning candidate, or
  – an aggregate ranking of all candidates
• Can vote over just about anything
  – political representatives, award nominees, where to go for dinner tonight, joint plans, allocations of tasks/resources, …
  – Also can consider other applications: e.g., aggregating search engine’s rankings into a single ranking
Example voting rules

- **Scoring rules** are defined by a vector \((a_1, a_2, \ldots, a_m)\); being ranked ith in a vote gives the candidate \(a_i\) points
  - **Plurality** is defined by \((1, 0, 0, \ldots, 0)\) (winner is candidate that is ranked first most often)
  - **Veto** (or anti-plurality) is defined by \((1, 1, \ldots, 1, 0)\) (winner is candidate that is ranked last the least often)
  - **Borda** is defined by \((m-1, m-2, \ldots, 0)\)

- **Plurality with (2-candidate) runoff**: top two candidates in terms of plurality score proceed to runoff; whichever is ranked higher than the other by more voters, wins

- **Single Transferable Vote (STV, aka. Instant Runoff)**: candidate with lowest plurality score drops out; if you voted for that candidate, your vote transfers to the next (live) candidate on your list; repeat until one candidate remains

- Similar runoffs can be defined for rules other than plurality
Pairwise elections

two votes prefer Obama to McCain

two votes prefer Obama to Nader

two votes prefer Nader to McCain

> 

> 

> 

> 

> 

> 

>
Condorcet cycles

two votes prefer McCain to Obama

two votes prefer Obama to Nader

two votes prefer Nader to McCain

“weird” preferences
Voting rules based on pairwise elections

- **Copeland**: candidate gets two points for each pairwise election it wins, one point for each pairwise election it ties
- **Maximin (aka. Simpson)**: candidate whose worst pairwise result is the best wins
- **Slater**: create an overall ranking of the candidates that is inconsistent with as few pairwise elections as possible
  - NP-hard!
- **Cup/pairwise elimination**: pair candidates, losers of pairwise elections drop out, repeat
Even more voting rules…

- **Kemeny**: create an overall ranking of the candidates that has as few *disagreements* as possible (where a disagreement is with a vote on a pair of candidates)
  - NP-hard!
- **Bucklin**: start with $k=1$ and increase $k$ gradually until some candidate is among the top $k$ candidates in more than half the votes; that candidate wins
- **Approval** (not a ranking-based rule): every voter labels each candidate as approved or disapproved, candidate with the most approvals wins
- … how do we choose a rule from all of these rules?
- How do we know that there does not exist another, “perfect” rule?
- Let us look at some criteria that we would like our voting rule to satisfy
Condorcet criterion

• A candidate is the Condorcet winner if it wins all of its pairwise elections
• Does not always exist…
• … but the Condorcet criterion says that if it does exist, it should win

• Many rules do not satisfy this
• E.g. for plurality:
  – b > a > c > d
  – c > a > b > d
  – d > a > b > c
• a is the Condorcet winner, but it does not win under plurality
Majority criterion

• If a candidate is ranked first by most votes, that candidate should win
  – Relationship to Condorcet criterion?

• Some rules do not even satisfy this

• E.g. Borda:
  – $a > b > c > d > e$
  – $a > b > c > d > e$
  – $c > b > d > e > a$

• $a$ is the majority winner, but it does not win under Borda
Monotonicity criteria

• Informally, monotonicity means that “ranking a candidate higher should help that candidate,” but there are multiple nonequivalent definitions

• A weak monotonicity requirement: if
  – candidate w wins for the current votes,
  – we then improve the position of w in some of the votes and leave everything else the same,
  then w should still win.

• E.g., STV does not satisfy this:
  – 7 votes b > c > a
  – 7 votes a > b > c
  – 6 votes c > a > b

• c drops out first, its votes transfer to a, a wins
• But if 2 votes b > c > a change to a > b > c, b drops out first, its 5 votes transfer to c, and c wins
Monotonicity criteria…

• A **strong** monotonicity requirement: if
  – candidate w wins for the current votes,
  – we then change the votes in such a way that for each vote, if a candidate c was ranked below w originally, c is still ranked below w in the new vote

  then w should still win.

• Note the other candidates can jump around in the vote, as long as they don’t jump ahead of w

• None of our rules satisfy this
Independence of irrelevant alternatives

• Independence of irrelevant alternatives criterion: if
  – the rule ranks a above b for the current votes,
  – we then change the votes but do not change which is ahead between a and b in each vote
then a should still be ranked ahead of b.

• None of our rules satisfy this
Arrow’s impossibility theorem [1951]

• Suppose there are at least 3 candidates
• Then there exists no rule that is simultaneously:
  – **Pareto efficient** (if all votes rank a above b, then the rule ranks a above b),
  – **nondictatorial** (there does not exist a voter such that the rule simply always copies that voter’s ranking), and
  – independent of irrelevant alternatives
Muller-Satterthwaite impossibility theorem [1977]

- Suppose there are at least 3 candidates
- Then there exists no rule that simultaneously:
  - satisfies **unanimity** (if all votes rank a first, then a should win),
  - is **nondictatorial** (there does not exist a voter such that the rule simply always selects that voter’s first candidate as the winner), and
  - is **monotone** (in the strong sense).
Manipulability

• Sometimes, a voter is better off revealing her preferences insincerely, aka. manipulating

• E.g. plurality
  – Suppose a voter prefers \(a > b > c\)
  – Also suppose she knows that the other votes are
    • 2 times \(b > c > a\)
    • 2 times \(c > a > b\)
  – Voting truthfully will lead to a tie between \(b\) and \(c\)
  – She would be better off voting e.g. \(b > a > c\), guaranteeing \(b\) wins

• All our rules are (sometimes) manipulable
Gibbard-Satterthwaite impossibility theorem

• Suppose there are at least 3 candidates
• There exists no rule that is simultaneously:
  – onto (for every candidate, there are some votes that would make that candidate win),
  – nondictatorial, and
  – nonmanipulable
Single-peaked preferences

- Suppose candidates are ordered on a line
- Every voter prefers candidates that are closer to her most preferred candidate
- Let every voter report only her most preferred candidate ("peak")
- Choose the median voter’s peak as the winner
  - This will also be the Condorcet winner
- Nonmanipulable!

Impossibility results do not necessarily hold when the space of preferences is restricted
Pairwise election graphs

- **Pairwise election** between $a$ and $b$: compare how often $a$ is ranked above $b$ vs. how often $b$ is ranked above $a$

- **Graph representation**: edge from winner to loser (no edge if tie), weight = margin of victory

- E.g., for votes $a > b > c > d, c > a > d > b$ this gives

![Diagram](image-url)
Kemeny on pairwise election graphs

- Final ranking = acyclic tournament graph
  - Edge (a, b) means a ranked above b
  - Acyclic = no cycles, tournament = edge between every pair
- Kemeny ranking seeks to minimize the total weight of the inverted edges

pairwise election graph

Kemeny ranking

\[(b > d > c > a)\]
Slater on pairwise election graphs

- Final ranking = acyclic tournament graph
- Slater ranking seeks to minimize the number of inverted edges

Pairwise election graph

Slater ranking

(a > b > d > c)
An integer program for computing Kemeny/Slater rankings

\[ y(a, b) \] is 1 if \( a \) is ranked below \( b \), 0 otherwise
\[ w(a, b) \] is the weight on edge \( (a, b) \) (if it exists)

in the case of Slater, weights are always 1

minimize: \( \sum_{e \in E} w_e \ y_e \)
subject to:

for all \( a, b \in V \), \( y(a, b) + y(b, a) = 1 \)

for all \( a, b, c \in V \), \( y(a, b) + y(b, c) + y(c, a) \geq 1 \)
Some computational issues in social choice

- Sometimes computing the winner/aggregate ranking is hard
  - E.g. for Kemeny and Slater rules this is NP-hard
- For some rules (e.g., STV), computing a successful manipulation is NP-hard
  - Manipulation being hard is a good thing (circumventing Gibbard-Satterthwaite?)… But would like something stronger than NP-hardness
  - Also: work on the complexity of controlling the outcome of an election by influencing the list of candidates/schedule of the Cup rule/etc.

- Preference elicitation:
  - We may not want to force each voter to rank all candidates;
  - Rather, we can selectively query voters for parts of their ranking, according to some algorithm, to obtain a good aggregate outcome

- Combinatorial alternative spaces:
  - Suppose there are multiple interrelated issues that each need a decision
  - Exponentially sized alternative spaces

- Different models such as ranking webpages (pages “vote” on each other by linking)