Computational social choice
The easy-to-compute axiom

Lirong Xia

Rensselaer
Fall, 2016
Last class: linear programming and computation

- **Linear programming**
  - variables are positive real numbers
  - all constraints are linear, the objective is linear
  - in P

- **(Mixed) Integer programming**
  - (Some) All variables are integer
  - NP-hard

- **Basic computation**
  - Big O
  - Polynomial-time reduction
Today’s schedule

- Computational social choice: the easy-to-compute axiom
  - voting rules that can be computed in P
    - satisfies the axiom
  - Kemeny: a full proof of NP-hardness
  - IP formulation of Kemeny
Polynomial-time reduction: convert an instance of A to an instance of another decision problem B in polynomial-time

- so that answer to A is “yes” if and only if the answer to B is “yes”

If you can do this for all instances of A, then it proves that B is HARDER than A w.r.t. polynomial-time reduction
NP-hard and NP-complete problems

- **NP-hard problems**
  - the decision problems “harder” than any problem in NP
  - for any problem A in NP there exists a P-time reduction from A

- **NP-complete problems**
  - the decision problems in NP that are NP-hard
  - the “hardest” problems in NP

![Diagram showing the relationships between P, NP, and NP-hard and NP-complete problems]
How to prove a problem is \text{NP-hard}

- How to put an elephant in a fridge
  - Step 1. open the door
  - Step 2. put the elephant in
  - Step 3. close the door

- To prove a decision problem \text{B} is \text{NP-hard}
  - Step 1. find a problem \text{A} to reduce from
  - Step 2. prove that \text{A} is \text{NP-hard}
  - Step 3. find a \text{p}-time reduction from \text{A} to \text{B}

- To prove \text{B} is \text{NP-complete}
  - prove \text{B} is \text{NP-hard}
  - prove \text{B} is in \text{NP} (find a \text{p}-time verification for any correct answer)
The first known NP-complete problem

3SAT

- Input: a logical formula F in conjunction normal form (CNF) where each clause has exactly 3 literals
  - F = (x₁ ∨ x₂ ∨ x₃) ∧ (¬x₁ ∨ ¬x₃ ∨ x₄) ∧ (¬x₂ ∨ x₃ ∨ ¬x₄)
- Answer: Is F satisfiable?

3SAT is NP-complete (Cook-Levin theorem)
Kemeny’s rule

- Kendall tau distance
  - \( K(R,W) = \# \{ \text{different pairwise comparisons} \} \)
    \[
    K( b > c > a , a > b > c ) = 2
    \]
- Kemeny(\( P \))=arg\min_w K(\( P,W \))
  \[
  = \arg\min_w \sum_{V \in P} K(V,W)
  \]
- For single winner, choose the top-ranked alternative in Kemeny(\( P \))
Example

Profile $P$

<table>
<thead>
<tr>
<th></th>
<th>$a &gt; b &gt; c &gt; d$</th>
<th>$b &gt; a &gt; c &gt; d$</th>
<th>$d &gt; a &gt; b &gt; c$</th>
<th>$c &gt; d &gt; b &gt; a$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

WMG

$$K(P, a > b > c > d) = 0 + 1 + 2 \times 3 + 2 \times 5 = 17$$
For each linear order $W$ ($m!$ iter)
  • for each vote $R$ in $D$ ($n$ iter)
    • compute $K(R,W)$

Find $W*$ with the smallest total distance
  • $W*=\arg\min_W K(D,W)=\arg\min_W \sum_{R\in D} K(R,W)$
  • top-ranked alternative at $W*$ is the winner

Takes exponential $O(m!n)$ time!
Kemeny

- Ranking $W \rightarrow$ direct acyclic complete graph $G(W)$

  \[ a > b > c > d \]

- Given the WMG $G(P)$ of the input profile $P$

- $K(P,W) = \sum_{a \rightarrow b \in G(W)} \# \{ V \subseteq P : b > a \ in \ V \}$
  
  \[ = \sum_{a \rightarrow b \in G(W)} (n + w(b \rightarrow a))/2 \]
  
  \[ = nm(m-1)/4 + \sum_{a \rightarrow b \in G(W)} w(b \rightarrow a)/2 \]

- $\arg\min_W K(P,W) = \arg\min_W \sum_{a \rightarrow b \in G(W)} w(b \rightarrow a)$
  
  \[ = \arg\min_W \text{Total weight on inconsistent edges in WMG} \]
Example

\[ W = a \succ b \succ c \succ d \]

Profile \( P \):

Total weight on inconsistent edges between \( W \) and \( P \) is: 20
Kemeny is $\text{NP}$-hard to compute

- Reduction from feedback arc set (FAC)
  - Given a directed graph $G$ and a number $k$
  - does there exist a way to eliminate no more than $k$ edges to obtain an acyclic graph?

J. Bartholdi III, C. Tovey, M. Trick, Voting schemes for which it can be difficult to tell who won the election, Social Choice Welfare 6 (1989) 157–165.
Proof

The KendallDistance problem:

• Given a profile $P$ and a number $k$,
• Does there exist a ranking $W$ whose total Kendall distance is at most $k$?

\[ \text{Instance of FAC} \xrightarrow{\text{P-time}} \text{Instance of KendallDistance} \]

\[
\begin{array}{c}
G \\
\begin{array}{ccc}
a & b \\
\downarrow & \downarrow \\
d & c
\end{array}
\end{array}
\quad
\begin{array}{c}
\text{WMG}(P): \\
\begin{array}{ccc}
a & 2 & b \\
\downarrow & \downarrow & \downarrow \\
d & 2 & c
\end{array}
\end{array}
\]

\[ k \xrightarrow{\text{Yes}} k' = 2k + nm(m-1)/4 - 5 \]

Yes \quad No
For any edge $a \rightarrow b \in G$, define

$$P_{a \rightarrow b} = \{a > b > \text{others}, \text{Reverse(others)} > a > b\}$$

![Diagram of directed edges](image)

$WMG(P_{a \rightarrow b}) = \{d, c\}$

$P = \bigcup_{a \rightarrow b \in G} P_{a \rightarrow b}$
Vertex cover (VC):

- Given a undirected graph and a natural number $k$.
- Does there exists a set $S$ of no more than $k$ vertices so that every edge has an endpoint in $S$.

Example: Does there exists a vertex cover of 4?
VC is NP-complete

Given $F = (x_1 \lor x_2 \lor x_3) \land (\neg x_1 \lor \neg x_3 \lor x_4) \land (\neg x_2 \lor x_3 \lor \neg x_4)$

Does there exist a vertex cover of $4+2*3$?
Notes

- More details:

- A yes to B must correspond to a yes to A
  - if yes↔no then this proves coNP-hardness

- The best source for NP-complete problems
  - by M. R. Garey and D. S. Johnson
  - cited for >46k times [Google Scholar]
    - vs the “most cited book” *The Structure of Scientific Revolutions* 59K
A voting rule satisfies the easy-to-compute axiom if computing the winner can be done in polynomial time.

- P: easy to compute
- NP-hard: hard to compute
- assuming P≠NP
The winner determination problem

- Given: a voting rule \( r \)
- Input: a preference profile \( P \) and an alternative \( c \)
  - input size: \( nm \log m \)
- Output: is \( c \) the winner of \( r \) under \( P \)?
If following the description of $r$ the winner can be computed in $p$-time, then $r$ satisfies the easy-to-compute axiom.

Positional scoring rule

- For each alternative ($m$ iter)
  - for each vote in $D$ ($n$ iter)
    - find the position of $m$, find the score of this position
- Find the alternative with the largest score ($m$ iter)
- Total time $O(mn+m)=O(mn)$
Computing the weighted majority graph

For each pair of alternatives $c,d$ \((m(m-1)\text{ iter})\)

- let $k = 0$
- for each vote $V \in P$
  - if $c > d$ add 1 to the counter $k$
  - if $d > c$ subtract 1 from $k$
- the weight on the edge $c \rightarrow d$ is $k$
## Satisfiability of easy-to-compute

<table>
<thead>
<tr>
<th>Rule</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positional scoring</td>
<td></td>
</tr>
<tr>
<td>Plurality w/ runoff</td>
<td>P</td>
</tr>
<tr>
<td>STV</td>
<td></td>
</tr>
<tr>
<td>Copeland</td>
<td></td>
</tr>
<tr>
<td>Maximin</td>
<td></td>
</tr>
<tr>
<td>Ranked pairs</td>
<td></td>
</tr>
<tr>
<td>Kemeny</td>
<td>NP-hard</td>
</tr>
<tr>
<td>Slater</td>
<td></td>
</tr>
<tr>
<td>Dodgson</td>
<td></td>
</tr>
</tbody>
</table>
Solving Kemeny in practice

- For each pair of alternatives $a, b$ there is a binary variable $x_{ab}$
  - $x_{ab} = 1$ if $a > b$ in $W$
  - $x_{ab} = 0$ if $b > a$ in $W$
- $\max \quad \sum_{a,b} w(a \rightarrow b) x_{ab}$
  
  s.t. for all $a, b$, $x_{ab} + x_{ba} = 1$ \hspace{1cm} No edges in both directions

  for all $a, b, c$, $x_{ab} + x_{bc} + x_{ca} \leq 2$ \hspace{1cm} No cycle of 3 vertices

- Do we need to worry about cycles of $>3$ vertices? Next homework
Advanced computational techniques

- Approximation
- Randomization
- Fixed-parameter analysis