Extending Condorcet's criterion and searching for forbidden words

Jen Townsend

Scripps College,
Harvey Mudd College

October 24, 2009

Extending Condorcet Winners

Subelection winners

Subelection winners

Definition

A *Condorcet winner* wins every pairwise race they are in.

Subelection winners

Definition

A *Condorcet winner* wins every pairwise race they are in.

Extension:

Condorcet k-winner: a candidate that wins every k-wise race they are in, regardless of the (positional) procedure used.

Subelection winners

Definition

A *Condorcet winner* wins every pairwise race they are in.

Extension:
Condorcet k-winner: a candidate that wins every k-wise race they are in, regardless of the (positional) procedure used.

A profile has an associated "word" giving the list of k-winners.

Subelection winners

Definition

A *Condorcet winner* wins every pairwise race they are in.

Extension:

Condorcet k-winner: a candidate that wins every k-wise race they are in, regardless of the (positional) procedure used.

A profile has an associated "word" giving the list of k-winners.

Example

The profile

3	$A \succ B \succ C$
1	$C \succ B \succ A$

has a 2-winner: A.

Example

The profile

3	$A \succ B \succ C$
1	$C \succ B \succ A$

has a 2-winner: A.
But no 3-winner.

Example

The profile

3	$A \succ B \succ C$
1	$C \succ B \succ A$

has a 2-winner: A.
But no 3-winner.
The word associated with this profile is $(A*)$.

Possible words

For 3 candidate races the possible word forms are

- (**)
- (A*)
- (*A)
- (AB)
- (AA)

Possible words

For 3 candidate races the possible word forms are

- (**)
- (A*)
- (*A)
- (AB)
- (AA)

Which words correspond to profiles? What is the dictionary of winner words?

Computing k-winners

To check that c wins an election of n candidates under every weighting vector, it suffices to check that c wins for a basis of weighting vectors $(\underbrace{1, 0, \ldots, 0}_{k})$, $(1, 1, 0, \ldots, 0), \ldots,$ and $(1, 1, \ldots, 1, 0)$.

Computing k-winners

To check that c wins an election of n candidates under every weighting vector, it suffices to check that c wins for a basis of weighting vectors $(\underbrace{1, 0, \ldots, 0}_{k})$, $(1, 1, 0, \ldots, 0), \ldots,$ and $(1, 1, \ldots, 1, 0)$.

Example

1	$A \succ D \succ B$
1	$B \succ A \succ D$
1	$A \succ B \succ D$

a·r·t harveymuddcollege

Computing k-winners

To check that c wins an election of n candidates under every weighting vector, it suffices to check that c wins for a basis of weighting vectors $(\underbrace{1, 0, \ldots, 0}_{k}), (1, 1, 0, \ldots, 0), \ldots,$ and $(1, 1, \ldots, 1, 0)$.

Example

1	$A \succ D \succ B$
1	$B \succ A \succ D$
1	$A \succ B \succ D$

$$\begin{pmatrix} & 1^{st} & 2^{nd} & 3^{rd} \\ A: & 2 & 1 & 0 \\ B: & 1 & 1 & 1 \\ D: & 0 & 1 & 2 \end{pmatrix}$$

Computing k-winners

To check that c wins an election of n candidates under every weighting vector, it suffices to check that c wins for a basis of weighting vectors $(\underbrace{1, 0, \ldots, 0}_{k}), (1, 1, 0, \ldots, 0), \ldots,$ and $(1, 1, \ldots, 1, 0)$.

Example

1	$A \succ D \succ B$
1	$B \succ A \succ D$
1	$A \succ B \succ D$

$$\begin{pmatrix} & 1^{st} & 2^{nd} & 3^{rd} \\ A: & 2 & 1 & 0 \\ B: & 1 & 1 & 1 \\ D: & 0 & 1 & 2 \end{pmatrix}$$

A is the only candidate that wins under every weighting vector.

a·r·t harveymuddcollege

Building a dictionary

We know the word $(A * \cdots *)$ is realizable from the profile

3	$A \succ B \succ C$
1	$C \succ B \succ A$

We know the word $(A * \cdots *)$ is realizable from the profile

3	$A \succ B \succ C$
1	$C \succ B \succ A$

But the word $(* * \cdots * A)$ is also realizable.

The word $(* * \cdots * A)$ is realizable.

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}}$$

The word $(* * \cdots * A)$ is realizable.

$A \succ D \succ B \succ C$
$B \succ C \succ A \succ D$
$C \succ A \succ B \succ D$

Examining 2-way races for a 2-winner:

- $A \succ B$

The word $(* * \cdots * A)$ is realizable.

$A \succ D \succ B \succ C$
$B \succ C \succ A \succ D$
$C \succ A \succ B \succ D$

Examining 2-way races for a 2-winner:

- $A \succ B$
- $B \succ C$

The word $(* * \cdots * A)$ is realizable.

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}}$$

Examining 2-way races for a 2-winner:

- $A \succ B$
- $B \succ C$
- $C \succ A$

The word $(* * \cdots * A)$ is realizable.

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}}$$

Examining 2-way races for a 2-winner:

- $A \succ B$
- $B \succ C$
- $C \succ A$
- $A, B, C \succ D$

The word $(* * \cdots * A)$ is realizable.

$A \succ D \succ B \succ C$
$B \succ C \succ A \succ D$
$C \succ A \succ B \succ D$

Examining 2-way races for a 2-winner:

- $A \succ B$
- $B \succ C$
- $C \succ A$
- $A, B, C \succ D$

Conclusion: There is no 2-winner.

The word $(* * \cdots * A)$ is realizable.

Examining 3-way races for a 3-winner:

The word $(* * \cdots * A)$ is realizable.

Examining 3-way races for a 3-winner:

The subrace with candidates A, B, D is the subrace we looked at before

The word $(* * \cdots * A)$ is realizable.

Examining 3-way races for a 3-winner:

The subrace with candidates A, B, D is the subrace we looked at before

1	$A \succ D \succ B$
1	$B \succ A \succ D$
1	$A \succ B \succ D$

when we concluded that A won for every weighting vector
$\Rightarrow B$ and D are not 3-winners.

The word $(* * \cdots * A)$ is realizable.

Examining 3-way races for a 3-winner:

The subrace with candidates A, B, D is the subrace we looked at before

1	$A \succ D \succ B$
1	$B \succ A \succ D$
1	$A \succ B \succ D$

when we concluded that A won for every weighting vector
$\Rightarrow B$ and D are not 3-winners.
Similarly, the subraces (B, C, D) and (A, C, D) show respectively that C and A are not 3-winners.
Thus there is no 3-winner.

The word $(* * \cdots * A)$ is realizable.

Examining the 4-way race for a 4-winner:

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}}$$

The word $(* * \cdots * A)$ is realizable.

Examining the 4-way race for a 4-winner:

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}}$$

$$\begin{pmatrix} A: & 1 & 1 & 1 & 0 \\ B: & 1 & 0 & 2 & 0 \\ C: & 1 & 1 & 0 & 1 \\ D: & 0 & 1 & 0 & 2 \end{pmatrix}$$

The word $(* * \cdots * A)$ is realizable.

Examining the 4-way race for a 4-winner:

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}} \qquad \begin{pmatrix} A: & 1 & 1 & 1 & 0 \\ B: & 1 & 0 & 2 & 0 \\ C: & 1 & 1 & 0 & 1 \\ D: & 0 & 1 & 0 & 2 \end{pmatrix}$$

So A is the 4-winner.

The word $(* * \cdots * A)$ is realizable.

Examining the 4-way race for a 4-winner:

$$\boxed{\begin{array}{c} A \succ D \succ B \succ C \\ B \succ C \succ A \succ D \\ C \succ A \succ B \succ D \end{array}} \qquad \begin{pmatrix} A: & 1 & 1 & 1 & 0 \\ B: & 1 & 0 & 2 & 0 \\ C: & 1 & 1 & 0 & 1 \\ D: & 0 & 1 & 0 & 2 \end{pmatrix}$$

So A is the 4-winner.
This constructs the word $(* * A)$.

In general, the profile

$$c_1 \succ c_n \succ c_2 \succ c_3 \succ \cdots \succ c_{n-1}$$

$$c_2 \succ c_3 \succ \cdots \succ c_{n-1} \succ c_1 \succ c_n$$

$$\vdots$$

$$c_{n-1} \succ c_1 \succ c_2 \succ \cdots \succ c_{n-2} \succ c_n$$

generates the word $(* * * \cdots * \, c_1)$.

In general, the profile

$$c_1 \succ c_n \succ c_2 \succ c_3 \succ \cdots \succ c_{n-1}$$

$$c_2 \succ c_3 \succ \cdots \succ c_{n-1} \succ c_1 \succ c_n$$

$$\vdots$$

$$c_{n-1} \succ c_1 \succ c_2 \succ \cdots \succ c_{n-2} \succ c_n$$

generates the word $(* * * \cdots * c_1)$.

And adding candidates always voted last produces the word $(* * \cdots * c_1 \ldots c_1)$.

In general, the profile

$$c_1 \succ c_n \succ c_2 \succ c_3 \succ \cdots \succ c_{n-1}$$

$$c_2 \succ c_3 \succ \cdots \succ c_{n-1} \succ c_1 \succ c_n$$

$$\vdots$$

$$c_{n-1} \succ c_1 \succ c_2 \succ \cdots \succ c_{n-2} \succ c_n$$

generates the word $(* * * \cdots * c_1)$.

And adding candidates always voted last produces the word $(* * \cdots * c_1 \dots c_1)$.

Important idea: Cycles (and profiles in the k level kernel) are useful as initial building blocks of profiles with interesting words.

Not everything is possible.

A forbidden word: $(ab\ldots a\ldots)$

A forbidden word: $(ab\ldots a\ldots)$

a is a 2-winner, so $[a \succ b] > [b \succ a]$.

A forbidden word: $(ab \dots a \dots)$

a is a 2-winner, so $[a \succ b] > [b \succ a]$.

Under the Borda Count $(k-1, k-2, \dots, 2, 1, 0)$, the score of c_1 in an election of $(c_1, \dots c_k)$, is

$$\sum_{i=2}^{k} [c_1 \succ c_i].$$

A forbidden word: $(ab \ldots a \ldots)$

a is a 2-winner, so $[a \succ b] > [b \succ a]$.

Under the Borda Count $(k-1, k-2, \ldots, 2, 1, 0)$, the score of c_1 in an election of $(c_1, \ldots c_k)$, is

$$\sum_{i=2}^{k} [c_1 \succ c_i].$$

b is the 3-winner, so $[a \succ b] + [a \succ x] \leq [b \succ a] + [b \succ x]$ for every candidate x

A forbidden word: $(ab \ldots a \ldots)$

a is a 2-winner, so $[a \succ b] > [b \succ a]$.

Under the Borda Count $(k-1, k-2, \ldots, 2, 1, 0)$, the score of c_1 in an election of $(c_1, \ldots c_k)$, is

$$\sum_{i=2}^{k} [c_1 \succ c_i].$$

b is the 3-winner, so $[a \succ b] + [a \succ x] \leq [b \succ a] + [b \succ x]$ for every candidate x

$\Rightarrow [a \succ x] < [b \succ x]$ for every candidate x.

A forbidden word: $(ab \ldots a \ldots)$

Under the BC for the k subrace $(abc_3, \ldots, c_k)$, the score of a is less than the score of b:

$$[a \succ b] + [a \succ c_3] + \sum_{i=4}^{k} [a \succ c_i] < [b \succ a] + [b \succ c_3] + \sum_{i=4}^{k} [b \succ c_i]$$

A forbidden word: $(ab \ldots a \ldots)$

Under the BC for the k subrace $(abc_3, \ldots, c_k)$, the score of a is less than the score of b:

$$[a \succ b] + [a \succ c_3] + \sum_{i=4}^{k} [a \succ c_i] < [b \succ a] + [b \succ c_3] + \sum_{i=4}^{k} [b \succ c_i]$$

$b \succ a$ under the k level Borda Count $\Rightarrow$ a is not a $(k > 2)$-winner.

Important tools for forbidding words

The Borda Count was useful because it is has a formulation as the sum of pairwise results: it *extends* the weighting vector $(1, 0)$.

Important tools for forbidding words

The Borda Count was useful because it is has a formulation as the sum of pairwise results: it *extends* the weighting vector $(1, 0)$.

Definition (Saari)

A weighting vector $\mathbf{w}$ *extends* the weighting vector $\mathbf{v}$ on k candidates if $\mathbf{w}$ assigns each candidate the sum of the weights they receive under $\mathbf{v}$ over each of the $\binom{n}{k}$ k subraces.

Important tools for forbidding words

The Borda Count was useful because it is has a formulation as the sum of pairwise results: it *extends* the weighting vector $(1, 0)$.

Definition (Saari)

A weighting vector $\mathbf{w}$ *extends* the weighting vector $\mathbf{v}$ on k candidates if $\mathbf{w}$ assigns each candidate the sum of the weights they receive under $\mathbf{v}$ over each of the $\binom{n}{k}$ k subraces.

Only extensions of k weighting vectors restrict possible results of larger elections based on k subraces. Yet even these restrictions are weak.

Important tools for forbidding words

The Borda Count was useful because it is has a formulation as the sum of pairwise results: it *extends* the weighting vector $(1, 0)$.

Definition (Saari)

A weighting vector $\mathbf{w}$ *extends* the weighting vector $\mathbf{v}$ on k candidates if $\mathbf{w}$ assigns each candidate the sum of the weights they receive under $\mathbf{v}$ over each of the $\binom{n}{k}$ k subraces.

Only extensions of k weighting vectors restrict possible results of larger elections based on k subraces. Yet even these restrictions are weak.

Important idea: Extensions are useful for finding forbidden words.

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden
  The number of unique k winners in a word is limited.

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden
  The number of unique k winners in a word is limited.
- What is the dictionary of "k-loser" words?
- How exactly do these dictionaries relate to Don Saari's dictionaries of possible rankings under assigned level weighting vectors?
- Are there forbidden words that do not rely directly on extensions?

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden
  The number of unique k winners in a word is limited.
- What is the dictionary of "k-loser" words?
- How exactly do these dictionaries relate to Don Saari's dictionaries of possible rankings under assigned level weighting vectors?
- Are there forbidden words that do not rely directly on extensions?
- How many weighting vectors must be considered to get the restrictive behavior we observe?

a·r·t
harveymuddcollege

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden
  The number of unique k winners in a word is limited.
- What is the dictionary of "k-loser" words?
- How exactly do these dictionaries relate to Don Saari's dictionaries of possible rankings under assigned level weighting vectors?
- Are there forbidden words that do not rely directly on extensions?
- How many weighting vectors must be considered to get the restrictive behavior we observe?
- What do these restrictions mean?

Possibilities for Future Work

- Completing the dictionary: characterizing completely the forbidden subwords.

  Conjectures from Linear Programming:
  $(\ldots a \ldots b \ldots a \ldots)$ is forbidden
  The number of unique k winners in a word is limited.

- What is the dictionary of "k-loser" words?
- How exactly do these dictionaries relate to Don Saari's dictionaries of possible rankings under assigned level weighting vectors?
- Are there forbidden words that do not rely directly on extensions?
- How many weighting vectors must be considered to get the restrictive behavior we observe?
- What do these restrictions mean?

Take Home Message

Take Home Message

Voting is bizarre- generally it is odd to find restrictions to possible results.

In particular, looking at pairwise elections does not impose many restrictions on the overall results

Take Home Message

Voting is bizarre- generally it is odd to find restrictions to possible results.

In particular, looking at pairwise elections does not impose many restrictions on the overall results

But forbidden winning words offer a panoramic view of the restrictions built up from subraces.

Thanks!