Learning Blocs and Slates from Ranked-Choice Ballots

MOON DUCHIN, DAVID SHMOYS, AND KRISTOPHER TAPP

Given a ranked choice election for political office, a natural question is whether the election is polarized, with two or more different *blocs* of voters exhibiting divergent preferences among *slates* of candidates. In this paper we develop basic machinery for finding clusters of voters with similar preferences, and for finding clusters of candidates who are preferred by similar voters. This requires some new results about metrics on complete rankings, and on weak rankings (i.e., incomplete or allowing ties).

ACM Reference Format:
Moon Duchin, David Shmoys, and Kristopher Tapp. 2025. Learning Blocs and Slates from Ranked-Choice Ballots. 1, 1 (April 2025), 26 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Author's address: Moon Duchin, David Shmoys, and Kristopher Tapp.

XXXX-XXXX/2025/4-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the discourse around elections for political representation, conventional wisdom says that voting is becoming increasingly polarized, with voters dividing into groups with more extreme and incompatible positions and preferences. Besides partisanship, particular forms of polarization are sometimes of interest: for example, voting rights law in the United States requires a determination of whether elections exhibit signs of *racial polarization*, where members of minority racial/ethnic groups have cohesive preferences that differ starkly from those of the majority bloc.

The purpose of this paper is to propose various natural methods for identifying *blocs* and *slates* as clusters of similar voters and candidates, respectively. We then introduce algorithmic approaches to finding optimal clusters for a given observed election. After introducing simplified synthetic election profiles, we make heavy use of a real-world dataset of over 1000 ranked-choice local government elections from Scotland from 2012 to 2022, which feature five major parties and dozens of minor ones. Importantly, the definitions do not presume that slates are aligned with political parties—and the methods fully apply in non-partisan elections—so we are able to identify patterns in voter preference that might be otherwise missed by party-focused analysis.

By a *ballot*, we mean a complete or partial ranking of n candidates. (In places, we also consider the fuller setting of weak rankings, allowing arbitrary ties.) We can embed ballots into a coordinate space (so that a coordinate vector serves as a proxy for the ballot) and then apply standard coordinate clustering methods, like k-means and k-medians, to the resulting point clouds. In addition to heuristic approaches to clustering, such as Lloyd's algorithm for k-means, we also present integer programming formulations to compute globally optimal solutions with respect to several distinct formulations.

Our first coordinate system assigns a ballot to a point in $\mathbb{R}^n$ by taking each candidate's coordinate to be the reverse of their rankings, so that a candidate earns a score of $n - j$ from a voter who ranks them jth. Because this system of scoring is sometimes called (standard) *Borda points* in the context of voting, we call this the *Borda embedding*. Our second system of coordinates maps a ballot to $\{-1, 0, 1\}^{\binom{n}{2}} \subset \mathbb{R}^{\binom{n}{2}}$, where each pairwise comparison is recorded as favoring one of the two candidates (± 1) or as a tie (0). We call this the *head-to-head embedding*. The corresponding distances d_B and d_H are defined by L^1 distance between vectors. These recover the classical ranking metrics called Spearman footrule and Kendall tau (also known as swap distance, or bubble sort distance), respectively. We explore the extent to which Spearman footrule and Kendall tau fit into a unifying framework of graph metrics. From these two embeddings, across a portfolio of multiple optimization models and methods, we obtain quite similar clusters when applied to real-world Scottish ranked choice elections.

We present a sparse graph called the *ballot graph* whose nodes are partial rankings and whose path metric recovers d_H; furthermore, we offer a novel construction that adds edges to create a *shortcut ballot graph* that recovers d_B. Our descriptions of d_B and d_H as graph metrics suggests another approach to clustering voters by *modularity maximization*, borrowed from network theory.

Clustering on the preference profile amounts to grouping the voters. An alternative is to group the candidates who have similar patterns of voter support. We develop methods to partition the candidates and to measure how well a candidate partition fits observed voting preference data.

There are several main contributions:

- Coordinate embeddings for (partial) rankings that unite previous metric results, are well-adapted to the voting context, and allow the measurement of how far a ballot is from a slate (or subset of candidates).
- A novel graph realization for partial rankings, with results connecting it to classical metrics.

- Integer programming formulations that return exactly optimal results for multiple clustering objectives.
- New visualization ideas for ranked-choice elections.

All data and code used in this paper are available on GitHub.

1.1 Related work

Well-studied metrics on full permutations (i.e., in this setting, complete ballots) include the *Spearman footrule D* and the *Kendall tau distance I* [Kendall, 1970, Kendall and Gibbons, 1990]. These are essentially identical to our metrics: $d_H = I$ and $d_B = \frac{1}{2}D$. A classic result of Diaconis and Graham from the 1970s [Diaconis and Graham, 1977] established that these two metrics are mutually bounded, satisfying $I \leq D \leq 2I$.[1] This result was generalized to partial permutations of fixed length (that is, top-k rankings) by Fagin et al. in [Fagin et al., 2003] and then to partial ballots and positional ties by a larger set of co-authors in [Fagin et al., 2006]. In the notation of Diaconis and Graham, the Spearman footrule distance between two complete ballots equals the L^1 distance between the vector of the symbol ranks. To generalize this to incomplete ballots of fixed length, and later to arbitrary ballots with ties allowed, the papers of Fagin et al. employ a convention that assigns to a tied set of symbols the average of the ranks that would have been achieved by resolving the tie linearly.

Since our setting is that of voting theory, we make the simple observation that the L^1 difference of ranks is the same as the L^1 difference of (standard) *Borda scores*, recalling that the highest-ranked position among n symbols earns $n-1$ points, the next position earns $n-2$, and so on down to the last position earning 0 points, so that the Borda score assigned to position j in the rankings is $n-j$. Completing Borda scores for partial rankings by an averaging convention is one of the standard options in the literature, but other alternatives exist, such as assigning the lowest of the available point totals rather than the average, sometimes called the "pessimistic" convention. This convention, resulting in zero points for all unmentioned candidates, is the most commonly used in real-world voting [Kamwa, 2022] for strong reasons that are explained below. We focus on this "pessimistic" convention below.

The application of these metrics in the literature most closely aligned with our aim to identify clusters and centroids is what is known as the *rank aggregation* problem. In the context of rank aggregation, the center is required to be a full ranking (whose summed distance to the input rankings is minimized) and is viewed as "consensus" ranking [Fagin et al., 2006, Van Zuylen and Williamson, 2009], so that rank aggregation can be viewed as a voting method. For example, [Fishburn, 1977] explores which Arrow-style axioms are satisfied by various methods of rank aggregation. In our work, we find it productive to use the slightly different interpretive framework that views the center (the centroid, median, or medoid) of each cluster as a useful summary statistic of the voting behavior within that cluster, whether or not it is a ranking and whether or not the voters are concentrated nearby.

1.2 Motivating example: A ranked-choice election in Pentland Hills

We can run through several methods for finding blocs and slates with a high-level preview of an election from Edinburgh Ward 2 (Pentland Hills) in 2017—a literally random selection from the large Scottish dataset. There were seven candidates:

1=Graeme Bruce (Con), 2 = Emma Farthing (LD), 3 = Neil Gardiner (SNP), 4 = Ricky Henderson (Lab),

[1] Note that $I \leq D \leq 2I \implies d_H \leq 2d_B \leq 2d_H \implies d_B \leq d_H \leq 2d_B$. In fact, they further define a metric T that counts the number of (not necessarily adjacent) transpositions needed to convert one permutation to another, and show the stronger result $I + T \leq D \leq 2I$.

5 = Ernesta Noreikiene (SNP), 6= Susan Webber (Con), 7 = Evelyn Weston (Grn).

Here, Con is the Conservative and Unionist Party, LD is Liberal Democrats, SNP is the (ruling, centrist) Scottish National Party, Lab is Labour, and Grn is the Green Party. These five are the most important players in local government, together holding 1071 out of 1227 seats at the time of writing.[2]

First, it is worth just describing the preference profile. The 11,315 cast ballots in this Pentland Hills contest ranged from bullet votes (length 1, occurring 967 times) to complete ballots (length 7, occurring 1431 times). Overall, the average ballot length was 3.2. There are 18 ballots cast over 100 times each, led by the ballot $(1, 6)$ ranking the conservatives in alphabetical order and leaving the rest blank—that ballot was cast by 1342 voters, about 11.8% of the electorate. Together, the ones receiving at least 100 votes account for more than half of the votes, but there is an extremely long tail, with 660 ballots receiving exactly one vote. Overall, 1238 different types were cast out of 8659 valid possible ballots.

For illustrative purposes, consider a trial that takes the best result from 200 runs of Lloyd's algorithm for the 2-means problem on the Borda coordinate embeddings. (See §2.1 and §3.1 for details.) This partitions the voters into two clusters of roughly equal size (5586 in Cluster 1 and 5729 in Cluster 2) that are visualized in Figure 1.

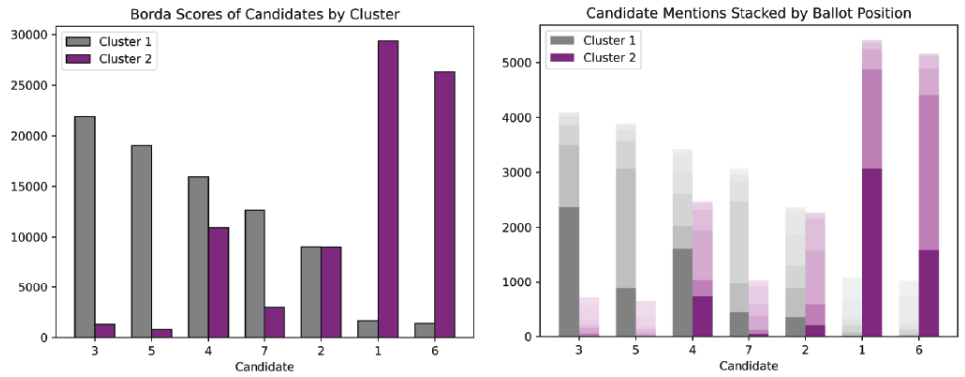

Fig. 1. Borda and mentions plots of the 2-means clusters. The candidates are listed in order of decreasing Borda score with respect to the first cluster. The total height of each bar in the mentions plot (right) is the number of ballots in the cluster that ranked the candidate in any position, with first-place mentions represented by the darkest shade and so on. Note that, up to scale, the heights of the bars in the Borda plot (left) can be read off as the components of the centroid of each cluster.

There are quite a few different modeling choices for clustering described below, like replacing Borda with the head-to-head embedding or doing L^1 distance minimization rather than L^2. For this Pentland Hills election, all options produce very similar clusters and nearly indistinguishable plots in the style of Figure 1. For all methods, one centroid is $(1, 6)$, while the other is 3 followed by 5 followed by 4 and/or 7.

Is $k = 2$ the right choice for the number of clusters in this Pentland Hills example? The choice $k = 3$ is also compelling because, as illustrated in Figure 2, the Lloyd's 3-means cluster assignments also look reasonable. In particular, they align closely with the party of the first-place vote; that is, the choice $k = 3$ best encodes the SNP–Lab–Con triangle that is visually striking in the MDS plot.

[2]Independent candidates, not to be confused with candidates for the Independence for Scotland Party (ISP), also receive significant support.

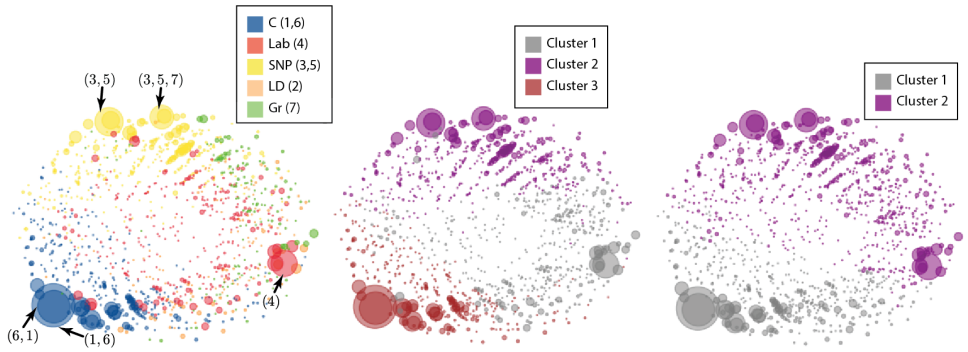

Fig. 2. A multi-dimensional scaling (MDS) plot of the ballots in Pentland Hills, with respect to the Borda embedding. Markers are sized by the ballot's weight (the number of voters who cast that ballot) and are colored by: party of first-place vote (left), a Lloyd's 3-means clustering assignment (middle), and a Lloyd's 2-means clustering assignment (right).

There are several common methods for identifying the "best" choice for the number of clusters in a dataset. For example, the method of comparing *silhouette scores* that compare within-cluster differences to between-cluster differences (see §4 and 5.2) identifies $k = 2$ as optimal for k-means clustering in Pentland Hills, but that score is not reasonably regarded as authoritative. Motivated by our graph-based descriptions of d_B and d_H, we can apply techniques from network theory that seek to maximize the modularity of a community partition of a graph (see §3.2). In Pentland Hills, the partition of maximum modularity has $k = 3$ pieces. Reassuringly, the modularity-maximizing partition that was identified by the heuristic algorithm in [Clauset et al., 2004] is almost identical to the Lloyd's 3-means partition that was visualized in Figure 2 (middle). So it is clear that from different perspectives one might legitimately prefer the 2-clustering or the 3-clustering.

Finally, we preview what happens when we sort the candidates rather than the voters. The optimal bipartition of the candidates in this election is

$$\mathcal{A} = \{1, 6\}, \quad \mathcal{B} = \{2, 3, 4, 5, 7\},$$

in the sense that this minimizes the average distance of the ballots to the closer of the two conditions $\mathcal{A} > \mathcal{B}$ and $\mathcal{B} > \mathcal{A}$, as explained in §3.4. In this particular case, 81% of total ballots turn out to be weakly consistent with one of those; that is, either no candidate from $\mathcal{A}$ is ranked lower than any candidate from $\mathcal{B}$ or the reverse.[3] The ballots can then be labeled according to the condition ($\mathcal{A} > \mathcal{B}$ or $\mathcal{B} > \mathcal{A}$) to which they are closest; in this election, the resulting "slate clusters" are very similar to the 2-means clusters illustrated in Figure 1. So although the many methods alluded to here give different specific answers, they all reinforce a generally consistent picture. We return to Pentland Hills in more detail in §5.1 below.

2 BALLOT GRAPHS AND COORDINATE EMBEDDINGS

Let Ω_n be the set of possible partial or complete ballots in an election with n candidates. The candidates can be denoted $\{A, B, \dots\}$ or $\{A_1, \dots, A_n\}$. We will adopt flexible notation for ballots, so that

$$(A, C, D, B), \quad A > C > D > B, \quad ACDB, \quad \text{and} \quad \begin{pmatrix} A \\ C \\ D \\ B \end{pmatrix}$$

[3]The convention is that any ranked candidate is considered to be higher than any unranked candidate, whereas two unranked candidates are considered to be tied.

all denote the same ballot. We will record partial ballots as having unmentioned candidates tied at the end of the ballot, so that $(A, B, -, -) = (A, B, \{C, D\})$.

2.1 Coordinate embeddings

Definition 2.1 (Borda and head-to-head distance). Suppose σ is a full ranking of the symbols $1, \ldots, n$, and consider that as a permutation with the convention that σ maps a candidate index to the rank of that candidate on the ballot, i.e., for candidates $A_1, \ldots, A_n$, we have that $\sigma(i)$ is the position in which the voter ranked A_i. In this notation, the ranking corresponding to σ is $\begin{pmatrix} \sigma^{-1}(1) \\ \vdots \\ \sigma^{-1}(n) \end{pmatrix}$.

Extend this to partial ballots by the rule that if k candidates are listed, and j is one of the unlisted candidates, then either $\sigma(j) = n$ (a *pessimistic* convention that amounts to ranking all unmentioned candidates in a tie for last), or by $\sigma(j)$ set to the *average* of the rankings that the $n - k$ unlisted candidates would have received if they had been ranked. Averaging maintains the total sum of ranks for any ballot, complete or partial, as $\sum_i \sigma(i) = \binom{n+1}{2}$, while the pessimistic convention lets that sum vary.

Define functions $\mathfrak{b} : \Omega_n \to \mathbb{R}^n$ and $\mathfrak{h} : \Omega_n \to \mathbb{R}^{\binom{n}{2}}$ by

$$\mathfrak{b}(\sigma)(i) = n - \sigma(i), \quad \mathfrak{h}(\sigma)(\{i, j\}) = \begin{cases} +1 & \sigma(i) > \sigma(j) \\ 0 & \sigma(i) = \sigma(j) \\ -1 & \sigma(i) < \sigma(j) \end{cases} \quad \text{(assuming } i < j\text{)}.$$

Then the Borda distance and head-to-head distance are defined by

$$d_B(\sigma, \tau) = \frac{1}{2}\|\mathfrak{b}(\sigma) - \mathfrak{b}(\tau)\|_1, \quad d_H(\sigma, \tau) = \frac{1}{2}\|\mathfrak{h}(\sigma) - \mathfrak{h}(\tau)\|_1.$$

Example 2.2. In Ω_4, the ballot AD is mapped to the Borda point $(3, .5, .5, 2)$. Since the 6 head-to-head comparisons are AB, AC, AD, BC, BD, CD, the ballot AD has the head-to-head image $(1, 1, 1, 0, -1, -1)$.

The head-to-head embedding uses an order on candidates, whether alphabetical or indexed, and the value +1 means that the first-indexed candidate is ranked higher.

For complete ballots, d_H is exactly the Kendall tau distance, counting neighbor swaps between permutations; for short ballots, d_H is the generalization of the Kendall tau metric discussed in [Fagin et al., 2006].

Because it is defined by adding the magnitudes of differences in rank, Borda distance clearly admits an interpretation as a *shift distance*, as illustrated in Figure 3.

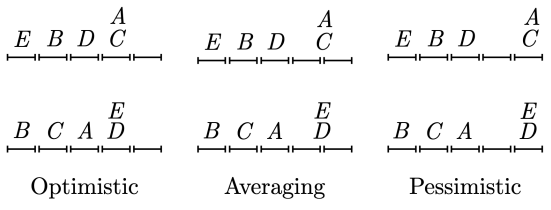

Fig. 3. For the ballots EBD and BCA in Ω_5, the Borda distance $d_B = 10$ with the averaging convention can be visualized as the sum of the magnitude of shifts. An exactly similar picture for the optimistic or pessimistic convention would give $d_{B\uparrow} = 8$ and $d_{B\downarrow} = 12$ for the same pair of ballots.

2.2 Ballot graphs

Any metric space with finitely many points can be given a graph realization in a trivial way: by adding all $\binom{n}{2}$ edges and giving the edge between x, y the length $d(x, y)$. For a graph realization to add value, it should be relatively sparse in terms of the edges needed to recover the metric as the path metric. We can achieve this with *ballot graphs* introduced here for the case of full and partial ballots and for d_H and $d_{B\downarrow}$. Though the number of nodes will grow as roughly $2n!$ for the graph on n candidates, the degree of each vertex will be only polynomial in n.

Definition 2.3 (Ballot graphs). The *ballot graph* $\mathcal{G}_n$ is a weighted, undirected graph with vertex set $V = \Omega_n$, adopting the convention that ballots of length $n-1$ are identified with their extension to complete ballots. There are two types of edges:

- Neighbor swaps: $(\sigma, \tau) \in E$ with weight one whenever $\sigma^{-1}\tau$ is a transposition $(i, i+1)$.
- Truncation and extension: $(\sigma, \tau) \in E$ with weight $\frac{n-k}{2}$ if they differ by an extension/truncation in the kth position and agree on the previous $k-1$ positions.

The *shortcut ballot graph* $\mathcal{G}_n^+$ adds a more general swap edge:

- General swaps: (σ, τ) is an edge with weight k whenever $\sigma^{-1}\tau$ is a transposition $(i, i+k)$.

In other words, any two candidates can be swapped, at a cost that is equal to the difference in positions.

The graphs are endowed with the path metric, so that two ballots have the distance equal to the lowest-weight path between them. (We identify nodes at distance zero to make this a true metric.)

Example 2.4 (Illustrating ballot graph construction).

- The ballot graph $\mathcal{G}_3$ is shown in Figure 4. For instance, the distance from ABC to ACB is one because they differ by a neighbor swap.
- In $\mathcal{G}_4$ (Figure 5), the distance from $ABCD$ to its reversal $DCBA$ equals 6. Note that this can either be realized through a path of 6 neighbor swaps or through a path that goes through bullet votes:

  $$ABCD \to ABC \to AB \to A \to AD \to DA \to D \to DC \to DCB \to DCBA.$$

  (That these two kinds of paths have equal length generalizes to connecting any complete ballot and its reversal in any $\mathcal{G}_n$.) In $\mathcal{G}_4^+$, the distance drops to 4 with the use of a shortcut edge $ABCD \to DBCA$.
- In $\mathcal{G}_6$, the edge from (A, B) to (A, B, C) has weight $\frac{3}{2}$.

2.3 Relating graphs and coordinate embeddings

Theorem 2.5 (Graphs match L^1 distances). *The head-to-head distance d_H is realized as the path metric on the basic ballot graph $\mathcal{G}_n$, while the "pessimistic" Borda distance $d_{B\downarrow}$ is realized as the path metric on the shortcut ballot graph $\mathcal{G}_n^+$.*

To prove this, it is convenient to introduce notions of strong and weak disagreement. We say that σ, τ have a strong disagreement on candidates i, j if one candidate is ranked higher on σ while the other is ranked higher on τ; a weak disagreement occurs when one ballot expresses a preference but the other ballot records a tie. Then for a pair of ballots σ, τ we can define $\mathrm{str}(\sigma, \tau)$ and $\mathrm{wk}(\sigma, \tau)$ as the total number of strong and weak disagreements, respectively. For example, if $\sigma = (A, B, C)$ and $\tau = (A, E)$ in Ω_5, then $\mathrm{str}(\sigma, \tau) = 2$ and $\mathrm{wk}(\sigma, \tau) = 4$. Since a strong disagreement causes a coordinate of the head-to-head vector to be $+1$ in one ballot and -1 in the other, it contributes 2 to the L^1 difference; similarly, a weak disagreement contributes 1. It is therefore immediate that

$$d_H = \mathrm{str} + \frac{1}{2}\,\mathrm{wk}$$

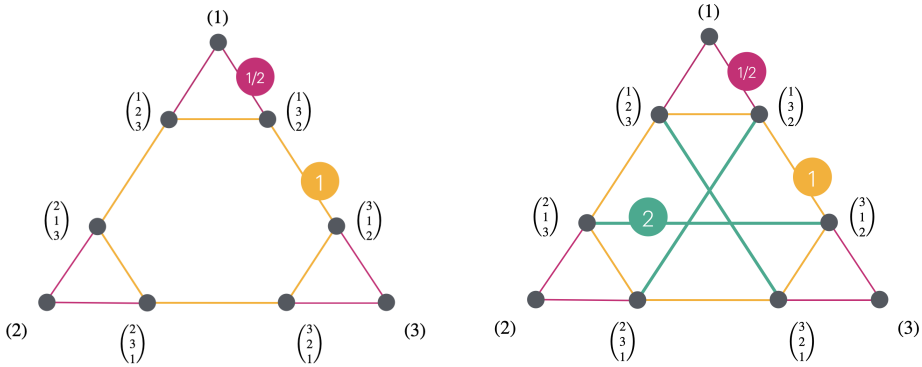

Fig. 4. The basic ballot graph $\mathcal{G}_3$ and shortcut ballot graph $\mathcal{G}_3^+$ on three candidates. Different edge styles in the graphs show truncation and swaps. Swaps of the first and last place correspond to new edges of length 2, providing shortcuts between nodes that would otherwise be 3 apart.

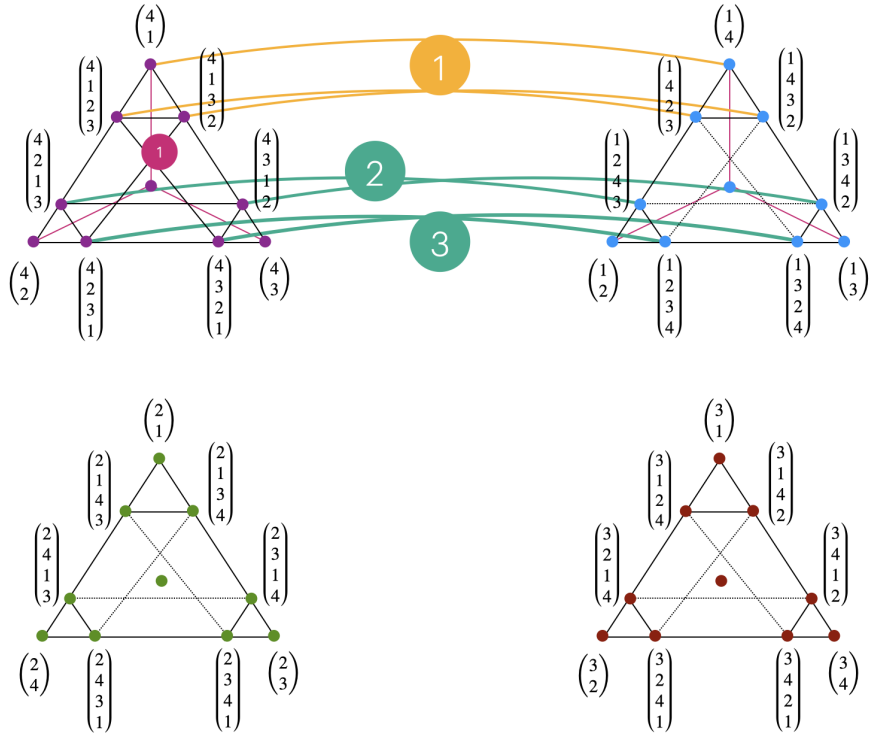

Fig. 5. An illustrative portion of the shortcut ballot graph $\mathcal{G}_4^+$, showing the connections between ballots headed by candidates 4 and 1. The construction of this picture shows the recursion $|\Omega_n| = n \cdot |\Omega_{n-1}| + n$.

for any pair of ballots. To prove that the graph distance matches d_H, it suffices to describe graph distance in terms of strong and weak disagreements.

Proof. For a pair of ballots that are adjacent in the basic ballot graph $\mathcal{G}_n$, it is straightforward to verify that the edge weight realizes the equality $w = \text{str} + \frac{1}{2}\,\text{wk}$ between endpoints of the edge. Now consider a geodesic in the graph visiting vertices $\sigma_1, \ldots, \sigma_r$ and note that the graph distance equals $\sum w(\sigma_i, \sigma_{i+1})$ by definition of geodesic. On the other hand, $d_H(\sigma_1, \sigma_r)$ is no more than the sum of d_H on the edges, because d_H is defined as an L^1 difference, so it satisfies the triangle inequality. Thus $d_H = \text{str} + \frac{1}{2}\,\text{wk}$ is less than or equal to the graph distance.

To show that this is an equality, we construct a path realizing equality between arbitrary ballots σ, τ. To do this, let σ' have the same initial part as σ, but add all unmentioned candidates to the end in the order in which they appear in τ; likewise, τ' starts like τ but adds candidates in the order of appearance in σ. (Note that if any candidates are unmentioned in both σ and τ, they remain tied at the bottom of σ' and τ'.) For the segment from σ' to τ', use any minimal-length sequence of neighbor swaps on the candidates in linear order. To see that this suffices, note that for two distinct ballots that are complete (or more generally that have the same candidate set), there must be at least one neighboring pair of candidates who have strong disagreement. Otherwise, every X ranked above Y by the first ballot would also be ranked above Y by the second ballot, forcing the ballots to be identical. This means that there is always a neighbor swap that reduces the strong disagreement count by 1, until that count is zero. Thus the length of the path $\sigma' \to \tau'$ is $\text{str}(\sigma, \tau)$. To compute the graph distance between σ and σ', let k be the number of candidates unmentioned in σ but linearly ranked in τ. Then successively adding those to the ranked part of σ has total edge weight $\frac{l+\cdots+(l+k)}{2} = \frac{1}{2}\left(\binom{k}{2} + kl\right)$, where l is the number of candidates missing from both σ and τ, while the number of weak disagreements involving those candidates is $\binom{k}{2} + kl$. Similarly between τ and τ' the total edge weight is one-half of the number of weak disagreements involving candidates unmentioned in τ but ranked in σ. Taken together, we've shown that the natural path $\sigma \to \sigma' \to \tau' \to \tau$ has total edge weight $\text{str}(\sigma, \tau) + \frac{1}{2}\,\text{wk}(\sigma, \tau)$, as desired.

The proof for pessimistic Borda is nearly the same, using the same σ', τ' as waypoints. There are no shortcuts possible between σ, σ' or between τ, τ', which are connected with the same sequence of extension/truncation moves as before. However, crucially, these moves are $d_{B\downarrow}$-geodesic but may not be d_B-geodesic. This is because in the pessimistic convention, moving a candidate from σ out of a tie at the end does not change the position of the remaining unmentioned candidates, so it moves σ closer to τ. But in the averaging convention, the extension move can improve the position of one candidate but worsen the position of others. Continuing with $d_{B\downarrow}$, it now suffices to find the shortest path σ', τ'. These have the same unmentioned candidates at the end, so it suffices to consider the number of shortcut moves needed between complete ballots. The difference of two Borda vectors records how far each candidate must shift (and in which direction), in order to transform one ballot to another. The sum of absolute values of this difference vector is d_B (which agrees with $d_{B\downarrow}$ on complete ballots). It suffices to give moves starting at σ' that reduce the distance by an amount equal to their length. To do this, consider the coordinate with positive difference closest to the top of the ballot. If the previous coordinate is negative, a neighbor swap makes geodesic progress. Otherwise, the previous coordinate is zero. In that case, it is part of a block of zeros preceded by a negative coordinate. In that case, the shortcut move that swaps the candidates on either side of the zero block makes geodesic progress. To visualize this, consider the difference vector $(\rightarrow, \rightarrow, \square, \square, \square, \leftarrow, \leftarrow, \square, \leftarrow)$, where red symbols show the necessary shortcut move, which in this case has length 4. □

Corollary 2.6. *$d_H(\sigma, \tau)$ equals the average (over all extensions of σ and τ to complete ballots) of the number of adjacent swaps needed to convert σ into τ.*

Proof. If σ and τ are complete, then the result is a standard interpretation of the Kendall tau distance. If σ and τ are not complete, then randomly completing them resolves each weak disagreement into either an agreement or a strong disagreement, with equal likelihood. Thus, the expected number of strong disagreements between their completions equals $\mathrm{str} + \frac{1}{2}\,\mathrm{wk}$. □

Theorem 2.7.

$$\frac{1}{2}\,\mathrm{str} + \frac{1}{2}\,\mathrm{wk} < d_{B\downarrow} \leq \mathrm{str} + \frac{1}{2}\,\mathrm{wk} = d_H$$

Proof. The above proof established that

$$d_{B_\downarrow}(\sigma, \tau) = \underbrace{d_{B_\downarrow}(\sigma, \sigma') + d_{B_\downarrow}(\tau, \tau')}_{=\frac{1}{2}\,\mathrm{wk}(\sigma,\tau)} + d_{B_\downarrow}(\sigma', \tau'),$$

so it remains to show that $d_{B_\downarrow}(\sigma', \tau')$ (which is the $\mathcal{G}_n^+$-path distance between σ' and τ') is more than half $\mathrm{str}(\sigma', \tau')$ (which is the $\mathcal{G}_n$-path distance). This follows from the fact that the weight of each edge of $\mathcal{G}_n^+$ is more than half the $\mathcal{G}_n$-distance between its endpoints. □

Corollary 2.8. $d_{B\downarrow} < d_H \leq 2d_{B\downarrow}$.

We note that [Fagin et al., 2006] defines a family of metrics they denote by $K^{(p)}$ as $\mathrm{str} + p \cdot \mathrm{wk}$, so that $d_H = K^{(1/2)}$. They prove that these are true metrics on the space of ballots with arbitrary ties when $p \in [1/2, 1]$, and that they are "near-metrics" (satisfying a relaxed triangle inequality) for $p \in (0, 1/2)$. However, d_B is not a multiple of any $K^{(p)}$. To see this, note that complete ballots have no weak disagreements. Comparing $\sigma = ABCDE$ to $\tau = EBCDA$ and $\pi = DBCAE$, which each differ from σ by one shortcut move of a different length, shows that d_B is not a fixed multiple of str, so $d_B \neq a \cdot \mathrm{str} + b \cdot \mathrm{wk}$ for any a, b.

Corollary 2.9. *For any pair $\sigma, \tau \in \Omega_n$ of distinct (possibly partial) ballots, $\dfrac{d_H(\sigma, \tau)}{d_{B\downarrow}(\sigma, \tau)} \in [1, 2)$, and these bounds are optimal.*

Proof. Every shortcut edge of length k represents two candidates jumping over $k-1$ stationary candidates located in between them; with neighbor swaps, this can be realized by moving the lower candidate k spots and the higher candidate $k-1$ spots, for $2k-1$ overall. Thus each shortcut speeds up a path segment by a factor less than two, proving that the bounds hold.

If $\sigma \in \Omega_n$ is a complete ballot, let $\hat{\sigma}$ be its reversal, with candidates listed in opposite order. We have $d_H(\sigma, \hat{\sigma}) = \binom{n}{2}$ because every pair will have strong disagreement. On the other hand, $d_{B\downarrow}(\sigma, \hat{\sigma}) = d_B(\sigma, \hat{\sigma}) = (n-1)(n+1)$, so we have a sequence with $\frac{d_H(\sigma,\hat{\sigma})}{d_B(\sigma,\hat{\sigma})} \to 2$. On the other hand, any pair of ballots that can be connected with neighbor swaps alone realizes $d_H = d_B$. □

In the previous proof, we noted that a shortcut edge of length k replaces a path of length $2k-1$ in the basic ballot graph. This means its distance savings is $k-1$. Thus if the shortcut edges used in a geodesic have lengths $k_1, k_2, \ldots, k_r$, we have the exact accounting $d_H - d_{B\downarrow} = (\sum k_i) - r$.

For complete ballots in S_n, Diaconis and Graham show that the expected Kendall tau distance to the identity is $I = n^2/4$ and the expected Spearman footrule distance is $D = n^2/3$ [Diaconis and Graham, 1977, Table 1]. Since $d_H = I$ and $d_B = \frac{1}{2}D$, this gives that the average value of d_H is 3/2 the average value of d_B.

As a final remark on natural ways to metrize the space of ballots, note that we can regard a ballot with ties as a collection of all complete ballots obtained by linear resolution of the ties. Rather

than computing the average as in Corollary 2.6, one could measure the maximum or the minimum distance between points in the resolution of one ballot to those representing the other. In [Critchlow, 1985], Critchlow also studied the Hausdorff distance between these point clouds $\bar{\sigma}$ and $\bar{\tau}$, namely the smallest ϵ so that the ϵ-neighborhood of $\bar{\sigma}$ contains $\bar{\tau}$, and likewise the ϵ-neighborhood of $\bar{\tau}$ contains $\bar{\sigma}$. Fagin et al. [Fagin et al., 2006] proved the following characterization. Let σ, τ be partial ballots and define $\mathrm{wk}(\sigma \to \tau)$ to be the number of candidates mentioned in τ but not σ while $\mathrm{wk}(\tau \to \sigma)$ is the reverse, so that $\mathrm{wk}(\sigma, \tau)$ is the sum of these two asymmetric values, then

$$d_{\mathrm{Haus}}(\sigma, \tau) = \mathrm{str}(\sigma, \tau) + \max\left(\mathrm{wk}(\sigma \to \tau), \mathrm{wk}(\tau \to \sigma)\right).$$

Because the max of two numbers is no greater than their sum but no less than half of their sum, we get $d_H \leq d_{\mathrm{Haus}} \leq 2d_B$.

3 CLUSTERING METHODS

3.1 Clustering cast ballots by distance optimization

The primary focus of our clustering work is to use the embeddings to cluster cast ballots; this has the advantage of producing summaries of the clusters that can be interpreted as identifying preferred candidates for each.

Once ballots are given coordinate representations by Borda ($\mathfrak{b}$ or $\mathfrak{b}_{\downarrow}$) or head-to-head ($\mathfrak{h}$) embeddings, we can employ standard clustering models and methods to group them into types. Though there are a great many clustering options, we will focus on three optimization models.

(1) Continuous k-means: minimize summed L^2 squared-distances to k centroids in $\mathbb{R}^n$ or $\mathbb{R}^{\binom{n}{2}}$;
(2) Continuous k-medians: minimize summed L^1 distance to k centroids in $\mathbb{R}^n$ or $\mathbb{R}^{\binom{n}{2}}$;
(3) Discrete k-medoids: minimize summed L^1 distance to k centroids in the dataset of points with positive weight in $\mathfrak{b}(\Omega_n)$ or $\mathfrak{h}(\Omega_n)$.

Heuristic methods are easily run to find approximate optima: we can use Lloyd's algorithm for k-means, or PAM (Partitioning Around Medoids) clustering for discrete k-medoids. While problem (1) admits the most efficient heuristic solution, problems (2) and (3) accommodate the L^1 distance, which is desirable due to its natural interpretations discussed above. Note that the cluster centroids in problem (1) with the Borda embedding can be visualized via bar charts like Figure 1 (left).

Provably optimal solutions for problems (2) and (3) can be computed via **integer programming** formulations with a solver such as Gurobi. For problem (3), the standard integer programming formulation uses binary variables y_i that indicate if a given data point i is selected as one of the centroids (i.e., medians), as well as assignment binary variables x_{ij} to indicate if data point j is within the cluster with median i.

For problem (2), the centroid of each cluster takes value, for each dimension in the embedding, equal to the median value of the points in that cluster. This allows an integer programming formulation in which binary variables z_{cvd} indicate that cluster c has median value v in dimension d, in addition to binary assignment variables x_{cj} that indicate to which cluster c each data point j is assigned. Again, the resulting integer programs can be easily solved by Gurobi. One consequence of this formulation is that it facilitates the solution of an alternate variant of problem (3) as well - when points corresponding to ballots that received no votes may also be selected as the centroid of a cluster. Details of the IP formulations can be found in Appendix §B.

3.2 Clustering in the ballot graph by network modularity

As a secondary consideration, the ballot graphs allow us to attempt *modularity clustering*, a method designed for community detection in social networks, which in principle has the advantage that the optimal number of clusters is not pre-determined but rather emerges from the calculation.

Given an undirected graph or multigraph G and a partition $\mathcal{P}$ of its vertices, Newman in [Newman, 2006] surveyed a family of measurements said to measure the *modularity* of the partition $\mathcal{P}$. Modularity roughly measures the share of the edges of G that lie within clusters compared to the share that would be expected to lie within the groups under a null model in which the same number of edges were redistributed randomly in G, in a manner that preserves the vertex degrees. The modularity of a partition can be positive or negative and its magnitude is at most 1. A high positive modularity indicates a natural community structure in the sense that more edges lie within clusters than in the null condition.

Modularity is designed for networks where edges represent relationships between nodes, but in our setting it is node weights that carry information about an election. So for a simple application of modularity, we pushed node weights and edge lengths to edge weights by forming *complete* graph $\tilde{\mathcal{G}}$ on the ballots in Ω_n of positive weight (the cast ballots). For each pair of cast ballots $\sigma, \tau \in \Omega_n$, we define the weight $\tilde{w}(e)$ of the edge $e = (\sigma, \tau)$ as $\tilde{w}(e) = \frac{w(\sigma)\, w(\tau)}{d_{B\downarrow}(\sigma,\tau)}$, where w denotes the number of times that the ballot was cast.

Many heuristic methods have been proposed to approximately identify the modularity-maximizing partition of a graph, including an elegant linear algebra based method [Newman, 2006]. We used the fast greedy algorithm described [Clauset et al., 2004] (as implemented in NetworkX) because of its speed and Python integration.

3.3 Clustering candidates by similarity

Another perspective on the problem is to introduce notions of similarity and dissimilarity on the candidates based on the preference profile. While there are several reasonable measurements that could be used here, a natural choice is to regard candidates as more similar when they tend to be ranked in closer positions by votes.

Definition 3.1 (Candidate dissimilarity by rank difference). For any pair i, j of candidates,

- $d(i, j)$ averages $|\sigma(i) - \sigma(j)|$ over the ballots σ.
- $\overline{d}(i, j)$ averages the average $|\overline{\sigma}(i) - \overline{\sigma}(j)|$ over the completions, then over the ballots.

Proposition 3.2. *d and $\overline{d}$ are metrics on the set of mentioned candidates.*

Proof. First, note that the contributions of a given ballot σ to these two measurements agree unless i, j are distinct candidates both missing from σ, in which case $|\sigma(i) - \sigma(j)| = 0$ while the average value of $|\overline{\sigma}(i) - \overline{\sigma}(j)|$ over the completions $\overline{\sigma}$ of σ equals $\frac{1}{3}(m+1)$, where m is the number of candidates missing from σ. Thus the restriction to mentioned candidates ensures a positive-definite metric. It is straightforward to see that d and $\overline{d}$ are symmetric and that d satisfies the triangle inequality. To see that $\overline{d}$ also satisfies the triangle inequality, let S denote the set of all completions of all cast ballots with the following redundancy convention. If the ballot σ from a given voter has length k, then the $(n-k)!$ distinct completions of σ will each appear $\frac{n!}{(n-k)!}$ times in S. This convention ensures that each voter is represented the same number (namely $n!$) times in S. Notice that $\overline{d}(i, j)$ equals the average over all $\overline{\sigma} \in S$ of $|\overline{\sigma}(i) - \overline{\sigma}(j)|$, from which the triangle inequality follows. □

3.4 Optimizing candidate bipartitions

Finally, consider optimizing a partition of the candidates into two sets, $\mathcal{A}$ and $\mathcal{B}$ (called slates), so that the ballots are most starkly divided into those preferring $\mathcal{A}$ over $\mathcal{B}$ and those preferring $\mathcal{B}$ over $\mathcal{A}$. We begin with notation.

Definition 3.3. Let σ be a ballot and let $\mathcal{A} \sqcup \mathcal{B}$ be a bipartition of the candidates into $\mathcal{A} = \{A_1, ..., A_s\}$ and $\mathcal{B} = \{B_1, ..., B_t\}$.

- $A_i >_\sigma B_j$ means that σ ranks A_i above B_j, while $A_i =_\sigma B_j$ means that neither candidate is ranked.
- $\mathcal{A} >_\sigma \mathcal{B}$ (respectively, $\mathcal{A} \geq_\sigma \mathcal{B}$) means that $A_i >_\sigma B_j$ (respectively, $A_i \geq_\sigma B_j$) for all $i \in \{1, ..., s\}$ and $j \in \{1, ..., t\}$.

Observe that if σ ranks at least one B-candidate, then the conditions $\mathcal{A} >_\sigma \mathcal{B}$ and $\mathcal{A} \geq_\sigma \mathcal{B}$ are equivalent and mean that σ begins with *all* of the A-candidates. By contrast, if σ ranks no B-candidates, then $\mathcal{A} \geq_\sigma \mathcal{B}$ is automatically satisfied, whereas $\mathcal{A} >_\sigma \mathcal{B}$ requires that σ ranks *all* of the A-candidates.

Definition 3.4 (Distance to slate). The distance from a ballot to a slate can now be defined as follows.

$$\text{dist}(\sigma, \mathcal{A}) := \frac{|\{i, j \mid B_j >_\sigma A_i\}| + \frac{1}{2} \cdot |\{i, j \mid B_j =_\sigma A_i\}|}{s \cdot t}.$$

In other words, $\text{dist}(\sigma, \mathcal{A})$ equals the portion of A_i-vs-B_j comparisons that σ makes in favor of the B-candidate, letting ties count with weight $\frac{1}{2}$. Notice that $\text{dist}(\sigma, \mathcal{A}) = 0$ if and only if $\mathcal{A} >_\sigma \mathcal{B}$.[4]

A natural alternative is based on strict preference.

$$\overline{\text{dist}}(\sigma, \mathcal{A}) := \frac{|\{i, j \mid B_j >_\sigma A_i\}|}{|\{i, j \mid B_j >_\sigma A_i\}| + |\{i, j \mid B_j <_\sigma A_i\}|}.$$

We see that $\overline{\text{dist}}(\sigma, \mathcal{A}) = 0$ if and only if $\mathcal{A} \geq_\sigma \mathcal{B}$. Also note that the definitions of dist and $\overline{\text{dist}}$ are normalized to ensure that for all σ,

$$\text{dist}(\sigma, \mathcal{A}) + \text{dist}(\sigma, \mathcal{B}) = 1, \quad \text{and} \quad \overline{\text{dist}}(\sigma, \mathcal{A}) + \overline{\text{dist}}(\sigma, \mathcal{B}) = 1.$$

We now describe our slate-clustering algorithm, which has two steps:

(1) For each bipartition $\mathcal{A} \sqcup \mathcal{B}$ of the candidates, calculate the sum (over all the ballots) of the distance to the condition $\mathcal{A} > \mathcal{B}$ or $\mathcal{B} > \mathcal{A}$, whichever is closest.
(2) Select the clusters induced by the bipartition for which this summed distance is minimized.

The method described here is specific to $k = 2$ clusters and can be quite effective for detecting certain classic polarization scenarios, but would fail to detect, for example, if all voters had the same first-choice candidate but disagreed starkly about the remaining candidates.

4 VALIDATION ON SYNTHETIC ELECTIONS

One challenge for testing clustering methods on real-world data is the lack of ground truth about whether voter behavior was polarized—not finding a good nontrivial clustering can indicate a poor clustering method or an unpolarized election. It is therefore natural to build a test set of synthetic elections with controlled polarization attributes, paying particular attention to whether our methods can correctly identify unbalanced clusters and determine the correct number of clusters that recovers the manner of construction.

Given $m \in \mathbb{N}$, $p \in (0, 1)$, and a complete ballot $\sigma \in \Omega_n$, we can randomly generate a cluster $C(\sigma, m, p) \subset \Omega_n$ comprised of m complete ballots, each obtained from σ by performing a number of

[4]If σ is complete, then $(s \cdot t) \cdot \text{dist}(\sigma, A)$ is the number of adjacent swaps necessary to convert σ into a ballot for which the A candidates are all listed first. If σ is not complete, then $(s \cdot t) \cdot \text{dist}(\sigma, A)$ equals the average (over all completions of σ) of the number of adjacent swaps necessary to convert the completion into a ballot for which the A candidates are all listed first, which can be proven just like Corollary 2.6. Thus $(s \cdot t) \cdot \text{dist}(\sigma, \mathcal{A})$ can be described as the "swap distance to solid A over B"; this viewpoint is developed in another new paper by overlapping authors that focuses on measurements of proportionality in ranked elections. [citation redacted]

random adjacent swaps that is drawn from the geometric random variable with parameter p. In other words, each ballot is obtained via a Bradley-Terry process: a random walk on the complete portion of the ballot graph starting from σ, with as many flips as a p-weighted coin comes up heads. A simple polarized election can then be formed from two or more such clusters, amounting to a mixed Mallows model. We first consider the election $\mathcal{E} = C(\text{ABCDE}, 300, 0.3) \cup C(\text{EDCBA}, 700, 0.3) \subset \Omega_5$, shown in Figure 6.

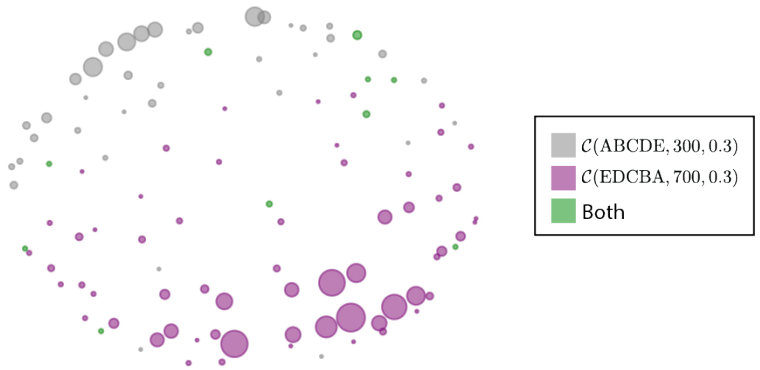

Fig. 6. Ballot MDS plot of synthetic election $\mathcal{E}$. Ballots marked green were hit from both centers.

All of our methods of 2-clustering perform well at identifying the unbalanced clusters in this election. The PAM 2-medoids algorithm correctly identifies the medoids as $\{ABCDE, EDCBA\}$ and produces corresponding clusters of sizes $\{315, 685\}$. In other words, it finds the correct centers and improves the partition by identifying each ballot with its closest center rather than the site from which it originated. Lloyd's 2-means algorithm produces a clustering that differs by only about 3% from the 2-medoids clustering, giving clusters of sizes $\{316, 684\}$. Modularity clustering produces a clustering that differs by about 8% from the ground truth, with cluster sizes $\{373, 627\}$. Slate clustering yields clusters of sizes $\{341, 659\}$ associated to the slates $\{\{A, B, C, D\}, E\}$.

Next, consider three 9-candidate elections $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$, defined below and shown in Figure 7.

$$\mathcal{E}_1 = C(\text{ABCDEFGHI}, 1000, 0.5); \qquad \mathcal{E}_2 = C(\text{ABCDEFGHI}, 300, 0.5) \cup C(\text{HGEIFCBAD}, 700, 0.5);$$

$$\mathcal{E}_3 = C(\text{ABCDEFGHI}, 200, 0.5) \cup C(\text{DFEAHBGCI}, 200, 0.5) \cup C(\text{HIGDEFCBA}, 600, 0.5)$$

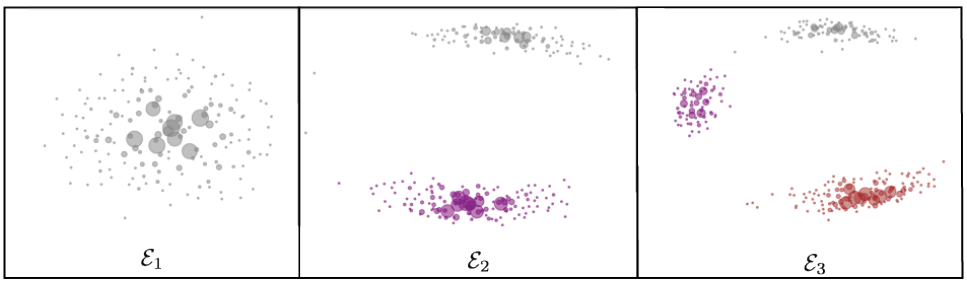

Fig. 7. Ballot MDS plots of $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$

As described above, silhouette scores are a standard off-the-shelf method of finding the optimal number of clusters k. Figure 8 shows that this method succeeds in clearly identifying that $\mathcal{E}_2$ as best described by two clusters and $\mathcal{E}_3$ as best described by three. For $\mathcal{E}_1$, all choices of $k \in \{2, 3, 4, 5\}$ yield visibly poor scores.

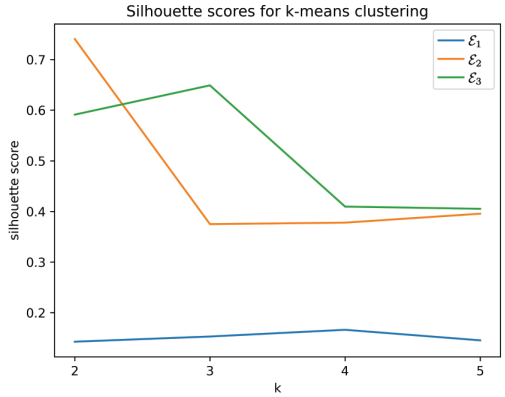

Fig. 8. Silhouette scores of k-means clusterings of $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$

This also allows us to confirm whether our method of measuring modularity is effective at distinguishing polarized from unpolarized test data, since the modularity of a 1-clustering (undivided dataset) equals zero. The maximal modularity for a clustering of $\mathcal{E}_1$ is about .009, which occurs at $k = 3$. By contrast, the maximum modularity for $\mathcal{E}_2$ is significantly above zero at about 0.215, and is attained by a 2-clustering, as expected. For $\mathcal{E}_3$, maximum modularity occurs for $k = 3$ at .241; however, the best 2-clusterings clocked in only slightly below this level. In these tests, modularity succeeds at identifying whether an election has more than one cluster, but if so, silhouette scores appear more reliable at determining how many.

Finally, we vary bloc sizes with a parameter $p \in (0, 1)$, defining

$$\mathcal{E}_2(p) = C(\text{ABCDEFGHI}, 300, p) \cup C(\text{HGEIFCBAD}, 700, p),$$

where larger p produces tigher clusters. Figure 9 shows the silhouette scores of the k-means clusterings of these elections. Even at small values of p, where $C(\text{ABCDEFGHI}, 300, p)$ and $C(\text{HGEIFCBAD}, 700, p)$ have significant overlap, $k = 2$ yields a silhouette score that is clearly better than the other options. Moreover, the modularity score is significancy above zero even for small values of p.

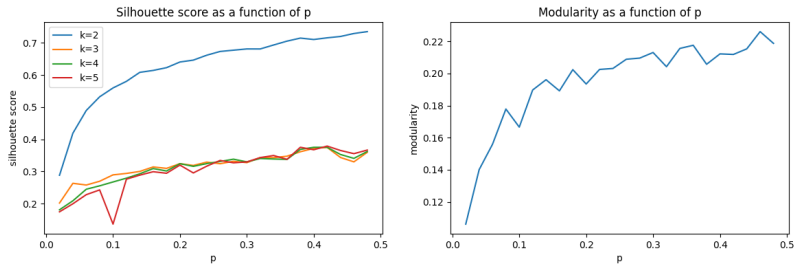

Fig. 9. Silhouette and modularity scores for $\mathcal{E}_2(p)$ for values of p from .02 to .5 by steps of .02.

5 RESULTS ON SCOTTISH ELECTIONS

5.1 Return to Pentland Hills

The 2017 Pentland Hills election was used above to preview many of the methods introduced here. We revisit that election now with some more detailed remarks, then pivot to an overview of the Scottish elections.

Modularity. In the Pentland Hills example, the subgraph of $\mathcal{G}_n$ corresponding to ballots cast at least once has 181 connected components, while the positive-weight subgraph of $\mathcal{G}_n^+$ has 60 components. This led us to run modularity on the full ballot graph, not restricted to positive weight. The maximum-modularity partition of Pentland Hills has $k = 3$ pieces, and it differs from the Lloyd's 3-means clustering by only about 2%. This means that only 2% of the total ballot weight needs to be moved to convert one partition into the other.

IP optimization compared to heuristics. One point of interest is to consider whether exact optimization agreed with the heuristic centroids. In this case, the PAM heuristic successfully located the globally optimal discrete k-medoids, and the global optimum while allowing to all ballots as potential centroids is identical. The IP models do allow us to compute globally optimal (arbitrary) centroids with respect to the L^1 metric, whereas the natural analog of Lloyd's algorithm struggles to do so. However, although there are slight differences in the resulting centroids, the essential underlying conclusions suggested by the heuristics is highly consistent with the conclusions using exact optima.

Candidate similarity and slate clustering. For the Pentland Hills election, slate-clustering yields the slates $A = \{1, 6\}, B = \{2, 3, 4, 5, 7\}$, which pits the two conservative candidates against everyone else, and produces a bipartition of the ballots that differs by less than 3% from that of Lloyd's 2-means clustering.

In Figure 10, the left-hand image shows the $d_{B_\downarrow}$ distances from the Pentland Hills election, restricted to ballots cast at least 10 times, colored by first-place vote. (That is, it is a close counterpart to Figure 2, which showed all ballots of positive weight.) The middle image is on the candidate level, showing the candidate dissimilarity $\bar{d}$-distances. The resemblance of these figures is notable, suggesting a consonance between the voter-centered and candidate-centered points of view.

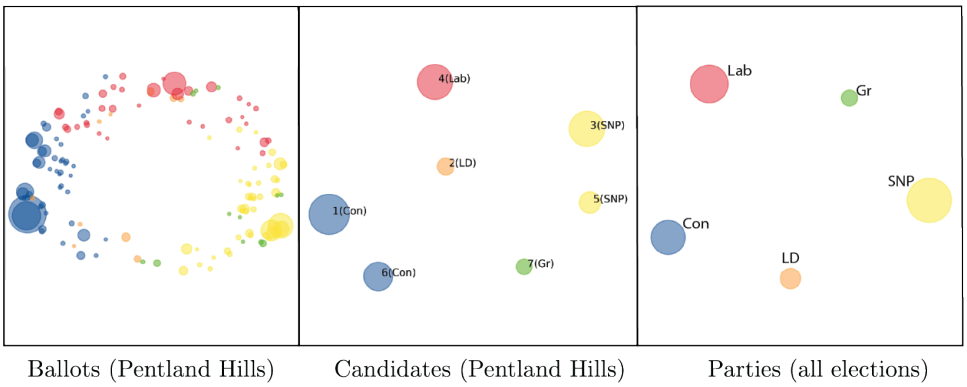

Fig. 10. Ballot, candidate and party MDS plots. The ballots (restricted to those cast at least ten times) are colored by the party of the first-place vote and sized by weight. The candidates are colored by party and sized by the number of first-place votes received. The parties are sized by the total number of first-place votes over all 1070 elections.

5.2 Overview across dataset

Extending candidate clustering to party clustering. The rightmost image in Figure 10 broadens the candidate view to the five major parties, constructed by averaging $\overline{d}$ over all 1070 elections. More precisely, we use the metric $\frac{1}{n} \cdot \overline{d}$ to compare candidates in an election with n candidates, so that the pairwise candidate distances all lie in $[0, 1]$. The distance between two parties, say Grn and SNP, is the average value of $\frac{1}{n} \cdot \overline{d}(A, B)$ over all pairings of a Green candidate A with an SNP candidate B over all 1070 elections. The visual similarities and differences between the three images in Figure 10 are interesting and not obviously predictable.

Modularity. In Pentland Hills, network modularity recovered a partition quite similar to the distance optimization methods and heuristics. However, in many of the Scottish elections, the modularity-optimizing partition differed substantially from the k-means clustering that used the same value of k. One qualitative difference is k-means clustering more often produced roughly equal-sized clusters. For example, among elections for which modularity clustering yielded $k = 3$ clusters, the pieces were often fairly balanced, but sometimes included one very small cluster.

In about 76% of the 1070 Scottish elections, the modularity optimizing algorithm yielded a partition with $k = 3$ components. About 13% had $k = 4$, 11% had $k = 2$ and exactly 2 elections had $k = 5$. In particular, *none* of the 1070 elections are indivisible; that is, none had $k = 1$.

Comparability of methods. To approach the clustering with heuristic methods on the Scottish dataset, we looped over runs that take the best of 200 trials of Lloyd's algorithm for 2-means, and ran single executions of the PAM clustering algorithm for discrete 2-medoids. Table 1 indicates a high degree of concordance among outputs over our database of 1070 Scottish council elections between 2012 and 2022. The table shows that the choice of embedding ($\mathfrak{b}_\downarrow$, $\mathfrak{b}$, or $\mathfrak{h}$) is generally of fairly mild consequence, and that multiple runs of the same heuristic method tend to produce partitions that agree on over 90% of cluster assignments.

	Lloyd $\mathfrak{b}_\downarrow$	Lloyd $\mathfrak{b}$	Lloyd $\mathfrak{h}$	PAM $\mathfrak{b}_\downarrow$	PAM $\mathfrak{b}$	PAM $\mathfrak{h}$
Lloyd $\mathfrak{b}_\downarrow$	0.000	0.016	0.017	0.070	0.084	0.069
Lloyd $\mathfrak{b}$	-	0.000	0.004	0.072	0.082	0.069
Lloyd $\mathfrak{h}$	-	-	0.000	0.072	0.082	0.068
PAM $\mathfrak{b}_\downarrow$	-	-	-	0.024	0.065	0.042
PAM $\mathfrak{b}$	-	-	-	-	0.016	0.061
PAM $\mathfrak{h}$	-	-	-	-	-	0.016

Table 1. The average over 1070 elections of the differences between the heuristic clustering methods. Each cell represents the portion of ballots in the election for which the two partitions disagree on average. Diagonal cells represent the average disagreement between two runs of the same method. All cells are rounded to 3 decimals.

Figure 11 (left) illustrates the distance between our clustering methods as measured by the entries of Table 1 expanded to include the slates method.

None of the clustering methods specifically target balanced bloc sizes, but the results in Scotland tend towards balanced blocs. Figure 11 (right) shows the distribution of bloc sizes for each clustering method. All of the methods typically produce fairly balanced blocs, although slate blocs are occasionally very lopsided.[5] For example, there are 10 elections for which slate clustering produces

[5]In this section, we report results for the slate-clustering algorithm with respect to dist, because we found that using $\overline{\text{dist}}$ too often resulted in slates that pitted a single candidate against all of the rest. Note that once the optimal slates are determined,

a partition whose smallest bloc size is less than 20%. The underlying reason is that, in most of these 10 elections, the slates partition the candidates into those with strong overall support and those with weak support. This partition solves the optimization problem but yields unbalanced blocs.

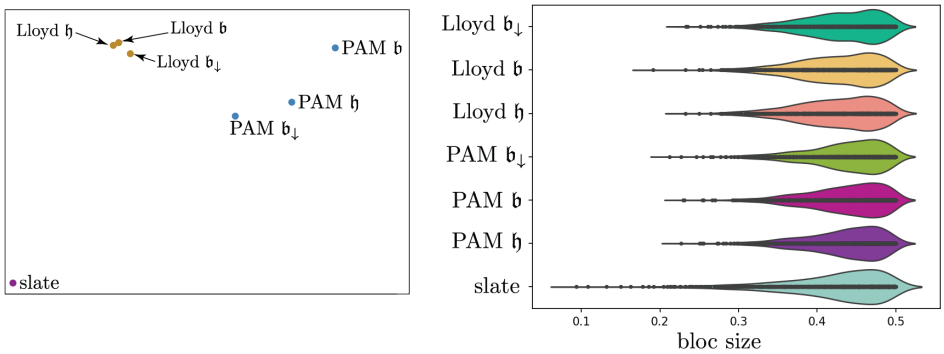

Fig. 11. LEFT: MDS plot of distances between clustering methods from Table 1. RIGHT: A visualization of how balanced the cluster sizes are in the 2-bloc case. Smaller bloc shown.

Finally, Figure 12 shows how each clustering methods performs as judged by three commonly used scoring functions: the silhouette score (defined for any $k > 1$) and the Calinski-Harabasz and Davies-Bouldin scores (only defined for $k = 2$). Regardless of the method used to to produce the clusters, we uniformly computed these three scores with respect to the ballot distances induced by the ɧ embedding.

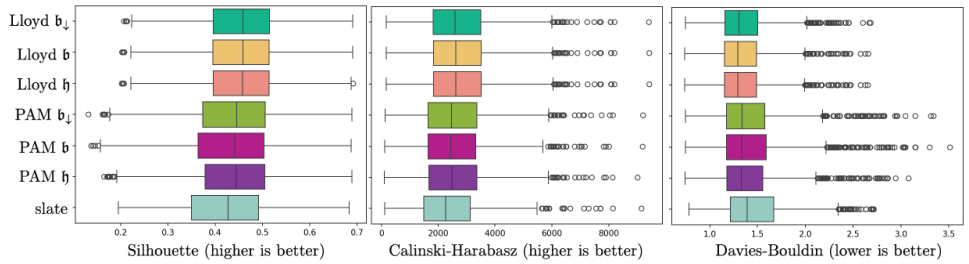

Fig. 12. Three common clustering scores applied to seven clustering methods averaged over 1070 elections. The performance is broadly similar, with slate clustering inducing voter divisions that are a small tick less crisp than the ballot-centered methods.

6 CONCLUSION

Though aggregating rankings is an extremely popular topic in the computing literature, surprisingly little of that work is adapted to the real-world setting of ranked choice elections for political office. This paper combines theory, visualization, and hands-on data analysis in the development of unsupervised techniques for learning structural information about observed elections.

dist and $\overline{\text{dist}}$ induce exactly the same sorting of the ballots into blocs, so the two approaches only differ when they produce different slates.

REFERENCES

Aaron Clauset, M. E. J. Newman, and Cristopher Moore. 2004. Finding community structure in very large networks. *Physical Review E* 70, 6 (Dec. 2004), 066111. https://doi.org/10.1103/PhysRevE.70.066111

Douglas E. Critchlow. 1985. *Metric methods for analyzing partially ranked data.* Springer-Verlag, Berlin; New York.

Persi Diaconis and R. L. Graham. 1977. Spearman's Footrule as a Measure of Disarray. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 39, 2 (Jan. 1977), 262–268. https://doi.org/10.1111/j.2517-6161.1977.tb01624.x

Ronald Fagin, Ravi Kumar, Mohammad Mahdian, D. Sivakumar, and Erik Vee. 2006. Comparing Partial Rankings. *SIAM Journal on Discrete Mathematics* 20, 3 (Jan. 2006), 628–648. https://doi.org/10.1137/05063088X

Ronald Fagin, Ravi Kumar, and D. Sivakumar. 2003. Comparing Top k Lists. *SIAM Journal on Discrete Mathematics* 17, 1 (Jan. 2003), 134–160. https://doi.org/10.1137/S0895480102412856

Peter C. Fishburn. 1977. Condorcet Social Choice Functions. *SIAM J. Appl. Math.* 33, 3 (Nov. 1977), 469–489. https://doi.org/10.1137/0133030

Eric Kamwa. 2022. Scoring rules, ballot truncation, and the truncation paradox. *Public Choice* 192, 1–2 (July 2022), 79–97. https://doi.org/10.1007/s11127-022-00972-8

Maurice G. Kendall. 1970. *Rank correlation methods* (4th ed ed.). Griffin, London.

Maurice G. Kendall and Jean Dickinson Gibbons. 1990. *Rank correlation methods* (5th ed ed.). E. Arnold; Oxford University Press, London: New York, NY.

M. E. J. Newman. 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103, 23 (June 2006), 8577–8582. https://doi.org/10.1073/pnas.0601602103

Anke Van Zuylen and David P. Williamson. 2009. Deterministic Pivoting Algorithms for Constrained Ranking and Clustering Problems. *Mathematics of Operations Research* 34 (Aug. 2009), 594–620. https://doi.org/10.1287/moor.1090.0385

A THE GENERALIZED BALLOT GRAPH

In this appendix, we generalize our previous metric results to include elections in which ballots are allowed to include ties. For example with 7 candidates, the *generalized ballot*

$$\mathcal{B} = (\{C\}, \{A, D\}, \{B, F, G\}, \{E\}) \tag{1}$$

ranks candidate C best followed by A and D tied in the next position and so on. If the candidates are partitioned into two sets (slates) S_1,S_2, then the conditions $S_1 > S_2$ and $S_2 > S_1$ can be regarded as generalized ballots, which is one motivation for this generality.

We can insist without loss of generality that all candidates appear on every generalized ballot, since leaving candidates off would be equivalent to grouping them at the end in the last position. Let $\tilde{\Omega}_n$ denote the set of generalized ballots in an election with n candidates.

The *generalized ballot graph*, $\tilde{\mathcal{G}}_n$, is the graph whose nodes are the elements of $\tilde{\Omega}_n$. There is an edge between two nodes whenever one ballot is obtained from the other by merging two adjacent candidate sets, with weight equal to $\frac{1}{2}$ times the product of the sizes of the sets that were merged. This simple edge rule correctly generalizes both edge rules for $\mathcal{G}_n$.

Example A.1. The graph $\tilde{\mathcal{G}}_4$ contains the following path.

$$(\{A\}, \{B\}, \{C\}, \{D\}) \xrightarrow{.5} (\{A\}, \{B, C\}, \{D\}) \xrightarrow{.5} (\{A\}, \{C\}, \{B\}, \{D\}).$$

This example generalizes to show that the swapping of any pair of adjacent singleton candidates in a generalized ballot corresponds to a distance of 1, as before. More generally, swapping any pair of adjacent candidate-sets corresponds to a distance equal to the product of the sizes of the sets.

We enlarge the domain of $\mathfrak{h}$ to $\mathfrak{h} : \tilde{\Omega}_n \to \mathbb{R}^{\binom{n}{2}}$ in the natural way. As before, its $\binom{n}{2}$ coordinates represent the pairs of candidates, and the value of each coordinate is -1 (lose), 0 (tie), or 1 (win). Here a tie means that the candidates belong to the same candidate-set. We also define weak and strong disagreements exactly as before; in particular, a pair of ballots has a weak disagreement between two candidates if the candidates tie (are in the same candidate-set) with respect to exactly one of the two ballots. Just like before, we define $d_H(\mathcal{B}_1, \mathcal{B}_2) = \frac{1}{2}\|\mathfrak{h}(\mathcal{B}_1) - \mathfrak{h}(\mathcal{B}_2)\|_1$ for all $\mathcal{B}_1, \mathcal{B}_2 \in \tilde{\Omega}_n$, and we observe that

$$d_H(\mathcal{B}_1, \mathcal{B}_2) = \mathrm{str}(\mathcal{B}_1, \mathcal{B}_2) + \frac{1}{2}\,\mathrm{wk}(\mathcal{B}_1, \mathcal{B}_2).$$

A portion of Theorem 2.5 generalizes as follows.

Proposition A.2. *The head-to-head distance d_H is realized as the path metric on the ballot graph $\tilde{\mathcal{G}}_n$.*

Proof. For a pair $\mathcal{B}_1, \mathcal{B}_2 \in \tilde{\Omega}_n$, we must construct a path from $\mathcal{B}_1$ to $\mathcal{B}_2$ in $\tilde{\mathcal{G}}_n$ whose edge weights sum to the weighted disagreement count. For this, the weight of each segment of our path must equal the weighted number of the original disagreements between $\mathcal{B}_1$ and $\mathcal{B}_2$ that the segment resolves (with no new disagreements ever created). We design such a path with waypoints

$$\mathcal{B}_1 \to \overline{\mathcal{B}}_1 \to \overline{\mathcal{B}}_2 \to \mathcal{B}_2,$$

where $\overline{\mathcal{B}}_1$ and $\overline{\mathcal{B}}_2$ share the same candidate-sets; namely the common refinement of the candidates-sets of $\mathcal{B}_1$ and $\mathcal{B}_2$. In other words, a pair of candidates is grouped in $\overline{\mathcal{B}}_1$ and $\overline{\mathcal{B}}_2$ if and only if they are grouped in both $\mathcal{B}_1$ and $\mathcal{B}_2$.

The path $\mathcal{B}_1 \to \overline{\mathcal{B}}_1$ is obtained by splitting (un-merging) each candidate set as much as necessary to achieve the common refinement. To describe the splitting procedure, denote one candidate set of $\mathcal{B}_1$ as $C = \{C_1, ..., C_l\}$, where each C_i is a candidate-set in the common refinement, which might be a singleton candidate. We assume without loss of generality that the indexing comes from the

unique strict order on C induced by $\mathcal{B}_2$. The splitting procedure un-merges C into the sets $C_1, ..., C_l$, *in that order*. This order (or any order) can be attained because we are free to choose the order in which to split off the C_i and whether to split each one off on the right or left.

We construct the path $\mathcal{B}_2 \to \overline{\mathcal{B}_2}$ analogously and then reverse it to obtain the desired path $\overline{\mathcal{B}_2} \to \mathcal{B}_2$. Finally, the path $\overline{\mathcal{B}_1} \to \overline{\mathcal{B}_2}$ is any minimal-weight sequence of adjacent swaps of the common refinement's candidate-sets (as in Example A.1).

The resulting path $\mathcal{B}_1 \to \overline{\mathcal{B}_1} \to \overline{\mathcal{B}_2} \to \mathcal{B}_2$ resolves every disagreement between $\mathcal{B}_1$ and $\mathcal{B}_2$ without ever creating any new disagreements. More precisely, the only head-to-head comparisons changed along $\mathcal{B}_1 \to \overline{\mathcal{B}_1}$ are the resolutions of all of the weak disagreements of the type that tie in $\mathcal{B}_1$. Similarly $\overline{\mathcal{B}_2} \to \mathcal{B}_2$ resolves all of the weak disagreements of the type that tie in $\mathcal{B}_2$, while $\overline{\mathcal{B}_1} \to \overline{\mathcal{B}_2}$ resolves all of the strong disagreements. □

The measurements d_B and $d_{B_\downarrow}$ generalize to $\tilde{\Omega}_n$, with the averaged and pessimistic conventions applied within each candidate set in the natural way. The following was proven in [Fagin et al., 2006].

Proposition A.3 ([Fagin et al., 2006]). *If $\mathcal{B}_1, \mathcal{B}_2 \in \tilde{\Omega}_n$ are distinct, then*

$$\frac{d_B(\mathcal{B}_1, \mathcal{B}_2)}{d_H(\mathcal{B}_1, \mathcal{B}_2)} \in [1, 2].$$

The following example shows that the corresponding result for $d_{B_\downarrow}$ is false.

Example A.4. If $\mathcal{B}_1 = (\{A_1, ..., A_k, X\}, \{B, Y\})$ and $\mathcal{B}_2 = (\{A_1, ..., A_k, Y)\}, \{B, X\})$ in $\tilde{\Omega}_{k+3}$, then we have $d_{B_\downarrow}(\mathcal{B}_1, \mathcal{B}_2) = 4$ and $d_H(\mathcal{B}_1, \mathcal{B}_2) = 2 + k$, so $\lim_{k\to\infty} \frac{d_{B_\downarrow}(\mathcal{B}_1, \mathcal{B}_2)}{d_H(\mathcal{B}_1, \mathcal{B}_2)} = 0$.

B INTEGER PROGRAMMING FORMULATIONS

We next present in detail the integer programming formulations used to find optimal solutions for each variant of our problem in which we are measuring distances with respect to an L_1 metric. There are two embeddings, Borda and head-to-head, and there are three natural choices to be the potential set of cluster centroids (in order of increasing flexibility): (1) points that are embeddings of ballots cast in the given election; (2) points that are embeddings of any valid ballot; (3) any point in the embedding space. We shall say that both (1) and (2) give rise to inputs of the discrete k-median problem (with respect to a distance function $d(u, v)$) whereas (3) gives rise to inputs for the continuous k-median problem. All six of these variations can then be solved by off-the-shelf optimization packages, such as Gurobi.

The discrete k-median problem has a classic integer programming formulation: given a set D of demand points, where each point $j \in D$ has a corresponding weight w_j, and a set C of possible cluster medians, where each pair of points $i \in C$ and $j \in D$ has an associated distance (or assignment cost) $d(i, j)$, we introduce a binary decision variable y_i for each $i \in C$ (to indicate if i is selected as a median) and binary assignment variables x_{ij} for each $i \in C$, $j \in D$ (to indicate if j is assigned to a cluster with i as its median. We then wish to minimize

$$\sum_{i \in C} \sum_{j \in D} w_i d(i, j) x_{ij}$$

subject to the constraints

$$\sum_{i \in C} y_i = k,$$

$$x_{ij} \le y_i, \quad \text{for each } i \in C,\ j \in D,$$

$$x_{ij} \in \{0, 1\},\ \ y_i \in \{0, 1\}, \quad \text{for each } i \in C,\ j \in D.$$

This formulation could, in principle, be used for either the setting (1) or (2), but in the latter case, for elections with more than 6 candidates, the size of $|C| \cdot |D|$ causes the formulation to be beyond memory limitations. (With a more sophisticated implementation that uses implicit column generation to solve the LP relaxations needed while preemptively forcing "unneeded" variables to 0, this might well facilitate all of the elections to be solved via this formulation.)

For the continuous k-median problem, for any distance function that is given by the L_1-norm with respect to some embedding into $\Re^d$, we can use the fact that, for any partition of the input points into k clusters, a minimizing median for each cluster can be found by computing, for each coordinate in the embedding, the median value of that coordinate. Thus, if each ballot is represented by a point $(v_1, \ldots, v_d)$, where each $v_i \in V$, for some finite set V, we can formulate the continuous k-median problem as an integer programming model, with the key decision variables now becoming a binary variable x_{jr} to indicate that each (ballot) $j \in D$ is assigned to the rth cluster, $r = 1, \ldots, k$, and then binary variables z_{irv} to indicate in the ith dimension of the embedding, whether the median value of the rth cluster is equal to $v \in V$. Although the main ideas for the details of the IP formulations are independent of the set V, it is somewhat cleaner for the head-to-head embedding (where $V = \{1, 0, -1\}$), and so we first give that formulation before giving the more general setting (which can be applied to the Borda embedding).

For a given cluster $r = 1, \ldots, k$, for each value $v \in V$, introduce an integer variable W_{irv}, which is defined to be the total weight of points in D assigned to cluster r with coordinate $i = 1, \ldots, d$ of value v. For notational convenience, let $HH(i, j) \in \{+1, 0, -1\}$ denote the value in coordinate $i = 1, \ldots, d$ for ballot $j \in D$ Thus,

$$W(i, r, v) = \sum_{j \in D: HH(i,j)=v} w(i) x_{jr}, \quad \text{for each } i = 1, \ldots, d,\ r = 1, \ldots, k,\ v \in \{+1, 0, -1\}, \tag{2}$$

are constraints in our IP formulation.

We wish to ensure that

$$W(i, r, 1) \geq W(i, r, 0) + W(i, r, -1) \Rightarrow z_{ir1} = 1, \quad \text{for each } i = 1, \ldots, d,\ r = 1, \ldots, k;$$

if this inequality holds, then the median for coordinate i and cluster r is clearly 1. Symmetrically, we also want

$$W(i, r, -1) \geq W(i, r, 0) + W(i, r, 1) \Rightarrow z_{ir,-1} = 1, \quad \text{for each } i = 1, \ldots, d,\ r = 1, \ldots, k.$$

In all other cases, we want $z_{ir0} = 1$. Of course, if either of these constraints holds with equality, we can also set $z_{ir0} = 1$ as well (since there are multiple optimal solutions). These can be written as the following integer programming constraints, using the traditional "big-M" approach, where M is a sufficiently large, precomputed value; in our case, $M = \sum_{j \in D} w(j)$ suffices: for each $i = 1, \ldots, d$, $r = 1, \ldots, k$,

$$W(i, r, 1) - W(i, r, 0) - W(i, r, -1) \leq M z_{ir1}, \tag{3}$$

$$-W(i, r, 1) + W(i, r, 0) + W(i, r, -1) \leq M(1 - z_{ir1}), \tag{4}$$

$$W(i, r, -1) - W(i, r, 0) - W(i, r, 1) \leq M z_{ir,-1}, \tag{5}$$

$$-W(i, r, -1) + W(i, r, 0) + W(i, r, 1) \leq M(1 - z_{ir,-1}). \tag{6}$$

Of course, we need to constrain that each ballot is assigned to a cluster: for each $j \in D$,

$$\sum_{r=1}^{k} x_{jr} = 1. \tag{7}$$

For each cluster and coordinate, there is exactly one median value: for each $i = 1, \ldots, d$, $r = 1, \ldots, k$,

$$\sum_{v \in \{1,0,-1\}} z_{irv} = 1. \tag{8}$$

To compute the cost of a given solution, observe that if the median is 1 for coordinate i in cluster r, then that coordinate contributes $W(i,r,0) + 2W(i,r,-1)$; if it is 0, then $W(i,r,1) + W(i,r,-1)$; and if it is -1, then $W(i,r,0) + 2W(i,r,1)$. One simple way to compute the actual contribution to the objective is simply to set it to the minimum of these three values. We will introduce integer variables C_{ir} to encode this computation. First, we ensure that C_{ir} is at most this minimum: for each $i = 1, \ldots, d$, $r = 1, \ldots, k$,

$$C_{ir} \leq W(i,r,0) + 2W(i,r,-1), \tag{9}$$

$$C_{ir} \leq W(i,r,1) + W(i,r,-1), \tag{10}$$

$$C_{ir} \leq W(i,r,0) + 2W(i,r,1) \tag{11}$$

We use the "big-M" approach to ensure that C_{ir} is exactly this minimum:

$$C_{ir} \geq W(i,r,0) + 2W(i,r,-1) - M(1 - z_{ir1}), \tag{12}$$

$$C_{ir} \geq W(i,r,1) + W(i,r,-1) - M(1 - z_{ir0}), \tag{13}$$

$$C_{ir} \geq W(i,r,0) + 2W(i,r,1) - M(1 - z_{ir,-1}). \tag{14}$$

And then the objective function becomes:

$$\sum_{i=1}^{d} \sum_{r=1}^{c} C_{ir}.$$

To extend this formulation to the Borda embedding, note that we now have a value set $V = \{0, 1, 2, \ldots, n-1\}$ in an n-candidate election. We can use the identical framework to devise an IP formulation. We can define $W(i,r,v)$ exactly as above, and then if v^* is the median value, the contribution of coordinate i to the cost of cluster r is equal to

$$\sum_{v \in V} |v - v^*| W(i,r,v). \tag{15}$$

Hence, we can extend the value C_{ir} to be the minimum of $|V|$ possible contributions.

The analogue of constraints (3)-(6) is based on the observation that v^* is the smallest value in V that satisfies

$$\sum_{v \leq v^*} W(i,r,v) \geq \sum_{v > v^*} W(i,r,v), \tag{16}$$

and so, exactly analogous constraints (using the "big-M" approach) can enforce the correctness that $z_{irv} = 1$, when v is the median value in cluster r for coordinate i.

One novelty in our work is that we observe that the approach used for the continuous k-median problem can be adapted to obtain relatively compact IP formulations for the variants in which we wish to optimize the discrete k-median problem over all potential ballots, not just those that were actually cast. In essence, this means enforcing consistency among the z_{irv} values so that they are binary encodings of proxies of ballots.

Suppose that the centroid for cluster $r = 1, \ldots, k$, is the proxy $\mathbf{v} \in \mathfrak{R}^n$ of a ballot with respect to the Borda embedding: one key feature of such a proxy is an integer $\ell_r \in \{0, 1, \ldots, n-1\}$, where each value $n-1, n-2, \ldots, \ell_r$ is the value of exactly one coordinate in $\mathbf{v}$, and each other coordinate has value 0. Of course, we have that each coordinate for each centroid has exactly one value:

$$\sum_{v \in V} z_{irv} = 1, \quad \text{for each } i = 1, \ldots, n, \ r = 1, \ldots, k.$$

And for each centroid, each value, other than 0, appears at most once:

$$\sum_{i=1}^{n} z_{irv} \leq 1, \quad \text{for each } r = 1, \ldots, k, \text{ for each } v = 1, 2, \ldots, n-1.$$

And, most importantly, if one of the coordinates is v (where $v \in \{1, \ldots, n-2\}$), then also one of the coordinates must be $v+1$:

$$\sum_{i=1}^{n} z_{irv} \leq \sum_{i=1}^{n} z_{ir,v+1}, \quad \text{for each } r = 1, \ldots, k, \ v = 1, 2, \ldots, n-2.$$

It is straightforward to argue that any point that satisfies these constraints corresponds to a valid ballot with respect to the Borda embedding.

We need to be sure to compute the objective function correctly with respect to the selected centroid, not the median value of points in each cluster. Again, we have a variable C_{ir} that represents the contribution of dimension i to the cost of cluster r. We know that if v^* is the selected centroid value, then the contribution of coordinate i to the cost of cluster r is equal to

$$\sum_{v \in V} |v - v^*| W(i, r, v); \tag{17}$$

note that this is no longer the same thing as letting v^* be the value that minimizes this quantity (over all possible choices of v^*). Enforcing this cost computation can again be done with a "big-M" approach by requiring that if $z_{irv^*} = 1$, then $C_{ir} = \sum_{v \in V} |v - v^*| W(i, r, v)$. More explicitly, for each $v^* \in V$,

$$C_{ir} \leq \sum_{v \in V} |v - v^*| W(i, r, v) + M(1 - z_{irv^*})$$

and

$$C_{ir} \geq \sum_{v \in V} |v - v^*| W(i, r, v) - M(1 - z_{irv^*}).$$

Now return to head-to-head embeddings, and consider a point $\mathbf{v} = (v_1, \ldots, v_d)$ where each $v_i \in \{-1, 0, 1\}$ and $d = n(n-1)/2$, and coordinate i corresponds to candidates $f(i)$ and $s(i)$ where $f(i) < s(i)$. Note that we can assume that either the ballot includes all n candidates or omits at least 2 candidates (since the ballot omitting exactly one candidate is completely equivalent to one that also lists the omitted candidate last). For any potential ballot, its head-to-head embedding $(v_1, \ldots, v_d)$ induces a bipartition of $\{1, 2, \ldots, n\}$ into O and N (indicating whether the candidate is on or not on the ballot), where $N = \bigcup_{i: v_i = 0} f(i) \cup s(i)$.

We introduce a binary variable $n_{\ell r}$ where a value of 1 means that the candidate ℓ is not on the ballot corresponding to the median of cluster r. Hence, we see that, for each coordinate i and cluster r,

$$z_{ir0} = 1 \Rightarrow n_{f(i)r} = 1 \text{ and } n_{s(i)r} = 1. \tag{18}$$

We also introduce binary variables $p_{\ell m r}$, where a value of 1 means that for the ballot corresponding to the median of cluster r, candidate ℓ is preferred to candidate m (for all ordered pairs of distinct candidates (ℓ, m) and all clusters r). If ℓ is not on the ballot, then it cannot be preferred to another candidate, and so for each other candidate m and cluster r,

$$n_{\ell r} = 1 \Rightarrow p_{\ell m r} = 0. \tag{19}$$

This preference relation must of course be transitive:

$$p_{\ell m r} = 1 \text{ and } p_{m t r} = 1 \Rightarrow p_{\ell t r} = 1. \tag{20}$$

If we now select a median for cluster r, for which $z_{ir1} = 1$, then the corresponding ballot for that cluster must prefer candidate $f(i)$ to candidate $s(i)$.

$$z_{ir1} = 1 \Rightarrow p_{f(i)s(i)r} = 1. \tag{21}$$

Analogously, we must also have that

$$z_{ir,-1} = 1 \Rightarrow p_{s(i)f(i)r} = 1. \tag{22}$$

We also want to ensure that if candidate m is not on the ballot, then any other candidate is either preferable to m, or also is not on the ballot: for each pair of distinct candidates ℓ, m

$$n_{mr} = 1 \Rightarrow p_{\ell mr} = 1 \text{ or } n_{\ell r} = 1 \tag{23}$$

Finally, we know that it is impossible to prefer both ℓ to m as well as m to ℓ. We can enforce that by requiring for each pair of distinct candidates ℓ, m that

$$p_{\ell mr} + p_{m\ell r} \leq 1. \tag{24}$$

We have argued that the head-to-head embedding of any ballot must satisfy each of the constraints (18)–(24). We also claim that they are sufficient to ensure that the values z_{irv} correspond to a ballot for cluster r. Consider the set S of candidates ℓ for which $n_{\ell r} = 1$. Consider a coordinate $i = 1, \ldots, d$, for which both $f(i)$ and $s(i)$ are in S. By (19), we have that both $p_{f(i)s(i)r} = 0$ and $p_{s(i)f(i)r} = 0$. But then by (21) and (22), we know that neither z_{ir1} nor $z_{ir,-1}$ can be 1, and hence $z_{ir0} = 1$.

Now consider any coordinate i for which exactly one of $f(i)$ and $s(i)$ is in S; suppose that ℓ is in S, but m is not. By (18), we know that $z_{ir0} = 0$. By (23), we know that $p_{m\ell r} = 1$ and by (24) that $p_{\ell mr} = 1$. But then constraints (21) and (22) assure that the z value merely indicates which of ℓ and m is $f(i)$ and $s(i)$, in giving the correct sign in the head-to-head vector for this preference.

Finally, consider the set T of coordinates such that both $f(i)$ and $s(i)$ are not in S. By (18), $z_{ir0} = 0$. But then one (21) and (22) must prompt one preference to be 1. By constraints (20) and (24) we then see that these preferences imply a linear ordering on T. Hence, we have z values that are the realization of the head-to-head encoding of a legitimate ballot.

We need now to translate the implications (18)-(23) into integer programming constraints. Constraint (18) is enforced by

$$z_{ir0} \leq n_{f(i)r} \text{ and } z_{ir0} \leq n_{s(i)r}.$$

The others follow analogously, though for constraints (20) we use

$$p_{\ell mr} + p_{mtr} \leq 1 + p_{\ell tr}.$$

Finally, for constraints (23), we use

$$n_{mr} \leq p_{\ell mr} + n_{\ell r}.$$

We must also correctly model the objective function. As for the Borda proxies, it will no longer be the case that the value selected in each coordinate is the median value present in that cluster. However, we now just insist that if $z_{irv^*} = 1$, the associated cost C_{ir} is exactly equal to (15). We retain constraints (12)-(14), and mirror them by adding

$$C_{ir} \leq W(i,r,0) + 2W(i,r,-1) + M(1 - z_{ir1}), \tag{25}$$

$$C_{ir} \leq W(i,r,1) + W(i,r,-1) + M(1 - z_{ir0}), \tag{26}$$

$$C_{ir} \leq W(i,r,0) + 2W(i,r,1) + M(1 - z_{ir,-1}). \tag{27}$$

These ensure that C_{ir} is computed correctly for our given centroid selection.

C MEASURING POLARIZATION

A promising future direction would build on the work here to measure the extent to which an election is *polarized* overall, including

- Silhouette score of the Lloyd's 2-means clustering of the ballots.
- Average over σ of $\min\{\mathrm{dist}(\sigma, A), \mathrm{dist}(\sigma, B)\}$, where A, B are the dist-slates.
- Average over σ of $\min\{\overline{\mathrm{dist}}(\sigma, A), \overline{\mathrm{dist}}(\sigma, B)\}$, where A, B are the $\overline{\mathrm{dist}}$-slates.

As an illustration, we selected a highly polarized and a highly unpolarized election from among the 282 Scottish elections with 7 candidates. More precisely, we selected elections for which all three of the above polarization measurements came close to attaining their extremum. The results are shown in Figure 13. Interestingly, all 7 candidates in the highly unpolarized election ran as independents. The middle figure shows the $\overline{\mathrm{dist}}$ slates, which in the South Ayrshire election pits the two SNP candidates against everyone else (whereas the dist-slates pit the three conservative candidates against everyone else).

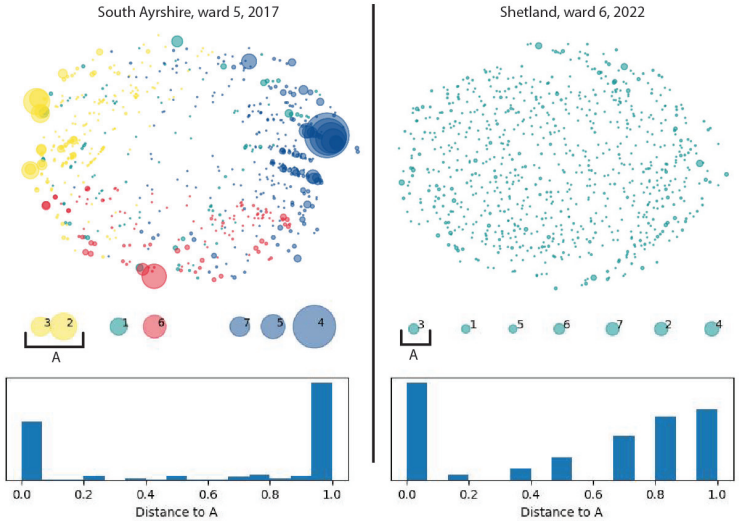

Fig. 13. Two elections with different polarization levels. TOP: two-dimensional ballot MDS plots colored by the party of the first-place vote. MIDDLE: 1-dimensional candidate MDS plots representing the $\overline{d}$-distance between pairs of candidates, colored by party and sized by number of first-place votes. BOTTOM: the distribution of $\overline{\mathrm{dist}}(\sigma, A)$.