The Journal of Open Source Software

VoteKit: A Python package for computational social choice research

Christopher Donnay[1], **Moon Duchin**[2], **Jack Gibson**[3], **Zach Glaser**[3], **Andrew Hong**[4], **Malavika Mukundan**[5], **and Jennifer Wang**[6]

1 The Ohio State University **2** Cornell University **3** MGGG Redistricting Lab **4** Stanford University **5** Boston University **6** Brown University

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

The scholarly study of elections, known as *social choice theory*, centers on the provable properties of voting rules. Practical work in democracy reform focuses on designing or selecting systems of election to produce electoral outcomes that promote legitimacy and broad-based representation. For instance, the dominant electoral system in the United States is a one-person-one-vote/winner-take-all system, sometimes known as PSMD (plurality in single member districts); today, there is considerable reform momentum in favor of ranked choice voting because it is thought to mitigate the effects of vote-splitting and to strengthen prospects for minority representation, among other claimed properties.[1] Across the world, systems of election—and prospects for system change—vary substantially. From both a scholarly and a practical perspective, many questions arise about comparing the properties and tendencies of diverse systems of election in a rigorous manner.

`VoteKit` (MGGG Redistricting Lab, 2023b) is a Python package designed to facilitate just that kind of analysis, bringing together multiple types of functionality. Users can:

1. Create synthetic *preference profiles* (collections of ballots) with a choice of generative models and behavioral parameters;
2. Read in real-world *cast vote records* (CVRs) as observed examples of preference profiles; clean and process ballots, including by deduplication and handling of undervotes and overvotes;
3. Run a variety of *voting rules* to ingest preference profiles and output winner sets and rankings; and
4. Produce a wide range of *summary statistics* and *data visualizations* to compare and analyze profiles and election outcomes.

Statement of need

There has been a significant gap in the literature of computational social choice—and in the landscape of software—between the theory and the practice of democracy. On the software side, researchers have built a multitude of different packages for generating and analyzing elections.[2]

[1]Recent ranked-choice voting reforms include the adoption of instant runoff voting (IRV) in Maine, Alaska, New York City, and single transferable vote (STV) in Portland, Oregon. Advocacy groups claiming various pro-democratic properties of ranked choice include Campaign Legal Center, FairVote, and many others.

[2]See for instance the extensive array of open-source tools on the Computational Social Choice (COMSOC) community page (Ulle Endriss and Simon Rey, 2024) including the widely used collection of ranked data called PrefLib (Ulle Endriss and Simon Rey, 2024). See also the materials provided by FairVote, including their DataVerse and GitHub (FairVote, 2024). The ArXiV preprint (Boehmer et al., 2024) provides an impressively

Most packages, to our knowledge, handle just one part of the research arc; for instance, `PrefSampling` (Boehmer et al., 2024) generates profiles but does not conduct elections, while `VoteLib` (Jan Šimbera, 2020) *only* conducts elections. Others, like `PrefLibTools` (Nicholas Mattei and Simon Rey, 2022) and `PrefVoting` (Eric Pacuit and Wesley H. Holliday, 2022), provide support for generating profiles and conducting single-winner elections. Packages with multi-winner capability, like `abcvoting` (Lackner et al., 2023) or `Apportionment` (Martin Lackner, 2022), do not support ranked voting. Note that single transferable voting (STV), a voting system actually used for political election in six countries, is curiously absent. `VoteKit` is built to provide an end-to-end pipeline that supports ranked, scored, and approval profiles as well as single- and multi-winner elections, with an emphasis on practical applicability.

Area of need: Generative models

For one concrete example of a literature and software gap, consider the construction of *generative models*. This term is often associated with large language models as paradigms of artificial intelligence; here, what is being generated is realistic voting rather than realistic language. In this setting, a generative model of voting is a probability distribution on the set of all possible ballots that can be cast in a given election style; profiles can be sampled from a generative model to produce simulated or synthetic elections. Having sources of rich, varied, and realistic data is essential to an empirically grounded research program to probe the properties of voting rules. Good generative models are also essential to advise reformers deciding between options in a new locality, as they enable generation of synthetic profiles keyed to the scale, demographics, and election specs of that specific place.

`VoteKit` implements many of the models typically used in computational social choice, as well as newer mathematical models that give users the ability to generate profiles that are designed to comport with real-world ranking behavior and particularly to generate polarized elections. Two leading choices are based on classic statistical ranking mechanisms, called the Plackett–Luce (PL) and Bradley–Terry (BT) models; another model called the Cambridge Sampler (CS) draws from historical ranking data in Cambridge, MA city council elections. These models have flexible parameters, allowing users to vary voting bloc proportions, candidate strength within slates, and polarization between blocs. These parameters can be specified or randomly sampled.

Area of need: Comparison and communication

In the realm of democracy reform, groups of stakeholders often ask researchers to provide modeling studies to decide on what shift to make in electoral systems, as the project list below makes clear. `VoteKit` implements voting rules that stakeholders often seek to compare, with parameters designed to be tailored by the user to the specific locality under study. Available voting rules include:

- **Ranking-based (ordinal).** Plurality/SNTV, STV and IRV, (generalized) Borda, Alaska[3], Top-Two, Dominating sets/Smith method, Condo-Borda[4], Sequential RCV.
- **Score-based (cardinal).** Range voting, Cumulative, Limited.
- **Approval-based (set).** Approval voting, Bloc plurality.

comprehensive list of numerical experiments on elections. The PRAGMA Project (https://perma.cc/2P6V-8ZER) echoes our statement of need, noting that the current literature and software falls short in practical applicability and that the understanding of real and synthetic data is "very limited."

[3]Our model of the Alaska method is an SNTV/STV hybrid that uses single non-transferable vote to choose a set of finalists, then runs STV on the same preference profile to fill the seats. Alaska's elections run this with four finalists and one seat; the top-two system runs this with two finalists and one seat.

[4]This system orders candidates within dominating sets by Borda score. Note that this is distinct from Black's method (Black, 1986), which uses Borda score as a backup system in case the smallest dominating set is not a singleton.

This list does not include every method that has attracted theoretical investigation; rather, it is oriented to methods used or considered for political representation, such as the final-four system in Alaska or the sequential RCV in Utah local elections. See generally (Amorós et al., 2016; Emerson, 2013; McCune et al., 2023; Reynolds et al., 2008; Tideman, 1995) for references. In addition, `VoteKit` is flexible enough to allow users to write custom voting rules.

Reform advocates also need to describe voting mechanisms and their likely outcomes effectively to members of their communities. The end-to-end pipeline provided by `VoteKit` allows advocates to toggle different system settings and compare expected outcomes. For example, Figure 1 is reprinted from a report on reform proposals for the chambers of the Washington state legislature, with Systems 0-3 as paired bicameral systems and Systems 4-5 as unicameral solutions. Using the codebase that formed the foundation of `VoteKit`, researchers compared the expected outcomes for minority representation under these six systems.

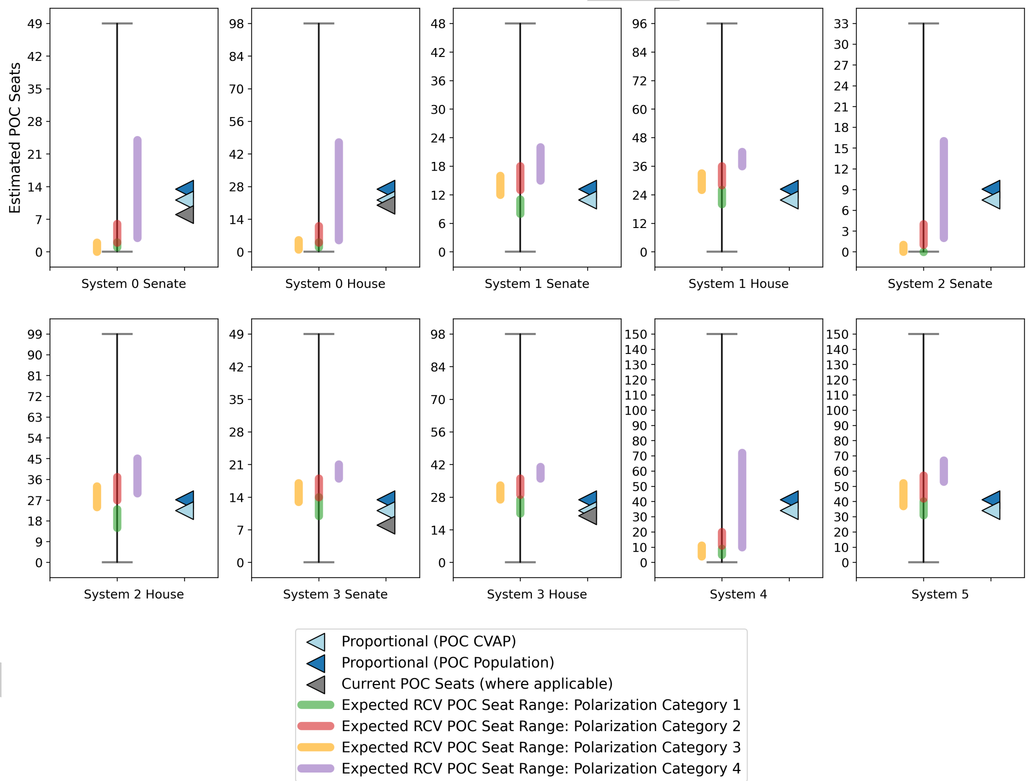

Figure 1: A comparison of a variety of electoral systems and their effect on minority representation in a case study of reform proposals for the Washington state legislature (MGGG Redistricting Lab, 2021).

Area of need: Resources for research

Previous research works such as (Elkind et al., 2017) have compared properties of earlier generative models; `VoteKit` has functionality to fully replicate this work and facilitates robust comparisons across a more comprehensive and up-to-date list of alternatives. It also offers new analytical tools that will support research on elections. Some examples are shown in Figure 2. At left is a *ballot graph*, where nodes are ballots weighted by their frequency in the profile; a recent research paper shows that ballot graphs can be metrized to realize classical statistical ranking distances, like Kendall tau and the Spearman footrule (Duchin & Tapp, 2024). `VoteKit` also implements a class of election distances, as surveyed in (Boehmer et al., 2022). Choices for measuring the difference between two profiles on the same set of candidates include L^p distance and Wasserstein (earth-mover) distance. At right is a multidimensional scaling (MDS) plot of a different set of data, showing mutual L^1 differences between generated profiles across various selections of model (shown in colors) and candidate strength parameters (shown with symbols), enabling comparisons in the style of (Szufa et al., 2020).

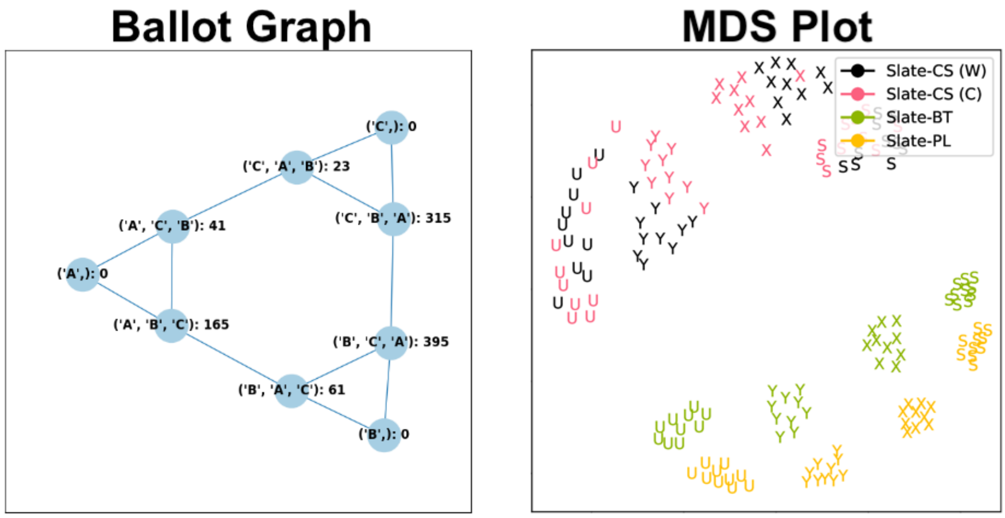

Figure 2: At left, the ballot graph for a 3-candidate election. There is one node per possible ballot, and the weights show the number of instances of that ballot in the profile. At right, a multidimensional scaling (MDS) plot for 160 synthetic profiles made with various generative models and candidate strength parameters for two slates of 3 candidates each. The MDS plot is a low-distortion planar embedding of those 160 profiles and their pairwise differences.

Finally, VoteKit interacts seamlessly with a wide range of actual vote data, such as thousands of political elections collected by FairVote and a cleaned repository of over 1000 Scottish STV local government elections (FairVote, 2024; MGGG Redistricting Lab, 2023a). Previously, the use of real data in election research was often extremely limited; for instance, a recent survey reports that the single most popular "real-life" dataset has been a survey of 5000 respondents' sushi preferences (Boehmer et al., 2024).

Projects

A significant number of white papers and scholarly articles have used VoteKit (and its predecessor codebase) in recent years. These include the following.

- A large number of case studies in ranked-choice modeling, such as studies for the city councils of Chicago, IL (MGGG Redistricting Lab, 2019b) and Lowell, MA (MGGG Redistricting Lab, 2019a). the state legislatures of Oregon and
- A study modeling the impact of proposed legislation called the Fair Representation Act, which would convert U.S. Congressional elections to the single transferable vote system (MGGG Redistricting Lab, 2022);
- A peer-reviewed article for an election law audience on the impact of STV elections on minority representation (Benadè et al., 2021);
- A peer-reviewed article for a CS/econ audience that probes whether STV delivers proportional representation (Benadè et al., 2024); and
- A peer-reviewed article for an CS/operations research audience on optimizing to "learn" blocs and slates in real-world elections (Duchin & Tapp, 2024).

The Journal of Open Source Software

Acknowledgements

This work was initiated in a research cluster in Summer 2023, funded by the Democracy Fund and graciously hosted at the Faculty of Computing and Data Sciences at Boston University and the Tisch College of Civic Life at Tufts University. Major contributors to the initiation of the project include Brenda Macias, Emarie De La Nuez, Greg Kehne, Jordan Phan, Rory Erlich, James Turk, and David McCune. Earlier code contributions were made by Chanel Richardson, Anthony Pizzimenti, Gabe Schoenbach, Dylan Phelan, Thomas Weighill, Dara Gold, and Amy Becker; recent code contributors include Peter Rock, Kevin Quinn, and Divij Sinha. The authors also thank Deb Otis, Jeanne Clelland, and Michael Parsons for helpful feedback. FairVote's data repository in Dataverse (https://dataverse.harvard.edu/dataverse/rcv_cvrs) and RCV Cruncher code on GitHub (https://github.com/fairvotereform/rcv_cruncher/) are open-source efforts that provided inspiration for the current project.

References

Amorós, P., Puy, M. S., & Martínez, R. (2016). Closed primaries versus top-two primaries. *Public Choice*, *167*, 21–35. https://doi.org/10.1007/s11127-016-0328-5

Benadè, G., Buck, R., Duchin, M., Gold, D., & Weighill, T. (2021). *Ranked choice voting and proportional representation*. https://doi.org/10.2139/ssrn.3778021

Benadè, G., Donnay, C., Duchin, M., & Weighill, T. (2024). Proportionality for ranked voting, in theory and practice. In *Preprint*. https://mggg.org/PRVTP.

Black, D. (1986). *The theory of committees and elections*. Springer Netherlands. https://doi.org/10.1007/978-94-009-4225-7

Boehmer, N., Faliszewski, P., Janeczko, Ł., Kaczmarczyk, A., Lisowski, G., Pierczyński, G., Rey, S., Stolicki, D., Szufa, S., & Wąs, T. (2024). Guide to numerical experiments on elections in computational social choice. In K. Larson (Ed.), *Proceedings of the thirty-third international joint conference on artificial intelligence, IJCAI-24* (pp. 7962–7970). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2024/881

Boehmer, N., Faliszewski, P., Niedermeier, R., Szufa, S., & Wąs, T. (2022). Understanding distance measures among elections. In L. D. Raedt (Ed.), *Proceedings of the thirty-first international joint conference on artificial intelligence, IJCAI-22* (pp. 102–108). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2022/15

Duchin, & Tapp. (2024). Learning blocs and slates from observed elections. *Preprint*.

Elkind, E., Faliszewski, P., Laslier, J.-F., Skowron, P., Slinko, A., & Talmon, N. (2017). What do multiwinner voting rules do? An experiment over the two-dimensional euclidean domain. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1). https://doi.org/10.1609/aaai.v31i1.10612

Emerson, P. (2013). The original borda count and partial voting. *Social Choice and Welfare*, *40*, 353–358. https://doi.org/10.1007/s00355-011-0603-9

Eric Pacuit and Wesley H. Holliday. (2022). *Pref_voting*. https://github.com/voting-tools/pref_voting

FairVote. (2024). *RCV cruncher*. https://github.com/fairvotereform/rcv_cruncher/

Jan Šimbera. (2020). *Votelib*. https://github.com/simberaj/votelib

Lackner, M., Regner, P., & Krenn, B. (2023). Abcvoting: A Python package for approval-based multi-winner voting rules. *Journal of Open Source Software*, *8*(81), 4880. https:

//doi.org/10.21105/joss.04880

Martin Lackner. (2022). *Apportionment*. https://github.com/martinlackner/apportionmentb

McCune, D., Martin, E., Latina, G., & Simms, K. (2023). *A comparison of sequential ranked-choice voting and single transferable vote*. https://doi.org/10.1007/s42001-024-00249-8

MGGG Redistricting Lab. (2019a). Findings on the city of lowell's election systems. In *White Paper*. https://mggg.org/Lowell-Detailed-Report.

MGGG Redistricting Lab. (2019b). Study of reform proposals for chicago city council. In *White Paper*. https://mggg.org/publications/Chicago.pdf.

MGGG Redistricting Lab. (2021). Analysis of election systems for washington state. In *White Paper*. https://mggg.org/Washington.

MGGG Redistricting Lab. (2022). Modeling the Fair Representation Act. In *White Paper*. https://mggg.org/FRA-Report.

MGGG Redistricting Lab. (2023a). *Scottish STV election repo*. https://github.com/mggg/scot-elex

MGGG Redistricting Lab. (2023b). *VoteKit*. https://github.com/mggg/VoteKit

Nicholas Mattei and Simon Rey. (2022). *PrefLib-tools*. https://github.com/PrefLib/preflibtools

Reynolds, A., Reilly, B., & Ellis, A. (2008). *Electoral system design: The new international IDEA handbook*. International Institute for Democracy; Electoral Assistance.

Szufa, S., Faliszewski, P., Skowron, P., Slinko, A., & Talmon, N. (2020). Drawing a map of elections in the space of statistical cultures. *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 1341–1349. ISBN: 9781450375184

Tideman, N. (1995). The single transferable vote. *Journal of Economic Perspectives*, *9*(1), 27–38. https://doi.org/10.1257/jep.9.1.27

Ulle Endriss and Simon Rey. (2024). *COMSOC community site*. https://comsoc-community.org/tools