Hacking the Election:

Measuring and Solving Gerrymandering in Today’s Political System

Kai Vernooy, Arin Khare, James Lian

Abstract

The goal of Hacking The Election was to quantify the political phenomenon of gerrymandering and create an algorithm to fairly redistrict based on this quantification. Our community generation algorithm groups geographical units called “precincts" obtained from census data into communities and districts. These districts are drawn based on partisanship, compactness, and population constraints using the iterative method, in which tests are run repeatedly until they satisfy all of our constraints. In order to measure gerrymandering, our quantification algorithm determines how political communities are cracked and packed into districts, whether actual or generated. Our community generation algorithm converged to a global optimum in both base political communities and in generated districts, and found not only significant gerrymandering in New Hampshire congressional districts but also new districts that, if implemented, would have halved the gerrymandering score measured by our quantification algorithm as compared to current districts. Through use and creation of our own open-source data and software, we managed to create districts that more fairly represented voters and reduced gerrymandering, and made important progress in finding a new method of detecting, quantifying, and solving gerrymandering as a result.

Introduction

Gerrymandering is defined as the practice of drawing districts in order to manipulate the constituency of those districts to give a political party an advantage. This began in 1812, when Massachusetts governor Elbridge Gerry drew district lines in order to favor the Democratic-Republicans in the legislature. Gerrymandering is a major problem in American democracy because it misrepresents the voice of voters by ensuring that certain districts always elect a member of one party, or that a constituency is unfairly split, essentially nullifying their votes. Redistricting by politicians occurs every ten years, and in recent decades gerrymandering has grown more and more blatant and common, by both Democrats and Republicans. Worse, the current methods of measuring gerrymandering are unreliable or computationally expensive, leading to a crucial gap in measurement for what is a major problem in America’s democratic system today.

The simplest method of measuring gerrymandering is through the “efficiency gap.” The efficiency gap uses the “wasted vote,” which is the number of votes that did not go towards electing the representative, and is defined as the difference between the wasted votes for Democrats and Republicans divided by the number of total votes. However, the efficiency gap can be unreliable, and flag what are perfectly normal scenarios of political transition while completely missing the most flagrant examples of gerrymandering. In 2014, roughly the same number of people voted for Democrats and Republicans in Illinois, but Democrats ended up winning 71-47 in the state house, even though the efficiency gap was only 2.3%.

Another approach is to calculate billions of maps and compare the votes / seats ratio for all of these hypothetical maps to the actual districts. While this is a reliable way of detecting gerrymandering, it involves the use of a supercomputer and months of data analysis, and offers no guidance on which of the maps it generates may be best as an alternative district map, which puts redistricting outside the reach of ordinary people.

Approach

Our approach seeks to address the limitations of current methods, create reliable maps that improve political representation, and accurately measure gerrymandering. To do so, we used a unit of geographical data called a "precinct". When districts are drawn, they are subdivided into precincts, which correspond to a group of people who vote at the same place. Precincts can be about the size of a neighborhood, and usually contain from a hundred to around three or four thousand voters. Thus, when we redistrict, we redistrict only along precinct lines - this gives a finite number of potential districts, and reduces computational complexity.

In order to implement an algorithm that groups precincts, we needed geographical data of precinct borders and data for the number of votes for each party within those precincts. To quantify the state of gerrymandering in current congressional districts, we additionally required geographical data of districts. These came in the form of GeoJson, an accessible and readily available format for coordinate data. However, because election records are different with every state, we often had to find separate election and population data, which came from various sources such as the 2010 Census, Harvard, State databases, open sources websites, and Github repositories. For convenience, storage size and reusability, we used a process called "serialization", which combines data from multiple sources into a single binary file. In the process of doing so, we split non-contiguous precincts into individual precincts and removed holes, making the problem of generating contiguous districts significantly easier. For five states, Oregon, Montana, Kentucky, Florida, and West Virginia, we were unable to gather all of the necessary data.

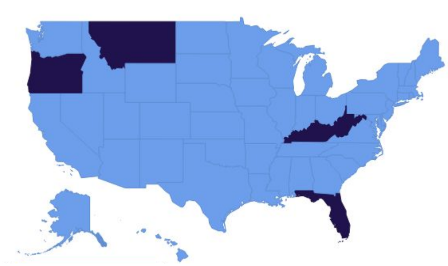


Fig 2.1 - A map shows the states for which we have election data, precinct geodata, and district geodata (in light blue).

The core of gerrymandering is how it breaks up or combines political communities of like-minded, partisan individuals. To measure gerrymandering, we needed to identify those political communities. For that we created an algorithm that will, given constraints, output collections of precincts that we call “communities” that follow the given constraints. These communities mainly depend on minimizing the standard deviation in the political opinion of the precincts inside of the communities. This means that the communities will be grouping people who think, and vote alike. Additionally, we define fair districts to be districts that minimize this same political diversity. This is the most important constraint in our algorithm for generating political communities.

Though it would be best to create districts with as little partisanship diversity as possible, there are constraints that limit the shapes that districts can take. According to federal law, districts cannot have populations that vary beyond 1% of each other. Districts also must be fairly compact, and although this varies according to state law, it is a good idea to consider compactness while redistricting. The measure of compactness that we are using is called the Schwartzberg ratio. The Schwartzberg ratio is a measure of how close a shape is to a circle, and is defined as the ratio between the perimeter of a shape and the circumference of a circle with the same area. We considered both population and compactness when creating our algorithms.

Our algorithm for generating political communities implemented the iterative method. This means it starts with a random initial guess, and then refines that guess for each of our constraints. It repeats these refinement processes as many times as necessary until it finds a solution that satisfies all of the constraints. The community generation algorithm is run with loose constraints for population and compactness, therefore allowing us to give importance to the partisanship standard deviation, so as to determine groups of political homogeneity. Thus, the average standard deviation of the partisanship of precincts in a community we set to be below 9%, while the minimum Schwartzberg compactness we set to 0.3, and the population for each community to be within 15% of every other community.

. . .

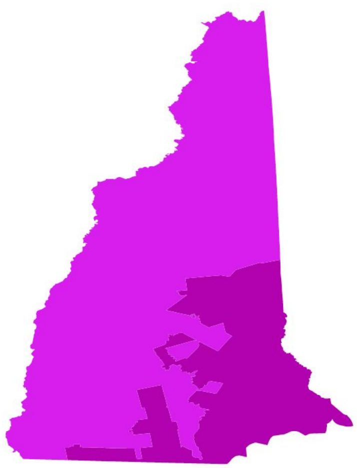
Once political communities are created, we then need a method to compare them to different district maps. For each district, our quantification method measures cracking and packing by finding the community that contains most of the district. We then determine the percent of the district’s area that is outside of this community, which becomes our gerrymandering score. This essentially identifies the community the district is trying to represent, and determines how well (in the form of a percentage) the district represents that community.

Our method of redistricting is to use our community generating algorithm with tighter population and compactness constraints, and a looser partisanship constraint. This is because, when generating actual districts, following population laws and increasing district compactness are more important than having districts with lower political diversity.

These algorithms were actually implemented in both C++ and Python, common languages for use in data science. They are open-source and can be found at https://github.com/hacking-the-election/gerrymandering.

Results

Despite having data for so many states, due to time limitations we were only able to run our algorithms on New Hampshire, creating both political communities and completely new districts. The following is data gathered from communities, redistricting and quantification algorithm



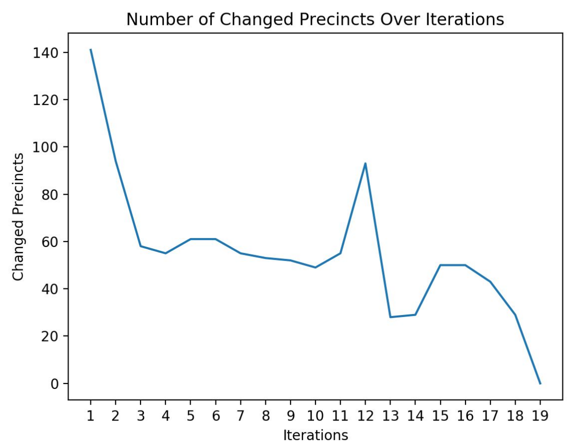
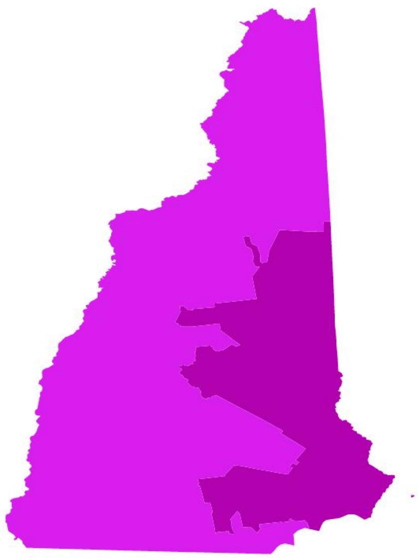


Fig 3.1. (Left) A picture of our generated base communities for New Hampshire. Fig 3.2. (Right) The number of precincts that changed communities over the iterations of our community generation algorithm when redistricting New Hampshire. The algorithm converged at iteration 19 when zero precincts moved and all the constraints were satisfied.

Using these base communities, our quantification algorithm returned a gerrymandering score for New Hampshire of about 24.96%, with its first congressional district getting a score of 36.7% and its second getting 13.18%.



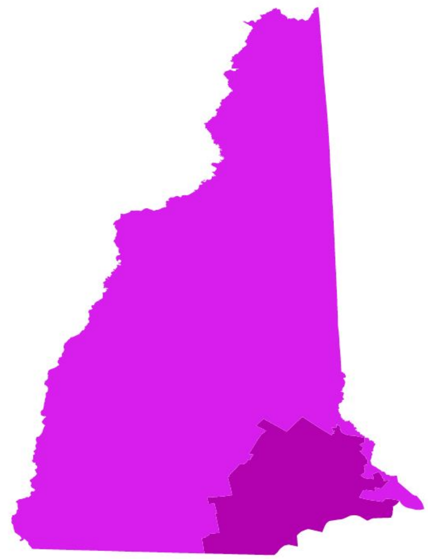


Fig 4.1 (Left) A picture of what the current congressional districts look like. Fig 4.2 (Right) A picture of what the districts we generated look like. Note that the state borders look different. This is due to the fact that precincts can be over water and have no people, but must be used due to the island in New Hampshire.

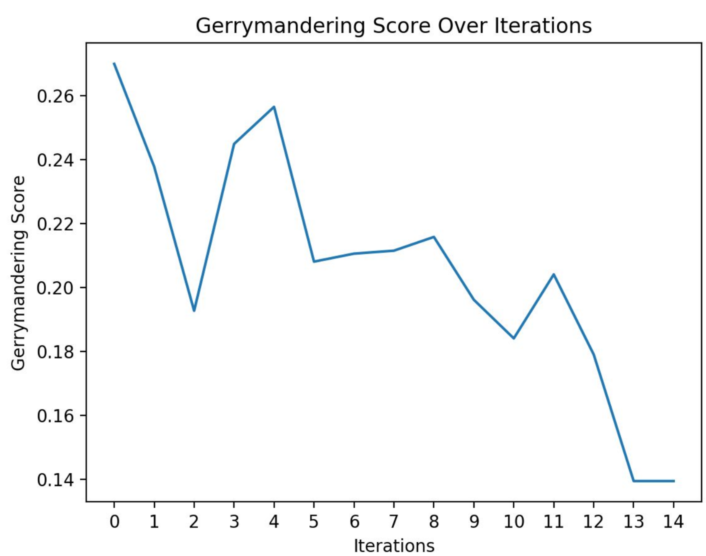


Fig 4.3 - Our algorithm started with a random guess and eventually almost halved its gerrymandering score through the refinement processes.

Discussion

When writing this algorithm, we attempted to find a solution where the communities "converge" - that is, the solution satisfies all of constraints and has no more precincts to exchange that benefit it. This is a concept similar to gradient descent, where by improving our solution every time, we find the global optimum, the point at which the map and the political communities cannot be changed further in order to improve some constraint. This was a success, as running the community generation algorithm multiple times with different initial configurations led to similar districts, showing that the districts we generated were indeed the best possible district for the state. We know this because we have written two different implementations of this algorithm (that allow us to randomly start on a different initial guess), and every attempt converges to the same solution. This is heuristic proof that our algorithm finds the optimum, but further testing would allow us to strengthen this conclusion.

Additionally, through trial and error, we attempted to find the constraints that enabled the algorithm to converge while being as tight as possible to create the best communities. If we had had greater computing power or more time, we could have used gradient descent to determine exactly what set of parameters would be as strict as possible but still converge.

Whenever attempting to create a scale for a previously considered qualitative concept, there arises the necessity of having a finite range of possible values. If either of the bounds of the scale are infinite or unknown, all the values lose their meaning because to put them into perspective, you have to attach a value to a scenario, (e.g. The definition of 24.9% gerrymandering is the current state of New Hampshire's congressional districts.) This revokes the scale of its status as quantitative, as it must have a qualitative frame of reference. Therefore our algorithm had to have a theoretical maximum, which we found can be determined mathematically. The maximum of n amount of districts means that a given district contains n equal sized communities. We can then apply the gerrymandering algorithm to this scenario, giving us the percentage of the area outside of one of these equally sized communities: $\frac{n-1}{n}$.

As the theoretical maximum varies depending on the state, it is possible to adjust the scale of output values depending on your state. For instance, New Hampshire redistricting ended at 12%, but the theoretical maximum for a state such as itself with two districts is only 50%, as calculated by the $\frac{n-1}{n}$ rule. The question is whether or not this 12% value should be adjusted as a fraction of the maximum, to determine that the gerrymandering score of New Hampshire is really 48% and our generated districts 24%, as a representation of how much possible gerrymandering could have been done.

However, the question still remains of how gerrymandered can a state get? The answer lies in our formula. The highest possible gerrymandering score as a function of *n* with a domain of $[1, \infty)$ increases as *n* does with a horizontal asymptote of $x = 1$, meaning that a theoretical state with an infinite number of districts would have the potential to be gerrymandered 100%.

Additionally, the gerrymandering value determined for the current New Hampshire districts shows significant gerrymandering and may be a sign that districts were intentionally drawn that way, a case to further look into. Furthermore, our redistricting algorithm, which bases its initial configuration on the communities, does come geographically closer to the political communities than current districts, meaning its gerrymandering score is significantly reduced.

Conclusion

In conclusion, this project successfully achieved its primary goals of quantifying gerrymandering and redistricting. We found New Hampshire's current congressional districts to be significantly gerrymandered, and we also created new districts that are much less gerrymandered. Our algorithm found the global optimum - an ideal solution to our constraints that can be reached from any initial guess. "Hacking the Election" introduces a new method of detecting and eliminating gerrymandering, one that is more successful in many aspects than the existing ones.

Future Steps

Though we were only able to run this algorithm on New Hampshire, this has enormous potential to identify gerrymandering in all states where other methods, such as the efficiency gap, fail. The algorithm we have developed is reliable, accurate, and not computationally intensive. However, there is room for it to be made faster, more rigorous, and more scalable, which would help as we tackle states with hundreds of times more precincts than New Hampshire, such as Texas and California.

Furthermore, 2020 is a Census year, which means states will be redrawing their districts, and thus introduce new opportunities for gerrymandering. We plan to apply our tools to this new data and compare gerrymandering this decade with gerrymandering the next, which will allow us to see which states have improved and which states have worsened, thus adding an extra dimension to our gerrymandering analysis.

Additionally, though a massive problem for our democracy, partisan gerrymandering is not currently illegal in the United States. However, one related crime is found in racially based gerrymandering, which splits up communities on the basis of race. In the future, we plan to apply our algorithm to detect racial communities, and use the same gerrymandering quantification algorithm to see how these communities are broken up. This could eventually lead to a court case, as our algorithms may be used as evidence in a racial gerrymandering

Finally, we plan to launch a website at https://hacking-the-election.github.io/. While currently it is being used to display videos and other media material not available for display in this paper, we plan to make it a fully developed website and redistricting tool, which will allow us to spread our work further and increase public awareness of the algorithms and code we are currently developing.

Bibliography

[1]. Adler, W. T.,& Koeze, E. (2019, July 9). One Way To Spot A Partisan Gerrymander. *FiveThirtyEight*. https://projects.fivethirtyeight.com/partisan-gerrymandering-north-carolina/.

[2]. Altman, Micah. “Modeling the Effect of Mandatory District Compactness on Partisan Gerrymanders.” *Political Geography*, vol. 17, no. 8, 1998, pp. 989–1012., doi:10.1016/s0962-6298(98)00015-8.

[3]. Bozeman, James R., et al. “Redistricting without Gerrymandering, Utilizing the Convexity Ratio, and Other Applications to Business and Industry.” *Applied Stochastic Models in Business and Industry*, vol. 34, no. 6, 16 Sept. 2018, pp. 835–851. *Wiley Online Library*, doi:10.1002/asmb.2396.

[4]. Chen, Jowei, and Jonathan Rodden. “Cutting Through the Thicket: Redistricting Simulations and the detection of Partisan Gerrymanders.” *Election Law Journal: Rules, Politics, and Policy*, vol. 14, no. 4, 17 Dec. 2015, pp. 331–345. *Election Law Journal*, doi:10.1089/elj.2015.0317.

[5]. Cottrell, David. “Using Computer Simulations to Measure the Effect of Gerrymandering on Electoral Competition in the U.S. Congress.” *Legislative Studies Quarterly*, vol. 44, no. 3, 19 Feb. 2019.

[6]. Feldman, M. (2017, April 14). “Can a Supercomputing Algorithm Kill Gerrymandering?” *Top500*. https://www.top500.org/news/can-a-supercomputing-algorithm-kill-gerrymanding/

[7]. Fleiner, Balázs, et al. “Optimal Partisan Districting on Planar Geographies.” *Central European Journal of Operations Research*, vol. 25, no. 4, 13 Sept. 2016, pp. 879–888., doi:10.1007/s10100-016-0454-7.

[8]. Forest, Benjamin. “Redistricting and the Elusive Ideals of Representation.” *Political Geography*, vol. 32, 2013, pp. 15–17., doi:10.1016/j.polgeo.2012.10.003.

[9]. Kelso, N. V. (2019, 3 February). “Nvkelso/Election-Geodata.” *GitHub* https://github.com/nvkelso/election-geodata

[10]. Liu, Yan Y., et al. “PEAR: A Massively Parallel Evolutionary Computation Approach for Political Redistricting Optimization and Analysis.” *Swarm and Evolutionary Computation*, vol. 30, 26 May 2016, pp. 78–92., doi:10.1016/j.swevo.2016.04.004.

[11]. Petry, Eric. “How the Efficiency Gap Works.” brennancenter.org, Brennan Center for Justice, www.brennancenter.org/sites/default/files/legal-work/How_the_Efficiency_Gap_Standard_Works.pdf

[12]. Ricca, Federica, and Bruno Simeone. “Local Search Algorithms for Political Districting.” *European Journal of Operational Research*, vol. 189, 27 June 2007, pp. 1409–1426., www.journals.elsevier.com/european-journal-of-operational-research.

[13]. Ricca, Federica, et al. "Political Districting: from Classical Models to Recent Approaches." *Ann Oper Res*, vol. 204, 9 Jan. 2011, pp. 271–299., doi:10.1007/s10288-011-0177-5.

[14]. Wasserman, D. (2018, January 25). Hating Gerrymandering Is Easy. Fixing It Is Harder. *FiveThirtyEight.* https://fivethirtyeight.com/features/hating-gerrymandering-is-easy-fixing-it-is-harder/

[15]. (2019, October 16). "Wasted Vote." *Wikipedia*. https://en.wikipedia.org/wiki/Wasted_vote