MCMC Redistricting

Dawson Eliasen, Jonathan Olavarria, Henrique Rio

12/11/2020

Motivation

Introduction

Fair elections are an integral part of democracy for, without it, the will of the people might not be respected and if that happens we no longer have a democracy. However, despite the huge importance of the elections in the US, there are still ways that a party can negatively impact the election by gaining an unfair advantage. One of the ways that this can happen, is gerrymandering. This is the practice for unfairly obtaining a political advantage in a state by manipulating district boundaries. This issue has come before the supreme court, however, it was ruled that it was up to Congress and the states to deal with it.

Due to the scale of this issue, the paper by Fifield et al, proposes the use of a Markov Chain Monte Carlo method to automate the redistricting process. Monte Carlo algorithms for redistricting have existed previously, but none of them have a strong theoretical background. Fifield et al algorithm proposes to solve the automation redistricting by formulating it as a Graph Cut problem, which allows the use of the Metropolis-Hastings algorthim to generate MCMC samples of districting plans from arbitrary distributions over the set of n contiguous districts. They also implemented equal population constraint, which is a main requirement in the real-world problem.

What is Redistricting?

"In the United States is the process of drawing electoral district boundaries. A congressional act passed in 1967 requires that representatives be elected from single-member districts, except when a state has a single representative, in which case one state-wide at-large election be held." (Wikipedia). Redistricting also has to follow 6 different criteria.

```
- Compactness
- Contiguity
- Equal population
- Preservation of existing political communities
- Partisan fairness
- Racial fairness
```

The compact criteria pertain to the size of each district where one district cannot be of enormous size and another one be really small, they all have to be more or less similar in size. The contiguity constraint stipulates that all of the districts have to be continuous in an area in the sense that one district cannot cut through the middle of the other one. The equal population constraint determines that the districts must divide the population of the state equally, which may seem simple at first, however, the implementation of this is quite complicated when the precinct's population are factored in since you have to account for each precinct population.

Gerrymandering

Gerrymandering Is the act of actively redrawing district lines so that it tilts as many districts as possible in favor of a party, for a party to cement themselves in power. The redistricting process happens every ten years, right after the U.S. census, in order to update the districts to the changes in the demographic. However, some parties use this opportunity to further cement their power in the state either by dividing the districts where the opposing party has more support to decrease their strength in said district or by adding other precincts district that is dominated by the opposition to increase their power.

One example extreme example of gerrymandering is in Texas in 2003, where, Republicans who had 15 seats in 2002, went to 21 seats in 2004 and Democrats only got 14 seats in 2004. This is a Bipartisan issue, and due to that it was later brought to the supreme court in the *"Rucho vs Common Cause"* case, However, it was decided that it was not a supreme court issue, and therefore, it must be dealt by the congress and the state officials to address it, and regulate it.

Automated Redistricting

Since the 1970s, researchers and scholars have been trying to automate the Redistricting process by using algorithms, since this process is extremely complicated and it is also critical in order to provide a fair election process. Firstly researchers tried to approach it as an optimization problem, where they tried to find a redistricting plan that optimizes a pre-established set of specifications. This can be used to effectively draw the district lines, However, this fails to account for some of the criteria that were previously mentioned (Compactness, Contiguity, etc), since in essence redistricting is a problem that has a lot of geographical properties involved, and requires the researchers to characterize the population distribution in order to fully judge whether a redistricting plan is fair and also meets the required criteria.

Due to the nature of the Redistricting problem, a simulation method would be a lot more appropriate since it allows researchers to better characterize the underlying population distribution, by simulating redistricting plans under different assumptions and constraints. Yet, until recently only a few algorithms existed and those algorithms are all pretty similar in terms of the way they try to tackle the redistricting problem and use almost the same Monte Carlo method. According to Fifield et. al, the way this algorithm work is by selecting a random geographical unit as a seed for each district, and then contiguous units are attached until the pre-defined population threshold is satisfied.

We choose to reproduce the paper from Fifield et. al, *Automated Redistricting Simulation Using Markov Chain Monte Carlo*. This paper implements the MCMC Redistricting Simulation in the state of Pennsylvania, where the goal of the paper is to reduce the partisan bias of redistricting plans to as close as possible to zero while also accounting for the equal population, compactness, and contiguity constraints. In our paper, we reproduce the algorithm in the state of Colorado, with the same goal of zero percent partisan bias. This algorithm uses a different method to simulate redistricting plans as it formulates the problem as a graph-cut problem.

Methodology

Redistricting Problem Representation

Redistricting is represented as a "graph cut" problem, where a state is represented as a graph G, and each voting precinct i is a node in the set of nodes V. There is an undirected edge $i \sim j$ connecting nodes i, j if the precincts i, j are contiguous (share a border). Then, we get voter districts by partitioning the set of nodes V. This leaves subgraphs that represent the congressional districts. Since districts are formed by "cutting" edges, we can think of redistricting as a "graph cut" problem. This is how the MCMC methods generate new samples of redistricting plans.

Metropolis-Hastings

The Metropolis-Hastings algorithm is a Monte Carlo Markov Chain algorithm based on the simple accept-reject algorithm, used to simulate redistricting plans.

- Define a function $f(x)$, proportional to the pdf of the desired distribution
- Define a proposal distribution $g(x|y)$ that provides a new sample x given the previous sample y
- Start with a valid sample, x_0
- At each iteration:
  - Generate a "candidate" sample x^* from g (perturb x_i)
  - Accept x^* with probability equal to $\frac{f(x^*)}{f(x_i)}$

Metropolis-Hastings Iteration Example

To simulate redistricting, "perturbation" looks like swapping precincts along existing district borders. **??** illustrates the perturbation process for an example sample in a Metropolis-Hastings chain. Perturbation works by identifying nodes that exist on congressional district borders and proposing swaps along the borders. This produces a perturbation of the previous sample which is accepted with acceptance ratio

$$\alpha(\pi', \mathbf{CP}|\pi) = \min\left(1, \left(\frac{|B(\mathbf{CP}, \pi)|}{|B(\mathbf{CP}, \pi'|}\right)^R \frac{F(|B(\mathbf{CP}, \pi)|)(1-q)^{C(\pi', \mathbf{V_{CP}})}}{F(|B(\mathbf{CP}, \pi')|)(1-q)^{C(\pi, \mathbf{V_{CP}})}} \cdot \frac{g(\pi')}{g(\pi)}\right)$$

where π is the current sample, π' is the perturbed sample, $\mathbf{CP}$ is the set of connected components formed by π, and C is a graph-cut function.

`redist` Package

Fifield et. al. use a package called `redist` which implements this Metropolis-Hastings algorithm for simulating redistricting. `redist.mcmc` is a function for generating redistricting samples, and requires precinct adjacency matrix, population data, and assignments to a valid set of congressional districts as function inputs. The authors of the paper shared some code for reproduction, but the code was limited to generating figures and it was not generalizable to different data, so we re-implemented the simulation code to generate redistricting simulations for the state of Colorado.

Data

Fifield et. al. use Pennsylvania shapefiles and election data from the Harvard Election Archive. However, the Harvard Election Archive Colorado dataset did not include congressional district assignments. Instead, we used a shapefile assembled by MGGG which was constructed from election data from Colorado Secretary of State's Office, 2010 Census data from the U.S. Census Bureau Data API, and precinct borders from U.S. Census Bureau TIGER/Line shapefile repository.

Simulation

Using the `sf` package, we read the Colorado shapefiles and calculated a precinct adjacency matrix with a package called `spdep`. Then, we used `redist.mcmc` to generate 3 MCMC chains of 10,000 samples each - this took a few hours. We initially tried 3 MCMC chains of 100,000 samples (this is what the authors of the paper do) - but this took to long and required more computing power than was available. We also tried applying a "compactness" constraint, but it requires providing a distance matrix of the nodes in the graph, which took too long to compute. Finally, we evaluate bias in the simulations using `redist.segcalc` and a modified version of the code shared by the authors of the paper.

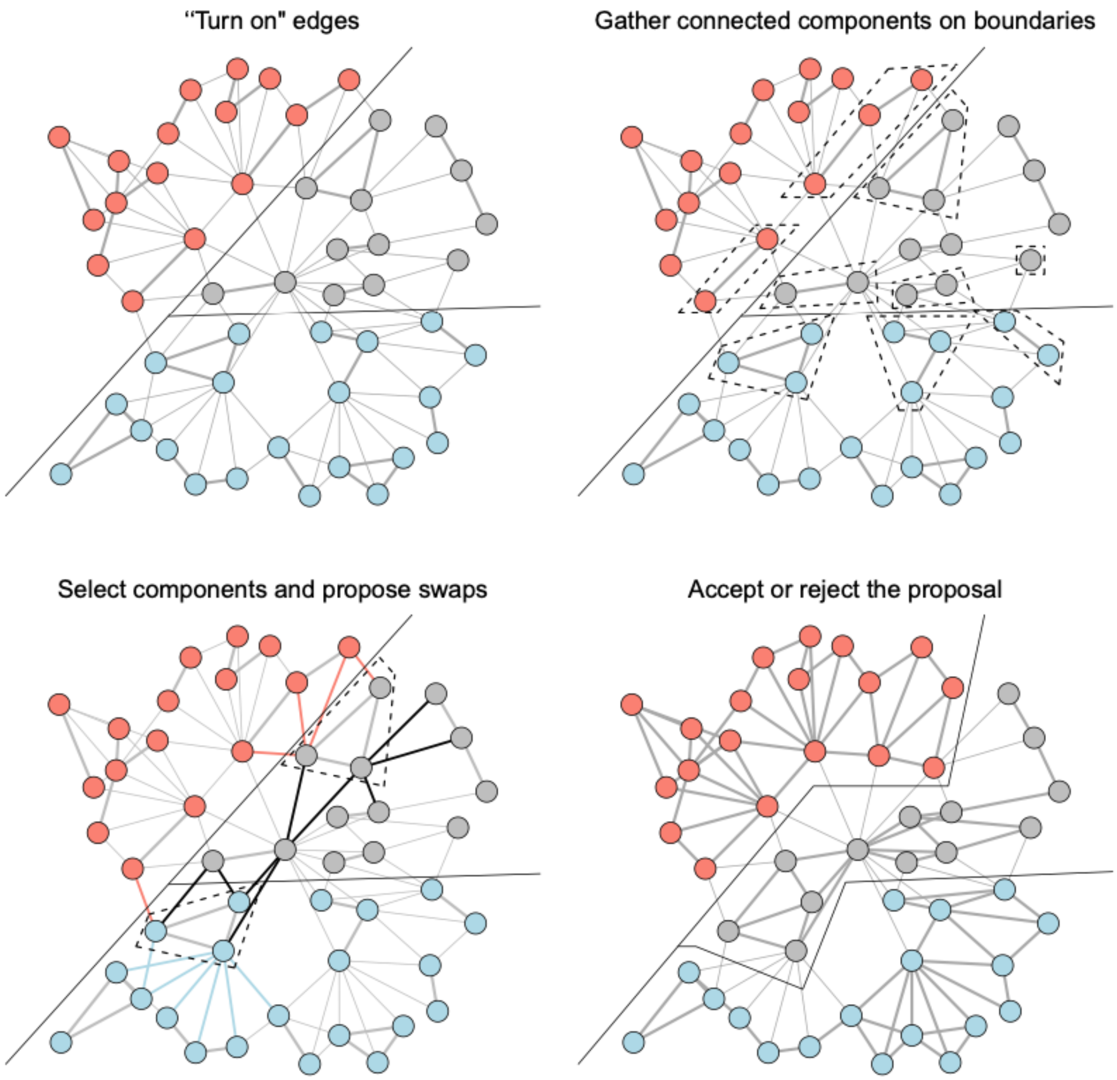

Figure 1: Metropolis-Hastings iteration example (Fifield et. al.)

Results

Starting Map (current districts)

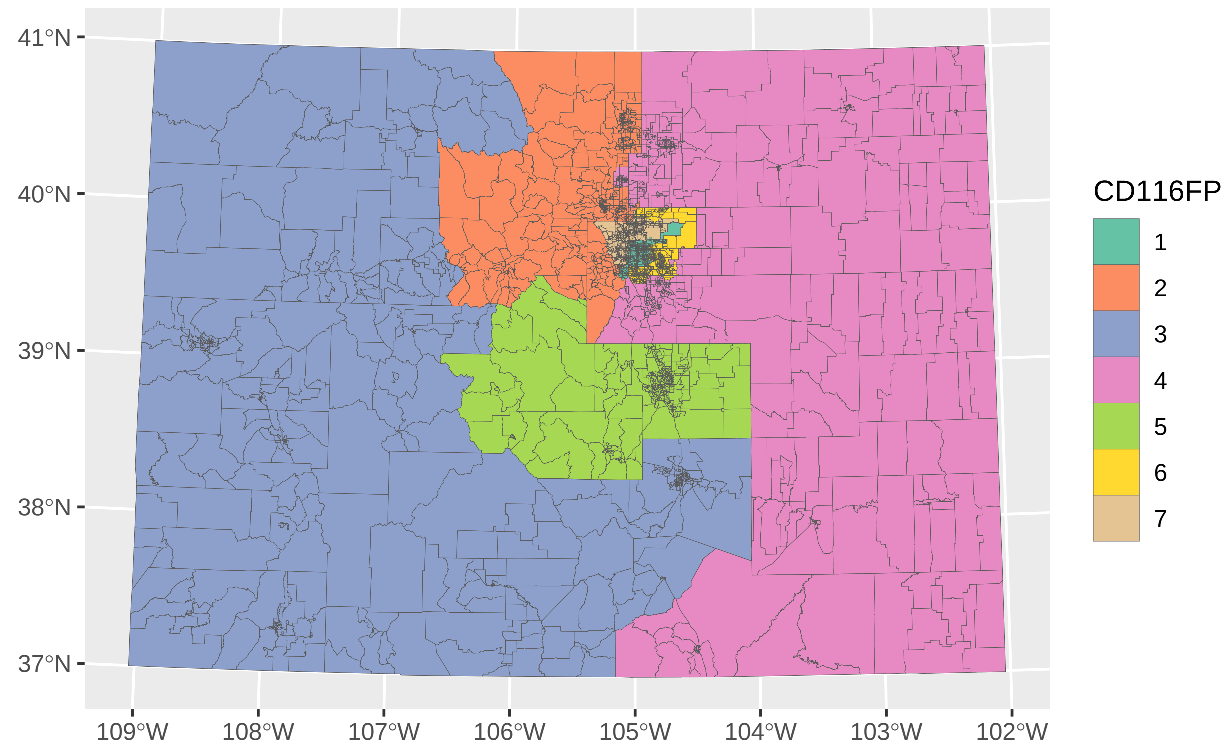

Figure 2: Current congressional districts in Colorado.

In figure 1 above we see the current district map for the state of colorado. The districts are distinguished by the different colors and the legend on the right. The opaque white lines show the precinct lines (there are over 3000 precincts in the state of Colorado), as mentioned in the introduction gerrymandering is the process of moving precincts from one district to another to sway political bias in certain districts or in an entire state. Each district shown above satisfies the equal population constraint stating that every district must have approximately the same population. At the moment politicians are doing this process by hand looking at each precinct's voting history and moving them around to benefit their political party. Redistricting happens once every 10 years after every census. The current district layout exhibits democratic bias which is as expected because democrats controlled the state legislature in 2010 which was the last census year.

MCMC Summary

Now we can look at the bias associated with our simulations. Our simulation consisted of 10,000 different and unique redistricting maps that the algorithm computed using our parameters and equal population constraint. Each point on the graph is a distinct Colorado district map. On the X-axis we have the percentage of precincts swapped for a particular simulation. On the Y-axis we have our calculated political bias. The red dotted line shows 0 bias, above that line is democratic bias and below that line is republican bias. The black line shows the current political bias which is slightly democratic.

There seems to be a connection between political bias and percentage of precincts swapped. When less than 10 percent of precincts are swapped the political bias changes very little and stays democratic but it can sway slightly closer to the ideal 0 bias line. However as more precincts get swapped we start to see a positive trend in the graph where partisan bias grows more and more in favor of democrats. According to our simulations in

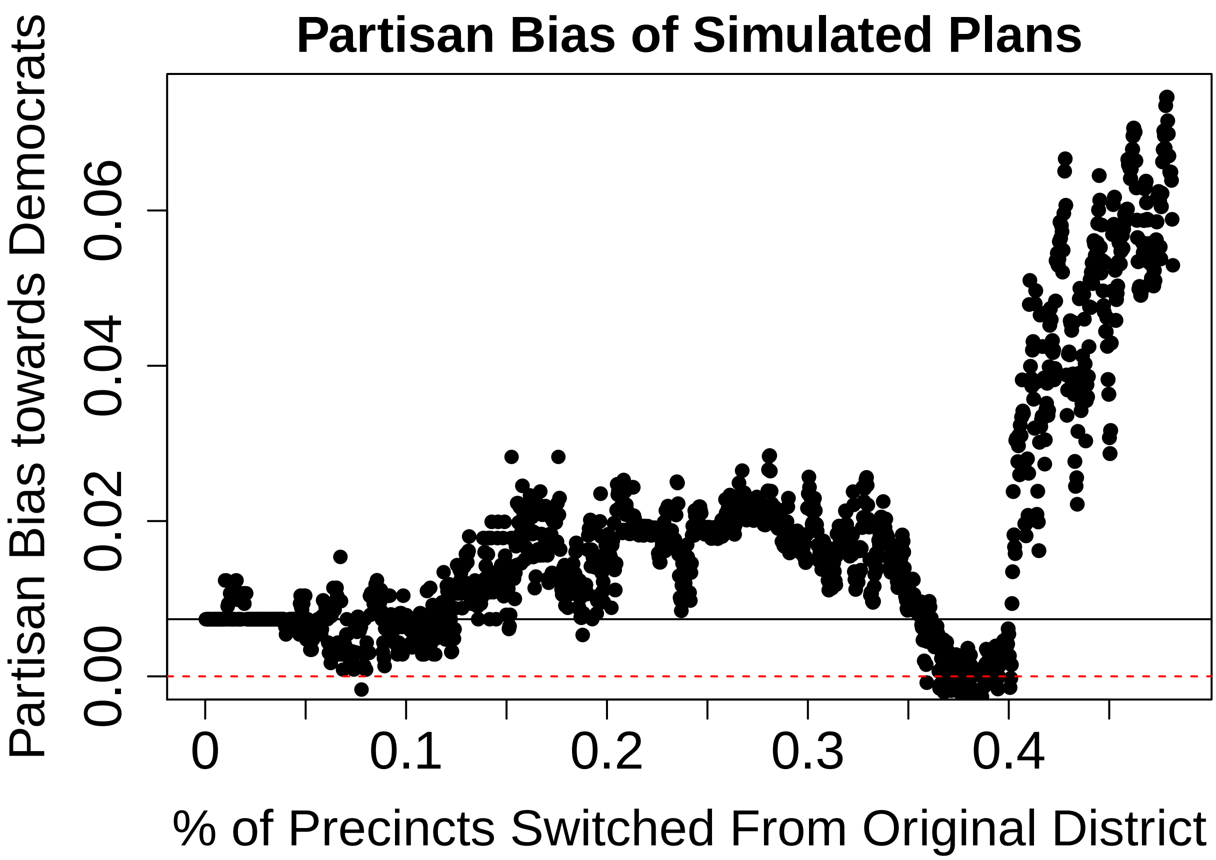

Figure 3: Redistricting samples, percent of precincts changed vs. bias.

order to achieve 0 bias approximately 37 percent of precincts need to be swapped from their original district. The political bias can even become slightly republican with this proportion of precincts swapped. However, the overall trend is that the vast majority of the simulations that were run return a district map of colorado that is even more democratically biased than it currently is. We believe this is due to the fact that Colorado has been trending away from being a swing state and becoming more of a solid “blue” shown by the outcomes of the presidential elections in the past 20 years.

Minimum Bias Map

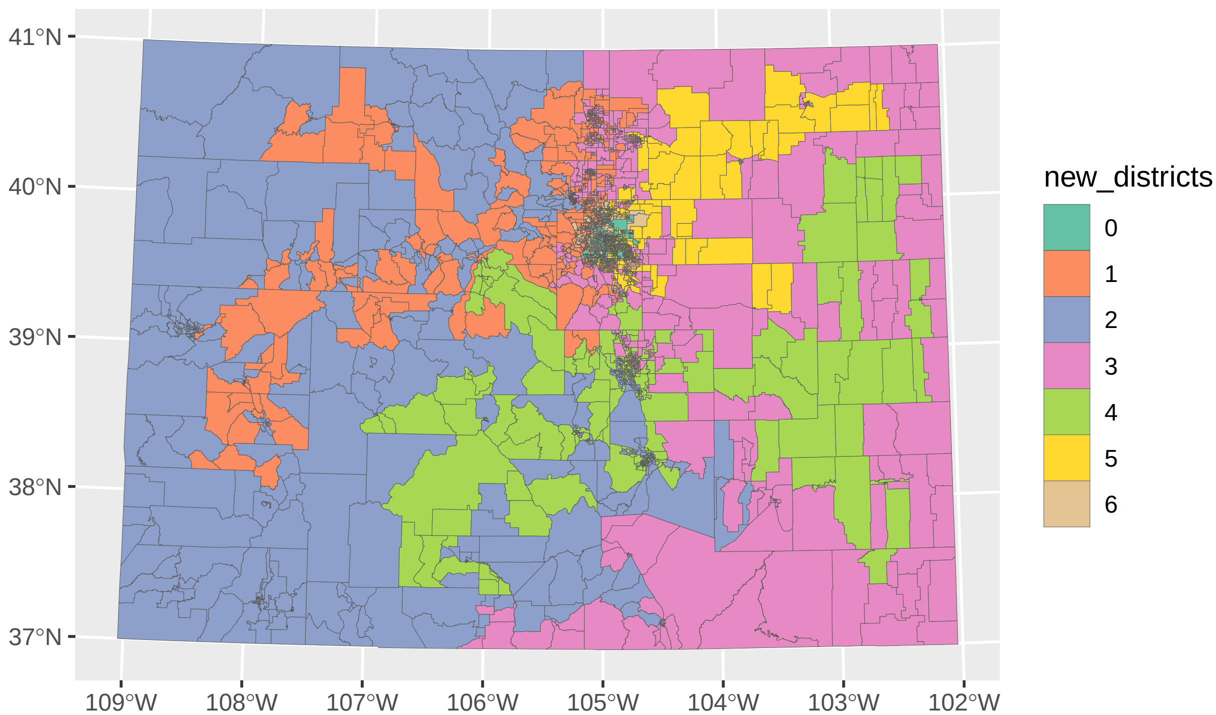

Figure 4: Redistricting sample with minimum bias.

Here is the minimum bias map. This map shows the simulation with the political bias that is closest to 0. This map does satisfy all of the constraints put forth by the state of colorado for redistricting however it may be slightly unrealistic as far as a real world district map. The districts developed a “tree branch pattern” where they seem to branch out across the entire state of Colorado from the center outwards. This is because of a compactness constraint that we were not able to implement due to our limited computing power. The compactness constraint takes in a matrix of pairwise distances between precincts and makes sure that all precincts in a district are not “too” far away from each other. We tried running our simulations using this constraint but it crashed our computers so we decided to run our simulations without compactness. However we think there is something to be learned from this minimum bias map, it shows just how different districts need to look in Colorado to obtain a 0 bias state.

Moving Forward

Moving forward we would like to explore the compactness constraint to attempt to get a more realistic redistricted map of Colorado. Also bumping up the number of simulations could prove to be useful, our limit with the computers that we have is 10,000 runs getting that number up to 100,000 would be ideal. However being able to do those two things would require access to more powerful computers. Also, running this analysis on another state could be very interesting, Texas for example, a historically republican state, has shown a trend of democratic voting increase. It would be very interesting to look into how gerrymandering has been affecting Texas politics.

Sources

Fifield, Benjamin, et al. "Automated Redistricting Simulation Using Markov Chain Monte Carlo." *Journal of Computational and Graphical Statistics* (2020): 1-14.

Altman, M. (1997). The computational complexity of automated redistricting: Is automation the answer. *Rutgers Computer & Tech. LJ*, 23, 81.

`redist` Manual

'sf' Manual

MGGG Colorado Election Shapefile

Harvard Election Data Archive

https://en.wikipedia.org/wiki/Gerrymandering