Detecting Partisan Gerrymandering: Stratified Sampling the Space of Possible North Carolina Congressional Redistricings

by

Luke Farrell, Jacob Shulman

Department of Computer Science
Duke University

Date: ____________________________
Approved:

____
Greg Herschlag, Dept. Mathematics, Supervisor

____
Jonathan Mattingly, Dept. Mathematics

____
Owen Astrachan, Dept.Computer Science

Thesis submitted in partial fulfillment of the requirements for
graduating with distinctions in the Department of Computer Science
of Duke University
2019

Abstract

We attempt to detect partisan gerrymandering by establishing a baseline distribution of electoral outcomes in North Carolinas 2016 Congressional election. We follow the model of Bangia et al. (2017) and introduce a Stratified Sampling method to this domain to improve upon state of the art sampling techniques.

Our implementation balances wide exploration and narrow sampling through a two step procedure. First we stratify the space of all possible NC redistricting plans and then sample from these strata. This procedure is highly parallelizable allowing for more rapid map generation and produces a statistically invariant measure of our final observables. By implementing Stratified Sampling in this domain for the first time, we produce a sample of redistrictings that is consistent with prior findings. Further, the resulting ensemble of 38,000 maps indicates that the 2012 and 2016 enacted plans in North Carolina are atypical and notably unresponsive to changes in voters' party preference.

Contents

1

Introduction to Gerrymandering

Responsive democracies are designed to translate voter preferences into governmental representation. In the locality based representative democracy of the United States, the geography of the districts that a voter belongs to influences this translation. Whether a voter falls in the majority or minority of their district can be directly influenced by how the geographic borders of that district are defined. In aggregate, given precisely the same vote counts in an election, different redistricting maps can produce dramatically different outcomes.

Consequently, the vote counts in an election can be thought to produce a distribution of possible electoral outcomes dependent on the particular district map that was chosen to do the aggregation. The electoral outcome that materializes is a consequence of the map that is chosen. Bad actors could work to game this phenomenon and choose maps that produce tail-end electoral outcomes in their favor. Understanding the underlying distribution of electoral outcomes across all maps is essential in judging the typicality of the outcome that materializes.

Gerrymandering is the practice by which the enacted maps intentionally produce these tail-end electoral outcomes. Whether it is choosing maps that elect fewer-

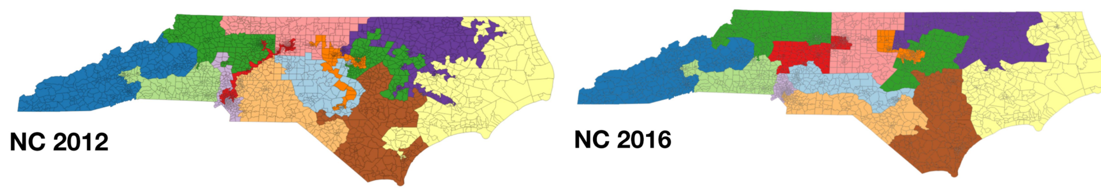

FIGURE 1.1: Both North Carolina Congressional Redistrictings in 2012 and 2016 produced 10R-3D seat splits in spite of approximately 50%R - 50%D popular vote splits. While previous work Herschlag (2018) has shown both to be the tail-end electoral outcomes, the 2012 enacted gerrymander appears to violate compactness criteria more clearly than the 2016 enacted gerrymander. Sophisticated data science techniques allowed for the creation of the more covert 2016 gerrymander. As a result, more sophisticated methods are required to detect it.

than-typical racial minorities or maps that elect more-than-typical incumbents, gerrymandering characteristically produces atypical electoral outcomes given the vote counts in an election. Here we focus on partisan gerrymandering: the practice by which redistricting maps are drawn to disadvantage the opposition party and make electoral outcomes unresponsive to changes in voter party preference.

Partisan gerrymandering often leads to disproportionality between the percentage of votes cast for a party and the percentage of seats won by the same party. For example in 2016 Democrats in North Carolina earned 48.3 percent of the total vote cast in House races but won three of thirteen possible seats, just 23 percent [Astor and Lai (2018)]. Discrepancies like these have motivated gerrymandering detection metrics like the efficiency gap [Stephanopoulost and McGheeft (2015)]. However, proportionality dependent metrics break down when considering the natural partisan geography of a state. For example in Massachusetts, Republicans won zero seats in 2018 despite garnering over 30 percent of the total vote. This discrepancy produces an efficiency gap of +14% in favor of Massachusetts Democrats [Bycoffe et al. (2018)], well above the 8% threshold recommended to detect partisan gerrymandering [Stephanopoulost and McGheeft (2015)]. However under closer examination, when following all legal compliance criteria, no majority Republican district can be produced in the state [Bycoffe et al. (2018)]. In fact, the observed outcome is not atypical at all. Rather the lack of proportionality is a feature of the high uniformity with which Massachusetts Republican voters distribute themselves throughout the state. A lack of proportionality is merely a downstream consequence of tail-end map selection; not a necessary nor sufficient condition to suggest atypicality of an electoral outcome. The same can be said for visual inspections for district compactness [Figure 1.1], which are not sufficient to spot more sophisticated partisan gerrymanders.

The failures of downstream metrics like proportionality and compactness highlight the difficulty of rigorously detecting this phenomenon [Chambers et al. (2017)]. Consequently, more complex techniques have been developed that better distill the problem of partisan gerrymandering to the detection of electoral outcome anomalies. The focus of this work is to build upon one such method designed by Bangia et al. (2017). Specifically we introduce a Stratified Sampling technique to this domain for the first time.

2

Detecting Gerrymandering: A Review

The method outlined by Bangia et al. (2017), models the underlying distribution of electoral outcomes by building an ensemble of possible redisticting maps. Using this method one can then directly quantify the typicality of the outcome produced by an enacted plan. If an enacted plan's outcomes fall significantly far on the tail ends of a distribution, it is suspected to be a partisan gerrymander.

Using this ensemble of possible redistricting maps has demonstrated an ability to capture the partisan geography of a state [Chen and Rodden (2013)] and avoids many of the shortcomings of other detection methods [Chen and Rodden (2015)]. As a result, it has found success as evidence in litigation challenging partisan gerrymandering in North Carolina, specifically in the Rucho v. Common Cause (2018) case.

This method uses a three step process. (1) It runs a Monte Carlo algorithm to randomly generate redistrictings, considering exclusively non-partisan criteria required for legal compliance. We refer to the collection of maps it produces as the *ensemble of maps*. (2) Historical voting data from an election is applied to the ensemble of maps to simulate the number of Democratic and Republican representatives that would be elected on each map in the ensemble. Since each map aggregates the voter counts differently, these variations in the electoral outcomes give a distribution of results. (3) Considering these outcomes in combination produces a distribution of expected election outcomes with which we can judge the typicality of enacted maps.

The first step of this process, building an ensemble of all possible North Carolina redistrictings, is both crucial and challenging. As the number of possible precinct to district assignments for North Carolina's US House of Representatives Congressional Map is intractably large [1], only a very small sample of possible maps can be taken. In spite of this, the sample produced must be *representative* of all possible compliant redistricting maps in order for subsequent claims about the distribution of electoral outcomes to be valid. Ensuring this representativeness is a significant challenge. It is the goal of the MCMC process, the sampling methods we outline below, and our contributions to construct such a sample.

2.1 Formalizing the Problem

We begin the process by creating an abstraction of North Carolina as a graph, $G = (V, E)$ where a node $v \in V$ is an individual precinct, and an edge $(v_1, v_2) \in E$ is present if v_1 and v_2 share a border. From this graph, the next step is to partition the graph into 13 districts, in the case of North Carolina. So, a redistricting plan can now be formalized as a partition of V into 13 sets. Let $\xi : V \to D$ be a function that maps each precinct to a district, so for a precinct $v \in V$, if $\xi(v) = i$ then precinct v is in the i^{th} district. Furthermore, let $D_i(\xi) = \{v \in V | \xi(v) = i\}$ be the set of precincts that compose the i^{th} district. More plainly, ξ is a Congressional redistricting map of North Carolina.

[1] There are more than 13^{2500} ways of partitioning the approximately 2500 precincts of North Carolina into 13 Congressional Districts. For scale there are an estimated 10^{80} atoms in the universe

Under this formulation, it is clear to see the combinatiorial nature and immense magnitude of this redistricting space. However, we are not interested in all possible maps, we are only interested in maps that are ”reasonable” [Bangia et al. (2017)]. In the case of North Carolina, Bangia et al. (2017) define a reasonable map as one in which:

- Districts have equal population (within .01%)
- Districts are reasonably compact
- Counties are split as infrequently as possible
- The redistricting does not violate the Voting Rights Act
- The redistricting is drawn **without the use of any partisan data** including demographic information or past voting data
- The redistricting must be contiguous

These remaining constraints are encoded into a score function J that takes in a redistricting ξ and returns a score, where a lower score represents a map that better follows these required constraints. For each of these constraints we develop an individual score function.

$J_p(\xi)$, the *population score* measures how equally spread the population is across the districts.

$$J_p(\xi) = \sqrt{\sum_{i=1}^{13} \left(\frac{\text{pop}\,(D_i(\xi))}{\frac{1}{13}\sum_{j=1}^{13} \text{pop}\,(D_j(\xi))} - 1 \right)^2}, \quad \text{pop}(D_i) = \text{total population of } i^{th}\text{district} \tag{2.1}$$

$J_I(\xi)$, the *isoparametric score* is a mathematical forumlation of compactness. It returns the sum of the isoparametric ratios of all districts: a function that is minimized by a circle.

$$J_I(\xi) = \sum_{i=1}^{13} \frac{\left[p\big(D_i(\xi)\big)\right]^2}{\text{area}\left(D_i(\xi)\right)}, \quad \text{p}(D_i) = \text{perimeter of } D_i, \quad \text{area}(D_i) = \text{area of } D_i \quad (2.2)$$

Additionally, Bangia et al. (2017) include a county splitting energy and a minority score energy. In our implementation of Stratified Sampling we did not include these scores.[2] $J_c(\xi)$, the *county-splitting score* measures and penalizes split counties. $J_m(\xi)$, the *minority score* measures the degree to which districts with the largest number of African-American voters reach the target percentages set by the Voting Rights Act. Next, we construct weights w_p and w_I that re-scale the output of these score functions, to ensure one does not dominate the other.[3] The score function we implemented $J(\xi)$ is defined as:

$$J(\xi) = w_p J_p(\xi) + w_I J_I(\xi)$$

2.2 Markov Chain Monte Carlo

These scores now give us a way to sift through the immense and intractable number of maps. The roadmap is as follows: we start at an initial map and propose a small incremental change. We decide whether or not to move to that incrementally changed map based on its score. If the changed map has a better score than our current map we are more likely to move to it. If we choose not to move to the changed map we propose another change and repeat. We continue this process, moving throughout the space until we have sampled it sufficiently. To start this process we first convert

[2] We chose not to include county splitting energy and minority score energy as minimize the number of parameters to tune, in order to more easily identify implementation level problems from parameter tuning problems.

[3] Without the inclusion of these weights one can imagine a scenario in which a very low (good) score in isoparametric overshadows a high population score resulting in what appears to be a good holistic score. Weighting ensures that they operate on the same scale.

a map's score into a probability. For each ξ in the set of possible redistrictings R, and for a tuning parameter β the probability of finding a given ξ is defined as :

$$P(\xi, \beta) = \frac{e^{-\beta J(\xi)}}{\sum_{\xi \in R} e^{-\beta J(\xi)}}$$

This parameter β is a positive constant that acts as a lever that tempers the effect of the score function[4]. When beta is low[5] two maps with a different score will have a more similar probability than they would with a high beta. As $\beta \to 0$, the probability distribution over all $\xi \in R$ limits towards a uniform distribution. Finally we set a compliance threshold, such that if the component scores for a given redistricting are below their respective thresholds, we declare the redistricting compliant.

We define the set of all possible compliant maps as R_c. Under this formulation, our goal is now to find a representative sample of all compliant plans $\hat{R}_c \subset R_c$. This ensemble of maps $\hat{R}_c$ is used to produce a distribution of observables $\hat{\pi}(x)$ which should approximate the distribution of observables over R_c, $\pi(x)$ if the ensemble $\hat{R}_c$ is representative. In this domain, the observable we are interested in is the distribution of electoral outcomes in the 2016 Congressional election. This is done using the Monte Carlo Markov Chain (MCMC) algorithm. In order to move through the space we need a way of transitioning from one redistricting to another. In practice we propose a new state by making a small change to our current redistricting: we create a new state ξ' by flipping a single precinct (node) in the graph of ξ from one district to another. The transition probability Q of moving from ξ to ξ' is defined

[4] It may be helpful to think of space as a landscape where maps with higher probability exist in valleys or wells, and maps with lower probability exist on hills or mountains. Under this analogy, increasing beta (decreasing temperature) has the effect of making the mountains taller and the wells deeper, incentivizing remaining in areas of good maps (the wells), while decreasing beta (increasing temperature) has the effect of shrinking the mountains, making them easier to traverse.

[5] Beta is often thought of as 'inverse temperature' where a temperature= $\frac{1}{\beta}$, so a high temperature would be equivalent to a low beta value and vice versa.

by the relative difference in their scores:

$$Q(\xi, \xi') \propto e^{-\beta(J(\xi')-J(\xi))}$$

Now that we can move between states, we can move through the redistricting space, sampling from R.

2.3 Previous Sampling Methods

Constant Temperature Sampling Constant temperature sampling is the most simplistic Metropolis-Hastings sampler. The constant temperature sampling scheme samples while keeping the value of β constant. This sampling scheme can be appealing for computational purposes; however, there are several disadvantages to this method. For example, at a high β the sampler is much more likely to 'downhill' towards higher probability maps. As such it may be practically impossible[6] to get out of a particular region of space. While at a low β it is easier to traverse through space, but the sampler is less likely to come across compliant maps. Because it appears that this energy landscape is highly irregular, a singular beta value is insufficient for producing a representative sample in a reasonable amount of time .

Simulated Annealing This ethos that a singular β value is insufficient leads directly into the next sampling method: Simulated Annealing. The intuition is that we can start with a low β value, at which we are more likely to move from a low score (likely compliant) to a high score (unlikely to be compliant) map[7]. This allows us to quickly move through the space, then once we are sufficiently far away from our last point,

[6] Given infinite time it will be possible to traverse the entire space at any β value, but ensuring that this happens in a reasonable time is the main onus for developing and testing new sampling methods.

[7] The higher beta is, the closer we are to taking a uniform random walk through the space. As beta decreases we are more likely to walk 'uphill' than we are with a high beta value.

we slowly start increasing our β value, descending into a 'lower'[8] area of space that is more likely to have compliant maps. Once in this area, we keep the sampler at a high β and sample the area for (hopefully) compliant maps. This process is then repeated until a sufficient sample of maps are found. This method also has a few drawbacks: first it is difficult to decide how long to stay in each of the three stages, and what to set the β value to for each stage. It is possible that with the first low β value, we are still unable to reach certain areas of space, or that we don't increase beta for long enough, meaning we are unlikely to end up in a region of compliant maps. Additionally Simulated Annealing lacks the *invariant measure* that we have on the space: when we swap the β values in the different stages of the Simulated Annealing, we no longer meet all the required assumptions for our MCMC algorithm. Since we alter the probabilities of transition without regard to the underlying distribution, the invariant measure is lost. This lack of an invariant measure loses us the ability to rigorously claim that we have sampled the space well and have converged to the underlying distribution of compliant maps, R_c.

Parallel Tempering This insight leads us to the next logical step in samplers: Parallel Tempering. The goal is to have the benefits of simulated annealing, where changing β helps us both to traverse the space and sample areas of compliant maps, but to also maintain our invariant measure on the space. The way this is done is by extending our search space to include a dimension for β. For Parallel Tempering, we have n samplers all working at once and in tandem. Each sampler begins at a different β value. As these samplers propose steps to traverse through the graph of redistrictings, they can also propose a 'step' through β space. They can propose to swap β values with one of the other samplers. Because they propose these swaps in

[8] Lower here again refers to the energy landscape analogy where maps with higher probability (lower score) are in wells or valleys, and maps with lower probability (higher score) exist on mountains or hills

β value with the same transition probability as one would between two maps, we do not lose our invariant measure on the space. The intuition here is that the samplers with higher beta values will traverse the space more easily, and through swapping, be able to descend into areas with higher probability maps. The sampler with the lowest β value is the sampler from which we actually gather our sample. This is the biggest drawback for Parallel Tempering: although we utilize many processors at once to run the samplers, we are only sampling from one at a time. This leads us to the sampling method we implemented: Stratified Sampling. With this method we have the ability to parallelize and sample from many areas of space at once.

3

Stratified Sampling

3.1 Overview

Canonically in statistics, Stratified Sampling is the procedure by which one can sample from a global distribution by splitting it into smaller groups. Under the canonical scheme these groups, called strata, must be homogeneous, and every member of the global distribution must be assigned to one stratum and only one stratum. After these strata are created, one can perform simple random sampling on each stratum. Traditionally, this process is done as a way of reducing variance, as it produces a weighted mean of many smaller samples, which has a lower variance than a simple arithmetic mean of a larger single sample. In the context of our MCMC algorithm, we take the flavor of Stratified Sampling but make substantial changes.

Stratified Sampling In looking at the short-comings of the previously explored sampling methods, we can develop a list of traits that we would like to have in a sampler. First, we want the ability to traverse through all regions of space without getting caught in enegry wells. Secondly, we want to maintain our invariant measure. Third, we want the ability to parallelize our sampler to reduce effective run time. Stratified

Sampling checks all of these boxes.

We follow the model of a Stratified Sampling approach originally proposed by Torrie Valleau (1977) that has found widespread use and success in computational chemistry problems where long posterior tails play an essential role (Boczko Brooks 1995; Berneche Roux 2001). Similar to the map sampling problem we are interested in, this class of application suffers from high dimensionality of sample space, multi-modal target distributions $\pi(x)$ and a particular interest in low-probability nodes. As a result both domains experience high computational cost of the $\pi(x)$ evaluation and slow convergence of the MCMC estimate.

The Stratified Sampling procedure seeks to address these challenges by dividing up or stratifying the sample space into many smaller MCMC sampling problems. Overlapping window functions, or strata $\psi_i(x)$, are defined to confine the MCMC walks to their corresponding distributions, $\pi_i(x) \propto \psi_i(x)\pi(x)$. If a selected window contains a high energy region sampling is confined to it by the window functions, which enables much more efficient coverage of low probability areas and ensure the discovery of widely separated peaks in multi-modal landscapes.

After sampling these diverse regions of state space independently, they must be combine to approximate the whole of the region sampled (Dinner et al. 2017). To do so, we use an implementation of the Eigenvalue Method for Umbrella Sampling (EMUS) described by Matthews (2016) and originally proposed by Thiede (1977) and Dinner et al. (1977).

Implementation Overview: Here we adapt the Stratified Sampling technique to this domain for the first time. Through this application we seek principally to produce a representative sample of all possible compliant maps in North Carolina. Since the space of possible North Carolina redistrictings is intractably large it cannot be subdivided prior to sampling. To solve this problem we introduce a two-phase

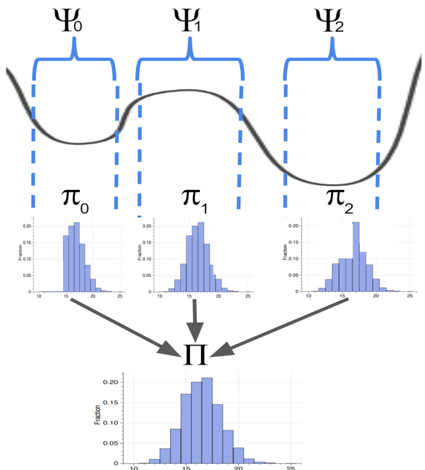

Figure 3.1: Stratified Sampling works by defining acceptance functions ψ_i that bound the space into strata. Within each of these bounded strata an MCMC sampler runs and produces a local distribution of observable $\hat{\pi}_i(x)$ (in this domain electoral outcomes). These local distributions are then merged into a global observable distribution $\hat{\pi}(x)$.

technique to produce the sample. In *phase1* we perform an MCMC walk and learn how to subdivide the space. These overlapping subdivisions of the space are called *strata*[1]. In *phase2* we perform an MCMC walk inside each strata. From each of these walks we collect a sample of maps and generate a corresponding distribution of electoral outcomes $\hat{\pi}_i(x)$ for each strata. Finally we merge these local distributions to produce our final global distribution of outcomes for the vote counts in the 2016 Congressional election in North Carolina $\hat{\pi}(x)$. We compare our final sample of maps and it's resulting global distribution with enacted redistricting plans and ensembles from other sampling methods.

3.2 Phase 1: Learning Strata

Goal: In *phase 1* we aim to generate numerous strata that subdivide the space of possible redistrictings. The goal is to define the regions of space that contain a representative sample of all compliant redistricting maps into strata. *Phase 1* does not seek to actually collect a sample of maps, so encouraging movement throughout the space is more important than focusing exclusively on low energy regions. Through the techniques implemented in *phase 1* we learn both the location and size of the

[1] In this case our strata will be defined as spheres that bound regions of space.

strata that will later be explored in *phase 2*.

3.2.1 Distance and Strata Definition

A strata is defined by a window function $\psi_i(\xi)$, and its centroid: C_i. $\psi_i(\xi)$ determines whether or not a given map ξ is contained within the i^{th} strata. And C_i is the map located at the center of the i^{th} stratum. Additionally below we define the Metric Space (R, ρ) as well our window function $\psi_i(\xi)$.

Defining Distance: To calculate the distance between two maps ξ_1 and ξ_2, we first create a 13-dimensional vector for each map. Each element in the vector is composed of a tuple of latitude and longitude pairs. Each of the 13 districts is assigned a latitude and longitude based on the average latitude and longitude of all of the voting precincts assigned to it. This generates a 26-dimensional vector (13 districts by 2 geographic-dimensions) with which we can embed states in space.

$$\rho(\xi, \xi') := \sum_{j=0}^{13} (\xi_{xj} - \xi'_{xj})^2 + (\xi_{yj} - \xi'_{yj})^2, \quad \xi_{xj} = \text{latitude}(\xi_j), \quad \xi_{yj} = \text{longitude}(\xi_j)$$

Defining Phase 1 Strata: With our metric space we can now compute the distance between a strata's centroid map C_i and some proposed map ξ which will be used by our window function ψ_i to constrain the MCMC walk within a strata. The function $\psi_i(\xi)$ rejects proposed redistricting plans if $\rho(C_i, \xi)$ falls beyond some predefined radius r. If a proposed redistricting falls within r it is accepted only if it meets contiguity criteria and would have been accepted if radius were not a factor[2].

$$\psi_i(C_i, \xi) := \begin{cases} \text{accept} & \rho(C_i, \xi) < r \\ \text{reject} & \rho(C_i, \xi) \geqslant r \end{cases}$$

[2] See A.1 for an analysis of radius definition in $\mathbb{R}^{26}$

In this implementation a strata centroid C_i is defined by a previous step that did not satisfy ψ_{i-1}. This previous plan, however, satisfied all other step acceptance criteria (e.g. contiguity). After the $i-1^{th}$ strata terminates we select the most recently rejected redistricting plan to become the centroid C_i for the i^{th} strata and the same sampling procedure begins, though this time confined by ψ_i. We use this strata creation criteria for simplicity, guaranteed overlap between strata, and for its tendency to follow energy gradients [3]. When instantiating the first strata, we begin with either the North Carolina Judge's Map[4] or the enacted North Carolina 2016 Map.

3.2.2 Dynamic Radius Allocation

Constraining strata by a constant predefined radius r proved ineffective at allowing for proper exploration. A constant radius relies on the fact that states are embedded throughout this $\mathbb{R}^{26}$ vector space in roughly equal density. This is not the case as seen in 3.2. Many steps in the MCMC process have order of magnitude variability in their distance from their strata centroid based on the region of space they are exploring. This leads to the case where the radius is either too large and the sampler never attempts to leave the stratum or the case where the radius is too small and nearly all steps are rejected due to being outside the boundary. To solve this problem, we introduce on-the-fly dynamic radius allocation. By monitoring the behavior exhibited inside a strata during the MCMC walk we allow for either the reduction or expansion of a stratum's radius during *phase1*. We define each ψ_i with a radius r_i equal to a

[3] In future implementations a better heuristic for new strata selection could be devised. Perhaps calculating the energy gradients of all exit points to inform strata selection could be possible. Additionally, we do not constrain ψ_i based on it's overlap with neighboring strata. In the future some optimal volumetric overlap could be calculated when defining new ψ_i

[4] The NC Judges map was created by a non-partisan panel of North Carolina Federal Judges following only legal compliance criteria. As a result it can be used as a good benchmark for what a possible non-gerrymandered map might look like.

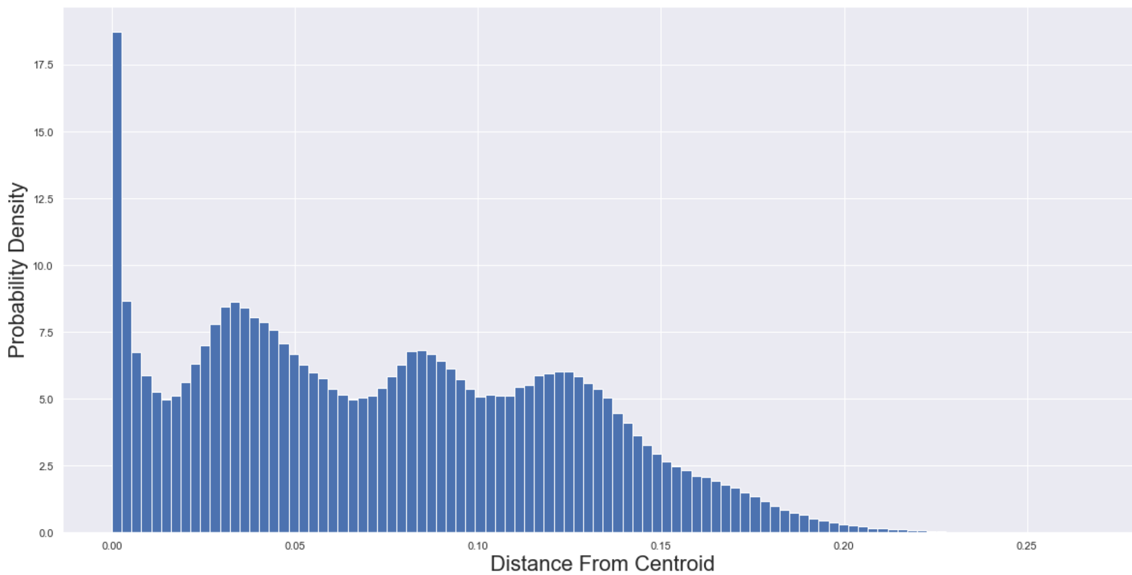

FIGURE 3.2: The average distance of steps taken from a given centroid C_i varies greatly in different regions of space. Some centroids have many proposed redistrictings densely packed in close proximity, while others exist in sparse regions of space. Here the distance between 2 million proposed steps and their corresponding centroid are presented in a probability density histogram. Not visible are the thousands of steps taken at an order of magnitude lower distance (all contained in the first bin). This variability in distance makes a fixed radius inappropriate.

default r and then adjust r_i if necessary as the MCMC walk proceeds. [5] We define an expansion factor EF and reduction factor RF to monitor this behavior.

$$EF = \left\lfloor \frac{|\text{exitSteps}|}{\text{exitThreshold}} \right\rfloor$$

$$RF = \left\lfloor \frac{|\text{consecutiveInsideSteps}|}{\text{insideThreshold}} \right\rfloor$$

$$r_i' = r * 2^{EF-RF}$$

consecutiveInsideSteps is incremented with each MCMC accepted step that is also

[5] After much parameter tuning we set a default radius of 0.1. This balanced the frequency with which reduction and expansion of r_i occurred during $phase1$.

accepted by ψ_i. *exitSteps* is incremented whenever ψ_i rejects a step and it is simultaneously rejected by the MCMC process. When this occurs *consecutiveInsideSteps* is reset to 0. Thresholds are set to allow the sampler to explore within the newly sized stratum and to globally balance expansion and reduction occurrences[6]. This function for r_i allows the definition of radius and consequently ψ_i to be dynamically updated. Empirically this ameliorated the problem of the MCMC process never attempting to exit the stratum in high density regions of space (a step that is required to continue making new strata) when radius was too big. Additionally, it prevented the case in which radius was too small that all of the samplers steps were outside the radius and therefore rejected, leading to a deadlock.

3.2.3 Traversing the Energy Landscape

During *phase1* we aim to subdivide regions of space into strata such that the regions contain a representative sample of all compliant redistricting maps. To ensure that we reach all of these regions we chose to run *phase1* at a hot temperature ($\beta = 0.01$).[7] We are not concerned with the sample of maps produced by *phase 1*, only that the strata it produces can produce a representative sampled in *phase 2*. In 1.1 low β consistently produces new strata, while MCMC processes run at higher β seem to get stuck and stop creating new strata. Furthermore, not only are more strata produced by low β but the centroids produced are more geographically diverse [8].

---

[6] Tuning these threshold parameters resulted in a *exitTheshold* of 500 steps, and insideTheshold of 10,000 consecutive internal steps

[7] High temperatures (low β) are good at quickly traversing regions of space and climbing large energy deltas. This behavior is ideal for *phase 1.* Low temperatures (high β) are good for generating many maps in low energy regions of space. This behavior is ideal for *phase 2.*

[8] Producing strata centroids that are more geographically diverse is a sign that we are "mixing" well. Suggesting that we are traversing the space effectively and finding regions that contain variable maps. See A.3 for more on centroid mixing

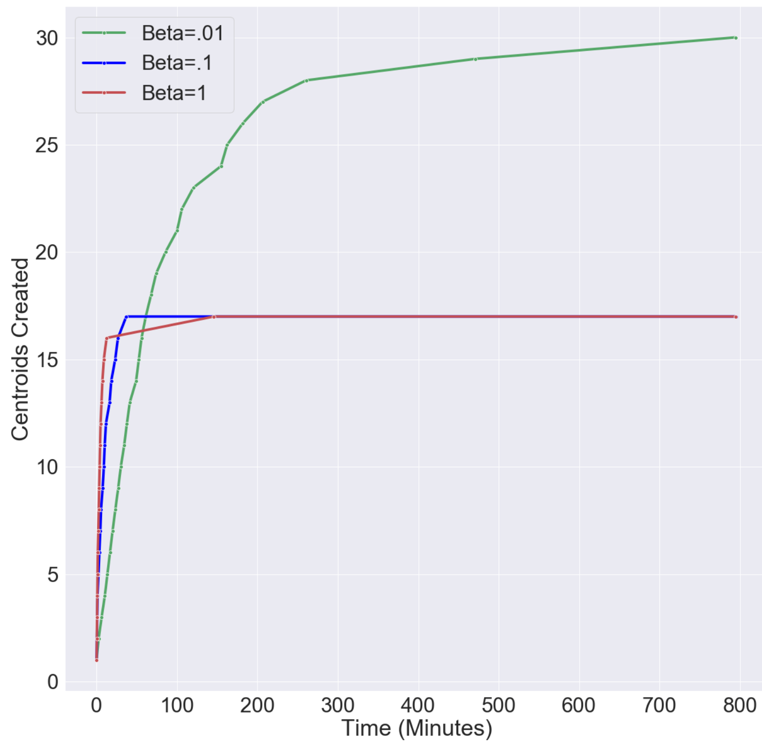

FIGURE 3.3: Here the effect of beta value on *phase 1*'s ability to create new strata to explore is shown. While low β at 0.01 is slower to create new strata early on, it ultimately creates more strata over the long-term. We hypothesize that this is a result of higher β (0.1 or 1.0) following energy gradients more closely and consequently move faster when the energy gradient is clear, but slow down once they reach energy wells. Since the lowest β most consistently explored the space and produced the most strata, β of 0.01 was the most aligned with the goals of *phase 1*

3.3 Phase 2: Sampling Strata

Goal: After defining the strata we want to explore in *phase1*, we aim to sample each strata in *phase 2*. Additionally, we seek to measure the overlap between strata and the movement of MCMC steps between them. This information is later used to merge the individual electoral outcome distributions for each strata $\pi_i(x)$ into a global distribution $\pi(x)$. In a highly parallelizable process we iterate through all of the strata, calculate this overlap, and explore them in depth to produce a sample of redistricting maps.

Algorithm 1 Stratified Sampling: Phase 1

Let X_n be the MCMC process
Let a strata be defined by ψ_i, the window function with centroid C_i and radius r_i
Let S be the collection of strata ψ_i discovered.
Let EF and RF be the expansion and reduction factors as defined previous.

Define first strata ψ_i
Begin X_n at starting map C_i
for proposed map $\xi \in X_n$ **do**:
    $r_i = 2^{EF-RF}$
    if $\psi_i(\xi)$ = Rejected **then**
        if ξ is Compliant **then**
            $C_j = \xi$
            $EF = RF = 0$
            Define new strata ψ_j
            $S+ = \psi_j$
        end if
    end if
end for

return S

3.3.1 Defining Phase 2 Strata:

Using the strata centroids C_i and radii r_i learned in *phase 1* along with the same metric space (R, ρ), we redefine ψ_i in *phase 2*. We now take a probabilistic approach to ψ_i. The function $\psi_i(\xi)$ rejects probabilistically based on the magnitude of $\rho(C_i, \xi)$. As $\rho(C_i, \xi)$ grows the probability of rejection increases. The radius r_i learned in *phase1* now serves as the standard deviation parameter[9] on the acceptance probability function:

$$P_{accept}(C_i, r_i, \xi) := e^{-\left(\frac{\rho(C_i,\xi)}{r_i}\right)^2}$$

$$\text{Let } x \sim \text{ Uniform}(0, 1)$$

$$\psi_i(\xi) := \begin{cases} \text{accept} & x < P_{accept}(C_i, r_i, \xi) \\ \text{reject} & x >= P_{accept}(C_i, r_i, \xi) \end{cases}$$

[9] The the PDF created is centered at $\rho = 0$ and with C_i at its center. As steps move away from the centroid the probability they they will be rejected increases with a standard deviation of r_i. This significantly softens the rejection criteria from *phase*1

Defining strata with a Gaussian probability ψ_i causes sampling to occur more heavily near the centroid[10] and allows for sampling to occur beyond the radius r_i with low probability. As compared to hard constraints, this allows for more overlap between strata and prevents behavior where the MCMC process over-samples the boundaries of a strata [11]. In addition to the acceptance probability from ψ_i a step must also be accepted by the MCMC transition probability, and must be contiguous. In order to combine all of $\hat{\pi}_i$ in the final step of the Stratified Sampling algorithm, we must record the overlap between each strata. In our current implementation we define this overlap as an overlap integral. Let X_{i_n} be defined as the n step markov chain walk taken by one sampler in strata i. Let F be our overlap matrix such that $F[i][j]$ represents the overlap between strata i and strata j.[12] For each $X_{i_k} \in X_{i_n}$ we calculate the probability that the current ξ would be in strata j and update the overlap matrix:

$$F[i][j] += P_{accept}(C_j, r_j, \xi)$$

We will ultimately use these values to appropriately weight each $\hat{\pi}_i(x)$ so that we can create the global $\hat{\pi}(x)$ distribution.

3.3.2 Building Local Ensembles of Maps:

For each strata, using this ψ_i, an independent MCMC sampler starts at the centroid and samples until a finite number of maps are discovered.[13] We run these samplers at a much cooler temperature ($\beta = .9$). The intuition here is that now that we have

[10] Since the sampler is more likely to be rejected while stepping towards the bounds, the Gaussian has the effect of pushing the sampler back towards the middle of the stratum

[11] We hypothesize this to be the case when strong energy gradients force the MCMC sampler to stick to the boundary of a strata instead of exploring the entirety of the space it is assigned.

[12] In theory $F[i][j]$ should equal $F[j][i]$, but due to the randomness of the walks this is not always the case. Empirically they seem to be reasonable close.

[13] Currently each strata is sampled until either 100 maps are generated or an exit criteria is reached (typically 10 million consecutive steps are rejected). Future implementations should sample strata until chain convergence is achieved [?]

defined the areas we want to explore, we want to find the compliant maps within each stratum. *Phase2* produces collections of redistricting maps local to each strata learned in *phase 1* and a single matrix defining the overlap between each pair of strata.

Algorithm 2 Stratified Sampling: Phase 2

```
Let ψ_i be the window function for strata S_i in the set of all strata S.
Let X_{i_n} be the MCMC process in S_i
Let R̂_{c_i} be the set of maps sampled in S_i
Let F be the strata overlap matrix.

for each S_i ∈ S do:
    Begin X_{i_n} at start point C_i
    for proposed map ξ ∈ X_{i_n} do:
        if ψ_i(ξ) = Accepted then
            for each S_j ∈ S do:
                F[i][j] += P_accept(C_j, r_j, ξ)
            end for
            if ξ is Compliant then
                R̂_{c_i} += ξ
            end if
        end if
    end for
end for

return R̂_c , F
```

3.4 Merging Observable Distributions

In order to merge the $\hat{\pi}_i(x)$ into our desired $\hat{\pi}_i(x)$ we have to use the overlaps we calculated during the sampling phase. Each $F[i][j]$ represents the overlap between strata i and strata j, calculated while in strata i. We first have to normalize this matrix columnwise, so each column $F[i]$ is a valid PDF for the distribution of overlap. After each column is normalized, we want to find the eigenvector z summing to one of F. Finally we can construct our desired global distribution:

$$\hat{\pi}(x) = \sum z_i \hat{\pi}_i(x)$$

Algorithm 3 Stratified Sampling: Merging Strata

Let F be the strata overlap matrix.
Let $\hat{R_{c_i}}$ be the ensemble of maps sampled from strata i.

for each column j in F **do**
  F[j] /= sum(j)
end for

Let $\hat{\pi}_i(x)$ = observable distribution on $\hat{R_{c_i}}$
Let z = Eigenvector of F
Let $\hat{\pi}(x) = \sum z_i \cdot \hat{\pi}_i(x)$

return $\hat{\pi}(x)$

4

Discussion

4.1 Results

The Stratified Sampling method we designed and implemented produced a final ensemble of 38,000 redistricting plans. We produced two samples following the entire Stratified Sampling procedure including *phase1*, *phase2*, and the merging procedure. Each of the two began with a different seed map from which sampling began, either the NC Judges Map or the NC 2016 enacted map. The *Phase1* run from the NC Judges seed produced 200 strata, the NC 2016 seed produced 180 strata, and in each subsequent *Phase2* 100 maps per stratum were generated for the ensemble. On the final ensemble of 38,000 maps we aggregate the 2016 Congressional vote counts to produce distributions of election outcomes for each of the 380 strata. We then weight each strata distribution by the eigenvector produced from the overlap matrix to generate the final global distribution of electoral outcomes.

To visualize these outcomes in a way that distills the characteristic packing and cracking behavior of gerrymandering, we borrow a visualization developed by Bangia et al. (2017) which can be seen in figure 4.1. This visualization orders districts from the most Democratic to the most Republican for each map in the ensemble. It then

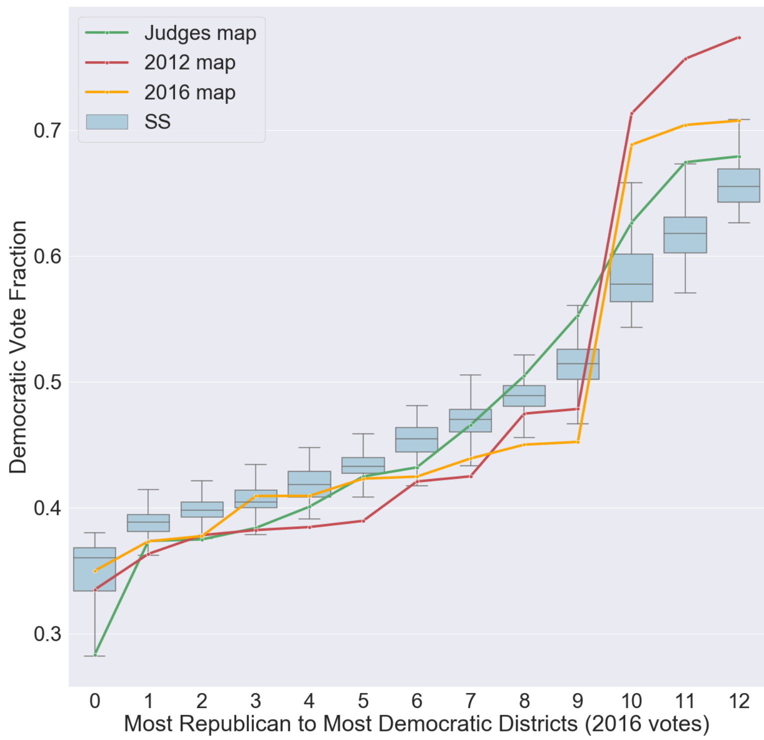

FIGURE 4.1: This figure displays our observable of interest: the distribution of the percentage of Democratic votes cast in each of North Carolina's 13 Congressional Districts. Ordered from most Republican to most Democratic, each boxplot is generated by aggregating the votes cast in 2016 Congressional election according to each redistricting plan in our ensemble of maps. Each boxplot shows the range of outcomes in each district that we consider typical. The non-partisan Judges plan seems to be typical in most districts while NC2012 and NC2016 have more Democrats than typical in the most highly Democratic districts (evidence of packing) and fewer Democrats in districts which are closer to parity between Democrat and Republican (evidence of cracking).

computes the Democratic vote fraction inside each of these ordered districts and produces a distribution of vote fraction across the ensemble. We exclude outliers from these distributions in order to make them more clear, but full distributions shown as violin plots can be found in A.4. In 4.1 the median values of the distribution of Democratic vote fraction across the 13 districts increase in a relatively smooth and linear fashion from most Republican to most Democratic. The same can be said for the NC Judges map. [1]. However the NC 2012 and NC 2016 maps follow a much less linear trend. In the three most Democratic districts these enacted plans are far above the range of the ensemble distribution, producing districts with much higher democratic vote fractions than typical (packing). In the districts that fall close to party parity in the ensemble and consequently could be flipped Democrat or Republican, the enacted plans fall below the range of the distribution, producing districts with much lower democratic vote fractions than typical (cracking). In combination a large jump between packed and cracked districts is present in the graph as compared to the medians of each distribution and the NC Judges map.

In 4.2 we compare the output of the ensemble produced by Stratified Sampling and the ensemble produced by a previously implemented technique; Simulated Annealing. The ensemble produced by the Simulated Annealing procedure is comprised of 2,000 compliant maps and serves as a good benchmark for validating our sample. As seen in the visualization 4.2, while the medians of both ensembles follow a similar trend and find NC2012 and NC2016 to be atypical, they are dissimilar in many instances. This dissimilarity between methods brings into question the representativeness of the sample we produce and its corresponding outcome distribution.

To further assess whether our method produced a representative sample of all possible compliant maps, we compare the final output distributions of sampling procedures started from different seed maps. The output distributions from samples starting at the NC Judges map and the NC 2016 enacted map are visualized in 4.3. While both show the NC 2016 and NC 2012 plans to be atypical and follow a relatively smooth trajectory, they are dissimilar in many districts. This dissimilarity suggests that we are not sampling the space sufficiently well in either sample alone. This may be a result of not running *phase1* sufficiently long to discover strata that

[1] Since the NC Judges map was created by a non-partisan panel of North Carolina Federal Judges following only legal compliance criteria, it is a good benchmark for what a possible non-gerrymandered map might look like.

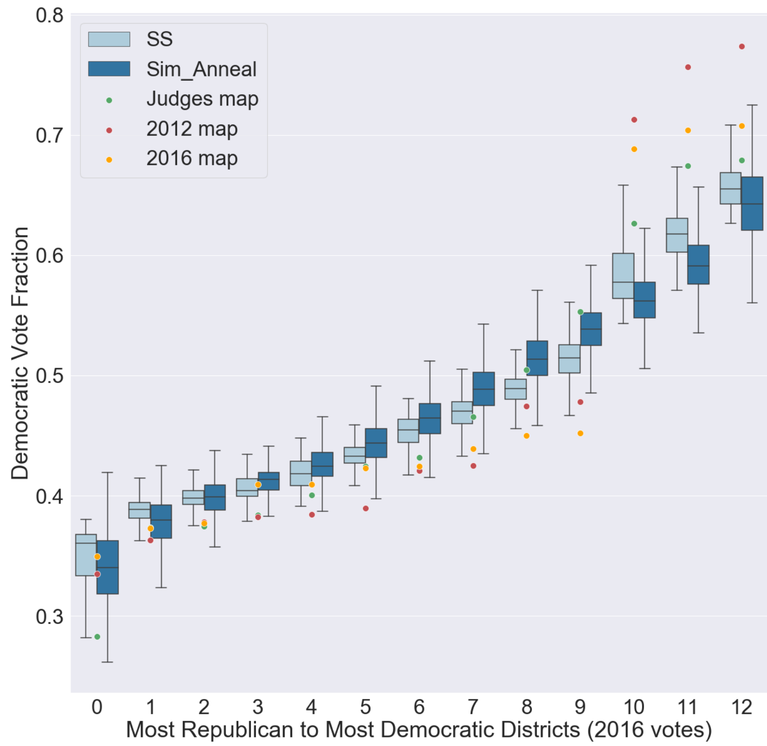

FIGURE 4.2: This figure displays the same electoral outcome distributions produced by the Stratified Sampling ensemble as in 4.1 in the light blue boxplots. In the dark blue boxplots we present the outcome distributions for an ensemble generated through Simulated Annealing produced by previous work [Herschlag (2018)] here we present the distribution of the percentage of Democratic votes cast in each of North Carolina's 13 Congressional District.

contain a representative sample of all compliant maps or it may be a result of not running *phase2* sufficiently long enough to explore each strata fully. While it is still unclear at this time which is the culprit, moving towards chain convergence [Raftery and Lewis (1992)] as stopping criteria will help ensure that we are sampling the space well. Hyper-parameter tuning may still require further attention. Specifically, while we develop a heuristic for how to set the β value in each phase in 3.3, more tuning is likely necessary.[2]

[2] We believe that *phase1* could be served by an even hotter temperature sampling at β below 0.01. Preventing or delaying the asymptotic behavior in strata creation seen in 3.3 will likely help full exploration of the space

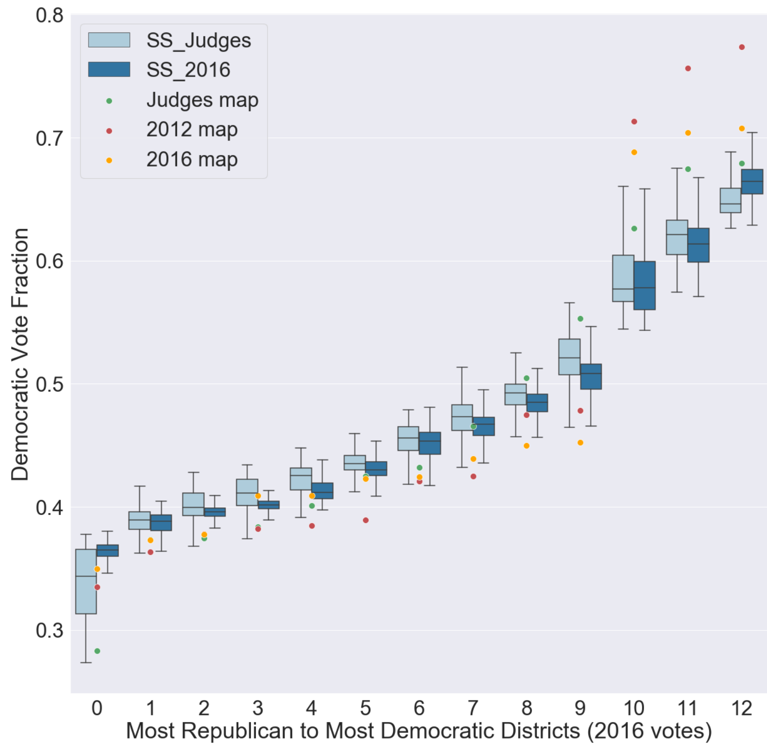

FIGURE 4.3: This figure displays the same electoral outcome distributions as 4.1 but breaks our ensemble into two halves based on the starting-point of the Stratified Sampling run. The light blue boxplots began at the non-partisan Judges redistricting plan, while the dark blue boxplots began at the NC2016 enacted redistricting plan. The similarity of the distributions created by each of these independent samples, suggests that we are sampling the space well.

4.2 Conclusion

The key challenge that any sampling method faces when attempting to collect a representative sample of all compliant redistrictings is to balance wide exploration and narrow sampling. The two phase Stratified Sampling technique we design and implement here addresses this problem directly. In *phase1* we perform wide exploration at low β in order to create strata. In *phase2* we focus in on those strata and sample them at high β. Innovations like dynamic radii and probabilistic strata window functions addresses the high variability of the energy landscape and the density of maps in space. As a result, this method produces electoral outcome distributions some-what consistent with previous methods and additionally detects atypical partisan advantage in the North Carolina 2012 and 2016 enacted maps. While the resulting distributions appear to require tuning, the application of Stratified Sampling to this domain shows great promise.

Appendix A

Appendix

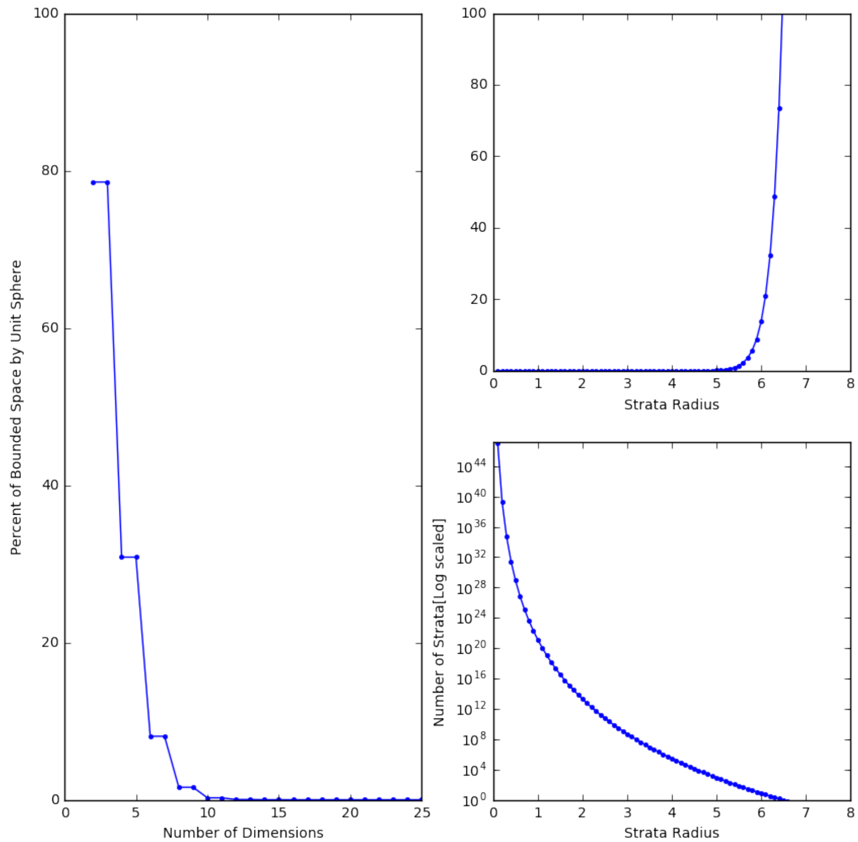

FIGURE A.1: Here we first show how high dimensionality affects the volume of a unit sphere in bounded space. In the first panel it is clear that as dimensionality climbs the percentage of volume contained by the unit sphere rapidly decreases. The panels on the right show how the volume contained by a sphere scales with radius in space defined by ρ in $\mathbb{R}^{26}$. Our default radius size of 0.1 would require approximately 10^40 different spherical strata to cover the entirety of the state space. This is okay because our goal is not to cover all of state space but rather to collect a representative sample of all compliant maps, which exist in a relatively small volume of space

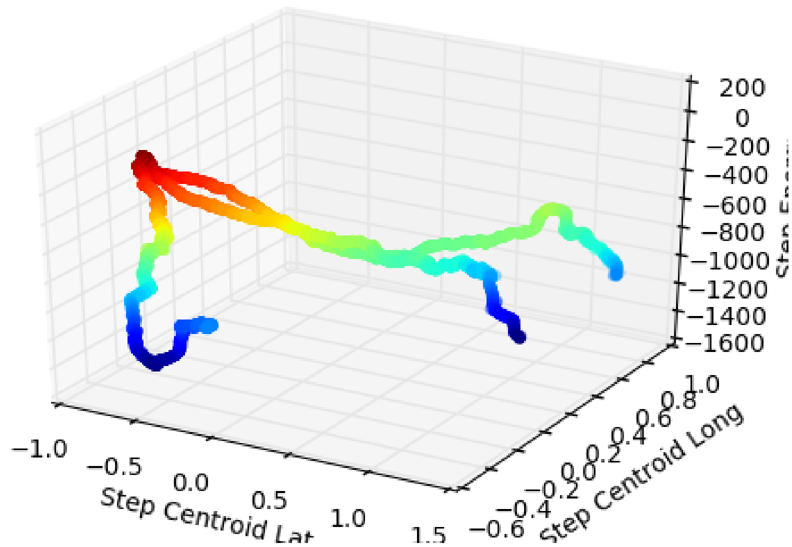

FIGURE A.2: Here we reduce ρ in $\mathbb{R}^{26}$ to $\mathbb{R}^2$ using principle component analysis (PCA) and plot them on the x and z axes and plot step delta energy on the y axis. This provides a 3D image of the MCMC process stepping through space following the energy gradient towards states with lower energy. The three different chains represent three different processors walking independently through state space. They each originate at the same start point (the 2016 judges map) and quickly diverge as they randomly walk. This divergence is characteristic of walking through high dimensional space. The walk illustrated is not bounded by any strata and occurs at constant temperature.

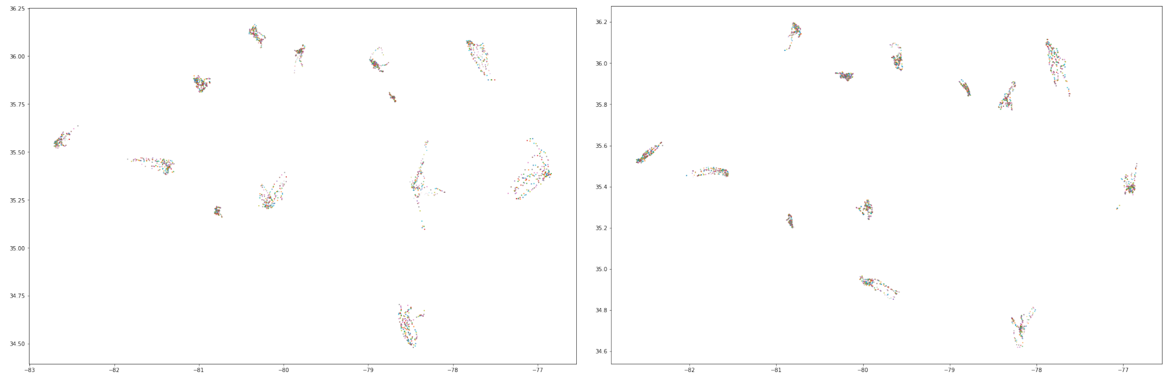

Figure A.3: Here we plot the latitude and longitude of the 13 districts for each stratum centroid. The variability of these points suggests mixing of maps and sufficient difference between strata. The mixing plot on the right was created by a *phase 1* run at a *Beta* = 0.1. The mixing plot on the left was created at a hotter temperature by a *phase 1* run at a *Beta* = 0.01. The strata centroids created by the hotter *phase 1* display better mixing and more heterogeneous strata. This supports our decision to run *phase 1* at a hotter temperature with a *Beta* = 0.01

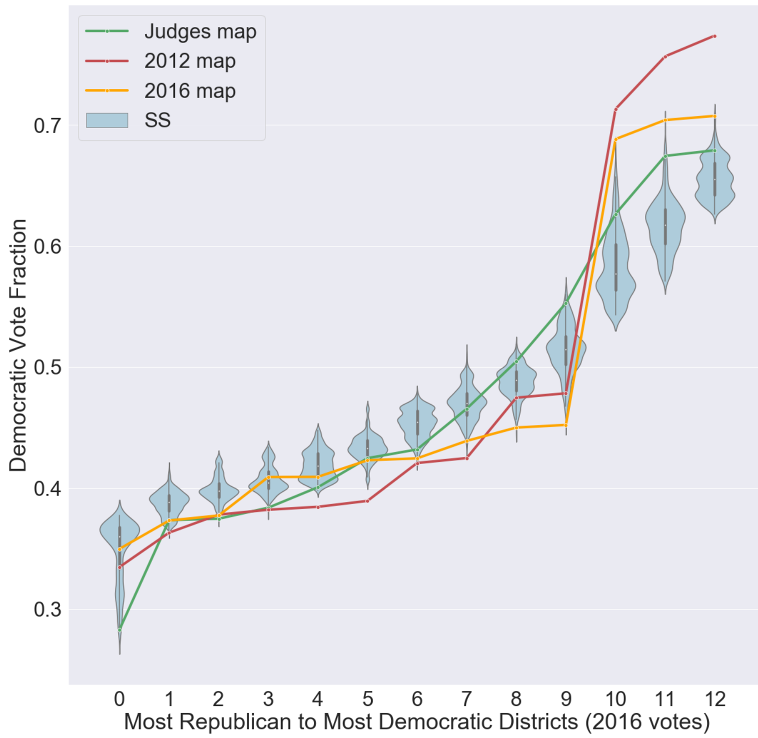

FIGURE A.4: These violin plots show the same data as 4.1 but in greater detail. The distributions show a bit of irregularity for several of the ordered districts. This is a signal that better mixing must occur and that this method may be oversampling some regions of space at this time.

Acknowledgements

We would like to thank our closest advisor on this project Dr. Greg Herschlag for his constant guidance and continuous input. Without his foundational work on the code-base and in our own understanding of the topic this work would have not been possible. We would also like to thank Professor Jonathan Mattingly for his guiding vision on this project and for his detailed help throughout.

This work was completed through the Quantifying Gerrymandering Bass Connections team. We thank the interdisciplinary team of students and faculty without whom this work could not have happened. In particular, we thank Tiffany Mei, Sam Eure, and Vinay Kshirsagar for their theoretical and material contributions to this work.

Professor Jonathan Weare's advice on the Stratified Sampling method was invaluable and we thank him for his willingness to share his expertise with us. Thank you to Professor Fritz Mayer for his help of our understanding of the political science surrounding this issue.

We would like to thank the Duke Computer Science Department, Duke Math Department, the Information Initiative at Duke (iID), and Bass Connections Program for financial and material support. Additionally thank you to the Data+ undergraduate research programs and its participants who laid the foundation for our work.

Bibliography

Astor, M. and Lai, K. R. (2018), "Whats Stronger Than a Blue Wave? Gerrymandered Districts," .

Bangia, S., Graves, C. V., Herschlag, G., Kang, H. S., Luo, J., Mattingly, J. C., and Ravier, R. (2017), "Redistricting: Drawing The Line," .

Bycoffe, A., Koeze, E., Wasserman, D., and Wolfe, J. (2018), "The Atlas Of Redistricting, fivethirtyeight.com," .

Chambers, P., C., Miller, D., A., and Sobel (2017), "Flaws in the Efficiency Gap," .

Chen, J. and Rodden, J. (2013), "Unintentional gerrymandering: Political geography and electoral bias in legislatures. Quarterly Journal of Political Science," .

Chen, J. and Rodden, J. (2015), "Cutting through the thicket: Redistricting simulations and the detection of partisan gerrymanders. Election Law Journal," .

Dinner, Thiede, Van-Koten, and Weare (1977), "Journal of Chemical Physics, 145, 084115," .

Herschlag, G. (2018), "Quantifying Gerrymandering in North Carolina," .

Matthews, C. (2016), "Umbrella sampling: a powerful method to sample tails of distributions," .

Raftery, A. E. and Lewis, S. M. (1992), "One Long Run with Diagnostics: Implementation Strategies for Markov Chain Monte Carlo," .

Stephanopoulost, N. and McGheeft, E. M. (2015), "Partisan Gerrymandering and the Efficiency Gap," .

Thiede (1977), "Journal of Chemical Physics, 145, 084115," .