scientific data

Check for updates

OPEN

DATA DESCRIPTOR

Simulated redistricting plans for the analysis and evaluation of redistricting in the United States

Cory McCartan[1], Christopher T. Kenny[2], Tyler Simko[2], George Garcia III[3], Kevin Wang[4], Melissa Wu[4], Shiro Kuriwaki[5] & Kosuke Imai[1,2] ✉

This article introduces the 50StateSimulations, a collection of simulated congressional districting plans and underlying code developed by the Algorithm-Assisted Redistricting Methodology (ALARM) Project. The 50StateSimulations allow for the evaluation of enacted and other congressional redistricting plans in the United States. While the use of redistricting simulation algorithms has become standard in academic research and court cases, any simulation analysis requires non-trivial efforts to combine multiple data sets, identify state-specific redistricting criteria, implement complex simulation algorithms, and summarize and visualize simulation outputs. We have developed a complete workflow that facilitates this entire process of simulation-based redistricting analysis for the congressional districts of all 50 states. The resulting 50StateSimulations include ensembles of simulated 2020 congressional redistricting plans and necessary replication data. We also provide the underlying code, which serves as a template for customized analyses. All data and code are free and publicly available. This article details the design, creation, and validation of the data.

Background & Summary

Redistricting—the process of redrawing electoral district boundaries following the constitutionally mandated decennial census—has substantial impacts on representation, voting rights, and governance in the American political system. As a fundamentally political process, redistricting has also been manipulated to fulfill partisan ends. The detection of gerrymandering, i.e., the intentional redrawing of district boundaries to unduly advantage or disadvantage a certain group of voters, is of immense importance among scholars, policymakers, federal and state courts, and the general public. Just within the 2020 redistricting cycle, at least 72 cases across 26 states have been filed to challenge congressional and state legislative redistricting plans as racially discriminatory and/or gerrymandered for partisan gain[1].

Evaluating a redistricting plan, however, is not a straightforward task. It requires the analyst to take into account federal requirements as well as each state's redistricting criteria and particular political geography. Comparing the partisan bias of a plan for Texas with that of a plan for New York, for example, is likely misleading given numerous differences in redistricting requirements, demographics, geography, candidates, and other state-specific factors. Comparing a state's current plan to its past plans to detect gerrymandering is also problematic because its demographics, politics, and institutions may have changed over the intervening time period between plans. Often, the implicit goal in these cross-state and cross-time comparisons is to assess how the redistricting plan compares to other possible plans in the same state. Under this approach, for example, a redistricting plan can be considered as a partisan gerrymander if it constitutes an outlier, relative to a sample of alternative plans that satisfy the same set of statutory guidelines and requirements, with respect to certain partisan bias metrics[2].

Simulation algorithms have been designed to generate these alternative redistricting plans. Recent advances in computing and methodologies, along with the increasing availability of granular data about voters and elections, have led to the development of redistricting simulation methods that can incorporate federal laws and guidelines, state-specific rules, and election data[3–7]. These simulation methods have gained widespread use in

[1]Department of Statistics, Harvard University, Cambridge, USA. [2]Department of Government, Harvard University, Cambridge, USA. [3]Department of Economics, Massachusetts Institute of Technology, Cambridge, USA. [4]Harvard College, Cambridge, USA. [5]Department of Political Science, Yale University, New Haven, USA. ✉e-mail: imai@harvard.edu

federal and court cases challenging redistricting plans. During this redistricting cycle alone, evidence based on simulation algorithms has been used in courts across a large number of states including Alabama, Georgia, New York, North Carolina, and Ohio. Indeed, simulation analyses have become a standard method for evaluating redistricting plans in academic and judicial settings.

For most analysts, however, performing a redistricting simulation analysis is a complex and laborious task. To begin, the analyst first must put together a detailed geographic dataset combining boundary geometries, legislative district plans, demographic information from the Census, and election data. These data generally come from different sources, and may not naturally overlap with each other. Once the necessary data are tidied and joined, the analyst must then identify the redistricting criteria that should constrain the alternative plans. These criteria are typically based on federal and state laws and need to be formalized as statistical constraints for redistricting simulation algorithms. Next, the analyst must implement a redistricting simulation algorithm to generate a representative sample of alternative redistricting plans that are both diverse and conform to the redistricting criteria. Lastly, they must compare the sampled plans on a wide-ranging set of metrics that past researchers have developed.

We created the 50stateSimulations to aid scholars, policymakers, data journalists, and citizen data scientists alike in performing redistricting simulation analyses. The stateSimulations offer a set of data and computing tools that drastically cut down the complexity and time required to conduct such analyses. The 50stateSimulations include tidied 2020 decennial Census joined with retabulated election data from the Voting and Election Science Team (VEST), a representative sample of simulated 2022 Congressional redistricting plans for all 50 states, and a suite of software packages to visualize, explore, and simulate redistricting plans[8,9]. Any analyst can use these alternative redistricting plans immediately to evaluate the potential bias of an enacted plan or any other counterproposal. In addition, the 50stateSimulations include the code used to generate these simulated plans, which can serve as templates for custom redistricting simulation analyses. Everything in the 50stateSimulations is open-source and reproducible. In this paper, we provide an overview of the workflow for building the 50stateSimulations, describe its contents, and illustrate its use.

Methods

When states enact Congressional district plans, they make a series of discretionary decisions. How many counties and towns should be split? How compact should districts be? A major benefit of using simulations is the ability to incorporate such redistricting criteria in a transparent fashion. We illustrate the simulation process that generates the 50stateSimulations by using Georgia's Congressional redistricting plan as an example. An overarching goal in this process is to generate a representative set of alternative plans that conforms to the redistricitng criteria of that state.

After the 2020 decennial Census, the state of Georgia was allocated 14 congressional districts, each of which elects a single member of the US House of Representatives by plurality vote in 2022 and subsequent general elections. The Georgia State Legislature has the authority to decide how these particular districts are drawn, and the plan is enacted after the Governor's signature. How exactly these districts are drawn is politically consequential, but neither the state constitution of Georgia nor the Georgia Code specifies legal requirements for Congressional redistricting. The map-drawing process, however, adheres in principle to the guidelines established by the state legislature, the authority responsible for redistricting in Georgia. Under the 2021–22 guidelines for Georgia's House Legislative and Congressional Reapportionment Committee[10], districts must: (1) be contiguous, (2) have equal populations, (3) be geographically compact, (4) preserve county and municipality boundaries as much as possible, and (5) avoid the unnecessary pairing of incumbents. Our simulations account for all of these criteria with the exception of incumbency pairing. The criteria used in Georgia are fairly standard, while other states like Colorado (which requires the creation of competitive districts) and Ohio (which has specific requirements about which counties and municipalities can be split and how often) require much more specific criteria that we incorporate into our simulations. Supplementary Table 1 shows the legal sources and redistricting criteria we included for every state.

Our simulations in Georgia also include a constraint based on the 1965 Voting Rights Act (VRA). According to the VRA, minority racial groups that are polarized from the majority should be arranged in districts that can elect their members of choice. In practice, this means that districts should have sufficient numbers of minority constituents as measured, for example, by the proportion of the Voting Age Population that is Black (BVAP). In Georgia, we incorporate a consideration of the VRA by penalizing districts that have a BVAP lower than 52 percent. We describe the reasoning behind this choice in the Technical Validation section, and list specifications for other states in Supplementary Table 1. This sets a soft, probabilistic target for the minority proportion in a district based on the enacted plan, but allows the proportion to vary below or above the value set. We then simulate alternative redistricting plans under these criteria.

The first step of the redistricting simulation workflow is to assemble precinct-level shapefiles with associated demographic data. The 50stateSimulations contains the ALARM Project's 2020 Redistricting Data Files that consist of the tidied 2020 decennial Census and statewide election data from the Voting and Election Science Team (VEST)[9]. The VEST data are widely used in academic research[11,12], litigation (e.g., "An Evaluation of the Partisan Fairness of Ohio's February 24, 2022 State Legislative Districting Plan", report of Christopher Warshaw in *League of Women Voters of Ohio v. Ohio Redistricting Commission* (2020)), and public projects (such as in Dave's Redistricting App or PlanScore). The election data are tidied by estimating VEST's collection of precinct shapes to their underlying 2010 blocks using[13], then crosswalked to 2020 blocks[14,15] and aggregated to voting districts. Election data are first matched to 2010 blocks, as precincts are largely made of census blocks, which are mostly stable for the decade[16]. We also include data about municipality boundaries, which are obtained from Census block files. We acquire the shapefiles for Georgia's enacted congressional plan from the General Assembly website (https://www.legis.ga.gov/joint-office/reapportionment) and join them to the rest of the data.

We incorporate the redistricting criteria set by the legislative committee into our simulation analysis using both hard constraints, which ensure that all simulated plans meet the specified criteria, and soft constraints, which encourage simulated plans to meet the specified criteria without enforcing hard limits. We use a Sequential Monte Carlo (SMC) redistricting algorithm[7] to obtain a representative sample of alternative redistricting plans under these redistricting criteria. The SMC algorithm enforces two hard constraints: contiguity and limited deviation from population parity, which reflect universal redistricting criteria. The algorithm by default also includes a soft constraint which encourages district compactness, according to a particular graph-theoretic measure of compactness (Specifically, the algorithm samples plans according to the number of spanning forests which can be drawn on the district adjacency graph. This graph-theoretic quantity correlates strongly with the total perimeter of the districts, and thus also with non-graph-based compactness measures which are based on perimeter length, such as Polsby–Popper). This compactness constraint can be tuned slightly upwards or downwards, which we do in states with specific compactness requirements (see the Appendix for details). The defaults, however, produce districts with a range of compactness values that generally span the range of historical values for congressional districts, and so in most states we do not tune the constraint. For the population equality constraint, we set the maximum deviation from the population parity to be 0.5%, which corresponds to approximately 3,826 people in Georgia. Although enacted redistricting plans often achieve exact population parity across districts, the use of the 0.5% threshold is appropriate because our simulation analysis is based on precincts (the smallest units for which the electoral data are available) rather than the more granular Census blocks.

Additionally, we add a hard constraint to the algorithm to limit the number of counties and municipalities that are split by districts. This hard constraint is incorporated as part of the SMC algorithm itself, and limits the number of splits of provided administrative units to one less than the number of districts. It is up to the analyst, however, to decide what administrative units to provide to the algorithm. Many states, including Georgia, value the preservation of both county and municipality boundaries, and we operationalize this constraint in the following way. We start by providing county boundaries to the SMC algorithm; this will limit the number of county splits. However, counties with populations larger than a congressional district must necessarily be split. In these counties (which are Cobb, Fulton, and Gwinnett counties, in Georgia), we use municipalities (specifically, Census Designated Place boundaries) as the administrative units. Across the whole state, then, the SMC algorithm will limit the number of county (in lower-population areas) and municipality (in large counties) splits to one less than the number of districts. We check the sampled plans against the enacted plan to ensure that the sampled plans perform as well or better, on average, than the enacted plan. In some other states, there are specific rules about the number of counties and municipalities that may be split, and we ensure every sampled plan meets those rules, either by redifining the administrative boundaries, adding further soft constraints to encourage fewer splits, or by subsetting the sampled plans to those which satisfy the requirements.

Lastly, we use a soft constraint to encourage sampling redistricting plans that have the same number of Black opportunity districts as the enacted plan. In other states, this constraint may be applied separately for Black and Hispanic voters, or collectively applied to minority voters overall if no one racial group has sufficient numbers to form a majority in a district. The strength of this constraint must be carefully tuned to ensure the algorithm functions correctly while producing the specified range of majority-minority or opportunity districts. The Technical Validation contains further details on how we consider the nuances of the VRA.

We simulate 20,000 alternative redistricting plans for Georgia, ensuring that the sample has converged into a stable distribution by comparing two parallel runs of the SMC algorithm. We then thin this set of plans to 5,000 plans for our final product.

In most states, we obtain two independent samples of 2,500 simulations in parallel and combine these samples to generate a sample of 5,000 simulated plans. No thinning is necessary in these states. In some large states such as Georgia that have more geographical complexity, we sample more than 5,000 plans in total so that the simulation algorithm converges, according to numerical diagnostics (see Technical Validation section). Once these diagnostics suggest convergence, we randomly sample 5,000 plans from the original set of simulated plans. This is done to maintain consistency across states and limit the memory usage for simulated data, eliminating a potential computational burden for users. A redistricting plan is essentially an assignment of each precinct in the state to a district number. Each alternative plan offers a distinct set of assignments (with some possible duplication), so that each simulated district covers a different geographic area and is therefore associated with different demographic and partisan characteristics. Finally, we post-process the plans so that their numbering matches that of the enacted plan—district 1 in any simulated plan will roughly correspond to district 1 in the actual plan, and so on.

As an example, we show 9 out of the 5,000 alternative plans of Georgia in Fig. 1. The district assignment that the Georgia state legislature finalized is also shown for comparison. We then characterize each simulated district by its demographics and partisan lean according to historical election data.

The final dataset is a series of alternative district assignments for each precinct in the state, along with the demographic and political characteristics of those alternative plans. We show how to extract and use these variables in the next section.

Data Records

The 50StateSimulations can be found in the Harvard Dataverse[17] at https://doi.org/10.7910/DVN/SLCD3E. Each state has four files that follow the `{state}_cd_{yyyy}_{type}` naming convention, with `state` indicating the state abbreviation, `cd` indicating the congressional districts, `yyyy` indicating the 4-digit year of the redistricting cycle, and `type` taking on (1) `doc` for the markdown documentation, (2) `map` for the `redist_map` object, (3) `plans` for the `redist_plans` object, and (4) `stats` for the summary statistics. These files are organized into folders by state in the *Tree View* option on the Dataverse website.

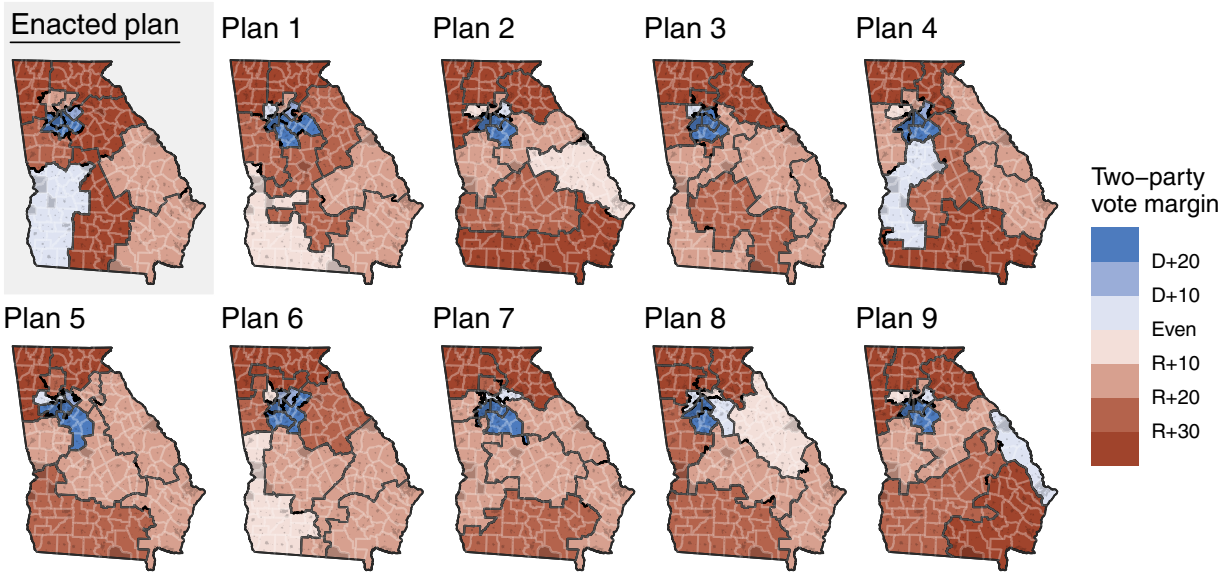

Fig. 1 The enacted Georgia congressional plan (top left) and 9 out of the 5,000 alternative plans we provide in this dataset. Gray areas indicate Census-designated places (including cities and towns), white lines indicate counties, and dark gray lines indicate district assignment. Each district is colored by its partisan lean computed as an average of historical statewide election results (2016–2020) within the sampled district, where D + 20 indicates the average Democratic candidate wins by 20 percentage points or more over the average Republican candidate. Because each alternative plan can assign different districts to each precinct, the district-level two-party vote can also differ.

Documentation for each state's redistricting simulation methodology is outlined in the `_doc.html` file. We record information about the state's legal redistricting requirements and a description of the algorithmic constraints that we implemented to comply with those requirements. We list the sources from which we obtain our geographical, population, election, and enacted plan data. We also describe any pre-processing notes, the number of plans simulated, and any special techniques needed to produce the sample. The documentation includes a brief description of each file in the state's folder and explanations of each summary statistic in a listed format.

Underlying each simulation is a `redist_map` object (a custom object defined in the *redist* package[18]), which has pre-merged geographic, demographic, and electoral data at each voting tabulation district (or precinct). It is a shapefile that contains the geographic coordinates and adjacency of each unit. The `redist_map` object contains geographical data such as `GEOID` (precinct/block unique identifier), `adj` (an adjacency graph that records which precincts are geographically adjacent to one another), state (state abbreviation), `county` (county name), `muni` (municipality identifier), `county_muni` (concatenation of `county` and `muni`), `cd_2010` (congressional district number assignment in the 2010 enacted plan), `cd_2020` (congressional district number assignment in the 2020 enacted plan), and `vtd` (voting district identifier).

The `redist_map` object also contains Census and election statistics measured at the precinct-level that will later be aggregated up to districts.

- Population and demographic information comes from the Census PL94–174 file: with `pop` indicating the entire population and `vap` indicating voting-age population. The size of demographic subgroups are also included for targeting effective minority districts in some states. Racial subgroups are denoted by `_hisp` (Hispanic or Latino of any race), `_white` (White alone, not Hispanic or Latino), `_black` (Black or African American alone, not Hispanic or Latino), `_aian` (American Indian and Alaska Native alone, not Hispanic or Latino), `_asian` (Asian alone, not Hispanic or Latino), `_nhpi` (Native Hawaiian and Other Pacific Islander alone, not Hispanic or Latino), `_other` (some Other Race alone, not Hispanic or Latino), and `_two` (population of two or more races, not Hispanic or Latino).
- Electoral information relies on the variables from VEST. Statewide offices and their election schedules vary by state. In Georgia, for example, VEST includes elections for President (2016, 2020), US Senate (2016, 2020), Governor (2018), Attorney General (2018), and Secretary of State (2018). The data from VEST uses the naming convention `{office}_{year}_{party}_{candidate}` to measure the vote totals of each candidate at the precinct-level: `office` indicates the office abbreviation, with options including President (`pre`), United States Senate (`uss`), Governor (`gov`), Attorney General (`atg`), Secretary of State (`sos`); `year` indicates the last two digits of the election year; `party` indicates either the Republican candidate (`rep`) or Democratic candidate (`dem`); `candidate` indicates the first three letters of the candidate's last name.
- We then summarize these VEST variables as `arv_{year}` (average vote counts for Republican candidates in that year), `adv_{year}` (average vote counts for Democratic candidates), `nrv` (average vote counts for

Republican candidates across all available election years), and `ndv` (average vote counts for Democratic candidates across all available election years).

We save the `redist_map` object as a compressed RDS file, a native R format that retains the basic attributes of the redistricting problem, such as the number of districts to draw and the population parity constraint they should satisfy. It also retains the shapefile and adjacency information for each precinct.

The set of plans simulated from this map object are stored in a `redist_plans` object (another custom object defined in the *redist* package). Each state's simulations are constructed differently, following the specific rules that govern redistricting in each state (see the Technical Validation for details). Each `redist_plans` object contains the assignment of districts to each unit of geography (in our case the precinct). Each alternative plan and a reference plan are encoded by `draw` and the district is indexed by `district`. For a redistricting problem of 1,000 precincts to be assigned to 5 districts, if we simulate 5,000 alternative plans and compare it with the plan enacted by the legislature, the `redist_plans` object contains $(5000 + 1) \times 5 = 25{,}005$ rows. Simulations are conducted over multiple independent runs of the redistricting algorithm, which is useful for diagnostic purposes, and these runs are identified by the `chain` column.

Our simulation output is most commonly used to compare a district-level or plan-level summary statistic of a particular plan to the entire distribution of that summary statistic from the simulated plans. We provide a plain-text comma-separated values (CSV) file that contains these summary statistics for each of the 5000 plans or simulated districts. This file does not include any R-specific attributes, and can be used in any programming or spreadsheet software. If users have a statistic for a proposal that is not in our data, e.g., the Efficiency Gap metric for a remedial map proposed in court, they can easily compare how that number compares to our reference distribution by loading the summary statistics in a spreadsheet or data analysis program of their choice.

The summary statistics file includes the totals of any variables included in the `redist_map` population and election data (such as the total population of racial subgroups in each district). The difference is that redist_map provides precinct-level measures of these data, while the `redist_plans` object and summary file show the district-level totals of these building blocks as they are aggregated into different district arrangements. It also contains plan-level statistics for traditional redistricting criteria such as: the maximum population deviation among plan districts (`plan_dev`), plan-level compactness according to the fraction of edges kept (`comp_edge`), compactness according to the Polsby-Popper score (`comp_polsby`)[19].

While many different compactness measures are used today, we have adopted these two as baseline summary statistics, in part because they are widely used in academic work and litigation and are computationally efficient to calculate. The Polsby-Popper score is perhaps the most widely-used compactness measure. The fraction of edges kept is a graph-theoretic measure and thus, unlike the Polsby-Popper score, it is invariant to changes in the resolution of the shapefile or the inclusion or exclusion of certain water areas from precinct boundaries. This compactness measure is also closely related to the graph-theoretic measure that is used in the SMC algorithm's built-in compactness constraint. We note that among the simulated plans, there is a high degree of correlation between both of these compactness measures. For users who require additional compactness information, the *redist* and *redistmetrics* packages provide functionality to easily compute many of these additional measures for the simulated plans.

Of interest for detecting partisan gerrymandering are normal Democratic share (`ndshare`), average Democratic vote share (`e_dvs`), probability that a seat is represented by a Democrat (`pr_dem`), expected number of Democratic seats for each plan (`e_dem`). We estimate the expected number of Democratic seats by first determining the winner of election within each simulated district using the precinct-level vote shares for a particular historical statewide race. We then compute the number of districts Democratic party candidates are expected to win. Taking the average of this number across all statewide races gives the expected number of Democratic seats. It is important to note that because a different set of statewide elections are available in each state, partisan summary statistics may not be directly comparable across states. We encourage users to consider modeling election results using a baseline partisanship measure derived from presidential elections, which have results available for all states.

These summary statistics of elections can be further transformed to compute common measures of partisan gerrymandering: The difference from the expected number of seats won by a party; the deviation from partisan symmetry[2], which estimates the difference in each party's seat share if they each won 50% of the statewide vote (the variable `pbias`); and the efficiency gap[20], which counts the difference in the number of wasted votes for each party averaged across all available elections (the variable `egap`). For partisan bias measures, a positive value implies a pro-Republican bias while a negative value indicates a pro-Democratic bias.

Technical Validation

In preparing each state's simulation, we studied the laws and regulations of each state's redistricting process and operationalized them through simulation constraints in the `redist` package. These goals for compliance must be balanced with maintaining a diverse sample of plans.

To ensure that the simulated plans are drawn from a population of valid alternatives, we check the population deviation at the voting district level, two measures of compactness (the fraction of edges kept compactness and Polsby-Popper compactness), number of county splits, number of municipality splits, and the minority voting age population in each district. We assess compliance with such traditional redistricting criteria by asking if the distribution of statistics of the alternative plans are in line with the enacted plan or historical plan.

The process is iterative. If initial simulations appear overly non-compact or appear to split administrative boundaries excessively, we consider strengthening the soft constraints for these metrics. However, we do not strictly tune our constraints to the precise compactness of the enacted map. As mapdrawers know that compactness will be used to evaluate redistricting maps, compactness can be used to disguise a gerrymander. For

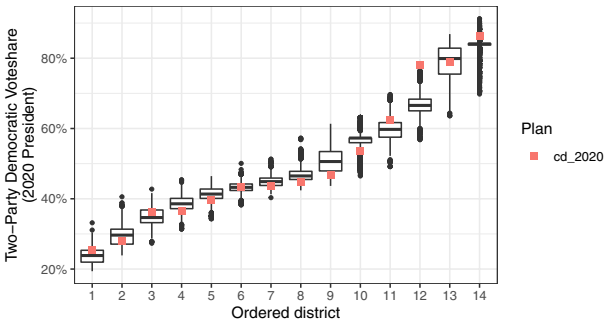

Fig. 2 Boxplots of the two-party Democratic vote share in the 2020 presidential election in each of our 5,000 simulated congressional districts. Districts are ordered by their rank ordering of the Democratic vote within a plan, ranging from the least Democratic (left) to the most Democratic (right) district. The solid bar shows the median Democratic vote, and black points indicate outlier values outside of the interquartile range. The red squares indicate the Democratic vote in the enacted plan.

example, in Florida and North Carolina, our final simulations are less compact than the enacted map. In contrast, in states such as Illinois, our final simulations are more compact, as compactness was subjugated to develop a Democratic advantage in the enacted plan.

Statistics about the size of the racial minority population in each district is fraught with minority vote dilution and racial gerrymandering litigation stemming from the implementation of the Voting Rights Act of 1965 (VRA). We ensure that the simulated plans would be *comparably* compliant with Section 2 of the VRA, which stipulates that minority voters cannot be deprived of the ability to be represented by their district representative of their choice[21]. While compliance is a legal question, we use this to guide the simulations under the assumption that the enacted plan complies with the VRA. In states with a history of litigation or preclearance, we therefore ensure that simulated plans have equal opportunity to elect candidates of choice as in the enacted plan by checking whether districts with large minority populations end up with a higher vote share for the party that the minority population is known to support. For example, in Georgia, external estimates suggest that Black voters overwhelmingly support the Democratic party, so we verify that districts with relatively more Black voters also elect Democratic candidates according to the historical election data.

Simulating plans which are VRA compliant can be a difficult procedure, as one must determine which districts are minority opportunity districts. There is neither academic nor legal consensus on best practices. Chen and Stephanopoulos (2021)[22] considers a three step test: (1) the minority-preferred candidate must win the general election, (2) there must be more minority voters for the winner than white voters, and (3) minority voters count towards criteria (2) only if the groups prefer the same candidate. Becker *et al.*[23] considers a different approach, defining a model of district performance on the logit scale. Our approach is closer to that of Chen and Stephanopoulos (2021)[22]. More precisely, we define a minority opportunity district as a district where (1) the minority voting age population proportion is in ranges which produce candidates of choice from minority voters (typically Democrats) and (2) minority voters are expected to contribute more to the winning coalition than white voters. This means that we do not target a particular racial lower bound while simulating districts. After plans are sampled, we check that all plans would have at least as many minority opportunity districts as the enacted plan, again ensuring that the simulated plans are at least as compliant with the VRA as the enacted plan. This should not be construed as a determination that the enacted plan is legally compliant with the VRA. Indeed, some states, like Mississippi, demonstrate that minority opportunity districts can be drawn with smaller concentrations of Black voters.

As we provide the full code and data necessary to replicate the simulations, the sampling constraints can be easily altered to use alternative methods of VRA compliance, by changing a single line of code for each VRA constraint. Indeed, this is just one approach to the VRA, which could alternatively include broader considerations, like primary elections, or more narrow considerations, like candidate-specific models of turnout. Our approach is rooted in the effects side of VRA compliance based on the enacted plan, in that minority preferred candidates can win elections in as many minority opportunity districts as in the enacted plan. It should not be construed as a determination of how many VRA districts a state should have drawn.

In simulating districts, there is often a trade-off between complying with constraints and obtaining a diverse sample of districts. The SMC algorithm samples from the space of all possible redistricting plans which are contiguous and meet the population balance threshold of 0.5%, and, depending on the state, an additional administrative unit splitting constraint. Within this space, plans which are more compact and better align with other specified criteria have a higher probability of being drawn. While the SMC algorithm has theoretical guarantees of sampling from the distribution specified by these constraints, in practice, it may become exceedingly difficult to draw plans which meet all the user-specified requirements and adhere to population parity and contiguity constraints. In these cases, the algorithm may be stuck in a handful of plans, which get duplicated many times across the simulated set. To ensure that the sampler is not getting stuck, or producing an unrepresentative sample, it is important to check several diagnostics. We conduct extensive diagnostic checks reflecting the best practices[7].

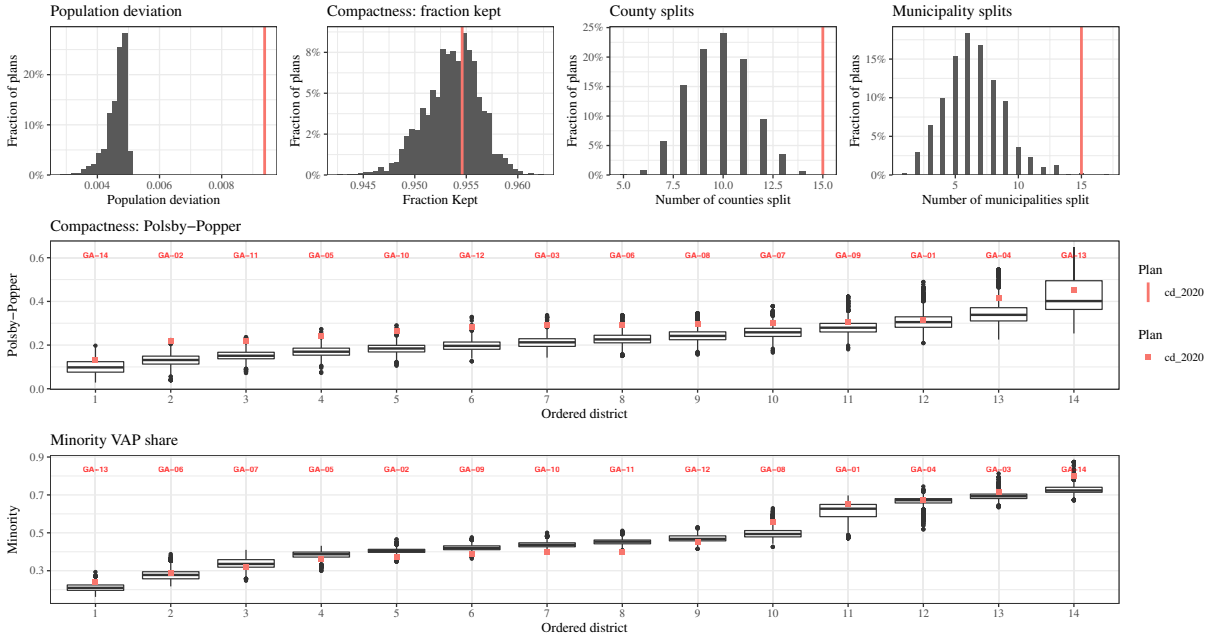

Fig. 3 An example validation plot from our Georgia analysis. Analogous validation plots were made for all states, and include information on metrics discussed above like compactness, diversity, boundary splits, and minority VAP. Validation plots for all analyses are available in the individual state pull requests on our public repository.

First, we check the final-stage resampling weights from the SMC algorithm to ensure that the constraints imposed on the sampling process are not too severe. Next, we evaluate how *diverse* the sample is by measuring the variation of information (VI) distance between plans. If two plans place most people into the same districts, then the VI distance between them is low; if two plans assign people to districts very differently, the VI distance is larger[24]. A diverse sample should have a wide variety of redistricting plans, and therefore a high average pairwise VI distance.

Conversely, when the algorithm becomes stuck, many duplicated or highly-similar plans will show up as a very low pairwise VI distance. After checking the weights and VI distance, we then use diagnostics from each iteration of the SMC procedure to ensure that there are no sampling bottlenecks or efficiency losses. These bottlenecks can lead to a small number of valid districts in a certain area of a state to appear an abnormally large portion of the time in the final sample. The *redist* software has heuristic checks for these bottlenecks built in to its diagnostic reporting. Finally, we evaluate the convergence of the algorithm for the specific set of summary statistics described above that are of interest to practitioners. This is accomplished by splitting the sample across at least two independent runs of the sampling procedure; from this, we calculate the Gelman-Rubin $\hat{R}$ statistic for each quantity of interest[25,26]. The $\hat{R}$ statistic will be large if the independent runs produce samples that are not alike. We check that $\hat{R}$ is around 1.05 or less for all of the calculated summary statistics, which indicates likely convergence.

After the simulations are completed, each diagnostic is presented in an automatically generated plot and summary report of the plan. These are reviewed by a different member of the team through a public "pull request" on the repository. Especially for larger states, many rounds of sampling and adjusting algorithm parameters such as the sample size and constraint strengths were often necessary to pass our battery of diagnostic checks. Only once these checks were passed and validated by another member of the team was the sample admitted to the dataset. Figure 3 in the Appendix display these diagnostic checks for our Georgia simulations.

Our simulations therefore serve as a realistic set of alternative plans. However, they do not represent our evaluation of the *legality* of the enacted and other plans. While simulations are used increasingly in litigation of districts, they are not always a deciding piece of evidence and simulations in cases often need to be tailored in specific ways that are difficult to characterize generally. Our parameters of the simulation are open-source and replicable, and serve as a useful template for those interested in constructing their own simulation.

Usage Notes

The 50StateSimulations are well-suited to assess what types of partisan or racial outcomes could have happened under alternative plans in a given state. Alternative plans should not be compared across states, for the same reason that comparing the metrics of enacted plans across states is also invalid. A variant of a cross-state comparison could be made through computing within-state differences between enacted and simulated alternative plans, and comparing those differences across several states as a measure of gerrymandering[27,28].

As an example, we continue our exploration of Georgia's 2020 Congressional redistricting plan. All of our data and analysis code is found in the open-source `R` package `alarmdata`[29]. After loading the package, we start

by downloading a redistricting `redist_map` object, which contains the basic information about the redistricting problem and Census and electoral variables as covariates.

```
library(redist)
library(alarmdata)

ga_map<- alarm_50state_map("GA")
```

Separately, users can download the simulated plans with the `alarm_50state_plans()` function. These plans come in the `redist_plans` object format described in the Data Records Section, such that each row is a district in one draw of the simulation. To pre-calculate common plan statistics for compactness, administrative boundary splits, and partisanship, we set the optional `stats = TRUE` argument.

```
plans<- alarm_50state_plans("GA", stats = TRUE)
```

Our redistricting objects are `tibble` objects, and therefore can be operated on by standard `tidyverse` functions[30]. For example, the simulation plans object is also a `tibble` dataframe where one row is a district in a simulation. Users can also create arbitrary statistics from the existing Census and VEST variables. For example, below we calculate the Democratic proportion of the two-party vote share in the 2020 presidential election for each plan in Georgia.

```
plans<- plans %>%
    mutate(dem_2020 = pre_20_dem_bid / (pre_20_dem_bid +  pre_20_rep_tru))
```

Often, analysts will want to compare a specific enacted map or a counterproposal to our simulations. The 50StateSimulations plan objects have the map enacted and finalized by each state as a reference plan in the draw `cd_2020`, but analysts can also add custom reference `plans`. Below, we add the 2010 Georgia Congressional plan to serve as a comparison for later analyses. We pass our `plans` simulation object to `alarm_add_plan()`, and add the `cd_2010` column as a reference redistricting plan. The `name` argument provides a `name` for the reference plan in the plans object.

```
plans<- plans %>%
  alarm_add_plan(map = ga_map, ref_plan = ga_map$cd_2010, name = "cd_2010")
```

Additionally, we present convenient visualization functions for immediate use with the 50StateSimulations. For example, maps are easily created with the `redist::redist.plot.map()` function. The `redist::redist.plot.hist()` function plots a variable in a plans object as a histogram while the `redist::redist.plot.distr_qtys()` function generates a district-level summary statistic as a boxplot.

For example, to display the distribution of the two-party 2020 Democratic vote in each alternative district, we run:

```
redist.plot.distr_qtys(plans, qty=dem_2020, geom="boxplot") +
    labs(y="2020 Presidential Two-Party Vote Share")+
    theme_bw()
```

Figure 2 shows the resulting boxplot, which indicates that, across most districts, the Democratic vote share in the enacted 2020 Congressional Plan falls within the range of our 5,000 simulated plans. The two exceptions are the 12th and 14th ordered districts, both of which have Democratic vote shares larger than the expected range under our simulations. This suggests that the enacted plan may be packing Democratic voters more than necessary by traditional criteria. A similar analysis can be done with the proportion of the district Voting Age Population that is Black (BVAP), to detect signs of racial gerrymandering. We hasten to note again that this result alone does not constitute conclusive evidence of partisan gerrymandering in the enacted plan. The court may also require the simulated plans to conform to other guidelines, such as avoidance of pairing incumbents, which we did not implement here. Nevertheless, the simulations serve as a diverse set of valid alternative maps that could be drawn to answer an otherwise intractable question: in what ways the enacted plan differs from other alternatives and by how much.

Further customization and visualization options are available in our companion packages `redist`[18], which implements the simulation algorithm, and `redistmetrics`[31], which efficiently computes metrics evaluating each plan.

Code availability

All code is publicly available on our GitHub repository for the 50StateSimulations project, https://github.com/alarm-redist/fifty-states. The package `alarmdata`[29] is a user-facing R package to download and work with our plans on the repository and our Dataverse.

Received: 28 July 2022; Accepted: 25 October 2022;
Published online: 11 November 2022

References

1. Brennan Center for Justice. Redistricting litigation roundup (2022). Last accessed September 18, 2022.
2. Katz, J., King, G. & Rosenblatt, E. Theoretical foundations and empirical evaluations of partisan fairness in district-based democracies. *American Political Science Review* **114**, 164–178 (2020).
3. Carter, D., Herschlag, G., Hunter, Z. & Mattingly, J. A merge-split proposal for reversible Monte Carlo Markov chain sampling of redistricting plans. *arXiv preprint arXiv:1911.01503* (2019).
4. DeFord, D., Duchin, M. & Solomon, J. Recombination: A family of Markov chains for redistricting. *Harvard Data Science Review* https://doi.org/10.1162/99608f92.eb30390f, https://hdsr.mitpress.mit.edu/pub/1ds8ptxu (2021).
5. Autry, E., Carter, D., Herschlag, G., Hunter, Z. & Mattingly, J. Multi-scale merge-split Markov chain Monte Carlo for redistricting. *Working paper* https://doi.org/10.48550/ARXIV.2008.08054 (2020).
6. Fifield, B., Higgins, M., Imai, K. & Tarr, A. Automated redistricting simulation using Markov chain Monte Carlo. *Journal of Computational and Graphical Statistics* **29**, 715–728 (2020).
7. McCartan, C. & Imai, K. Sequential Monte Carlo for sampling balanced and compact redistricting plans. *arXiv*, (2021). arXiv:2008.06131.
8. U.S. Census Bureau. 2020 Census (2022). Last accessed April 29, 2022.
9. Voting and Election Science Team. Precinct-level election results. https://dataverse.harvard.edu/dataverse/electionscience (2022). Last accessed April 29, 2022.
10. George House of Representatives Legislative and Congressional Reapportionment Committee. 2021–2022 guidelines for the house legislative and congressional reapportionment committee. https://perma.cc/87pv-vj45.
11. Allcott, H. *et al*. Polarization and public health: Partisan differences in social distancing during the coronavirus pandemic. *Journal of Public Economics* **191**, 104254 (2020).
12. Grossman, G., Kim, S., Rexer, J. M. & Thirumurthy, H. Political partisanship influences behavioral responses to governors' recommendations for covid-19 prevention in the united states. *Proceedings of the National Academy of Sciences* **117**, 24144–24153 (2020).
13. Kenny, C. T. geomander: Geographic tools for studying gerrymandering. Available at The Comprehensive R Archive Network (CRAN) (2021).
14. Amos, B. 2020 Census Block Crosswalk Data. *Harvard Dataverse* https://doi.org/10.7910/DVN/T9VMJO (2021).
15. McCartan, C. & Kenny, C. T. *PL94171: Tabulate P.L. 94–171 Redistricting Data Summary Files*. R package version 1.0.1 (2021).
16. Amos, B., McDonald, M. P. & Watkins, R. When Boundaries Collide: Constructing a National Database of Demographic and Voting Statistics. *Public Opinion Quarterly* **81**, 385–400, https://doi.org/10.1093/poq/nfx001. https://academic.oup.com/poq/article-pdf/81/S1/385/13942323/nfx001.pdf (2017).
17. McCartan, C. *et al*. 50-State Redistricting Simulations, *Harvard Dataverse* https://doi.org/10.7910/DVN/SLCD3E (2021).
18. Kenny, C. T., McCartan, C., Fifield, B. & Imai, K. redist: Simulation methods for legislative redistricting. Available at The Comprehensive R Archive Network (CRAN) (2021).
19. Polsby, D. D. & Popper, R. D. The third criterion: Compactness as a procedural safeguard against partisan gerrymandering. *Yale Law & Policy Review* **9**, 301 (1991).
20. Stephanopoulos, N. O. & McGhee, E. M. Partisan gerrymandering and the efficiency gap. *University of Chicago Law Review* **82**, 831 (2015).
21. Grofman, B., Handley, L. & Lublin, D. Drawing effective minority districts: A conceptual framework and some empirical evidence. *North Carolina Law Review* **79** (2000).
22. Chen, J. & Stephanopoulos, N. O. The Race-Blind Future of Voting Rights. *The Yale Law Journal* **130**, 862-946 (2021)
23. Becker, A., Duchin, M., Gold, D. & Hirsch, S. *Computational Redistricting and the Voting Rights Act* https://doi.org/10.1089/elj.2020.0704 (2021).
24. Guth, L., Nieh, A. & Weighill, T. Three applications of entropy to gerrymandering. *Political Geometry* 275 (2020).
25. Gelman, A. & Rubin, D. B. Inference from iterative simulation using multiple sequences. *Statistical Science* **7**, 457–472 (1992).
26. Vehtari, A., Gelman, A., Simpson, D., Carpenter, B. & Bürkner, P.-C. Rank-normalization, folding, and localization: An improved R for assessing convergence of MCMC. *arXiv preprint arXiv:1903.08008*, (2019).
27. Chen, J. & Cottrell, D. Evaluating partisan gains from Congressional gerrymandering: Using computer simulations to estimate the effect of gerrymandering in the U.S. House. *Electoral Studies* **44**, 329–340, https://doi.org/10.1016/j.electstud.2016.06.014 (2016).
28. Kenny, C. T., McCartan, C., Simko, T., Kuriwaki, S. & Imai, K. Widespread partisan gerrymandering mostly cancels nationally, but reduces electoral competition. *arXiv preprint arXiv:2208.06968* (2022).
29. McCartan, C., Kenny, C. T., Simko, T., Zhao, M. & Imai, K. *alarmdata: Download, Merge, and Process Redistricting Data*. R package version 0.0.0.9000 (2022).
30. Wickham, H. *et al*. Welcome to the tidyverse. *Journal of Open Source Software* **4**, 1686, https://doi.org/10.21105/joss.01686 (2019).
31. Kenny, C. T., McCartan, C., Fifield, B. & Imai, K. redistmetrics: Redistricting metrics. Available at The Comprehensive R Archive Network (CRAN) (2021).

Acknowledgements

The authors thank the Harvard Data Science Initiative and Microsoft for computational support.

Author contributions

C.T.K., C.M., T.S. and K.I. conceived the project, C.T.K., C.M., T.S., G.G., K.W., M.W. and S.K. conducted the data analysis. All authors contributed to the writing.

Competing interests

C.T.K. has served as a paid expert for the Maryland Redistricting Commission. K.I. has served as a paid expert witness in the court cases related to the 2020 Congressional redistricting in Alabama, Kentucky, Ohio, and South Carolina.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41597-022-01808-2.

Correspondence and requests for materials should be addressed to K.I.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.