The Essential Role of Empirical Validation in Legislative Redistricting Simulation Replication Guide

Benjamin Fifield* Kosuke Imai† Jun Kawahara‡ Christopher T. Kenny§

June 2020

Abstract

As granular data about elections and voters become available, redistricting simulation methods are playing an increasingly important role when legislatures adopt redistricting plans and courts determine their legality. These simulation methods are designed to yield a representative sample of all redistricting plans that satisfy statutory guidelines and requirements such as contiguity, population parity, and compactness. A proposed redistricting plan can be considered gerrymandered if it constitutes an outlier relative to this sample according to partisan fairness metrics. Despite their growing use, insufficient effort has been made to empirically validate the accuracy of the simulation methods. We apply a recently developed computational method that can efficiently enumerate all possible redistricting plans and yield an independent uniform sample from this population. We show that this algorithm scales to a state with a couple of hundred geographical units. Finally, we empirically examine how existing simulation methods perform on realistic validation data sets.

Keywords: enumeration, gerrymandering, graph partition, Markov chain Monte Carlo, redistricting, zero-suppressed binary decision diagram

*Affiliated Researcher, Institute for Quantitative Social Science, Harvard University, Cambridge, MA 02138. Email: benfifield@gmail.com, URL: https://www.benfifield.com

†Professor, Department of Government and Department of Statistics, Institute for Quantitative Social Science, Harvard University, Cambridge MA 02138. Phone: 617–384–6778, Email: Imai@Harvard.Edu, URL: https://imai.fas.harvard.edu

‡Associate Professor, Graduate School of Informatics, Kyoto University. Email: jkawahara@i.kyoto-u.ac.jp, URL: http://www.lab2.kuis.kyoto-u.ac.jp/jkawahara/index-en.html

§Ph.D. Student, Department of Government, Harvard University. Email: christopherkenny@fas.harvard.edu URL: https://www.christophertkenny.com

Prior to attempting to download this entire project, please beware that there are at least 300GB worth of files when uncompressed. Files created by the scripts may more than double the amount of free storage space necessary to run the entire project. It is recommended to begin with the 70 precinct map ("largemap_enum_all") if storage space is a concern.

1 Overview of Replication Archive

This replication guide serves to clarify the order of the files contained in the dateverse and which figures they correspond to. Multiple files are needed to be run to replicate individual figures. These scripts were run on a Linux cluster, often requiring over 100GB to run the scripts and even to create the figures themselves from saved data. File paths are relative to the file structure used on the cluster.

The code folder contains the code necessary to run the primary analysis of the paper. See the instructions below for figure correspondence. Due to the size and time required to replicate, there is not one single script for each figure, instead having scripts which break up the task into tasks which will take at most one week each. Within the code folder, there is a slurm folder which contains some slurm scripts used to submit the jobs. These will need to be updated to match the system that any replication is run on. Files that were run interactively do not have slurm files. For those jobs, typically requesting 120GB to run the job is sufficient. The enumpart_private folder contains the C++ implementation of the enumeration method. There is a README file within that directory with additional information. At a minimum, you must run make within that directory to use the enumeration method. The data folder contains the necessary input data for running the scripts. The lab folder provides the sampled districts for the Iowa and 250 precinct maps.

2 Instructions for Figure Replication

Figures 1-5

Figures 1-5 are drawn in LaTeX using tikz, thus there is no code necessary.

Figure 6

1. Run runtime_test.R
2. Run runtime_test_analyze.R

Figure 7 and Figure 8

1. Run largemap_enum_all.R
2. Run largemap_enum_all_validation.R

3. Run fh_enum_all.R
4. Run fh_enum_rsg.R
5. Run largemap_enum_all_analyze_compact.R

Figure 9

1. Run qq_test.R
2. Run qq_test_sample.R
3. Run qq_test_analyze.R

Figure 10 and Figure 11

1. Run ia_enumerate.R
2. Run ia_enumerate_rsg.R
3. Run ia_enumerate_analyze.R

Figure 12 and Figure 13

1. Run largemap_enum_sample.R
2. Run largemap_enum_sample_rsg.R
3. Run largemap_enum_sample_analyze.R

3 R Version Info

```
> sessionInfo()
R version 3.6.1 (2019-07-05)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: CentOS Linux 7 (Core)

Matrix products: default
BLAS:   /usr/lib64/libblas.so.3.4.2
LAPACK: /usr/lib64/liblapack.so.3.4.2

Random number generation:
 RNG:     Mersenne-Twister
 Normal:  Inversion
 Sample:  Rounding

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
```

```
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] grid      parallel  stats     graphics  grDevices utils     datasets
[8] methods   base

other attached packages:
 [1] patchwork_1.0.0    ggrepel_0.8.2      RColorBrewer_1.1-2 doMC_1.3.6
 [5] iterators_1.0.12   foreach_1.5.0      gridExtra_2.3      spdep_1.1-3
 [9] sf_0.9-3           spData_0.3.5       redist_2.0.0       igraph_1.2.4.2
[13] maptools_0.9-8     sp_1.4-2           forcats_0.4.0      stringr_1.4.0
[17] dplyr_0.8.4        purrr_0.3.3        readr_1.3.1        tidyr_1.1.0
[21] tibble_3.0.1       ggplot2_3.3.1      tidyverse_1.3.0

loaded via a namespace (and not attached):
 [1] httr_1.4.1          jsonlite_1.6.1     splines_3.6.1      modelr_0.1.8
 [5] gtools_3.8.2        assertthat_0.2.1   expm_0.999-4       cellranger_1.1.0
 [9] LearnBayes_2.15.1   pillar_1.4.4       backports_1.1.7    lattice_0.20-38
[13] glue_1.4.1          rvest_0.3.5        colorspace_1.4-1   Matrix_1.2-17
[17] pkgconfig_2.0.3     broom_0.5.6        raster_3.1-5       haven_2.2.0
[21] gmodels_2.18.1      scales_1.1.1       gdata_2.18.0       generics_0.0.2
[25] ellipsis_0.3.1      withr_2.2.0        cli_2.0.2          magrittr_1.5
[29] crayon_1.3.4        readxl_1.3.1       deldir_0.1-25      fs_1.4.1
[33] fansi_0.4.1         doParallel_1.0.15  nlme_3.1-140       MASS_7.3-51.4
[37] xml2_1.2.2          foreign_0.8-71     class_7.3-15       tools_3.6.1
[41] hms_0.5.2           lifecycle_0.2.0    munsell_0.5.0      reprex_0.3.0
[45] compiler_3.6.1      e1071_1.7-3        rlang_0.4.6        classInt_0.4-3
[49] units_0.6-6         rstudioapi_0.11    boot_1.3-22        gtable_0.3.0
[53] codetools_0.2-16    DBI_1.1.0          R6_2.4.1           lubridate_1.7.8
[57] KernSmooth_2.23-15 stringi_1.4.6       Rcpp_1.0.4.6       vctrs_0.3.0
[61] dbplyr_1.4.3        tidyselect_1.1.0   coda_0.19-3
```

4 Additional Inclusions

The "Additional Data" folder contains congressional district matrices for the full enumeration, broken into 44 files with 1 million columns and 1 file with the remaining columns. This is broken into smaller pieces to allow for local testing on the 70 precinct validation map within the paper. Unlike other files included as ZDD outputs, these are transformed such that each row is a precinct and each column is a map, which makes it easier to use but comes at a very large storage efficiency loss.

The file redist.7z contains a compressed version of the developer version of the redist package. This can be built to ensure replicability, as this version was not published on CRAN. This corresponds to Version 2.0.0 listed above. For all other figures, the CRAN version 1.3-3 is sufficient.