Replication Archive for

"Racial Disparities in Voting Wait Times: Evidence from Smartphone Data"

Keith Chen, Kareem Haggag, Devin Pope, Ryne Rohla

Review of Economics and Statistics

Overview

The analysis in this paper primarily relies on confidential and proprietary smartphone geolocation data from SafeGraph. The paper has been granted an exemption from the full data posting requirement. We cannot post the raw ping-level data, however, we have provided the final analysis datasets, public files used in analysis, and the code. Researchers interested in access to the ping-level Safegraph data may contact Veraset (formerly a business unit of SafeGraph) to purchase it: https://www.veraset.com/contact-sales

This archive contains three directories:

1. `Scripts`: This directory contains the Stata do-files that generate the data for analysis (1a-1d, 2), do-files that generate the primary tables and figures in the paper (3), and do-files that generate the primary tables and figures in the appendices (4, 5). The 0-Master.do file sets the working directories called upon in the subsequent do-files.

2. `Data-Raw`: This directory contains the raw data that we are able to distribute to the public. Due to privacy agreements, we cannot share ping-level data acquired from SafeGraph, nor the derived data. Researchers interested in acquiring these data may reach out to SafeGraph – without these data, you will not be able to run the pre-processing files (1a-1d or 2).

3. `Data-Extract`: This directory contains cleaned data used in the analysis do-files (including the primary smartphone-level dataset: `voterwaittimes.dta`). These data will allow you to reproduce all figures and tables in the text (except for Panel C of Figures 1 & 4) as well as the majority in the Appendix by running the primary analysis files (3 & 4).

Instructions

Run the following do-files in order (more complete documentation may be found in the do-files, but brief summaries are provided below):

1. `0-Master.do`: This file sets local directories and defines macros used in graphs.

2. `1a-PullandProcessPings.do`: This file assigns polling place GPS coordinates to appropriate buildings and then pulls smartphone pings associated with each polling place. To begin, the file loads (state-by-state) the set of polling place locations.[1] First, each polling place's GPS coordinates are assigned to the nearest building within that state (using the `geonear` command). Then, the coordinates are matched to building rooftops and building convex hulls (using the `geoinpoly` command). The building ID used for a polling place is set to our best guess for the polling place building based on this matching: If a roof match is found, that building is defined as the polling place. If a roof match can't be found, it is assigned to the building that satisfies a convex hull match. If neither a roof match nor a convex hull match can be found, it is assigned to the nearest building within 99 meters. If the distance to the nearest building is greater than 99 meters, then the GPS coordinates themselves are used as a "Pseudo Building". Then, we match to the Census block group of the building's centroid. Finally, we create shapefiles for the matched buildings and their convex hulls.

   The file then pulls in relevant cellphone pings. Pings are first assigned to their closest polling place building centroid, and then the dataset is limited to only those pings that fall within 100 meters of a polling place building. Finally, it creates a dummy variable equal to 1 if the ping falls within the convex hull of the building (for "Pseudo Building" cases, this equals 1 if the ping is within 35 meters of the building GPS coordinates).

3. `1b-DefineRadius.do`: This file appends each day of pings that fell within 100 meters of a polling place (November 1 – 16) and then merges to polling place information, time zones, and county/state FIPS codes. The date/time stamps are then adjusted from Greenwich time to the local time and appropriate hour offsets

[1] Note: These polling place locations were originally sourced from state and county election authorities or public resources, and we then geocoded the addresses using Google Maps API (in Python) in the majority of cases.

are applied (with non-Daylight Savings Time offsets for dates before November 6). We keep only those pings falling between November 1 – 15, as we have incomplete coverage on November 16. Variables are labeled and the data is saved as `Pings_all_days.dta`. Finally, we calculate the number of unique devices that fall within 10 meters, 20 meters,…,100 meters of the polling place building centroids on each day and save this as `radiusdata.dta` – this will be later used in Figure 1, as this information guided our decision to use 60 meters as the radius around the polling place building centroid.

4. `1c-IdentifyLikelyVoters.do`: This file begins by loading all pings that fell within 60 meters of a polling place building centroid (November 1 – 15). We create a lower bound wait time measure for each individual / polling place / day combination that is equal to the time between the earliest and latest pings within a polling place circle (it is equal to zero if there is only 1 ping in the polling place circle). An upper bound measure is composed of this lower bound measured added to: (the number of seconds between the earliest ping in a polling place and the immediately preceding ping just outside the polling place) and (the number of seconds between the latest ping in a polling place and the immediately following ping just outside the polling place). Next, we drop any of these spells that have an upper bound of less than 60 seconds. We then construct two different versions of a "likely voter" filter. The weaker version (the one primarily used in our analysis, i.e. `likelyvoter_v1`) is set equal to zero if (1) an individual shows up at more than 1 polling place on Election Day, (2) if the person shows up at their polling place on any other day than November 8. The stricter version further sets this equal to zero if that individual shows up at *any* polling place on *any* day other than November 8. A file with the device IDs and these filters is saved as `likelyvoters.dta`. We then repeat this filter classification for all other radii (for use in Figure 4).

5. `1d-IdentifyRegularPingers.do`: This file begins with all pings on November 8 (not limited to those that fall within a radius of a polling place). For each device, we calculate the number of unique hours of that day on which at least one ping was recorded. This creates a summary measure `uniquepinghours` that is used for the `consistentpinger` filter (i.e. `uniquepinghours>=12`) in the next file. This saves a file with the device ID and the variables used for that filter, `IdentifiedRegularPingers.dta`.

6. `2-CreateAnalysisDatasets.do`: This file does the bulk of the data cleaning for producing the primary analysis dataset (`voterwaittimes.dta`) and a few supplementary files (e.g. `county_voterwaittimes.dta`, `cces.dta`, etc.). The file begins by loading all pings within 60 meters of a polling place centroid on Election Day (November 8). The files used to create our primary filters are then merged with this dataset. The wait time measure is then constructed as the midpoint between the upper bound and the lower bound measures (as described in item 4 above). We additionally define a `reasonablevalues` filter that is equal to 1 if the upper bound measure is between 1 to 120 minutes (this is adjusted in the Figure 4 to examine robustness to this assumption). We then merge to Census Demographic data for the polling place block group, as well as to information on state and county characteristics/laws and create county, congressional district, and state level datasets from both our own data and from the CCES (survey data). Finally, we create a set of datasets to be used in maps.

7. `3-PrimaryAnalysis.do`: This file produces most of the figures and tables in the body of the text (except Figure 1c and 2b). The code is arranged in the order in which the tables/figures appear in the paper.

8. `4-Appendix.do`: This file produces most of the figures and tables in the Online Appendix (except Figures A1, A2, and A8 and Table A4). The code is arranged in the order in which the tables/figures appear in the Appendix.

9. `5-PlaceboDays.do`: This file produces figures that use data from non-Election Days in addition to Election Day (Figures 1c, A2, and A8).

Software and hardware requirements

- Stata (code was last run with version 15.1).

- The following Stata packages may be installed by typing "`ssc install`" followed by the package name below:
  - `estout`
  - `geonear`
  - `geoinpoly`
  - `statastates`
  - `shp2dta`
  - `spmap`
  - `maptile`

- gtools
- plotplain

Additionally, the `ebayes.ado` package is included in the "Data-Raw" folder (from http://sacarny.com/wp-content/uploads/2015/08/ebayes.ado)

- File "1a-PullandProcessPings.do" was run on a Dell Precision workstation with two 3.4 GHz 12-core Intel Xeon processors and 512 GB 2133 MHz DDR4 memory. Files "1b-DefineRadius.do" through "5-PlaceboDays.do" were run on a Macbook Pro laptop with a 2.4 GHz 8-Core Intel Core i9 processor and 64 GB 2667 MHz DDR4 memory.

CONTENTS

File Name	Description	Provided
	Stata Do-Files (Located in Scripts Directory)	
0-Master.do	Sets the directories used in subsequent scripts, as well as macros used in plots.	Yes
1a-PullAndProcessPings.do	Matches polling place GPS coordinates to buildings & pull pings within 100 meters of those building centroids.	Yes
1b-DefineRadius.do	Appends several days of pings, merges to polling place information, time zones, and county/state FIPS codes, adjusts time variables, labels variables, and constructs the number of unique devices by various radii used later to define the primary radius around a building centroid (60m).	Yes
1c-IdentifyLikelyVoters.do	Constructs the likelyvoter filter.	Yes
1d-IdentifyRegularPingers.do	Constructs data used to later define the regularpinger filter.	Yes
2-CreateAnalysisDatasets.do	Performs the bulk of the data cleaning for producing the primary analysis dataset (voterwaittimes.dta) and a few supplementary files (e.g. county_voterwaittimes.dta, cces.dta, etc.).	Yes
3-PrimaryAnalysis.do	Produces the primary figures and table of the paper (Table 1, Figures 1 – 4) except for Figure 1c (produced in 5-PlaceboDays.do) and 2b (produced in ArcGIS).	Yes
4-Appendix.do	Produces the primary tables and figures of the Online Appendix except for Appendix Figures A1 (produced in ArcGIS), A2 and A8 (produced in 5-PlaceboDays.do), and A4 (produced in 3-PrimaryAnalysis.do)	Yes
5-PlaceboDays.do	Produces Figures 1c and Appendix Figures A2 and A8.	Yes
	Raw Data Files (Located in Data-Raw Directory)	
Pings_100m_PollingPlace_11_**_16.dta (** = 1 to 16), Pings_all_days.dta, 11_16_All_SortGH_Day8.dta	Datasets of Pings (e.g. Pings_100m_PollingPlace_11_8_16.dta is all pings within 100 meters of a polling place building centroid on November 8, 2016).	No
PollingPlaces2016_w_TimeZones_and_Buildings.dta	Processed file of polling place locations and associated information.	Yes
block_group_data_2017.dta	Census Block Group level demographics from the 2013-2017 ACS	Yes
countypres_2000-2016.dta	County and State Vote Shares From:https://dataverse.harvard.edu/file.xhtml?persistentId=doi:10.7910/DVN/VOQCHQ/FQ9NBF&version=5.0	Yes

NCSL_2016laws.csv	State Law variables compiled manually from Internet Archive snapshots of the National Conference of State Legislatures website (links listed in 2-CreateAnalysisDatasets.do)	Yes
online_table4-2.dta	Chetty & Hendren (2018) County-Level measures. From: https://opportunityinsights.org/wp-content/uploads/2018/04/online_table4-2.dta	Yes
2013-17-ACS-county-data.csv	County level demographics from the 2013-2017 ACS	Yes
cb_2016_us_cd115_500k.shp	115th Congress Congressional Districts shapefiles. From: https://www.census.gov/geographies/mapping-files/time-series/geo/carto-boundary-file.2016.html	Yes
CCES16_Common_OUTPUT_Feb2018_VV.dta	Cooperative Congressional Election Study (CCES) Common Content, 2016 Wave. From: https://doi.org/10.7910/DVN/GDF6Z0	Yes
Polling_Places_Voters_AA_Nums.dta	Processed data from L2's 2016 General Election national voter file (proprietary). Contains the number of registered voters assigned to vote at each polling place and the number of votes cast.	No
Analysis Data Files (Located in Data-Extract Directory)		
voterwaittimes_filtered.dta	This is the smartphone-level analysis dataset corresponding to the sample used in the majority of the analysis (i.e. 154,489 wait times after applying the four filters).	Yes
voterwaittimes.dta	Smartphone-level analysis dataset before any of the four filters are applied.	Yes
radiusdata.dta	Cleaned dataset used to produce Panels A and B of Figure 1.	Yes
county_voterwaittimes.dta	Cleaned dataset used to produce Appendix Tables B2, B3, C1, and C3.	Yes
cces_comparison.dta	Cleaned dataset used to produce Appendix Tables A4 and C2.	Yes