Data and Code for: “Measuring Racial Discrimination in Bail Decisions”

DATA ACCESS:

The data for this paper contains confidential information about criminal defendants in New York City and so is restricted-use. Accessing the data can be done by entering a data sharing agreement with the New York State Division of Criminal Justice Services and Office of Court Administration. Inquiries can be sent to:

DCJS Research Request Team
Office of Justice Research and Performance, New York State Division of Criminal Justice Services
80 South Swan St., Albany, NY 12210
DCJS.ResearchRequests@dcjs.ny.gov
www.criminaljustice.ny.gov

Authors will assist with any reasonable replication attempts.

CODE:

The code for the project is included in three subfolders: build_stata, results_stata, and results_matlab. We will discuss the files in each of these subfolders separately.

“build_stata” Folder

The code in this folder first constructs the analysis samples from various datasets provided by the New York State Division of Criminal Justice Services (DCJS). From the microdata, aggregated judge-level datasets are created which are the primary datasets analyzed in the paper.

1. 1_casedata.do – Cleans the raw data to construct the main case-level dataset.

2. 2_judge_bw.do – Uses the microdata to estimate release and misconduct rates by race at the judge-level and then saves a judge-level dataset named judge_bw_main.dta. This analysis sample is used extensively in the results section of the paper and code.

3. 3_judge_bw_leadlag.do -- This file splits at the median judge-specific case and estimates disparate impact within each split (i.e. in the first half of judge’s cases and the second half of judge’s cases).

4. 4_judge_bw_baelsines.do – Uses the microdata to estimate predicted misconduct rates (based on defendant observables) by race at the judge-level.

5. 5_judge_bw_moneybail.do – Uses the microdata to estimate release on recognizance (ROR) and misconduct rates by race at the judge-level.

6. 6_judge_bwh.do -- Uses the microdata to estimate release and misconduct rates by race at the judge-level. The key difference relative to 2_judge_bw.do is that the comparison by race is now between Black and/or Hispanic defendants vs. Non-Hispanic white defendants.

7. 7_judge_bw_borough.do -- Uses the microdata to estimate release and misconduct rates by race at the judge-level. The key difference relative to 2_judge_bw.do is the estimation is done entirely separately by borough.

8. 8_judge_bw_xvar.do -- Uses the microdata to estimate release and misconduct rates by race at the judge-level. The key difference relative to 2_judge_bw.do is that the analysis is done entirely separately by different characteristics of the defendants (for example, by the type of crime the defendant is arrested for).

9. 9_judge_bw_controls.do -- Uses the microdata to estimate release and misconduct rates by race at the judge-level. The key difference relative to 2_judge_bw.do is that the estimation also includes control variables (such as the criminal history of defendants) when estimating a judge's average misconduct and release rate.

10. 10_judge_bw_timefe.do -- Uses the microdata to estimate release and misconduct rates by race at the judge-level. The key difference relative to 2_judge_bw.do is that the estimation allows for time effects that vary at the judge level in various ways.

"results_matlab"

The code in this folder estimates the hierarchical marginal treatment effects (MTE) model and performs simulations related to the MTE model. Additionally a number of subroutines are called, descriptions of which are described below.

1. 1_convert.do -- Translates the first-step estimates (i.e. release and misconduct rates by race at the judge level) to MATLAB format.

2. 2_estimate.m -- Applies the simulated minimum distance (SMD) estimation procedure to the first-step judge-specific estimates.

3. 3_make_table.m -- Formats the SMD estimates to make the paper and appendix tables that utilize the hierarchical model estimates (Table 4: Hierarchical MTE Model Estimates and Appendix Table A18: Hierarchical MTE Model Hyperparameter Estimates).

4. 4_make_lealag_table.m – Constructs the lead and lag estimates that are utilized in tables that regress characteristics of judges on model parameters (Appendix Table A20-A22).

5. 5_est_posts.m -- Applies the posterior calculation to the SMD estimates.

6. 6_get_posts.do -- Translates the posteriors to a format Stata can read.

7. 7_get_policy_sims.m -- Simulates the New York city data generating process and then retrieves posteriors that are used in the policy simulations.

8. 8_compile_policy_sims.m – Formats the policy simulations to the versions in the paper (Table 5: Disparate Impact Decompositions and Table 6: Policy Simulations).

SUBROUTINES

1. fit_psi – Function for fitting posterior judge-level parameters.
2. get_estimates – Function that obtains estimates from the simulated minimum distance (SMD) procedure.
3. get_g – Function used to get the gradient of the moment vector, for getting standard errors for the SMD estimates.
4. get_m – Function that computes target moments from simulated and real data.
5. get_posts – Function that computes posteriors from hyperparameter estimates and judge-specific estimates.
6. get_ud – Function that translates a set of SMD estimates to measures of disparate impact.
7. get_v – Function that estimates the variance of the moment vector, for getting standard errors for the SMD estimates.
8. initial_posts – Function that obtains initial posteriors.
9. post_lhood – Likelihood used to compute posteriors.
10. signalcdf – Function that returns the cumulative distribution of risk signals under the model's parameterization.
11. smd – Function that computes the simulated minimum distance objective.
12. translate_params – Function that translates a given set of hyperparameter estimates to the interpretable moments in Table 4.
13. translate_posts – Function that translates posteriors to judge-level bias and signal quality estimates.
14. wmean – Function for computing weighted means.

<u>"results_stata" Folder</u>

Includes code to estimate the majority of the tables and figures in the paper.

1. 1_tables.do – Estimates and creates all the main tables in the paper (excluding those created in "results_matlab", for example, Table 4 and Table 5). Line 118 contains the contents of what is produced in the file.

2. 2_figures.do – Estimates and creates all the main figures and appendix figures in the paper.

3. 3_appendix_tables.do – Estimates and creates all the appendix tables in the paper.

4. 4_text_stats.do – Generates statistics that are cited in the paper, but are not taken directly from a table or figure.

Replication Instructions
The file “run_build_stata.do” executes all of the files in the folder “build_stata.do”. There is also commented out code at the top of the file that installs the required packages in order to replicate the results in the paper. This file must be run first as it builds all of the analysis datasets that are used in the results folders. In order to run run_build_stata.do, the directory at the top of the file must be edited. The files within this directory (1_casedata.do through 10_judge_bw_timefe.do do not set any additional directories).

Next, all of the files in results_matlab must be run in the order that they appear (i.e. 1_convert.do first, followed by 2_estimate.m, and so on). In order to run these files, the directory at the top of each matlab file must be edited. Once this is done, the file run_results_matlab can be executed to run all of the contents of the results_matlab folder.

Lastly, the file “run_results_stata.do” executes all of the files in the folder “results_stata” in the necessary order and produces tex files for all tables as well as all the figures in the paper and online appendix. The directory at the top of the file must be edited to successfully replicate the results.

Controlled Randomness
Some estimation codes uses random numbers. For example, standard errors are often estimated via bootstrap. Therefore, in order to replicate the results a seed must be set. The seeds used for this version of the results are set within individual files. For example, the seed in 1_tables.do is set on line 40. Seeds are set within each individual file that produces results, but if you are generating a subset of the results within a file, there may be slight differences in standard errors values.

Computational and Software Requirements
The Stata code was last run in Stata-MP version 16 on a Mac Pro (2019) with 3.5 GHz 8-core Intel Xeon W and 192 GB 2666 MHz DDR4. The MATLAB code was last run in MATLAB version 2018b in Windows 10 Pro with Intel Core i7-7600U @ 2.80GHz. The total runtime of all the code is approximately

Additionally, the following Stata packages are required to estimate the results or produce the figures: reghdfe, ivreg2, blindschemes, psacalc and unique. There is code to install these (which is commented out) in the file run_build_stata.do.

DATA:

Inside the data folder there are three different folders: cleandata, rawdata, and simulations. The rawdata contains the raw data that was received from DCJS. The cleandata folder contains analysis samples created from this raw data. The folder simulations contain model estimates from the hierarchical marginal treatment effects model. All of these folders are empty in the replication package, as they require access to confidential, restricted-use data.

The first table we present summarizes all the datasets needed to create the main case-level analysis sample (i.e. the raw data received from DCJS (which are used in the file /build_stata/1_casedata.do), not the analysis samples constructed from the raw data. We will discuss the various analysis data samples below.

Dataset name	Description
NamesBail.dta	Arraignment-level dataset that contains limited information about the outcome of a given arraignment.
Arraignment.dta	Arraignment-level dataset that contains additional information about the outcome of an arraignment, such as
CriminalHistory.dta	Defendant-level dataset that contains information on prior criminal history, such as prior arrests and prior failure to appears.
Warrant.dta	Contains information on bench warrants issued which is used to define failure to appear (FTA).

Analysis Samples

The table below creates a table of all data sets that are generated from the raw data and used in the analysis in the paper, as well as in which files these analysis samples are created in.

File Name	Description	Created in?
main_analysis.dta	Arraignment-level dataset. Many analysis restricts to sampleMain==1, which are the set of arraignments that meet the conditions described in the section “Sample and Summary Statistics”. This file is used as the	/build_stata/ 1_casedata

	input to the judge-level datasets that are described below.	
judge_bw_main.dta	Judge-level estimates of misconduct rates and release rates, displayed in Figure 2 of paper.	/build_stata/ 2_judge_bw
judge_bw_leadlag_mcase.dta	Judge-level estimates, but separately for before-median and after-median case, as discussed in Section 5.2.	/build_stata/ 3_judge_bw_leadlag
judge_bw_baselines.dta	Judge-level estimates that compute predicted misconduct rates, rather than actual misconduct rates.	/build_stata/ 4_judge_bw_baselines
judge_bw_moneybail.dta	Judge-level estimates of ROR rates rather than release rates.	/build_stata/ 5_judge_bw_moneybail
judge_bwh.dta	Judge-level estimates of misconduct rates and release rates, comparing Black and Hispanic defendants to white non-Hispanic defendants.	/build_stata/ 6_judge_bwh
judge_bw_borough_all.dta	Judge-level estimates of misconduct rates and release rates separately by borough.	/build_stata/ 7_judge_bw_borough
judge_bw_xvar.dta	Judge-level estimates of misconduct rates and release rates for various subsamples of the data.	/build_stata/ 8_judge_bw_xvar
judge_bw_controls.dta	Judge-level estimates of misconduct rates and release rates controlling for defendant characteristics.	/build_stata/ 9_judge_bw_controls
judge_bw_borough_y_by_m.dta	Judge-level estimates of misconduct rates and release rates controlling for judge x year interactions and judge x month interactions.	/build_stata/ 10_judge_bw_timefe
judge_bw_borough_ym.dta	Judge-level estimates of misconduct rates and release rates controlling for judge x year-month interactions	/build_stata/ 10_judge_bw_timefe
judge_bw_borough_ym2.dta	Judge-level estimates of misconduct rates and release	/build_stata/ 10_judge_bw_timefe

	rates controlling for judge x year-month squared interactions	
both_nobeta0_nosigma0.csv	Estimates of hierarchical marginal treatment effects model	/results_matlab/ 3_estimate.m
lealag_both_nobeta0 _nosigma0_param_out.csv	Estimates of hierarchical marginal treatment effects model, separately estimated on first half of judge cases and second half of judge cases	/results_matlab/ 5_make_lealag_table.m
judge_posteriors.dta	Posterior estimates of judge-level parameters for the hierarchical marginal treatment effects model.	/results_matlab/ 7_get_posts.do

Additional Intermediate Data Files Created

Many of the build files create intermediate files that are never used in the analysis. For example, 2_judge_bw.do creates three intermediate files: judge_bw_block1.dta, judge_bw_block2_`outcome'.dta, and judge_bw_block3.dta, where `outcome' is either any misconduct, case FTA, rearrest, or violent rearrest (see Appendix Table A15 for the alternative outcomes). These different blocks are aggregated in order to construct the analysis sample which is judge_bw_main.dta. Many files share a similar structure of creating a limited number of intermediate files that are aggregated to create the final analysis sample. These temporary files are stored in the folder /data/tempdata.

Empty Directories

The directories "/replication/logs" and "/replication/results" are included in the replication package, but both are empty. The "logs" file collects all the log files that are generated from executing various Stata files. The directory "results" is where all tables and figures are placed.

References

New York State Division of Criminal Justice. 2018. "NYC DCJS Administrative Data." (accessed in 2018).