Simulation Study Overview

November 22, 2021

Contents

1 Data

We have two main data sources: sentencing data and judge schedules, both of them are for the 2001 fiscal year. We obtained the data from the authors of Hester and Hartman (2017). We briefly describe both datasets below.

1.1 Sentencing Data

The sentencing data contains information about 17,671 sentencing events in South Carolina from August 2000-July 2001. Each sentencing event contains an identifier for the judge who heard the case, the county the case was heard in, categorical variables describing the offense, and some defendant characteristics (e.g. race, age, and criminal history). There are 50 judges and 46 counties in the dataset. A full list of the variables[1] we use can be found in Table 1. Figure 1 contains a histogram of the number of cases sentenced each county. Figure 2 contains a histogram of the number of cases sentenced by each judge. Figure 3 contains a histogram of the share of cases that went to trial for each judge. A more detailed description of the original sentencing data, can be found in Appendix A. There, we describe the original data files, the data cleaning, and how some of the variables were created.

[1]The original dataset we received also has eight additional fields that correspond to binary variables indicating whether the offense is of a certain type. They are created from the *offtyped* field in the data. We omit them, but include the *offtyped* field.

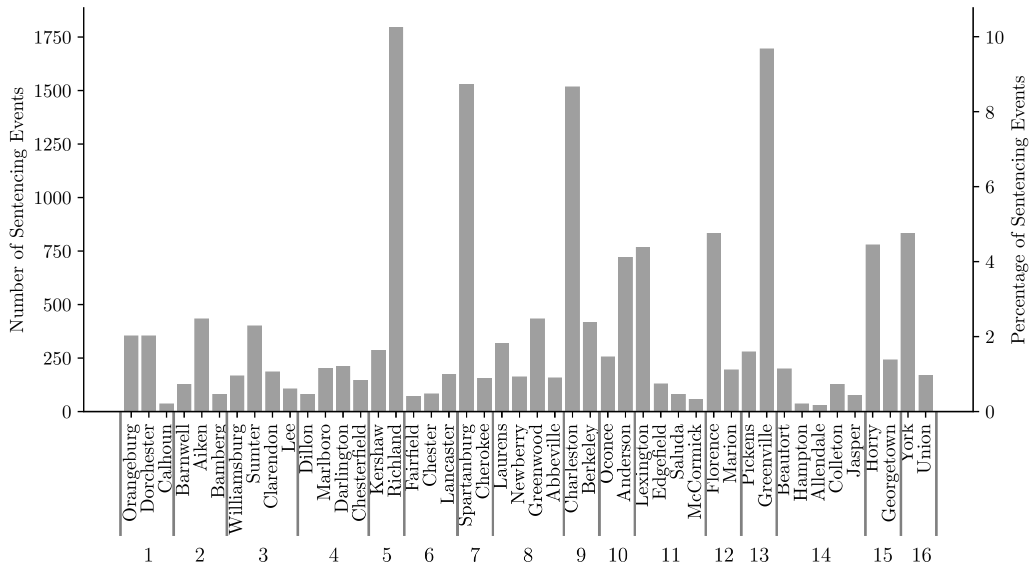


Figure 1: A histogram of the sentencing events by county. The counties are grouped into 16 circuit courts. The number underneath each group indicates the corresponding circuit court.

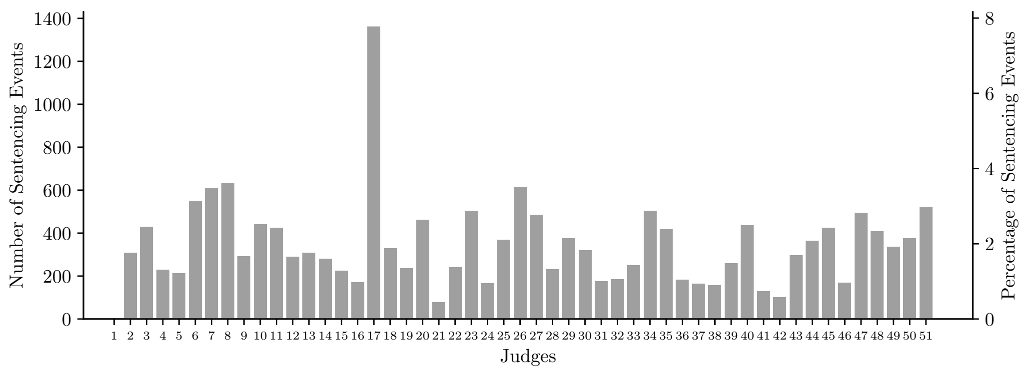


Figure 2: A histogram of sentencing events by judge.

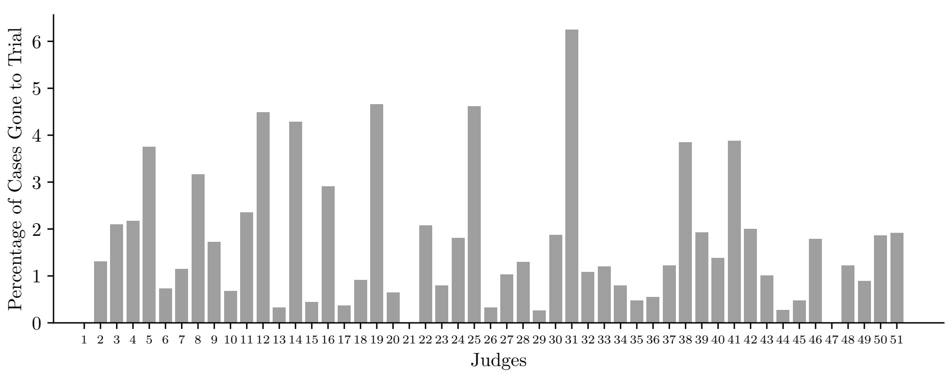


Figure 3: Percentage of sentencing events for each judge that resulted in trial.

Table 1: Sentencing Data Variables

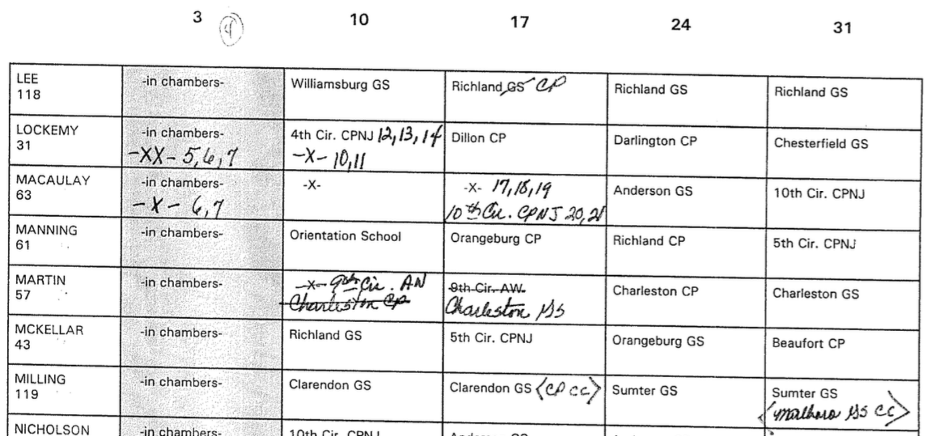
Variable	Description
date	date of sentence
county	county where the sentence was decided
circuit	circuit court where sentence was decided
judge	numerical identifier for the judge who head the case
trial	binary variable indicating whether the case went to trial
incarc	binary variable indicating whether the sentence includes incarceration
statute	the code of law that the offender broke (e.g. 56-05-0750(B)(1))
offdescr	a description of the offense (e.g. driving under the influence)
counts	numerical variable
statute_first	the code of law that the offender broke, identical to 'statute' variable
offdescr_first	a description of the first offense, similar to offdescr
offtyped	detailed categorical offense type, there are 10 values (rape, assault, theft, etc).
sgc_offcode	a numerical code that is used to determine the minimum sentence multiplier
offtypeLibHyp	categorical offense type (property, violent, drug, other)
offser	offense seriousness, ranges from 1-8
ccpnts	numeric values between 1 and 68
ccpts99	commitment score, an alternative measure of offense seriousness.
crimhist	defendant criminal history, takes 5 categorical values
ppoints	numerical values between 0-1014
male	binary variable indicating sex
age	age of defendant at sentencing, ranges 15-81
black	binary variable indicating whether defendant is black
sentence	length of sentence in months, ranges 0-11988
expmin	expected minimum sentence

Sample In the raw data, there are 51 distinct judge ID's. However, according to Hester and Hartman (2017), Judge 1 is a combination of several judges that had few sentencing events. As a result, we exclude Judge 1 from our sample, which leaves us with 17,516 observations. Additionally, after removing the sentencing events head by Judge 1, there are 1,482 sentencing events with the *date* field missing. We impute the dates for these events as described in Appendix A.

1.2 Master Calendar Data

This dataset contains information about each judge's assignment for each week of the fiscal year 2001. A snapshot of the calendar can be seen in Figure 4. The first column indicates the judge names. The remaining columns correspond to the weeks in that month. The number written on top of each column is the date of the Monday of that week. Each row gives the schedule of a judge in that week. The master calendar data is cleaned as described in Appendix B. Each assignment generally contains the county each judge was assigned to and the assignment type. The assignment type refers to the kind of cases a judge is scheduled to hear (civil or criminal).

Figure 4: Snapshot of Judge Calendar for November 2000



	3	10	17	24	31
LEE 118	-in chambers-	Williamsburg GS	Richland ~~GS~~ CP	Richland GS	Richland GS
LOCKEMY 31	-in chambers- -XX- 5,6,7	4th Cir. CPNJ 12,13,14 -X- 10,11	Dillon CP	Darlington CP	Chesterfield GS
MACAULAY 63	-in chambers- -X- 6,7	-X-	-X- 17,18,19 10th Cir. CPNJ 20,21	Anderson GS	10th Cir. CPNJ
MANNING 61	-in chambers-	Orientation School	Orangeburg CP	Richland CP	5th Cir. CPNJ
MARTIN 57	-in chambers-	~~-X-~~ 9th Cir. AW ~~Charleston CP~~	~~8th Cir. AW.~~ Charleston GS	Charleston CP	Charleston GS
MCKELLAR 43	-in chambers-	Richland GS	5th Cir. CPNJ	Orangeburg GS	Beaufort CP
MILLING 119	-in chambers-	Clarendon GS	Clarendon GS <CP CC>	Sumter GS	Sumter GS <Marlboro GS CC>
NICHOLSON	-in chambers-	10th Cir. CPNJ			

1.3 Assignments

The judges were generally assigned to either specific counties or to circuit courts. In the master calendar data, 75% of all judge assignments were to either a specific county or to a circuit court. The remaining assignments were to one of the categories in Table 2. As can be seen, outside of "in chambers" they generally pertain to special circumstances.

Table 2: Assignments that are not to a specific county or circuit court

Other Assignment
In chambers
Orientation School
Medical
Family Death
Sick
Military
X

1.3.1 Assignment Types

Judge calendar assignments generally have an acronym indicating the type of the assignment (e.g. Marion GS). There are eight different assignment types, each with a corresponding acronym.

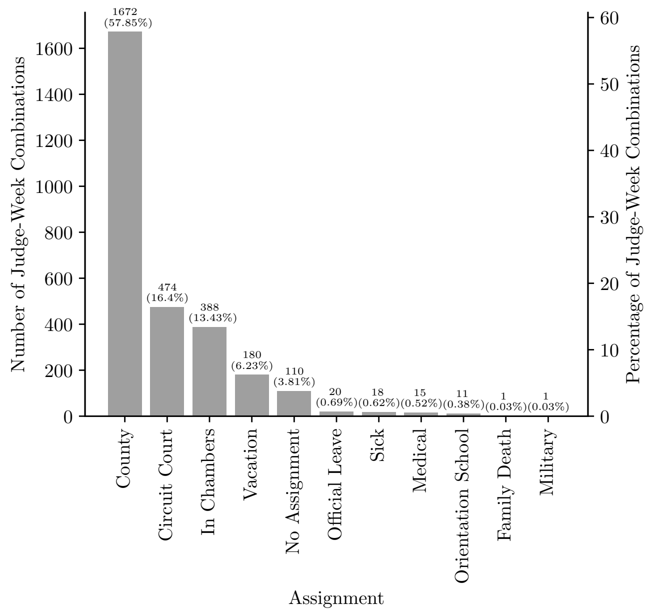


Figure 5: A histogram of the assignments observed for a judge-week combination in the master calendar.

Assignments can have more than one type (e.g. Marion GS CC). Our study focuses on criminal cases, which are heard in general sessions (GS). As a result, much of our analysis focuses on days in which judges only had GS assignments. In other words, in much of our analysis we exclude all assignments that are not exclusively of type GS (for example GS CC would be excluded). Some assignment types, like CP (common pleas), are exclusively for civil cases. A full list of the assignment type acronyms and their meanings can be found in Table 3. Figure 5 contains a histogram of the different assignments. Figure 6 contains a histogram of the assignment type (according to the master calendar) during which sentencing events occurred. A description of how Figure 6 was created can be found in Appendix B.

Table 3: Assignment Type Acronyms. The share column specifies the share of all assignments that include that assignment type.

Acronym	Meaning	Share
GS	General Session	0.36
CC	Circuit Court	0.02
SGJ	State Grand Jury	0.01
CP	Common Pleas	0.26
CPNJ	Common Pleas Non-Jury	0.13
PCR	Post Conviction Relief	0.03
Capital PCR	Capital Post Conviction Relief	< 0.01
AW	Administrative Week	0.01

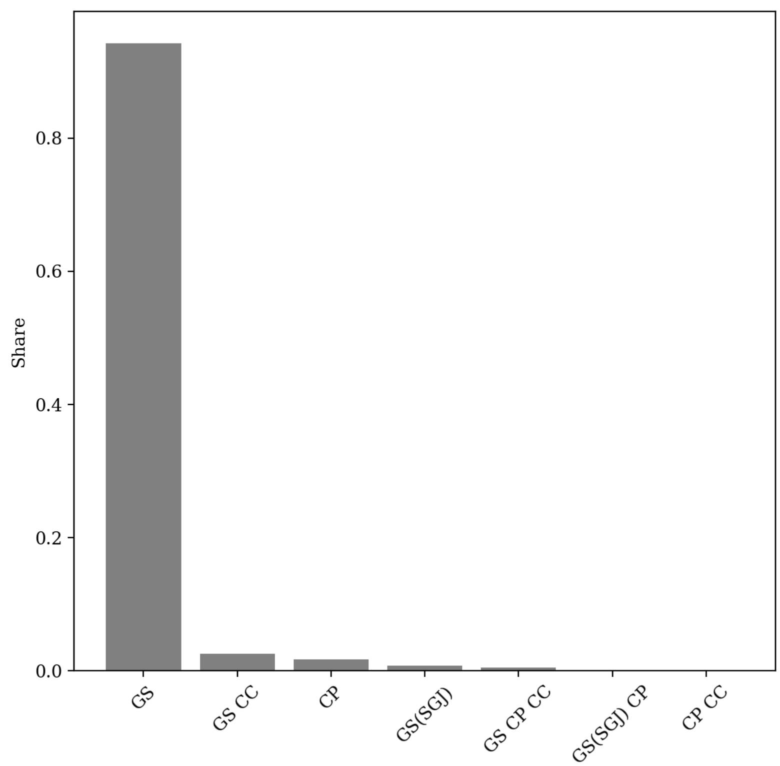


Figure 6: A histogram of the assignment type (on the master calendar) during which the sentencing events occurred. A description of how this figure was created can be found in Appendix B.

1.3.2 Parsing

As can be seen from Figure 4, the assignments in the raw calendar data are on a weekly basis. However, several parts of our analysis stood to benefit from having the daily assignments for each judge, as opposed to only the weekly assignment[2]. In this section, we describe how we parsed the raw calendar data to determine each judge's scheduled location and assignment for each day of the year. In the resulting data, the unit of observation is judge, day, county.

A judge can be assigned to one or multiple counties in a week, and we observe 5 different kinds of assignments: single assignment with no dates, single assignment with dates, multiple assignments with no dates, multiple assignments with some dates, and multiple assignments with all dates. Table 4 contains examples of each kind of assignment.

Table 4: Assignment Types

Assignment	**Example**
Single with no dates	Marion
Single with dates	Marion 23, 24, 25
Multiple with no dates	Marion, Horry
Multiple some dates	Marion 23, 24, Horry
Multiple all dates	Marion 23, 24, Horry 25, 26, 27

Note that the only potentially ambiguous assignment types are multiple assignments and multiple assignments with some dates[3]. By ambiguous, we mean that the calendar data does not specify a unique location for a judge. For example, in the single assignment case the judge is supposed to be only in one county, so we set the judge to be in that county for every day in that week. However, in the multiple assignment case, it is not clear in which county the judge should be. For these ambiguous assignment types, we set the judge to be in all counties that he is scheduled to be in.

Our analysis involves counting the number of days that a judge was assigned to a specific county. For days in which a judge is assigned to multiple counties, we normalize the number of days he is assigned to each county so that the sum of the assignments for each day is equal to one. For example, if Judge 2 is scheduled for both Marion and Horry in the week of March 19-March 23, for each day

[2]For example, this allowed us to more accurately count the total number of days a judge was assigned to a county, or to more accurately determine if there was discrepancy between the sentencing data and the calendar data.

[3]Figure 7 contains the relative frequencies of the different assignment types, and shows that ambiguous assignments are rare.

of that week our data would include an observation for Judge 2 in Horry and an observation for Judge 2 in Marion. Furthermore, for each of these days we would say that Judge 2 was assigned to 0.5 days in Horry and 0.5 days in Marion. An example of what our parsed data looks like can be seen in Table 5.

Table 5: Example of Parsed Calendar Data, here Judge 3 is scheduled for both Horry and Greenville on March 19.

Judge	**Day**	**County**	**Days Assigned**
2	2001-03-19	Marion	1
3	2001-03-19	Horry	0.5
3	2001-03-19	Greenville	0.5

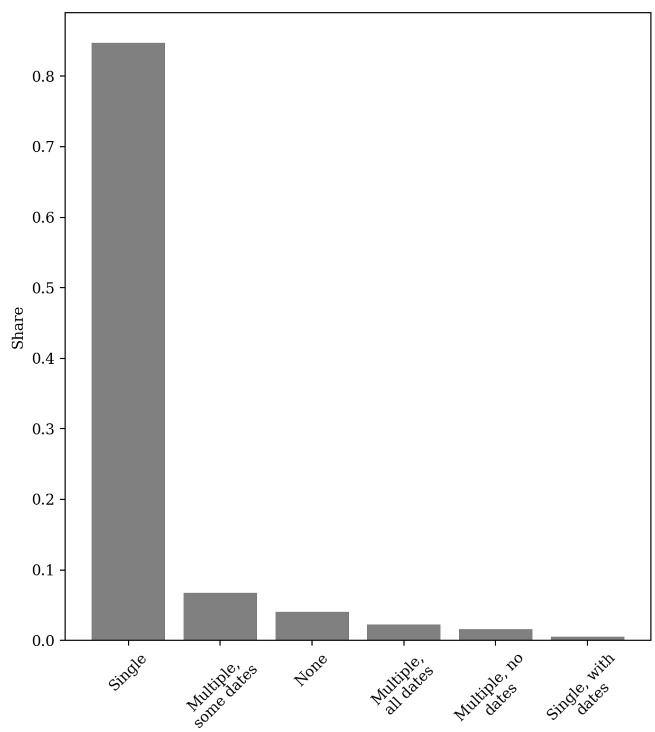


Figure 7: A histogram of the relative frequency of the different assignment types from the master calendar.

1.4 Merging the Sentencing Data and the Calendar Data

One of the main challenges of using the two datasets is that they don't share a common identifier for judges. The sentencing data uses numerical IDs to identify judges, whereas the calendar data uses judge names to identify them. There is no mapping from judge numbers to judge names in the raw data. We map judge names to judge IDs as follows.

First, for each judge name in the calendar data, we construct a sequence of the locations he was scheduled to be in in each week. This is a sequence with 52 elements, where the i^{th} element of the sequence contains the judge's scheduled location in the i^{th} week of the year. Similarly, for each judge ID in the sentencing data, we construct a sequence of the locations in which that judge ID appeared[4]. Next, we define a similarity metric for these sequences. We define the similarity between two sequences, $a = (a_i)$ and $b = (b_i)$, as the number of times there is componentwise agreement between a and b (i.e. the number of times $a_i = b_i$). For example, if $a =$(Horry, Aitken, Richmond) and $b =$(Dorchester, Aitken, Richmond) their similarity would be 2.

We then calculate the similarity between every possible judge name-judge ID pair. We map each judge ID to the judge name that has the most similar sequence. We observe no ties using this approach. That is, for each judge ID there is a unique judge name that whose sequence is most similar to the judge ID, and every judge ID is mapped to the judge name with the most similar sequence. The resulting mapping is alphabetical with respect to judge last names and is given in Appendix D.

We also tried two other methods, and they yielded the same results. A description of the alternative approaches can also be found in Appendix D. After merging the two data sources, we noticed that there are certain cases in which they conflict. For example, sometimes the sentencing data will indicate that a judge sentenced a case in a county he was not scheduled to be in according to the master calendar data. However, less than 3% of the events in the sentencing data conflict with the master calendar data. A more thorough analysis of the discrepancies between the two data sets can be found in Appendix C.

[4] Recall that every sentencing event contains the ID of the judge who heard the case as well as the county in which the case was heard.

2 Model

We build on the model developed by Wang (2019). There are three agents: the judge, the defendant, and the prosecutor. The prosecutor proposes a plea offer, and the judge and the defendant choose whether to accept. The game evolves in the following steps:

1. The defendant chooses the judge/decides when to go to court
2. The prosecutor makes a plea offer
3. The defendant decides whether to accept the plea offer or go to trial

Each defendant is characterized by the following quantities: θ - the probability of conviction at trial, and τ - the expected sentence length if convicted in trial. Each defendant additionally has an idiosyncratic cost of going to trial, $c_d > 0$. A defendant will accept a plea offer, s, if $s \leq \theta\tau + c_d$.

Judges are modeled by their harshness, h. Given a defendant with an expected sentence length of $\theta\tau$ and a plea offer, s, the lowest plea offer the judge will accept is denoted by $l_j(\theta\tau)$ and the maximum plea offer the judge will accept is denoted by $u_j(\theta\tau)$.

As a result, given a specific judge j, the optimal sentence for the prosecutor to offer is $s^* = \min(\theta\tau + c_d, u_j(\theta\tau))$. This quantity is also the defendant's cost. Finally, the case goes to trial if $\theta\tau + c_d < l_j(\theta\tau)$.

The model has two additional parameters: r-the maximum number of weeks a defendant can delay their hearing, and v, the defendant's visibility into the judge schedules. Currently, South Carolina posts judge schedules well into the future on its website[5]. As such, we will set $v = \infty$ for simplicity. Thus, a defendant can choose among the next r weeks for his/her sentencing.

3 Estimation

For the simulation, we have to estimate several of the model's parameters. In this section, we provide a detailed description of how we estimate each parameter, including the sample used.

[5]www.sccourts.org/calendar/dspCCJudgeAsgMenu.cfm. Accessed on November 4, 2021.

3.1 Assumptions/Restrictions

Table 6: Assumptions/Restrictions Used

Assumption/Restriction	Section Used
We only use the trial sentencing events to estimate θ and τ	3.3
We add the point $(0, 0)$ to the convex hull of every judge	3.5
We assume (TBD) when using regression model	**??**

3.2 Judge Weekly Availability

We use the following two assumptions to calculate the number of days in each week that each judge was available to hear criminal cases.

- We only consider a judge to be available if he has an assignment of type 'GS' (i.e. days with assignments of 'GS/CC' or 'GS/CP' would be excluded).
- If a judge is assigned to multiple counties in a day, we assume his availability is evenly split across counties. For example, if a judge is assigned to n counties in a day, and m of those assignments are of type 'GS', then we would say that that judge was available for m/n 'GS' days on that day.
- We assume that judges are not available on South Carolina state holidays[6].

Using the restrictions/assumptions described above, we created an auxiliary dataset with each judge's availability for criminal cases for each week of the year.

3.3 Conviction Probability and Expected Sentence Length - $\theta\tau$

Following Hester and Hartman (2017), we estimate θ, τ using a hurdle model. In a hurdle model, the probability of a random variable, y, is modeled using two separate models: one model is for modeling probability that $y = 0$ and a separate model is used for modeling $f(y|y > 0)$. Our hurdle model estimates $P(y = 0)$ using a logit model. As in Hester and Hartman (2017), we use a negative

[6]We get the list of South Carolina's holidays from: https://www.public-holidays.us/US_EN_2001_South20Carolina. Accessed on November 8, 2021

binomial model to estimate $f(y|y>0,x)$. The full hurdle model is then:

$$\Pr(y_i=0|x_i)=\frac{1}{1+\exp(-\beta^T x_i)}=(1-\theta_i)$$
$$\Pr(y_i|x_i)=\theta_i f_{NB}(y_i|x_i) \text{ for } y_i>0$$

where $f_{NB}(y|x)$ is the conditional probability of y given x under the negative binomial model. Here, x is a vector of covariates for the defendant including offense type, offense seriousness, and race. We use the exact same model as Hester and Hartman (2017). For a given defendant, x_i, $\theta_i=\Pr(y_i=0|x_i)$ and $\Pr(\tau|x_i)=\theta_i f_{NB}(\tau|x_i)$.

3.3.1 Sample

We only use the observations in the data that went to trial. There are only 258 observations in the data meeting this criteria. Of these 258 observations, 247 resulted in a conviction.

3.3.2 Probability of Conviction at Trial - θ

We estimate this using a logit model. In the logit model, the optimization problem is as follows:

$$\min_{w,c}\sum_{i=1}^{n}\log(\exp(-y_i(X_i^T w+c))+1)$$

We split our sample into train and test sets using a 67/33 split to evaluate the performance of our classifier. We use the following variables to predict this quantity: Black, Offense Type, and Offense Seriousness. We represent all variables as binary variables. Our classifier's performance can be seen in table 7. The classifier outputs prediction probabilities, which is what we use for θ. The classifier's prediction probability can be interpreted as the probability that the observation will have a value of 1 for the incarceration variable. The final classifier we use to predict θ for all observations in our dataset is trained on the full training set.

Table 7: Evaluation Metrics for Classifier

Metric	**Score**
AUC	0.91
F1	0.96
Accuracy	0.93

3.4 Defendant Cost of Trial - c_d

Recall that $i = 1, \ldots, I_j$ are the plea bargains judge j oversaw, and that their sentence is given by $s_i = \min(\theta_i \tau_i + c_d(i), u_i)$, where $u_i = u_j(\theta_i \tau_i)$ and $u_j(\cdot)$ is defined above. We define the sets $\mathcal{I}_j^1$ and $\mathcal{I}_j^2$ as follows:

$$\begin{aligned} \mathcal{I}_j^1 &= \{i = 1, \ldots, I_j : s < u_i\}, \\ \mathcal{I}_j^2 &= \{1, \ldots, I_j\} \setminus \mathcal{I}_j^1 = \{i = 1, \ldots, I_j : s_i = u_i\}. \end{aligned}$$

Then, we can infer $c_d(i) = s_i - \hat{\theta}_i \hat{\tau}_i$ for $i \in \mathcal{I}_j^1$. On the other hand, we can only infer that $c_d(i) \geq u_i - \hat{\theta}_i \hat{\tau}_i$ for $i \in \mathcal{I}_j^2$.

Next, we let K_j denote the number of trials judge j oversaw, which are indexed by $k \in \mathcal{K}_j = \{1, \ldots, K_j\}$. Recall that a case goes to trial if $\theta_i \tau_i + c_d(i) < l_i$. Thus, for $i = 1, \ldots, K_j$, we infer that $c_d(i) < l_i - \hat{\theta}_i \hat{\tau}_i$.

In summary, although we can impute $c_d(i)$ exactly for $i \in \mathcal{I}_j^1$, we can only infer that $c_d(i)$ falls into an interval for $i \in \mathcal{I}_j^2 \cup \mathcal{K}_j$. However, assuming a parametric distribution for c_d, e.g., $c_d \sim N(\mu, \sigma^2)$, we can estimate its parameters using maximum likelihood. To this end, we let F and f denote the cdf and pdf of the distribution of c_d, and define the likelihood function $\mathscr{L}_j$ as follows:

$$\mathscr{L}_j = \prod_{i \in \mathcal{I}_j^1} f(s_i - \hat{\theta}_i \hat{\tau}_i) \prod_{i \in \mathcal{I}_j^2} \bar{F}(u_i - \hat{\theta}_i \hat{\tau}_i) \prod_{i \in \mathcal{K}_j} F(l_i - \hat{\theta}_i \hat{\tau}_i).$$

Then, we let $\mathscr{L} = \prod_{j=1}^{J} \mathscr{L}_j$ and $\operatorname{argmax} \mathscr{L}$ helps us choose the parameters of the distribution F. We set F to be the Normal distribution.

3.5 Judge minimum and maximum plea - $l_j(\theta\tau), u_j(\theta\tau)$

Mathematical Description. Recall that a key quantity for us is $\theta\tau$. Fix a judge, say judge j, and focus only on the pleas he sentenced. For each such case, we can calculate $\hat{\theta}\hat{\tau}$ using the estimates from the hurdle model described in Section (tbd). Suppose judge j handled I_j such cases, indexed by $i = 1, \ldots, I_j$. Let $\mathcal{A}_j$ denote the convex hull of the origin $(0, 0)$, the point $(\max_i \hat{\theta}_i\hat{\tau}_i, \max_i s_i))$ and the points $(\hat{\theta}_i\hat{\tau}_i, s_i)$ for $i = 1, \ldots, I_j$. Given the set $\mathcal{A}_j$, we impute $l_j(\cdot)$ and $u_j(\cdot)$ as functions of $\theta\tau$ as follows: For $\theta \in [0, \max_{i=1,\ldots,I_j} \hat{\theta}_i\hat{\tau}_i]$

$$
\begin{aligned}
l_j(\theta\tau) &= \min\{y : (\theta\tau, y) \in \mathcal{A}_j\}, \\
u_j(\theta\tau) &= \max\{y : (\theta\tau, y) \in \mathcal{A}_j\}.
\end{aligned}
$$

3.5.1 Sample

Here, to estimate $l_j(\theta\tau), u_j(\theta\tau)$ for a specific judge, j, we use all of pleas judge j heard. In other words, we exclude all of j's trials. We also add the point $(0, 0)$ to the convex hull of every judge.

3.6 County Arrival Rates - λ_c

We set each county's arrival rate to the average number of sentencing events per week in that county, as observed in the data. Let N_{cw} denote the number of sentencing events in county c in week w, county c's arrival rate is defined as: $\lambda_c = \frac{\sum_w N_{cw}}{\sum_w 1}$. We are currently using all pleas and all weeks in our data to calculate this.

3.7 Service Rates - μ_p, μ_t

3.7.1 Model

Estimating the plea and trial service rates turned out to be one of the most challenging aspects of this project. The difficulty in this problem lies in the fact that judges work on a variety of different tasks (e.g. administrative work, hearing civil cases, hearing criminal cases). This prevents us from taking a simple average of number of pleas/trials per day, because doing so would introduce the implicit assumption that judges only work on pleas/trials. Doing so would also ignore any idle time judges may experience. In an ideal world, we would be able to lock judges in a room, have them work exclusively on pleas and trials without incurring any idleness, and then simply measure their service rates. This is impossible, but it illustrates what we were aiming for. To get as close as possible to the ideal scenario, we restricted our attention to days in our data in which judges plausibly spent most of their time working on either pleas or trials. We use the master calendar data to determine which were these days. More specifically, we restrict our attention to days that are of type "GS" in the master calendar data. The "GS" acronym stands for "General Session", which are dedicated exclusively to criminal cases (both pleas and trials). As a result, it is likely that on these "GS" days, judges mostly spent their time working on either pleas or trials, or they were idle.

We use the sentencing data to construct an auxiliary dataset where the unit of observation is a judge-county combination. For each judge-county combination, v, with judge j and county c, we count the number of days of type "GS" that judge j was assigned to be in county c in the master calendar. We also count the number of pleas and trials judge j processed on "GS" days in county c. So, we obtain a dataset where for each judge-county combination we have the total amount of work completed (that is the number of pleas and trials), and the total time it took to complete.

In our exploratory analysis of the sentencing data, we noticed that there exists considerable heterogeneity across counties in the number of pleas and trials sentenced in each county. For example, we observe 1,633 pleas in Richmond county in the data, but only 62 in Fairfield county. This suggests that some of the variation in judge idleness can be explained by the county the judge is in. For example, a judge assigned to Fairfield is more likely to be idle because there is less work to be done there than in Richmond. To account for these differences in demand for pleas in trials across counties, we decided to include fixed effects for each county. The model we fit is as follows:

$$\text{Days}_v = \alpha_c + \beta_T \text{Trials}_v + \beta_P \text{Pleas}_v + \epsilon_v$$

The estimates from this model can be seen in Table 8. In this model, β_T, β_P are the trial and plea service rates, and α_c can be interpreted as the average idle time in each county. This could include time to get set up or administrative tasks. The model estimates that judges can process about 10 pleas per day, and that it takes about 4 days for judges to hear each trial[7]. The estimated α_c's can be found in Appendix F.12.

Table 8: County Fixed Effects Model

	Dependent variable:
	Days
Plea	0.091^{***}
	(0.008)
Trial	4.336^{***}
	(0.339)
Observations	278
R^2	0.735
Adjusted R^2	0.681
Residual Std. Error	7.552 (df = 230)
Note:	$^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

4 Simulation Plan

In this section, we provide a detailed description of the simulation we run. We start by describing the state variables we use and how we represent them. Next, we describe the dyanmics of the simulation, including how we assign judges to counties and what happens in each iteration.

4.1 State Variables

The state variables for our simulation are judges and counties. In what follows, suppose that our simulation is of T time periods. We index judges by j and counties by c.

Judges. We represent each judge, j, as a tuple $(x_j, \mathcal{A}_j)$ where x_j is an array of length T whose i^{th} element is judge j's remaining capacity for the i^{th} time period. $\mathcal{A}_j$ is the convex hull of judge

[7]We also estimated a judge fixed effects model, and it yielded similar estimates. The estimates can be found in Appendix F.1. Although it gave similar estimates, we didn't opt for a model with both county and judge fixed effects because it would have entailed estimating about 100 parameters with roughly 200 data points.

j's past sentencing events, $(\hat{\theta}_i \hat{\tau}_i, s_i)$, as described in Section 3.5. We set each judge's capacity in the following way. Let s_j be judge j's share of GS days. In other words, $s_j = N_j^{GS}/N$ where N_j^{GS} is the number of days of type 'GS' the judge was scheduled to work in the data, and N is the total number of working days in the year. Let $\mu_P = 1/\beta_P$ be the number of pleas per day a judge can hear on GS days. Here, β_P is the parameter we estimated in Section 3.7. Judge j's capacity for each week would then be $C_j = \mu_P \cdot s_j \cdot 5$.

Counties. We represent each county c as a tuple $(s, \lambda_c, \mathcal{D}_c, \mathcal{B}_c, \mathcal{R}_c)$. Here, s is an array of length T whose i^{th} element is the judge that is scheduled to work in county c on the i^{th} time period. λ_c is county c's average arrival rate, which we calculate as described in Section 3.6. $\mathcal{B}_c$ is a list of county c's backlogged defendants. $\mathcal{D}_c$ is a set containing all of the sentencing events observed in county c in our sentencing data. $\mathcal{R}_c$ is a set containing the sentencing events that occur in county c in our simulation. That is, $\mathcal{R}_c$ contains the sentencing events that occur in county c in our simulation.

4.2 Dynamics

4.2.1 Judge Schedules

Trials We would schedule trials in the following way. First, for each county, we would set the trial dates to be as observed in the sentencing data. Then, in each time period, we would randomly assign judges to counties as described below, if a judge is assigned to county on a day in which a trial is scheduled in that county, then the judge would stay there for 1 week (since our estimates indicate processing trials takes a little over 4 days). As a result, when assigning judges and counties for the next period, the county and the judge would be removed from the list of available counties/judges. In other words, the assignment for that judge and county would already be determined for the next week as well. We could then assign the remaining judges/counties using the method described below.

Assigning Judges to Counties Here, we describe a method to assign a set of judges to a set of counties for T time periods. The time periods here are discrete and we think of them as weeks. We describe how the assignment would work for each individual week, but in practice all the assignments would be determined before running the rest of the simulation. First, for each judge, we first calculate the share of all weeks in which they had an assignment, call this α_j. Since there are 50 judges and 46 counties, in each period we would randomly draw, without replacement, 46 judges from the list of 50. For each judge, the probability of being chosen in this step will be proportional to α_j. Then, each of the selected judges will stay in his "home" county with probability η and with probability $1 - \eta$ he will be assigned to another county. We refer to judges that don't stay in their home county in a specific week as "rotating judges", and we refer to counties whose home judge will be rotating as "rotating counties". We assign rotating judges to rotating counties as follows: we randomly shuffle the rotating judges and the rotating counties are sorted alphabetically. So the rotating judge in the first position after shuffling would be assigned to county A, the second to County B, and so on.

4.2.2 Iteration/Simulation Step/Action

Defendant Arrivals. In each time period, t, we iterate over the different counties. We simulate defendant arrivals for a given county, c, as follows: first, we determine the number of arrivals, n_{ct} by drawing from a Poisson distribution with mean λ_c. We then draw n_{ct} defendants from county c's past defendants, $\mathcal{D}_c$ as observed in the sentencing data. If the county has any backlogged defendants in that week, that is if $\mathcal{B}_c \neq \emptyset$, the backlogged defendants are added to that week's list of defendants, and are served first.

Defendant Choice. Each defendant then chooses from the available judges that will be in county c in the next r weeks. The defendant chooses the judge, j that minimizes his expected cost, $\min(\theta\tau + c_d, u_j(\theta\tau)) + k(j)d$, where c_d is the defendant's cost of going to trial, $k(j)$ is the number of time periods until judge j will be in county c, and d is the cost of delay. Once a defendant chooses a judge, we reduce that judge's capacity for the week in which he will sentence that defendant, and we add that sentencing event to the county's history of sentencing events. More concretely, if the judge is supposed to sentence the defendant on week k, we set $x_j[k] = x_j[k] - 1$ and we add the sentencing event to $\mathcal{R}_c$. If there are no judges available in the next r weeks, the defendant is added to the county's list of backlogged defendants, $\mathcal{B}_c$ and processed again next week. If a defendant chooses to go to trial, that defendant disappears from the simulation, and all judges' capacity remains unchanged.

4.2.3 Calibration

Once the details of the simulation are decided, we can simulate the above model to calibrate the values of d (the cost of delaying by a week) and r (the maximum delay a defendant can choose) so as to match some key performance metrics in the data. We can then use it for the counterfactual analysis. To be specific, for each possible d, we will try different r values and see where the relevant metrics flatten. We choose the smallest such r, denoted by $r(d)$. Then we vary d to match the data.

5 Next Steps

- Implement hurdle model for $\tau\theta$.
- Implement MLE estimation for c_d
- Implement changes to simulation

References

Rhys Hester and Todd K Hartman. Conditional race disparities in criminal sentencing: A test of the liberation hypothesis from a non-guidelines state. *Journal of quantitative criminology*, 33(1): 77–100, 2017.

Can Wang. *Three Data-driven Model-based Policy Analyses in Criminology.* PhD thesis, Stanford University, 2019.

A Sentencing Data Set

This section provides a more in-depth description of the sentencing data we used. All of the sentencing events in the data correspond to felonies or serious misdemeanors.

A.1 Variables

There are two versions of the sentencing dataset: the CSV file and the STATA file. There are 17,671 sentencing events (rows) in both files. The CSV file has 33 data fields. This is the first dataset Larry obtained. The STATA file has 21 data fields. This is the second dataset Larry obtained.

Table 9: Data fields of the CSV and STATA files.

Only CSV	Both	Only STATA
unnamed	datedisp/data File	expmin
statute_first (identical to statute)	circuit	jud_no
offdescr_first	county	
offtypeLibHyp	counts	
ccpnts	offser	
ccpts99	sgc_offcode	
trial	of_hom	
incarc	of_rape	
crimhist	of_rob	
ppoints	of_asslt	
male	of_burg	
age	of_dstrb	
black	of_possn	
judge	of_theft	
	of_fraud	
	of_other	
	realsent/sentence	
	statute (99.95% match)	
	offdescr (99.21% match)	

Table 9 lists the data fields that are common to the CSV file and the STATA file as well as the data fields that are unique to either the CSV file or the STATA file. As shown in Table 9, the CSV file and the STATA file have 18 data fields in common; see the middle column of Table 9. Of these 18 data fields that are in common across the two files, 16 of them are identical.[8] The remaining two common data fields are statute and offdescr. We observe that 99.95% of the statute entries and 99.21% of the offdescr entries are identical between the CSV and STATA files. Moreover, the 10 non-identical statute entries and the 141 non-identical offdescr entries seem to refer to the same statutes and offense descriptions, respectively.

There are 14 data fields that appear only in CSV file; see the first column of Table 9. Similarly, there are 2 data fields that only appear in the STATA file: expmin and jud_no. The expmin data

[8] To be specific, after sorting the rows of the two files based on the 16 commons data fields, the entries of these 16 data fields match exactly.

field is computed by Rhys; see item 1 in the STATA file description below. The jud_no data field of the STATA file provides a numeric identifier for the judge that sentenced each offender. Although these numeric identifiers are not identical to the numeric identifiers used in the judge data field of the CSV file, there is a one-to-one mapping between them. Consequently, we use the CSV file in our analysis (in the following sections) but supplement it with the expected minimum sentence data field from the STATA file, which is calculated using Equation (1); see Section A.1.2. Next, we describe the data fields in the CSV and STATA files in detail.

A.1.1 CSV File

The data fields of the CSV file are as follows:

Table 10: Variable descriptions for CSV file

Variable	Description
	An unnamed numeric identifier for the sentencing events.
date	Date of the sentence, ranging from 2000-07-07 to 2001-06-29.
county	County where the sentence was decided. There are 46 counties.
circuit	The circuit the county belongs to. There are 16 circuits (1-16).
judge	A numeric identifier for the judge who heard the case. There are 51 judges.
trial	Binary variable indicating whether the case went to trial.
incarc	Binary variable indicating whether the sentence included incarceration.
statute	The law the offender is accused of breaking, for example 56-05-0750(B)(1).
offdescr	A description of the offense (e.g. "Failure to Stop for a Blue Light")
counts	There are 29 distinct values, ranging between 0 and 60.
statute_first	The description of the first offense of the offender. This field is identical to 'statute'.
offdescr_first	A description of the first offense (e.g. "Pointing and Presenting Firearms at a Person")
sgc_offcode	286 distinct numeric values between 4 and 2878.
of_hom	Binary variable indicating the offense type is homicide.
of_rape	Binary variable indicating the offense type is rape
of_rob	Binary variable indicating the offense type is robbery
of_asslt	Binary variable indicating the offense type is assault
of_burg	Binary variable indicating the offense type is burglary
of_dstrb	Binary variable indicating the offense type is drug distribution
of_possn	Binary variable indicating the offense type is drug possesion
of_theft	Binary variable indicating the offense type is theft
of_fraud	Binary variable indicating the offense type is fraud
of_other	Binary variable indicating the offense type belongs to the category "other".
offtypeLibHyp	Categorical offense type. The categories: property, violent, other, and drug.
offser	Offense seriousness. There are 8 categories: 1 to 8. Categories are increasing in seriousness.
ccpnts	Numeric values between 1 and 68.
ccpts99	Commitment score. Numeric values between 1 and 12 in increasing order of seriousness.
ppoints	Numeric values between 0 and 1014.
male	Binary variable indicating whether defendant is male.
age	Age of defendant at sentencing.
black	Binary variable indicating whether defendant is black.
sentence	The length of the sentence in months. Ranges between 0 and 11988.

A.1.2 Stata File

Next, we describe the data fields of the STATA file. We only describe the two data fields that are specific to the STATA file. The remaining data fields have already been discussed above.

Table 11: Variable descriptions for STATA file

Variable	Description
expmin	The expected minimum sentence. Ranges between 0 and 470.
jud_no	A numeric identifier for the judges. There are 50 judges, numbered 1-54.[9]

A.1.3 Expected Minimum Sentence

The expected minimum sentence (reported in the STATA file) is calculated as follows:[10] The set of sgc_offcode values is partitioned into four subsets: $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{C}_3$, and $\mathcal{C}_4$; see Hester and Hartman (2017) for the partitions. Let s denote the sentence length in months, c denote the sgc_offcode, and

$$m(c) = \begin{cases} 1.00 & \text{if } c \in \mathcal{C}_1, \\ 0.85 & \text{if } c \in \mathcal{C}_2, \\ 0.33 & \text{if } c \in \mathcal{C}_3, \\ 0.25 & \text{if } c \in \mathcal{C}_4, \end{cases}$$

denote the expected minimum sentence multiplier, i.e., the fraction of the sentence that has to pass before the defendant is eligible for parole. Then,

$$\text{Expected Minimum Sentence}\,(s, c) = \min\left\{\left\lceil m(c) \cdot s \right\rceil, 720\right\}. \tag{1}$$

The exceptions to this are as follows. Define the sets

$$\mathcal{A}_1 = \{267, 383, 2362\},$$

$$\mathcal{A}_2 = \{454, 2360, 2362\},$$

$$\mathcal{A}_3 = \{79, 90, 113, 114, 139, 217, 280, 283, 312, 368, 387, 388, 389, 392, 395, 402, 457, 2356, 2359\},$$

$$\mathcal{A}_4 = \{116, 148, 281, 284, 349, 370, 452, 456, 2417\},$$

where $\mathcal{A}_i \subset \mathcal{C}_i$ for $i = 1, 2, 3, 4$. According to Hester and Hartman (2017), a mandatory minimum

[10] See the code provided on https://tkhartman.netlify.app (accessed on October 17th, 2020) for Hester and Hartman (2017).

sentence is imposed if $c \in \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4$.

A.2 Missing Data

A.2.1 Variables with missing data

After excluding all of the sentencing events heard by Judge 1, the fictitious judge, there are 17,516 sentencing events. Amongst these sentencing events, the only data field with missing values are the *date* data field and the *expmin* data field. 1482 sentencing events are missing the *date* data field and 28 sentencing events are missing the *expmin* data field.

A.2.2 Imputation of Missing Dates

After removing the sentencing events heard by Judge 1, there are 1,482 sentencing events with the *date* field missing. We impute the dates for these events in the following way. Recall that the judge name and county are still available for sentencing events with missing dates. As a result, we can use the calendar data to see what dates the judge was assigned to be in that county. For example, say that Judge 1 has a sentencing event with a missing date in Aiken county. If according to the calendar data Judge 1 was only assigned to Aiken county on January 25, 2001, then it is highly likely that the sentencing event happened on that date. Similarly, if Judge 1 was only assigned to Aiken county on two days, then the event must have happened in one of those two days. In general, when imputing the dates for sentencing events with missing dates, we first use the master calendar to determine a set of possible dates in which the sentencing event could have occurred. The set of possible dates in which the event could have occurred are those days in which the judge in question was assigned to the county in question (or to the same circuit). Figure 8 provides the histogram of the number of possible weeks during which a sentencing event (with a missing date) might have occured. Once we have the set of possible dates, we assign the events with missing dates as evenly as possible across these possible dates. Let n be the number of events with missing dates, and let m be the number of possible days in which these events could have happened. First, we sort the possible days by date, earliest to latest. Suppose $n/m = l$ with remainder k. In this case, each of the m possible days would be assigned l events with missing dates, and the k remaining events with missing dates would be assigned to the first k possible days. So, for example, if there are 5 sentencing events with missing dates and 4 potential dates, then each of the potential dates would get assigned a sentencing event, and the earliest of the potential dates would get the remaining one.

To elaborate further, suppose that judge j has m sentencing events with missing dates in county c (which is in circuit k). The set of potential dates we would assign these sentencing events to would evolve in the following way. In each category, the days are ordered from earliest to latest.

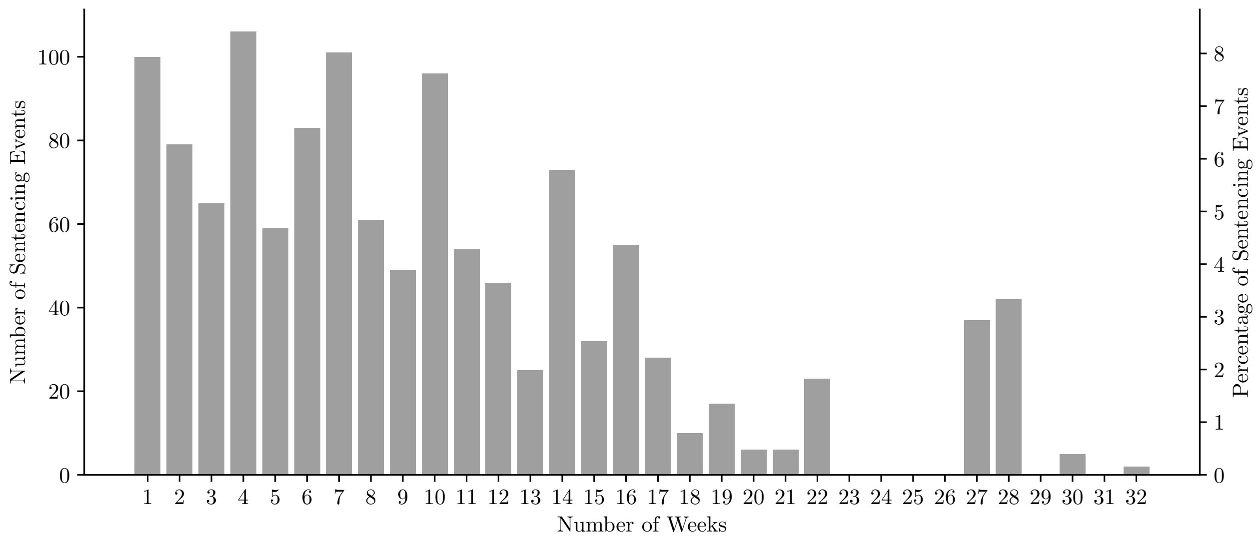


Figure 8: A histogram of the number of weeks the judge visited the county in which the sentencing events missing the *date* data field occurred.

1. **Matching county, GS:** Days in the master calendar in which judge j had a "GS" assignment to county c.

2. **Matching county, non-GS:** Days in the master calendar in which judge j had a non-"GS" assignment to county c.

3. **Matching circuit, GS:** Days in the master calendar in which judge j had a "GS" assignment to a county in circuit k.

4. **Matching circuit, non-GS:** Days in the master calendar in which judge j had a non-"GS" assignment to a county in circuit k.

5. **Sentencing data match:** Days in which we observe sentencing events in the sentencing data

for judge j in county c.

6. **Any day, GS:** Days in the master calendar in which judge j had a "GS" assignment to any county.

So, first, we would try to assign the sentencing events to days in the first set, if that set is empty, we would move on to the next set until we found a non-empty set. Using this method, we are able to impute the missing dates for all sentencing events with missing dates. Table 12 contains the distribution of the imputation method used for pleas with missing dates.

Table 12: Distribution of missing events

Imputation Group	**Share of Pleas**
1. Matching county, GS	0.825
2. Matching county, non-GS	0.024
3. Matching circuit, GS	0.02
4. Matching circuit, non-GS	0.006
4. Sentencing data match	0.026
5. Any day, GS	0.098

B Master Calendar Data

B.1 Data Cleaning

We do two things to clean the calendar data:

1. We discard all judge names in the master calendar that are not mapped to a judge number in the sentencing dataset. We discard these judge names because we do not observe any sentencing events for them. The discarded judge names are as follows: Bergdorf, Couch, Drew, Gier, Peeples, Simmons, Watts, and Young. Since we are using the same sentencing data as Hester and Hartman (2017), we can conclude that they also excluded these judges.

2. We combine the schedules of Judges "Cooper" and "Cooper, TW" in the calendar data. There are three judges in the calendar data with last name Cooper: (i) Cooper, (ii) Cooper, TW, and (iii) Cooper, GT. If we run the algorithm proposed in Section D using the master calendar with three distinct judges with last name Cooper, Judge 14 is mapped to either Judge

Cooper (under the perfect match criterion) or Judge Cooper, TW (under the overlapping match criterion). Table 13 depicts the schedules of the three judges with last name Cooper (according to the master calendar) as well as the counties visited by Judge 14 (according to the sentencing dataset). The counties to which either Judge Cooper or Judge Cooper, TW are assigned are depicted in blue. The counties to which Judge Cooper, GT is assigned are depicted in green. The counties to which neither Judge Cooper, nor Judge Cooper, TW or Cooper, GT are assigned are depicted in red. Table 13 shows that the schedules of Judges Cooper and Cooper, TW do not overlap. Moreover, their combined schedules resemble the schedule of Judge 14 (the list of counties visited by Judge 14) the most. This suggests that Judge Cooper and Judge Cooper, TW are (most likely) the same person. Therefore, we combine the schedules of Judges "Cooper" and "Cooper, TW".[11]

[11] Incidentally, only two judges with last name Cooper are listed on the South Carolina Judicial branch's website. These judges are Thomas W. Cooper, Jr. and G. Thomas Cooper, Jr.; please see https://www.sccourts.org/circuitCourt/displaycirjudge.cfm?judgeid=2054 and https://www.sccourts.org/circuitCourt/displaycirjudge.cfm?judgeid=2126 (accessed on December 14th, 2020) for their bios.

Week	Judge 14	Cooper	Cooper, TW	Cooper, GT
July 10		Aiken		
July 17				Berkeley
July 24				Richland
July 31		Sumter		Greenville
August 7	Sumter	Sumter		Greenwood
August 21	Greenville	Sumter		Greenville
August 28	Williamsburg, Sumter	Williamsburg, Orangeburg		Richland, Greenville
September 4	Williamsburg, Greenville	Williamsburg		Greenville
September 11	Sumter, Clarendon, Greenville	Sumter		Greenville
September 18				Spartanburg
September 25	Greenville	Williamsburg		Greenville
October 9	Lee, Sumter	Lee		Richland
October 16	Williamsburg	Williamsburg, Aiken		Richland
October 23	Williamsburg, Horry, Georgetown	Williamsburg		
October 30	Sumter	Sumter		Richland
November 6		Williamsburg		Richland
November 13	Sumter	Sumter		
November 27		Clarendon		Newberry
December 4	Lee, Sumter	Lee		
December 11	Lee	Lee, Aiken		
January 1			Aiken	
January 8	Richland		Richland, Aiken	McCormick
January 15			Kershaw	
January 22			Aiken	
January 29	Aiken		Beaufort, Aiken	Lexington
February 5			Orangeburg	Laurens
February 12				Newberry
February 26	Richland		Richland, Orangeburg	Greenville
March 5	Richland, Edgefield			McCormick, Edgefield
March 12			Orangeburg	
March 19	Orangeburg		Orangeburg	
March 26	Richland		Richland	Edgefield
April 9	Richland		Richland	Lexington
April 16	Richland			
April 23			Richland	Lexington
April 30	Kershaw		Kershaw	Lexington
May 14				Lexington
May 21	Richland		Richland	Edgefield
May 28	Richland		Richland	
June 4	Richland		Richland	Edgefield
June 11				Edgefield
June 18	Kershaw		Kershaw	Lexington
June 25			Richland	McCormick

Table 13: The list of counties visited by Judge 14 (according to the sentencing dataset) and the counties visited by Judges Cooper, Cooper, TW, and Cooper, GT (according to the master calendar). The counties visited by Judge 14 to which either Judge Cooper or Judge Cooper, TW were assigned are written in green font. The counties visited by Judge 14 to which Judge Cooper, GT was assigned are written in blue font. The counties visited by Judge 14 to which neither Judge Cooper, nor Judge Cooper, TW or Cooper, GT were assigned are written in red font. For visual clarity, only the weeks in which at least one judge has an assignment or sentencing event are depicted.

B.2 Descriptive Statistics

For each judge, the percentage of weeks with a GS (CP) assignment on the master calendar is depicted in Figure 9 (10). Figures 9-10 jointly show that GS and CP assignments are the most common assignments for the judges. Figure 11 shows the overlap, i.e., the percentage of weeks (with an assignment) in which the judge has at least one assignment of type GS and at least one assignment of type CP. It shows that it is not common for a judge to have both GS and CP assignments in a week.

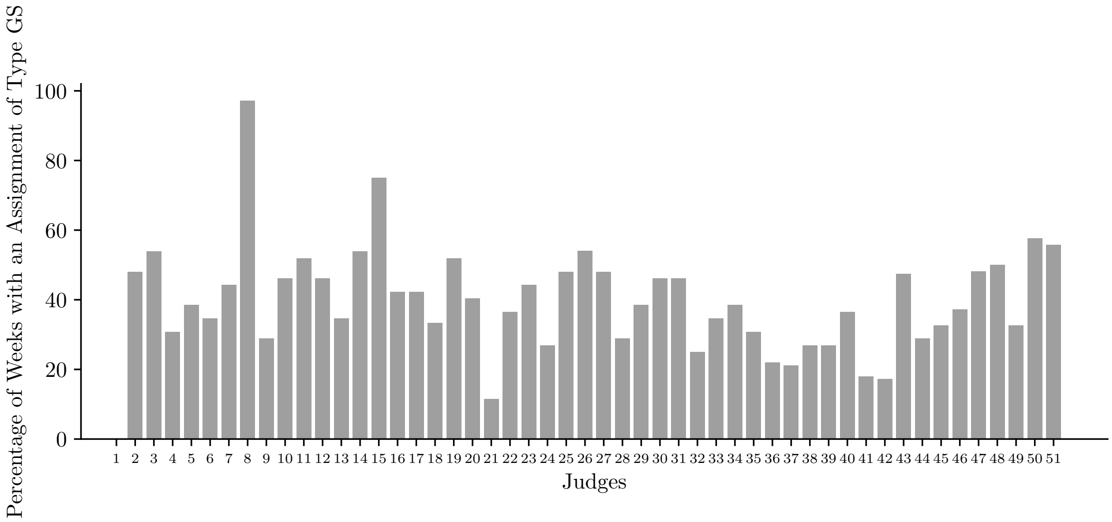


Figure 9: The percentage of weeks (with an assignment) in which the judge has at least one assignment of type GS.

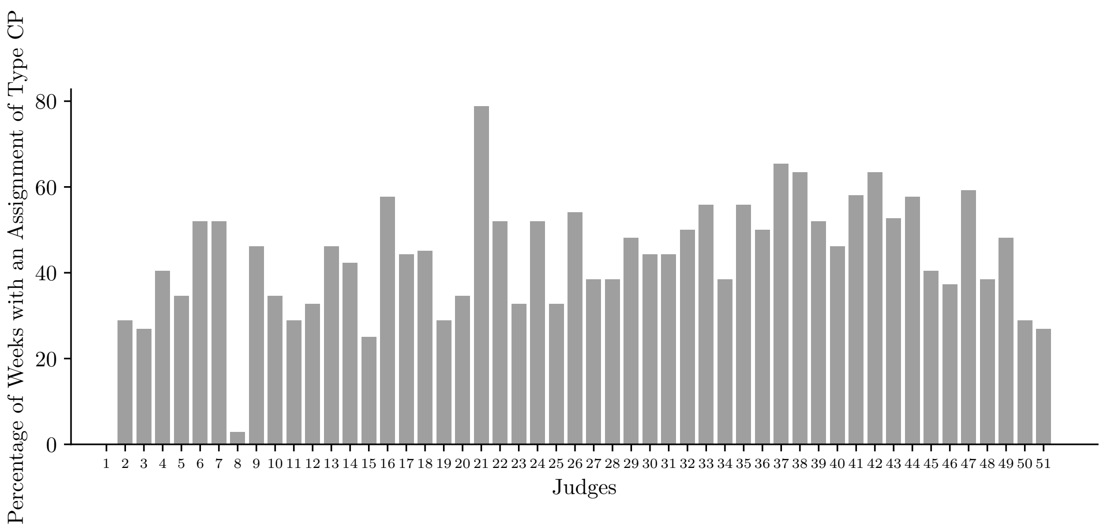


Figure 10: The percentage of weeks (with an assignment) in which the judge has at least one assignment of type CP.

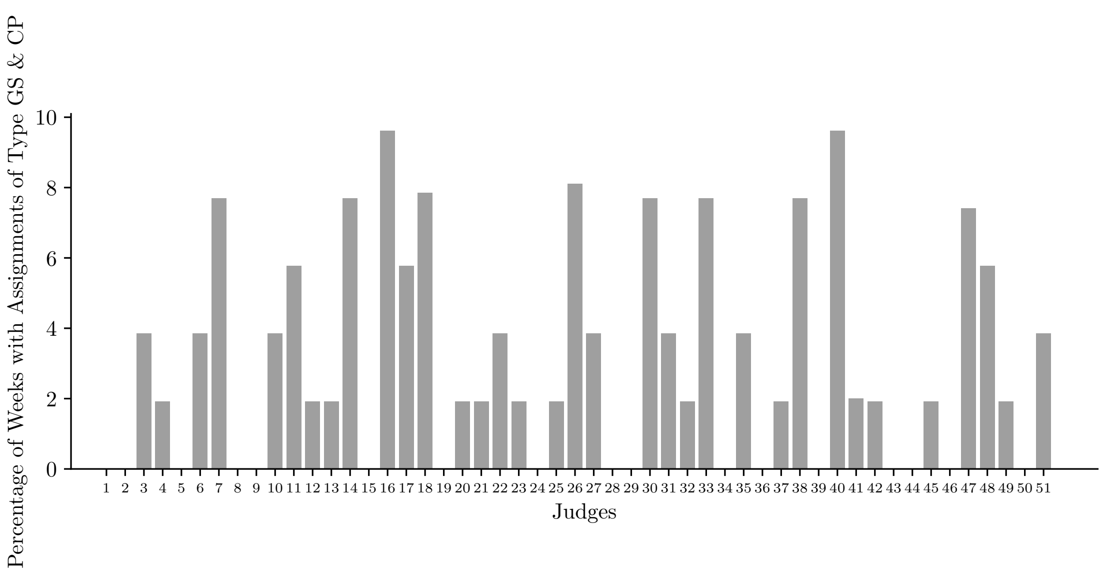


Figure 11: The percentage of weeks (with an assignment) in which the judge has at least one assignment of type GS and at least one assignment of type CP.

B.3 Creation of Figure 6

Figure 6 depicts the frequency of the assignment types based on the sentencing dataset. Figure 6 is generated as follows: First, we create a list of the sentencing events undertaken by Judges $2, \ldots, 51$ that have a date. Then, for each sentencing event on the list, we proceed as follows:

1. We read the judge number, the week, and the county corresponding to each sentencing event (from the sentencing dataset). For example, Judge 2 on the Week of January 8 in Chester.[12]
2. We look for the judge's assignment for this week on the master calendar and report the type of the assignment in the figure. We sometimes observe multiple assignments in a week. For example, Judge 2 had the following assignments in the week of January 8: 2nd Cir. CPNJ 8, Chester GS 9,10,11, and Lancaster 12. In this case, the assignment in Chester county is the one we are interested in, because that is where the sentencing event of interest occurred. Thus, we use this assignment (and its type, i.e., GS) to record the sentencing event in the figure. For 4.12% of the sentencing events (without a date), we cannot find the county assignment we are interested in due to inconsistencies between the master calendar and the sentencing dataset; see Section C for a detailed description of the judge-week combinations in which these sentencing events occurred.

We repeat this procedure for all sentencing events and count the number of assignment types. Finally, we plot the histogram in Figure 6.

C Discrepancies between the Sentencing Data and the Master Calendar Data

This section studies the discrepancies between the sentencing data and the calendar data. In doing so, we take advantage of the mapping from judge numbers to judge names (discussed in Section 1.4). We conducted this analysis before imputing the missing dates. As a result, this

[12]The subsequent analysis in this section shows that there are some inconsistencies between the master calendar and the sentencing dataset. If we use the sentence date (to generate Figure 6), we will run into a problem for some sentencing events, e.g., the sentencing events in Table 14. Therefore, we make a weaker assumption and use the sentence week (as opposed to sentence date).

analysis excludes those events. We excluded these events because any conflicts between the two datasets resulting from the imputation would have been artificial.

C.1 No Assignments Case

There are 110 judge-week combinations in this category. We observe no sentencing events (in the sentencing dataset) for 101 out of these 110 (91.8%) judge-week combinations. Recall that the judges in these combinations have no assignments on the master calendar. Therefore, there are no inconsistencies between the master calendar and the sentencing dataset for these 101 judge-week combinations. The remaining 9 judge-week combinations are listed in Table 14. We observe at least one sentencing event (in the sentencing dataset) for these judge-week combinations. Each judge visits only one county (according to the sentencing dataset) in these 9 judge-week combinations.

Table 14: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - "No Assignment" category. The first number in the parenthesis depicts the number of pleas and the second number depicts the number of trials.

			Sentencing Data Set				
	Judge	Week	Monday	Tuesday	Wednesday	Thursday	Friday
1	7	2000-07-03					Dorchester (0,1)
2	7	2000-07-24	Orangeburg (1,0)	Orangeburg (1,0)		Orangeburg (1,0)	
3	7	2001-01-29	Dorchester (1,0)				
4	7	2001-06-04	Orangeburg (5,0)		Orangeburg (2,1)	Orangeburg (4,0)	
5	25	2000-10-30			Greenwood (1,0)		
6	25	2001-04-23	Greenwood (1,0)				
7	42	2001-03-12			Greenville (1,0)		
8	42	2001-04-16		Greenville (1,0)			
9	46	2000-10-23	Charleston (1,0)				

C.2 Single Assignment Case

C.2.1 Single assignment, no dates

We start with the single assignment, no date category. There are 2,214 judge-week combinations in this category. Although in the single assignment, no date judge-week combinations (according to the master calendar) the judges have an assignment the entire week, they do not necessarily sentence every day of the week. For 1871 of the 2200 (85%) judge-week combinations, we only observe sentencing events in the county to which the judge is assigned, i.e., we do not observe sentencing events in a county other than the county to which the judge is assigned. In other words, there are no inconsistencies between the master calendar and the sentencing dataset for 1871 of these 2200 judge-week combinations. The remaining 328 judge-week combinations are illustrated in Table 15. We observe at least one sentencing event (in the sentencing dataset) in a county to which the judge is not assigned in these judge-week combinations.

Table 15: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - single assignment, no date category. The counties written in green font are the counties to which the judge is assigned. The counties written in red font are the counties to which the judge is not assigned. The counties written in blue font are the counties to which the judge is not assigned, however, he is assigned to the circuit court containing these counties. So, the county assignment in the master calendar and this county belong to the same circuit court.

				Sentencing Data Set				
	Judge	Week	Master Calendar	Monday	Tuesday	Wednesday	Thursday	Friday
1	1	2000-08-07	York GS		York (2,0)	York (2,0)	York (4,0)	Kershaw (1,0), York (7,0)
2	1	2000-08-21	York GS		York (1,0)			Union (1,0), York (2,0)
3	1	2000-09-11	York GS	York (6,0)	York (2,0)		York (2,0)	Spartanburg (1,0), York (5,0)
4	1	2000-10-09	York GS			Fairfield (1,0), York (1,0)	York (5,0)	York (4,0)
5	1	2000-10-23	York GS	York (2,0)	York (3,0)	York (9,0)	York (1,0)	Kershaw (1,0), York (6,0)
6	1	2001-01-08	Chester GS	Chester (2,0)	York (1,0)			Chester (2,0)
7	1	2001-02-12	Chester CP				Lancaster (1,0)	
8	1	2001-06-04	Chester GS			Chester (0,1)	Chester (2,0), Spartanburg (1,0)	Chester (3,0)
9	2	2000-07-24	Richland GS	Richland (4,0)	Lexington (1,0), Richland (4,0)	Richland (2,0)	Richland (7,0)	
10	2	2000-09-11	Richland GS		Kershaw (1,0), Richland (6,0)	Richland (7,0)	Richland (3,0)	Richland (1,0)
.	.	.	.	.	.	.	.	.
320	49	2001-06-25	Richland GS	Lexington (1,0), Richland (1,0)				
321	50	2000-09-25	Richland GS	Richland (7,0)	Clarendon (1,0), Lexington (1,0)	Richland (2,1)	Richland (3,0)	Richland (1,0)
322	50	2000-10-02	in chambers					Richland (1,0)
323	50	2000-10-16	Orangeburg GS	Orangeburg (14,0)	Orangeburg (2,0)	Orangeburg (5,0)	Charleston (1,0), Orangeburg (3,0)	Orangeburg (1,0)
324	50	2000-12-18	in chambers	Richland (1,0)	Richland (24,0)			
325	50	2001-01-08	Orangeburg GS	Calhoun (2,0), Orangeburg (14,0)	Orangeburg (3,0)	Orangeburg (4,0)	Orangeburg (3,0)	Aiken (1,0), Orangeburg (2,0)
326	50	2001-01-22	Sumter GS	Sumter (6,0), York (1,0)	Sumter (2,0)	Sumter (3,0)	Sumter (1,0)	Sumter (3,0)
327	50	2001-02-05	Clarendon GS	Clarendon (3,0)	Clarendon (2,0)	Clarendon (1,0)		Clarendon (3,0), Sumter (1,0)
328	50	2001-04-16	Sumter GS	Clarendon (0,1), Sumter (5,0)	Sumter (3,0)	Sumter (2,0)	Sumter (3,0)	Sumter (3,0)
329	50	2001-05-28	3rd Cir. CPNJ/PCR				Sumter (1,0)	Williamsburg (1,0)

C.2.2 Single assignment with dates

There are 14 judge-week combinations in the "single assignment with specific dates category". In 10 of the 14 judge-week combinations, we either observe no sentencing events or sentencing events only in the same county as listed in the master calendar (2 of them). In other words, there are no inconsistencies between the master calendar and the sentencing dataset for 10 of these 14 (71.4%) judge-week combinations. The remaining 4 judge-week combinations are listed in Table 16. In these 4 judge-week combinations, exactly one of the following conditions holds:

1. At least one sentencing event occurred in a county not stated on the master calendar (3 judge-week combinations); see the last three rows of Table 16.

2. At least one sentencing event occurred in the county stated on the master calendar but on a date not stated on the master calendar (1 judge-week combination); see the first row of Table 16.

Table 16: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - single assignment, with dates category. The counties written in green font are the counties to which the judge is assigned. The counties written in red font are the counties to which the judge is not assigned.

				Sentencing Data Set				
	Judge	Week	Master Calendar	Monday	Tuesday	Wednesday	Thursday	Friday
1	6	2001-03-19	Horry GS CP CC 21,22,23		Horry (6,0)		Horry (5,0)	
2	27	2000-09-25	Bamberg GS 25,26,27,28,29	Bamberg (10,0), Lee (1,0)	Bamberg (6,0)			
3	46	2000-10-30	Barnwell GS 1,2,3			Barnwell (1,0)	Allendale (1,0), Barnwell (2,0)	
4	48	2000-10-23	13th Cir. CPNJ 23,24,25,26					Clarendon (1,0)

C.3 Multiple Assignments Case

C.3.1 Multiple assignments, no dates

We start with the category multiple assignment, no dates. There are 19 judge-week combinations in this category. For 17 of the 19 (89.4%) judge-week combinations in this category, we only observe sentencing events in one of the two counties stated on the master calendar. In other words, there are no inconsistencies between the master calendar and the sentencing dataset for 17 of these 19 judge-week combinations. The remaining 2 judge-week combinations are listed in Table 17. The judge-week combinations in this category that satisfy (at least) one of the following conditions are potentially problematic:

1. At least one sentencing event occurred in a county not stated on the master calendar (Both rows of Table 17).
2. The judge sentenced in more than one county on a day (First row of Table 17).

Table 17: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - multiple assignment, no dates cateogry. The county written in green font is the county to which the judge is assigned. The counties written in blue font are the counties to which the judge is not assigned, however, he is assigned to the circuit court containing this county. So, the county assignment in the master calendar and this county belong to the same circuit court.

				Sentencing Data Set				
	Judge	Week	Master Calendar	Monday	Tuesday	Wednesday	Thursday	Friday
1	2	2001-04-30	Aiken GS CC, 2nd Cir. CP/PCR			Aiken (5,0), Bamberg (1,0)		
2	9	2001-01-08	Florence GS, Lee GS			Marion (1,0)	Marion (1,0)	

C.3.2 Multiple assignments, some dates

There are 181 judge-week combinations in the multiple assignment, some dates category. For 155 of these 181 (85.6%) judge-week combinations, we only observe sentencing events in one of the two counties stated on the master calendar and only on the days stated on the master calendar. In other words, there are no inconsistencies between the master calendar and the sentencing dataset for 155 of these 181 judge-week combinations. The remaining 26 judge-week combinations are listed in Table 18. There is only one judge-week combination in the "multiple assignment, some dates ambiguous" category, and we do not observe any conflicts in that week.

Table 18: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - multiple assignment, some dates category. The counties written in green font are the counties to which the judge is assigned. The counties written in red font are the counties to which the judge is not assigned. The counties written in blue font are the counties to which the judge is not assigned, however, he is assigned to the circuit court containing these counties. So, the county assignment in the master calendar and this county belong to the same circuit court.

				Sentencing Data Set				
	Judge	Week	Master Calendar	Monday	Tuesday	Wednesday	Thursday	Friday
1	6	2000-10-16	Florence CP, Florence GS CC 20				Horry (1,0)	Florence (8,0)
2	6	2000-12-18	in chambers, 12th Cir. CPNJ 18,19	Marion (1,0)				
3	9	2001-06-18	Florence GS, Florence CP 21	Florence (5,0)	Florence (3,0)	Darlington (1,0), Florence (17,0)	Florence (12,0)	Florence (5,0)
4	10	2000-10-02	in chambers, 7th Cir. CPNJ 2,3,5				Spartanburg (3,0)	
5	10	2001-06-04	8th Cir. CPNJ, Spartanburg GS CC 8	Laurens (1,0)				
6	12	2001-03-05	McCormick GS, Edgefield GS CC 8	McCormick (3,0)			Saluda (1,0)	
7	13	2000-08-28	Williamsburg GS, Orangeburg GS CC 1	Williamsburg (3,0)	Williamsburg (4,0)	Sumter (1,0), Williamsburg (2,0)	Williamsburg (6,0)	
8	13	2000-12-11	Lee GS, Aiken GS CC 15	Lee (1,0)	Lee (1,0)			Lee (0,2)
9	13	2001-01-08	Richland GS, Aiken GS CC 12	Richland (8,0)	Richland (3,0)	Richland (1,0)		Richland (0,1)
10	17	2001-03-12	Spartanburg CP, Cherokee GS CC 15			Pickens (1,0)	Cherokee (1,0)	
11	25	2001-02-26	Greenwood GS, 8th Cir. CPNJ CC 26	Greenwood (4,0)	Greenwood (10,0)	Greenwood (7,0)	Abbeville (1,0), Laurens (1,0)	Greenville (1,0), Greenwood (5,0)
12	26	2000-11-13	Greenwood GS, Medical 17	Anderson (1,0)		Greenwood (9,1)	Greenwood (5,0)	
13	28	2000-07-31	Richland GS, Lexington GS CC 3	Lexington (1,0), Richland (13,0)	Kershaw (1,0), Marlboro (1,0), Richland (10,0)	Richland (12,0)	Richland (4,0)	
14	28	2001-03-19	Darlington GS, Dillon GS CC 23	Darlington (1,0)	Darlington (8,0)	Chesterfield (1,0), Darlington (9,0)	Darlington (3,0)	
15	29	2001-06-04	16th Cir. CPNJ, Clarendon GS CC 8		Union (1,0)			Clarendon (1,0)
16	30	2000-10-02	in chambers, Greenville GS 6			Anderson (1,0)		
17	32	2001-04-09	Charleston CP, Horry GS CC 9	Charleston (1,0)				
18	32	2001-04-30	Berkeley GS, Richland GS CC 2	Berkeley (6,0)	Berkeley (1,0)	Berkeley (2,0)	Berkeley (6,0)	
19	34	2001-04-16	Florence GS, 5th Cir. CPNJ CC 20	Florence (12,0)	Florence (8,0)	Florence (16,0)	Florence (5,0)	Florence (5,0)
20	34	2001-05-14	Marion GS, 5th Cir. CPNJ CC 14	Marion (7,0)		Marion (1,0)		Marion (8,0)
21	37	2001-01-15	Georgetown CP, Georgetown GS CC 19				Marion (0,2)	
22	39	2000-08-28	Anderson CP, Oconee GS CC 1					Anderson (1,0)
23	39	2000-11-06	Anderson GS, 10th Cir. CPNJ CC 8	Anderson (3,0), Greenville (1,0)		Anderson (8,0)	Anderson (10,0)	
24	40	2001-05-07	in chambers, Greenville GS(SGJ) 7, 13th Cir. CPNJ 8	Spartanburg (1,0)				
25	43	2001-05-07	in chambers, 9th Cir. CPNJ 8		Colleton (0,1)			
26	47	2000-12-04	15th Cir. CPNJ, Horry GS CP CC 5	Horry (1,0)				

The judge-week combinations listed in Table 18 satisfy (at least) one of the following conditions, which are problematic (some satisfy multiple conditions):[13]

1. At least one sentencing event occurred in a county not stated on the master calendar, e.g., judge-week combination 2 of Table 18 (17 such judge-week combinations are listed in Table 18).

2. At least one sentencing event occurred in the county stated on the master calendar with specific dates, but on a date other than the dates specified, e.g., judge-week combination 23 of Table 18 (4 such judge-week combinations are listed in Table 18.).

3. At least one sentencing event occurred in the county stated on the master calendar without specific dates. However, this sentencing event occurred on a date specified for the other county, e.g., judge-week combination 8 of Table 18 (8 such judge-week combinations are listed in Table 18.).

4. The judge sentenced in more than one county on a day, e.g., judge-week combination 3 of Table 18 (5 such judge-week combinations are listed in Table 18.).

C.3.3 Multiple assignments, all dates

There are 57 judge-week combinations in the multiple assignment, all dates category. For 47 of these 57 (82.4%) judge-week combinations, we only observe sentencing events in one of the two counties stated on the master calendar and only on the days stated on the master calendar. In other words, there are no inconsistencies between the master calendar and the sentencing dataset for 47 of these 57 judge-week combinations. The remaining 10 judge-week combinations are listed in Table 19.

[13] For example, judge-week combination 12 satisfies all four conditions. In judge-week combination 12, the assignment of Judge 29 in the week of July 31 is Richland GS and Lexington GS CC 3. Therefore, we would expect Judge 29 to be in Lexington on Thursday, and to be in Richland on Monday, Tuesday, Wednesday, and Friday. Since we observe sentencing events in Marlboro and Kershaw (alongside Richland) on Tuesday, judge-week combination 12 satisfies Conditions 1 and 4. Since we observe sentencing events in both Lexington and Richland on Monday, judge-week combination 12 satisfies Condition 3. Since we observe sentencing events in Richland on Thursday, judge-week combination 12 satisfies Condition 2, as well.

Table 19: Judge-week combinations in which the judge has sentencing events in a county to which he is not assigned - multiple assignment, all dates category. The counties written in green font are the counties to which the judge is assigned. The counties written in red font are the counties to which the judge is not assigned. The counties written in blue font are the counties to which the judge is not assigned, however, he is assigned to the circuit court containing these counties. So, the county assignment in the master calendar and this county belong to the same circuit court.

				Sentencing Data Set				
	Judge	Week	Master Calendar	Monday	Tuesday	Wednesday	Thursday	Friday
1	5	2000-10-16	15th Cir. AW 16,19,20, Aiken 17,18		Aiken (11,0)	Aiken (11,0)	Horry (6,0), Williamsburg (1,0)	Horry (1,0)
2	9	2000-07-10	Darlington GS 11,12,13,14, XX 10		Darlington (2,0)	Darlington (5,0)	Chesterfield (1,0), Darlington (9,0)	
3	16	2000-11-27	XX 1, Aiken GS 27,28,29,30	Aiken (3,0)		Aiken (1,1)	Aiken (7,0), Spartanburg (1,0)	
4	17	2001-04-16	13th Cir. CPNJ 16, Lexington GS 17,18,19,20			Lexington (12,0)	Lexington (11,0), Richland (1,0)	
5	26	2001-05-28	10th Cir. CPNJ/PCR 28,29,30, Lexington GS 31,1			Anderson (1,0)	Lexington (15,0)	Lexington (1,0)
6	30	2001-02-26	Anderson CP 26,27,28,1, Cherokee GS 2, 10th Cir. CPNJ/GS 2					Oconee (6,0)
7	34	2001-01-08	12th Cir. CPNJ 8,9,11,12, X 10	Richland (1,0)				
8	39	2000-10-30	Anderson CP 1,2,3, Barnwell GS 30,31	Barnwell (3,0)	Aiken (1,0), Barnwell (13,0), Newberry (1,0)			
9	43	2001-05-14	Colleton GS 14,15,16, X 17,18	Colleton (4,0)	Colleton (2,0)	Colleton (3,0)		Beaufort (1,0)
10	49	2000-07-17	Lexington GS 17,18,19,20, X 21			Lexington (14,0)	Lexington (10,0)	Lexington (2,0)

The judge-week combinations listed in Table 19 satisfy (at least) one of the following conditions, which are problematic (some satisfy multiple conditions):[14]

1. At least one sentencing event occurred in a county not stated on the master calendar, e.g., judge-week combination 1 of Table 19 (8 such judge-week combinations are listed in Table 19).

2. At least one sentencing event occurred in the county stated on the master calendar with specific dates, but on a date other than the dates specified (only one such judge-week combination is

[14]For example, judge-week combination 1 satisfies Conditions 1 and 3. In judge-week combination 1, the assignment of Judge 6 in the week of October 16 is 15th circuit court AW 16,19,20 and Aiken 17,18. Therefore, we would expect Judge 6 to be in the 15th circuit court on Monday, Thursday, and Friday, and to be in Aiken on Tuesday and Wednesday. Since we observe sentencing events in Horry and Williamsburg on Thursday, judge-week combination 1 satisfies both Conditions 1 and 3.

listed in Table 19; see the last row of the table).

3. The judge sentenced in more than one county on a day, e.g., judge-week combination 1 of Table 19 (5 such judge-week combinations are listed in Table 19).

D Merging the Sentencing Data and the Calendar Data

D.1 Alternative mapping methods

Recall that the master calendar lists the judge names and the counties they are assigned to each week. Also, recall that the judge names are not recorded in the sentencing dataset. Instead, the judges are numbered. This section maps the judge numbers in the sentencing dataset to the judge names in the master calendar. To do so, for each judge, we create a sequence of weekly county assignments. We do so using both the sentencing dataset (using the sentencing events and their dates) and the master calendar, resulting in two sequences of weekly county assignments for each judge. We compare these two sequences to find the mapping from the judge numbers to the judge names under the following assumptions. The justification for this assumption is discussed below after presenting the matching algorithm and the resulting mapping.

Assumption 1. We restrict attention to the county name in the assignments listed on the master calendar when constructing the mapping from the judge numbers in the sentencing dataset to the judge names in the master calendar.

For each judge, in order to measure how close the aforementioned two sequences of weekly county assignments are, we start off by comparing these sequences week by week, i.e., componentwise. To do so, fix a judge and a week, and consider the list of counties for that week in each sequence. In particular, we consider the following two notions of match:

Perfect Componentwise Match. If the set of counties visited by the judge number (obtained from the sentencing dataset) is a subset of the set of counties assigned to the judge name (obtained from the master calendar), the two sets constitute a perfect componentwise match. Otherwise, they do not constitute a perfect componentwise match. They constitute a mismatch (under the perfect componentwise matching criterion). For example, let us consider Judge 25 and Judge Hayes in the week of July 10th. In the week of July 10th, Judge 25 had sentencing events in {York} and Judge Hayes was assigned to {York}. These two sets (of counties) constitute a perfect componentwise match.

Overlapping Componentwise Match. If the intersection of the two sets is non-empty, they constitute an overlapping componentwise match. Otherwise, they constitute a mismatch (under the overlapping componentwise matching criterion). For example, let us consider Judge 25 and Judge Hayes in the week of July 17th. In the week of July 17th, Judge 25 had sentencing events in {Union,Spartanburg} and Judge Hayes was assigned to {Union}. These two sets (of counties) constitute an overlapping componentwise match. Note that this would be a mismatch under the perfect componentwise matching criterion.

Then, we use the following algorithm to map the judge numbers (in the sentencing dataset) to the judge names (in the master calendar). The algorithm seeks to map each judge number to the judge name for which the number of matching weeks is maximal.

Algorithm. Choose the matching criterion.

Step 0. Fix a judge number. Find the set of judge names that have zero weeks of mismatch with the judge number under consideration. Then, map each judge number to the judge name with which it has zero weeks of mismatch. Repeat this process for all judge numbers. Next, drop the mapped judge numbers and judge names from the consideration list and proceed to the next step.

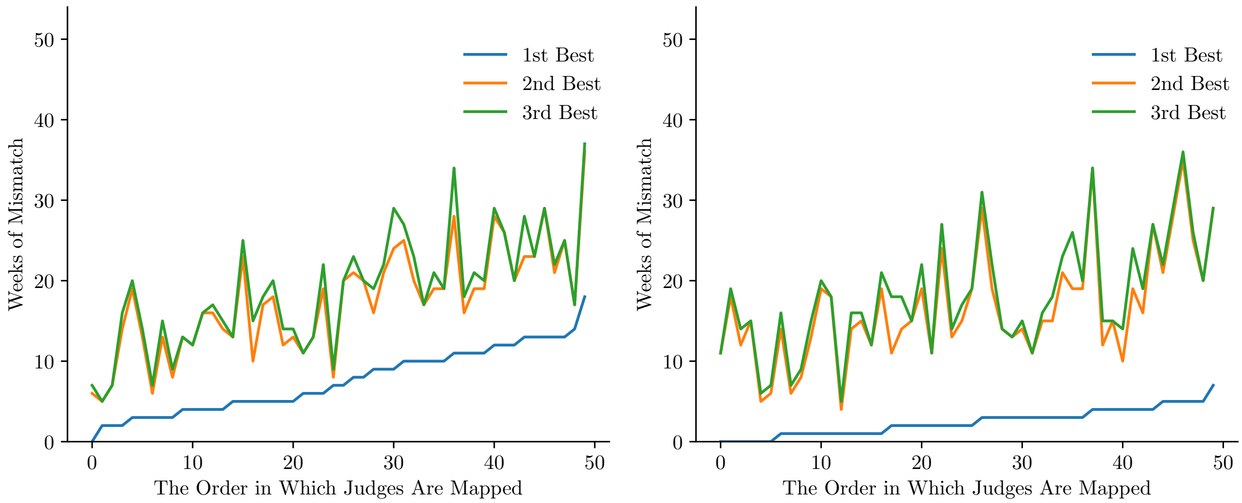
Step 1. Fix an unmapped judge number. Find the set of unmapped judge names that have 1 week of mismatch with the judge number under consideration. Then, map each unmapped judge number to the judge name with which it has 1 week of mismatch. Repeat this process for all unmapped judge numbers. Next, drop the mapped judge numbers and judge names from the consideration list. If any of the (non-fictitious) judge numbers are not yet mapped, proceed to the next step.

Step $n > 1$. Fix an unmapped judge number. Find the set of unmapped judge names that have n weeks of mismatch with the judge number under consideration. Then, map each unmapped judge number to the judge name with which it has n weeks of mismatch. Repeat this process for all unmapped judge numbers. Next, drop the mapped judge numbers and judge names from the consideration list. If any judge numbers are not yet mapped, proceed to the next step. Continue this process until all judge numbers are mapped to a judge name.

We obtain the same alphabetical mapping under both matching criterion; see Table 20. Figure 12 depicts the number of weeks of mismatch between the judge numbers and the judge names. In particular, it shows that for all judge numbers, the mapped judge name has considerably fewer weeks of mismatch than any other judge name. In particular, there were no ties.

Table 20: The mapping from the judge numbers to the judge names.

Judge Number	Judge Name	Judge Number	Judge Name
2	ALFORD	27	JOHNSON
3	BARBER	28	KEESLEY
4	BAXLEY	29	KINARD
5	BEATTY	30	KING
6	BREEDEN	31	KITTREDGE
7	BROGDON	32	LEE
8	BROWN	33	LOCKEMY
9	BUCKNER	34	MACAULAY
10	BURCH	35	MANNING
11	CLARY	36	MARTIN
12	COLE	37	MCKELLAR
13	COOPER, GT	38	MILLING
14	COOPER, TW	39	NEWMAN
15	COTTINGHAM	40	NICHOLSON
16	DENNIS	41	PATTERSON
17	FEW	42	PIEPER
18	FLOYD, H.	43	PYLE
19	FLOYD, S.	44	RAWL
20	GOODE	45	SAUNDERS
21	GOODSTEIN	46	SHORT
22	GREGORY	47	SMOAK
23	HALL	48	THOMAS
24	HARWELL	49	WATSON
25	HAYES	50	WESTBROOK
26	HUGHSTON	51	WILLIAMS



(a) Perfect Match. (b) Overlapping Match.

Figure 12: Weeks of mismatch for the judge name mapped to each judge number. The judge names with the second and third fewest weeks of mismatch are also depicted.

D.2 Evaluating the mapping

Next, we assess the performance of the (derived) mapping (from the judge numbers to the judge names). Under this mapping, we observe that only in 4.1% of the sentencing events with a date, the judge was sentencing in a county to which he was not assigned. These 4.1% problematic sentencing events belong to one of two categories: About 0.9% belong to judge-week combinations in which the judge did not have any county assignments on the master calendar. The remaining 3.2% belong to judge-week combinations in which the judge had county assignments on the master calendar. However, he was not assigned to the county in which the sentencing event occurred. These problematic sentencing events are discussed in detail in Section C.

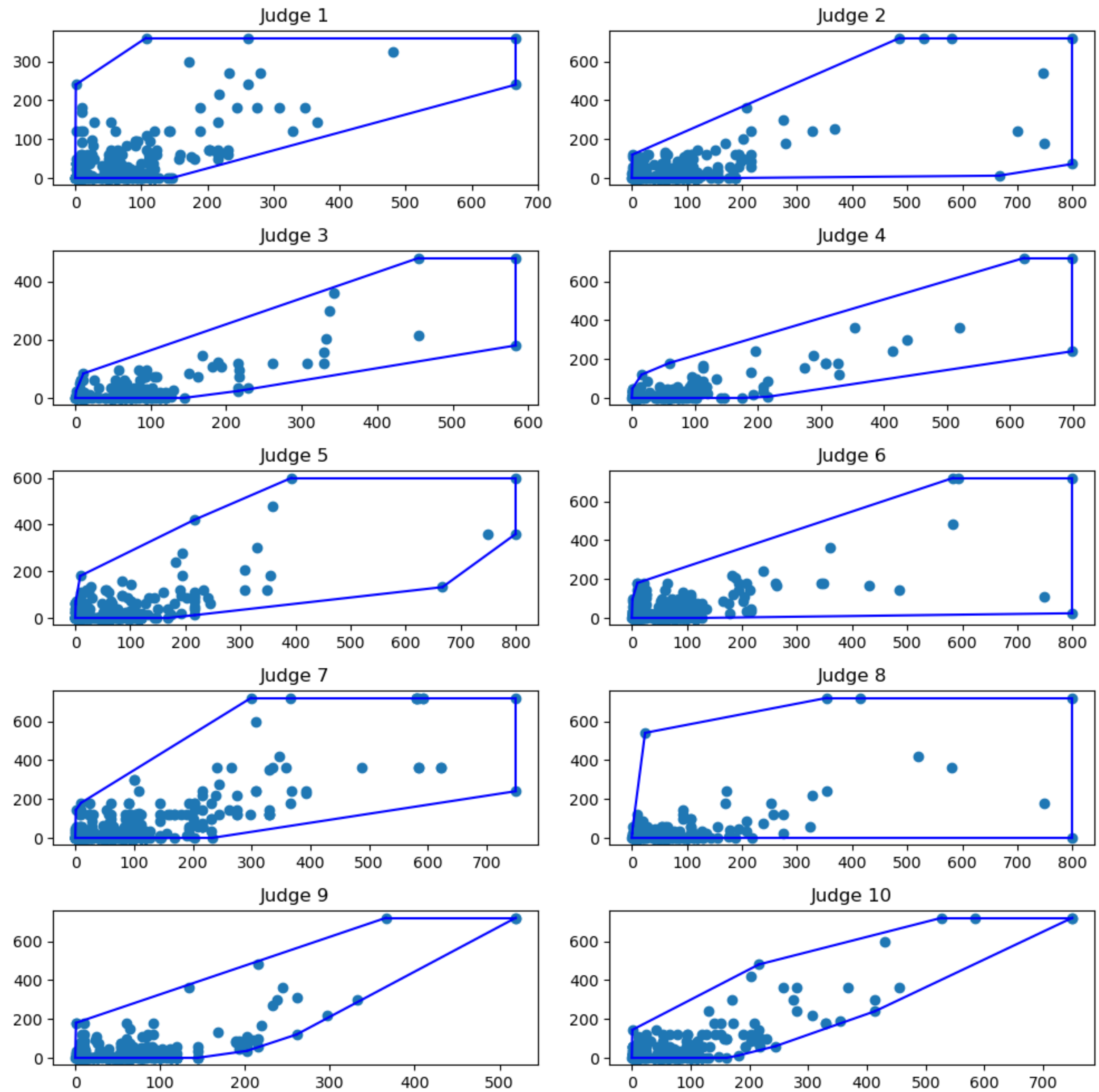
D.3 Limitations

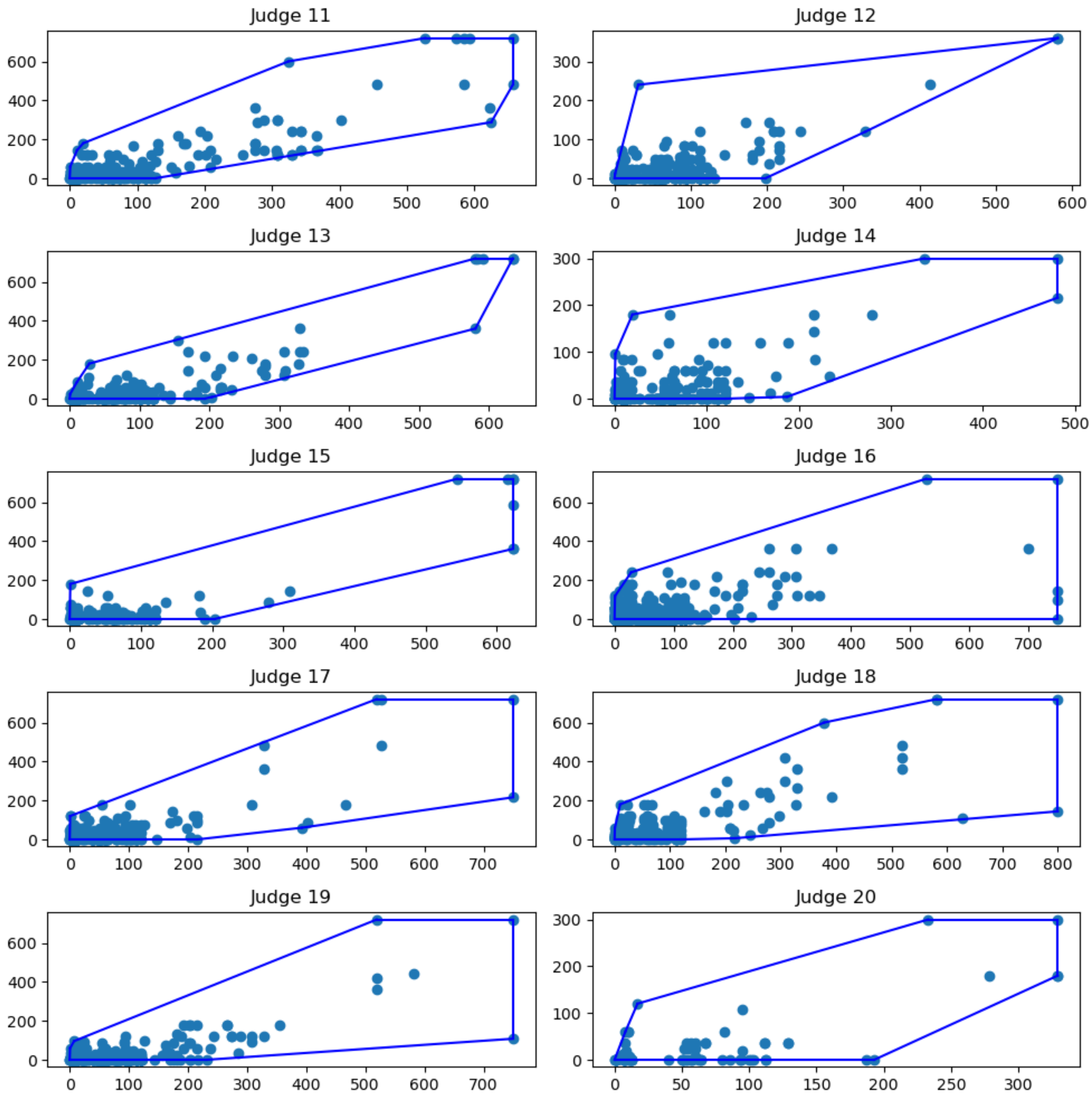
Assumption 1 ignores the "circuit" assignments on the master calendar. Each circuit contains multiple counties. Instead, our matching algorithm works with the most granular geographic unit of interest, the county. As mentioned above, 3.2% of the sentencing events with dates are at odds with the master calendar under the alphabetical mapping depicted in Table 20. Including "circuit" assignments and matching them on the basis of circuit courts as opposed to counties addresses only 12% of the problematic sentencing events (corresponding to 0.4% of the total number of sentencing events). As such, we stick with the choice of county for the matching purposes because it is more granular, and likely more accurate. Other assignments we ignore should be harmless because judges would not be involved in sentencing events or trials in those cases; see Figure 5.

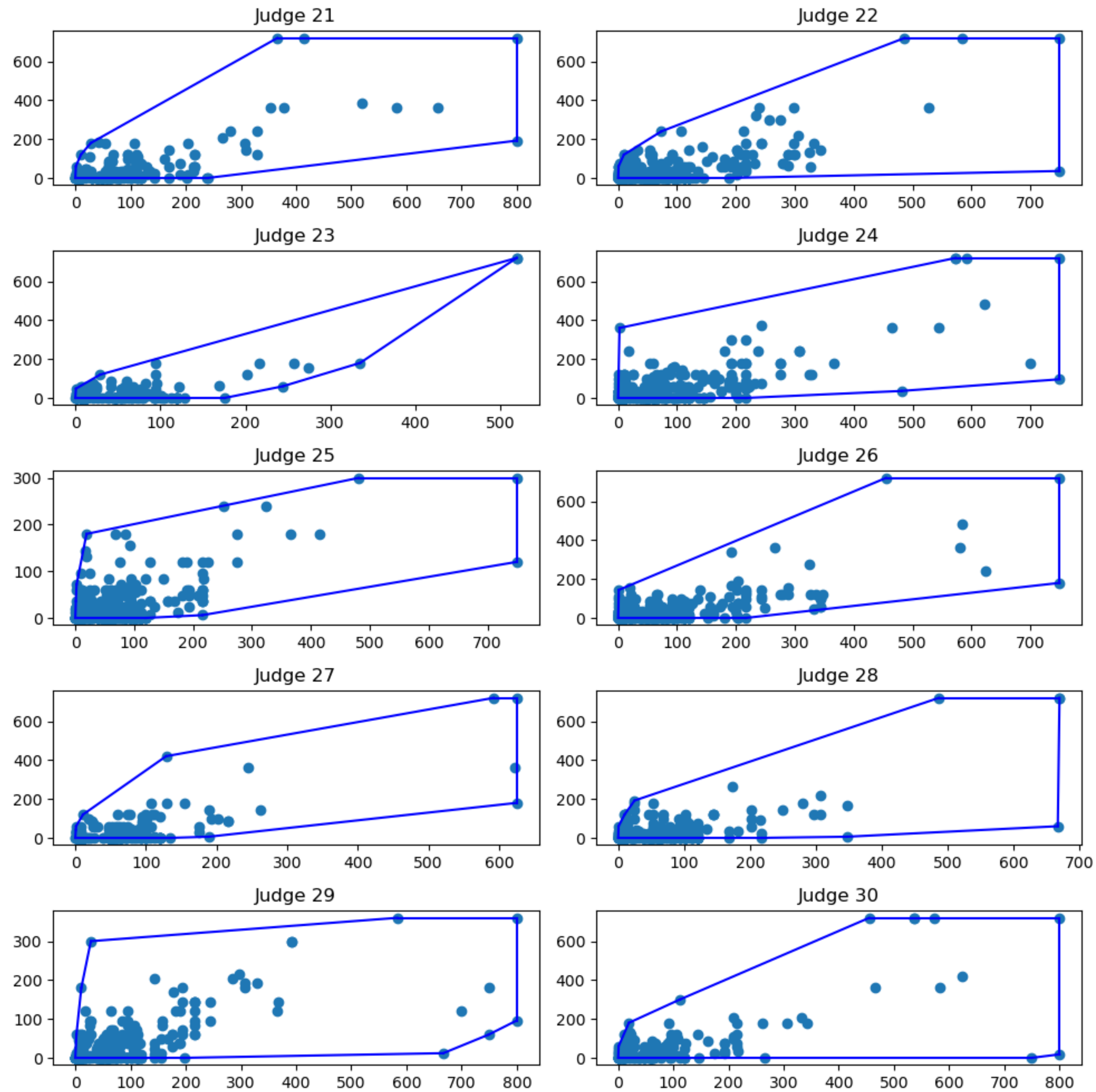
In addition, we focus only on the county names as opposed to the county name and the type of the assignment, which may also be listed; see Figure **??**. For example, Georgetown GS and Georgetown CP are both taken as Georgetown. There are several reasons for this: First, the type of the assignment is not always available. More importantly, all sentencing events should happen only in the general sessions (GS); and the common plea (CP) designation could be an error if a sentencing event appears in the sentencing dataset for the corresponding judge-week-county triple. Moreover, if no sentencing event occurs, then it is harmless to omit the CP designation. Consequently, focusing on the county name only makes the match easier.

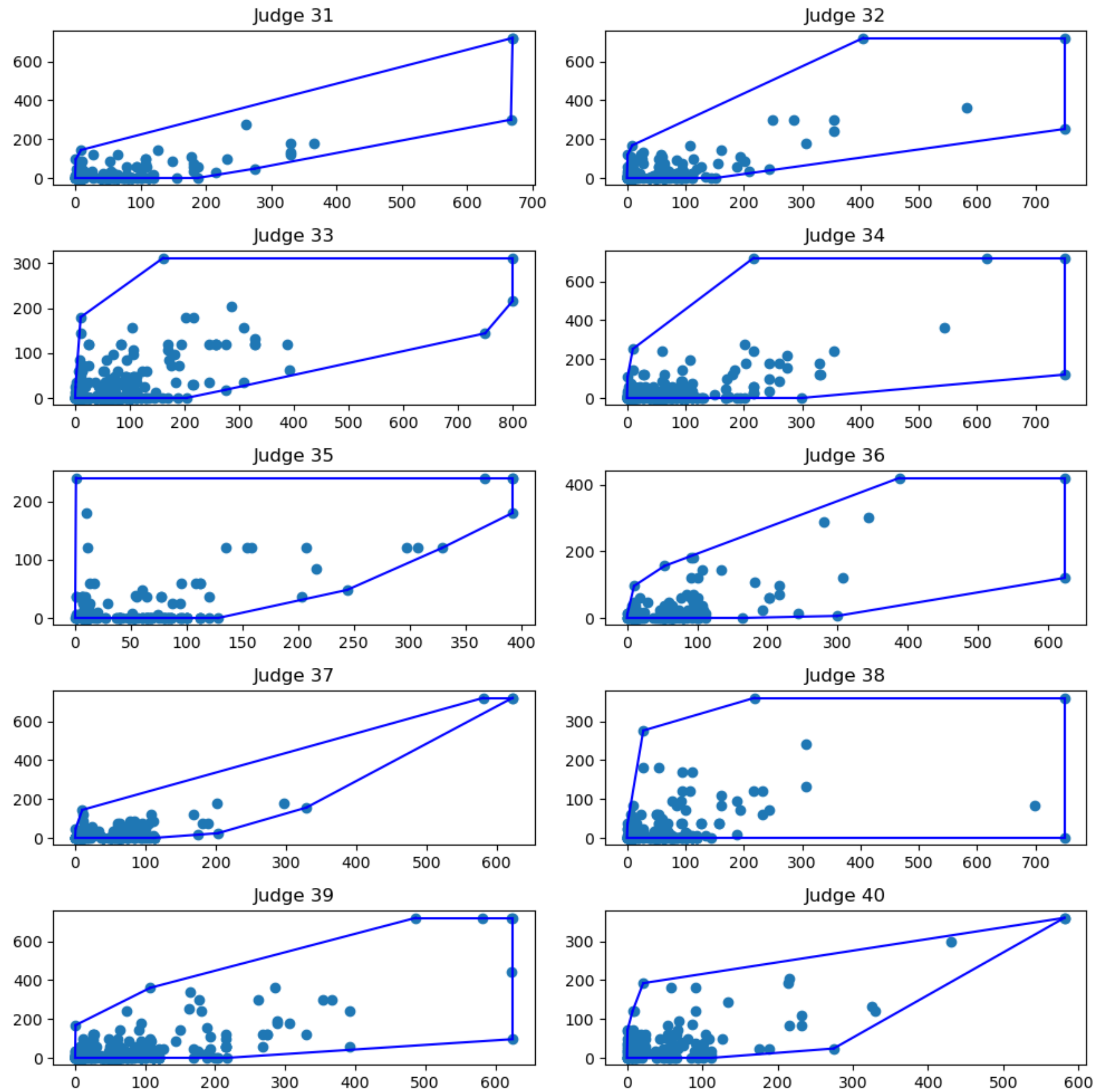
E Judge Convex Hulls

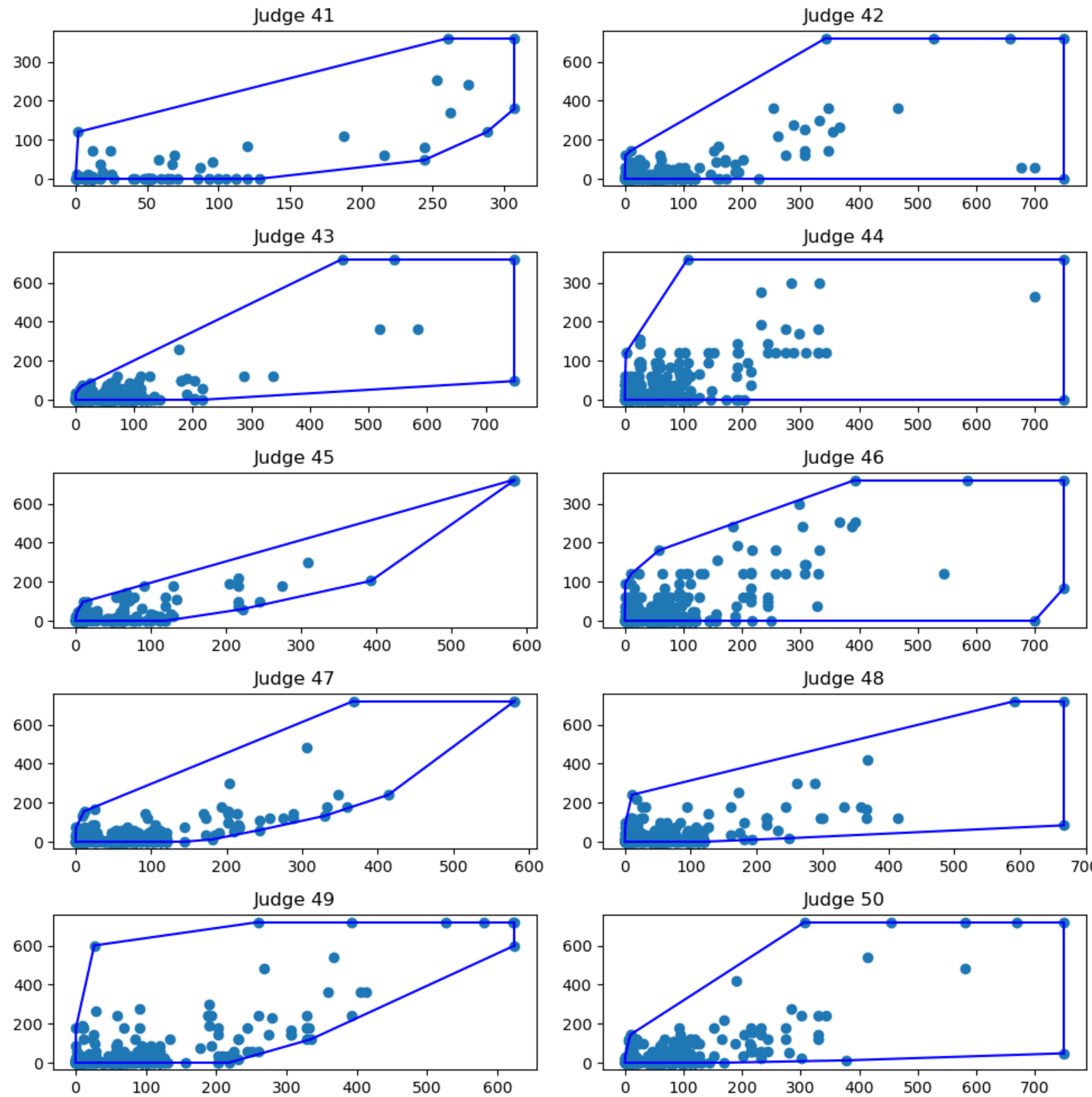
We created the convex hulls for each judge using the sentence (truncated at 720 months) as done in Hester and Hartman (2017).


Judge 1
Judge 2
Judge 3
Judge 4
Judge 5
Judge 6
Judge 7
Judge 8
Judge 9
Judge 10


Judge 11
Judge 12
Judge 13
Judge 14
Judge 15
Judge 16
Judge 17
Judge 18
Judge 19
Judge 20


Judge 21
0 200 400 600
0 100 200 300 400 500 600 700 800
Judge 22
0 200 400 600
0 100 200 300 400 500 600 700
Judge 23
0 200 400 600
0 100 200 300 400 500
Judge 24
0 200 400 600
0 100 200 300 400 500 600 700
Judge 25
0 100 200 300
0 100 200 300 400 500 600 700
Judge 26
0 200 400 600
0 100 200 300 400 500 600 700
Judge 27
0 200 400 600
0 100 200 300 400 500 600
Judge 28
0 200 400 600
0 100 200 300 400 500 600 700
Judge 29
0 100 200 300
0 100 200 300 400 500 600 700 800
Judge 30
0 200 400 600
0 100 200 300 400 500 600 700 800


Judge 31
0 200 400 600
0 100 200 300 400 500 600 700
Judge 32
0 200 400 600
0 100 200 300 400 500 600 700
Judge 33
0 100 200 300
0 100 200 300 400 500 600 700 800
Judge 34
0 200 400 600
0 100 200 300 400 500 600 700
Judge 35
0 100 200
0 50 100 150 200 250 300 350 400
Judge 36
0 200 400
0 100 200 300 400 500 600
Judge 37
0 200 400 600
0 100 200 300 400 500 600
Judge 38
0 100 200 300
0 100 200 300 400 500 600 700
Judge 39
0 200 400 600
0 100 200 300 400 500 600
Judge 40
0 100 200 300
0 100 200 300 400 500 600



F Service Rate Estimation: Other Attempts

In this section, we describe other methods we tried for estimating the service rates. A brief overview is provided below.

- **Judge Fixed Effects Model:** this model is the same as the county model we ended up going with, except that instead of having county fixed effects it has judge fixed effects. The model estimates that judges can process about 10 pleas per day, and that it takes about 4 days for judges to hear each trial.

- **Nonlinear Model:** our motivation for this model was that we wanted to model idleness as a proportion of total time, as opposed to a fixed amount of time. However, the model was not identified. We added constraints to make it identified, but it gave unreasonable estimates: $\beta_T = 1.58$ and $\beta_P = 0.02$.

- **Iterative Algorithms:** these algorithms iteratively tried to estimate the service rates as well as each judge/county's idleness. These methods yielded estimates ranging from 7 pleas per day to 21 pleas per day, and 2 days per trial to 6 days per trial.

- **Ad-hoc Algorithm:** this was an iterative algorithm to jointly estimate the two service rates. However, it yielded estimates of 56 pleas per day and about 9 days per trial. One of the major problems with this approach is that it requires many additional restrictions/assumptions.

- **EM Algorithm:** This was an expectation maximization algorithm for jointly estimating the service rates. The estimates from this model diverged. This may be due to the fact that the model is not identified.

F.1 Judge Fixed Effects

Overview of Results. This model is very similar to the county model we ended up going with. It reassuringly yields similar estimates to our county model. The model estimates that judges can process about 10 pleas per day, and that it takes about 4 days for judges to hear each trial. We ended up deciding for the county model because there was more cross-county variation in demand for criminal cases.

Motivation. A qualitative review of the system (Hester and Hartman (2017)) suggests that there could also be considerable heterogeneity in the plea processing rate across judges. According to Hester and Hartman (2017), this is because some judges develop a reputation for harshness or leniency, and since defendants have some choice with regards to the judge that hears their case, harsher judges often hear fewer cases than lenient judges. Guided by this knowledge, we decided to also fit a model with judge fixed effects. The judge fixed effects model is as follows:

$$\text{Days}_v = \alpha_j + \beta_T \text{Trials}_v + \beta_P \text{Pleas}_v + \epsilon_v$$

The estimates from this model can be seen in Table 21. The model estimates are similar to the county fixed effects model. The model estimates that judges can process about 10 pleas per day, and that it takes about 4 days for judges to hear each trial.

Table 21: Judge Fixed Effects Model

	Dependent variable:
	Days
Plea	0.105***
	(0.007)
Trial	3.861***
	(0.349)
Observations	278
R^2	0.771
Adjusted R^2	0.720
Residual Std. Error	7.083 (df = 226)
Note:	*p<0.1; **p<0.05; ***p<0.01

F.2 Nonlinear Model

Overview of Results. The models in this section yielded unreasonable estimates. On the one hand, they estimated that judges could process between 50 and 100 pleas per day. They also estimated that judges could hear trials in less than 2 days. There were several problems with these models, the main one being that they were not statistically identified. The problem could be ameliorated by adding constraints, however the resulting estimates were still unreasonable.

Description. Let g be whatever group we decide to calculate idleness for (Judge or County) and let i be our current observation, which is a judge-county combination. The non-linear model is:

$$\mu_g \text{Days}_i = \beta_P \text{Plea}_i + \beta_T \text{Trial}_i + \epsilon_i$$

Where we interpret μ_g to be the utilization of group g. To implement this, we define the function $f(\mu_g, \beta_P, \beta_T) = (\mu_g \text{Days}_i - \beta_P \text{Pleas}_i - \beta_T \text{Trials}_i)^2$ and we use a standard nonlinear optimizer to minimize it. To ensure the model is identified, we set $\mu_g = 1$ for a random judge/county. The baseline judge was Judge 41, and the baseline county was Aiken.

F.2.1 Results

The model runs and yields estimates of μ_g between 0 and 1 for all judges and counties, however, the estimates of β_T and β_P it gives seem unreasonable. The β_T estimates seem too low and the β_P estimates seem too high.

County Model The results in this subsection correspond to the model: $\mu_c \text{Days}_i = \beta_P \text{Plea}_i + \beta_T \text{Trial}_i + \epsilon_i$. Where i is a judge-county combination and c is a county. The county model yields $\beta_T = 1.58$, which would imply that it takes a little over a day and a half to process a trial. The county model also yields $\beta_P = 0.02$, which would imply judges can process 50 pleas per day.

Table 22: County Model

Parameter	Estimate
Abbeville	0.16
Aiken	1.00
Allendale	0.01
Anderson	0.26
Bamberg	0.06
Barnwell	0.19
Beaufort	0.08
Berkeley	0.19
Calhoun	0.05
Charleston	0.16
Cherokee	0.24
Chester	0.07
Chesterfield	0.18
Clarendon	0.09
Colleton	0.07
Darlington	0.06
Dillon	0.04
Dorchester	0.37
Edgefield	0.08
Fairfield	0.19
Florence	0.17
Georgetown	0.21
Greenville	0.22
Greenwood	0.16
Hampton	0.04
Horry	0.24
Jasper	0.16
Kershaw	0.10
Lancaster	0.07
Laurens	0.13
Lee	0.15
Lexington	0.18
Marion	0.13
Marlboro	0.08
McCormick	0.05
Newberry	0.12
Oconee	0.15
Orangeburg	0.15
Pickens	0.14
Richland	0.18
Saluda	0.15
Spartanburg	0.27
Sumter	0.16
Union	0.14
Williamsburg	0.06
York	0.26
BetaP	0.02
BetaT	1.58

Judge Model The results in this subsection correspond to the model: $\mu_j \text{Days}_i = \beta_P \text{Plea}_i + \beta_T \text{Trial}_i + \epsilon_i$. Where i is a judge-county combination and j is a judge. The judge model yields $\beta_T = 0.06$. The judge model also yields $\beta_P = 0.01$, which would imply judges can process 100 pleas per day.

Table 23: Judge Model

Parameter	Estimate
Judge 1	0.03
Judge 10	0.04
Judge 11	0.03
Judge 12	0.04
Judge 13	0.03
Judge 14	0.04
Judge 15	0.04
Judge 16	0.17
Judge 17	0.04
Judge 18	0.03
Judge 19	0.06
Judge 2	0.04
Judge 20	0.03
Judge 21	0.02
Judge 22	0.07
Judge 23	0.03
Judge 24	0.05
Judge 25	0.07
Judge 26	0.06
Judge 27	0.04
Judge 28	0.04
Judge 29	0.04
Judge 3	0.04
Judge 30	0.02
Judge 31	0.03
Judge 32	0.05
Judge 33	0.07
Judge 34	0.06
Judge 35	0.04
Judge 36	0.03
Judge 37	0.03
Judge 38	0.06
Judge 39	0.07
Judge 4	0.03
Judge 40	0.04
Judge 41	1.00
Judge 42	0.09
Judge 43	0.04
Judge 44	0.07
Judge 45	0.02
Judge 46	0.12
Judge 47	0.04
Judge 48	0.04
Judge 49	0.03
Judge 5	0.08
Judge 50	0.05
Judge 6	0.06
Judge 7	0.05
Judge 8	0.05
Judge 9	0.06
BetaP	0.01
BetaT	0.06

F.3 Iterative Algorithms

Overview of Results. We developed these problems to try to address the fact that judges idle. These models iteratively update the estimates of the judge/county idleness as well as the overall service rates. The estimates from these models were for the most part similar to the estimates from our fixed effects models. Table 24 contains a summary of the results of these models. In Table 24, the column 'Group' refers to the unit of observation in the dataset used to estimate the model.

Table 24: Summary of Results

Model	Group	Pleas per Day	Days per Trial
Min	County	6.92	4.41
Min	Judge	9.66	6.28
Min	Judge-County	14.97	4.25
All time Min	County	7.12	3.65
All time Min	Judge	16.53	4.52
All time Min	Judge-County	15.05	4.09
Utilization	County	7.56	3.25
Utilization	Judge	13.99	3.12
Utilization	Judge-County	20.90	1.73

Description. The main problem that we are trying to tackle is that the simple model, $\text{Days}_j = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$, doesn't account for idling judges. Hester's qualitative interviews with the judges indicates that harsher judges idle more often than more lenient judges.

F.3.1 Iterative Idleness Estimation Using Expected Utilization

Step 0: We estimate the model, $\text{Days}_j = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$.

Steps 1-n: We then use the estimates of $\beta_t^{(1)}$ and $\beta_p^{(1)}$ to estimate the expected number of days it would take each judge to complete their work. Mathematically: Expected $\text{Days}_j^{(1)} = \beta_p^{(1)} \cdot \text{Plea}_j + \beta_t^{(1)} \cdot \text{Trial}_j$. Then, the utilization for each judge would be: $\text{Utilization}_j^{(1)} = \frac{\text{Expected Days}_j^{(1)}}{\text{Days}_j}$. Let $\gamma^1 = \max_j \text{Utilization}_j^{(1)}$, be the maximum utilization amongst all judges. Each judges idleness will be: $\text{Idleness}_j^{(1)} = \frac{\text{Utilization}_j^{(1)}}{\gamma^{(1)}}$. We then set $\text{Days}_j^{(1)} = \text{Days}_j \cdot \text{Idleness}_j^{(1)}$. We then estimate the model $\text{Days}_j^{(1)} = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$ and repeat until convergence.

Table 25: Judge Model

Dep. Variable:	y	**R-squared (uncentered):**	1.000
Model:	OLS	**Adj. R-squared (uncentered):**	1.000
Method:	Least Squares	**F-statistic:**	4.033e+32
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	0.00
Time:	12:00:21	**Log-Likelihood:**	1541.8
No. Observations:	50	**AIC:**	-3080.
Df Residuals:	48	**BIC:**	-3076.
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Plea	0.0715	5.92e-18	1.21e+16	0.000	0.071	0.071
Trial	3.1154	3.49e-16	8.94e+15	0.000	3.115	3.115

Omnibus:	29.779	**Durbin-Watson:**	0.611
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	60.328
Skew:	-1.784	**Prob(JB):**	7.94e-14
Kurtosis:	7.027	**Cond. No.**	85.5

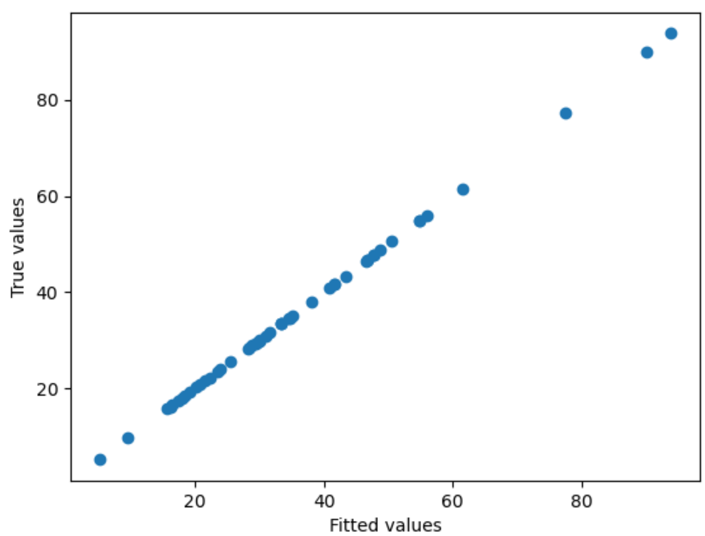


Figure 13: True vs Fitted Values, Judge Model

Table 26: Judge Model

Iteration	Beta P	Beta T
0	0.15	6.48
1	0.07	3.12
2	0.07	3.12

Table 27: Utilization at convergence, judge model

JudgeID	Plea	Trial	Days	TrialDays	PleaDays	Utilization	Idleness
Judge 16	1041	5	90.00	15.58	74.42	1.00	1.00
Judge 24	341	17	110.50	52.96	24.38	0.70	0.70
Judge 42	283	3	44.00	9.35	20.23	0.67	0.67
Judge 46	443	0	48.67	0.00	31.67	0.65	0.65
Judge 5	492	4	76.00	12.46	35.17	0.63	0.63
Judge 6	505	6	89.00	18.69	36.10	0.62	0.62
Judge 39	389	6	76.00	18.69	27.81	0.61	0.61
Judge 40	112	5	40.00	15.58	8.01	0.59	0.59
Judge 7	572	17	167.00	52.96	40.89	0.56	0.56
Judge 11	244	12	107.00	37.38	17.44	0.51	0.51
Judge 33	479	4	98.00	12.46	34.24	0.48	0.48
Judge 50	469	9	129.50	28.04	33.53	0.48	0.48
Judge 22	480	4	98.50	12.46	34.32	0.47	0.47
Judge 38	247	4	64.00	12.46	17.66	0.47	0.47
Judge 25	527	1	87.00	3.12	37.68	0.47	0.47
Judge 10	315	9	108.50	28.04	22.52	0.47	0.47
Judge 26	450	5	103.00	15.58	32.17	0.46	0.46
Judge 2	390	9	121.00	28.04	27.88	0.46	0.46
Judge 44	395	2	78.00	6.23	28.24	0.44	0.44
Judge 47	388	5	100.00	15.58	27.74	0.43	0.43
Judge 30	147	10	96.50	31.15	10.51	0.43	0.43
Judge 8	215	5	72.00	15.58	15.37	0.43	0.43
Judge 17	288	3	73.00	9.35	20.59	0.41	0.41
Judge 3	193	5	72.00	15.58	13.80	0.41	0.41
Judge 27	204	3	60.00	9.35	14.58	0.40	0.40
Judge 32	226	3	64.00	9.35	16.16	0.40	0.40
Judge 18	202	11	123.00	34.27	14.44	0.40	0.40
Judge 4	162	7	85.50	21.81	11.58	0.39	0.39
Judge 19	404	2	92.00	6.23	28.88	0.38	0.38
Judge 34	355	1	75.00	3.12	25.38	0.38	0.38
Judge 37	112	3	46.50	9.35	8.01	0.37	0.37
Judge 13	228	7	105.50	21.81	16.30	0.36	0.36
Judge 48	317	2	80.00	6.23	22.66	0.36	0.36
Judge 29	293	4	98.50	12.46	20.95	0.34	0.34
Judge 36	139	2	49.00	6.23	9.94	0.33	0.33
Judge 9	398	2	105.50	6.23	28.45	0.33	0.33
Judge 31	171	2	58.00	6.23	12.23	0.32	0.32
Judge 15	144	2	52.00	6.23	10.29	0.32	0.32
Judge 21	170	3	70.00	9.35	12.15	0.31	0.31
Judge 49	321	6	137.00	18.69	22.95	0.30	0.30
Judge 23	139	3	64.00	9.35	9.94	0.30	0.30
Judge 28	353	1	97.00	3.12	25.24	0.29	0.29
Judge 35	176	1	54.00	3.12	12.58	0.29	0.29
Judge 43	283	0	72.00	0.00	20.23	0.28	0.28
Judge 1	293	4	122.00	12.46	20.95	0.27	0.27
Judge 12	268	1	82.00	3.12	19.16	0.27	0.27
Judge 14	208	1	68.00	3.12	14.87	0.26	0.26
Judge 41	91	1	38.00	3.12	6.51	0.25	0.25
Judge 45	161	3	90.00	9.35	11.51	0.23	0.23
Judge 20	72	0	23.00	0.00	5.15	0.22	0.22

Judge Model

Table 28: County Model

Dep. Variable:	y	**R-squared (uncentered):**	1.000
Model:	OLS	**Adj. R-squared (uncentered):**	1.000
Method:	Least Squares	**F-statistic:**	1.261e+33
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	0.00
Time:	12:00:22	**Log-Likelihood:**	1409.1
No. Observations:	46	**AIC:**	-2814.
Df Residuals:	44	**BIC:**	-2810.
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Plea	0.1324	8.1e-18	1.63e+16	0.000	0.132	0.132
Trial	3.2526	5.3e-16	6.14e+15	0.000	3.253	3.253

Omnibus:	36.252	**Durbin-Watson:**	1.962
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	192.576
Skew:	-1.678	**Prob(JB):**	1.52e-42
Kurtosis:	12.445	**Cond. No.**	149.

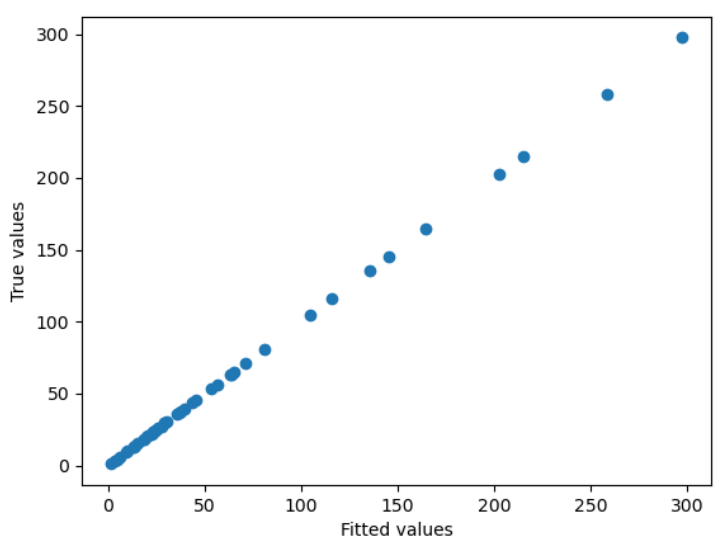


Figure 14: True vs Fitted Values, Judge-County Model

Table 29: Judge Model

Iteration	Beta P	Beta T
0	0.17	4.23
1	0.13	3.25
2	0.13	3.25

Table 30: Utilization at convergence, county model

County	Plea	Trial	Days	TrialDays	PleaDays	Utilization	Idleness
Spartanburg	1063	19	202.50	61.80	140.70	1.00	1.00
Greenville	1608	14	272.00	45.54	212.84	0.95	0.95
Dorchester	291	10	78.00	32.53	38.52	0.91	0.91
Anderson	655	15	161.00	48.79	86.70	0.84	0.84
Oconee	230	2	46.50	6.51	30.44	0.79	0.79
Florence	754	5	146.50	16.26	99.80	0.79	0.79
Aiken	343	11	105.50	35.78	45.40	0.77	0.77
Berkeley	382	4	83.00	13.01	50.56	0.77	0.77
York	777	19	216.50	61.80	102.85	0.76	0.76
Lexington	644	6	141.00	19.52	85.24	0.74	0.74
Charleston	1329	12	291.50	39.03	175.91	0.74	0.74
Barnwell	113	1	25.00	3.25	14.96	0.73	0.73
Horry	655	18	199.50	58.55	86.70	0.73	0.73
Greenwood	370	5	90.00	16.26	48.97	0.72	0.72
Richland	1633	25	417.00	81.31	216.15	0.71	0.71
Cherokee	128	7	57.50	22.77	16.94	0.69	0.69
Georgetown	230	7	79.00	22.77	30.44	0.67	0.67
Sumter	353	5	100.00	16.26	46.72	0.63	0.63
Pickens	254	3	73.00	9.76	33.62	0.59	0.59
Laurens	293	2	82.00	6.51	38.78	0.55	0.55
Orangeburg	327	4	104.00	13.01	43.28	0.54	0.54
Saluda	74	1	25.00	3.25	9.79	0.52	0.52
Kershaw	268	0	68.00	0.00	35.47	0.52	0.52
Jasper	73	1	25.00	3.25	9.66	0.52	0.52
Chesterfield	131	4	59.00	13.01	17.34	0.51	0.51
Marion	171	1	51.00	3.25	22.63	0.51	0.51
Union	147	3	59.00	9.76	19.46	0.50	0.50
Abbeville	118	2	45.00	6.51	15.62	0.49	0.49
Fairfield	62	5	50.00	16.26	8.21	0.49	0.49
Newberry	149	1	48.00	3.25	19.72	0.48	0.48
Lee	91	2	40.50	6.51	12.04	0.46	0.46
Edgefield	116	0	34.00	0.00	15.35	0.45	0.45
Clarendon	158	2	63.50	6.51	20.91	0.43	0.43
Darlington	172	2	68.00	6.51	22.77	0.43	0.43
Bamberg	70	0	22.67	0.00	9.27	0.41	0.41
Colleton	124	1	50.50	3.25	16.41	0.39	0.39
Lancaster	158	1	67.00	3.25	20.91	0.36	0.36
Marlboro	174	0	66.50	0.00	23.03	0.35	0.35
Beaufort	183	4	109.00	13.01	24.22	0.34	0.34
Williamsburg	156	0	61.00	0.00	20.65	0.34	0.34
McCormick	47	0	23.00	0.00	6.22	0.27	0.27
Calhoun	27	0	14.00	0.00	3.57	0.26	0.26
Chester	79	1	59.00	3.25	10.46	0.23	0.23
Hampton	33	0	19.00	0.00	4.37	0.23	0.23
Dillon	74	0	48.00	0.00	9.79	0.20	0.20
Allendale	8	0	14.00	0.00	1.06	0.08	0.08

County Model

Table 31: County Model

Dep. Variable:	y	**R-squared (uncentered):**	1.000
Model:	OLS	**Adj. R-squared (uncentered):**	1.000
Method:	Least Squares	**F-statistic:**	1.306e+33
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	0.00
Time:	12:00:22	**Log-Likelihood:**	8998.6
No. Observations:	278	**AIC:**	-1.799e+04
Df Residuals:	276	**BIC:**	-1.799e+04
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Plea	0.0478	1.72e-18	2.79e+16	0.000	0.048	0.048
Trial	1.7321	8.7e-17	1.99e+16	0.000	1.732	1.732

Omnibus:	127.483	**Durbin-Watson:**	1.817
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	1614.656
Skew:	-1.487	**Prob(JB):**	0.00
Kurtosis:	14.426	**Cond. No.**	61.2

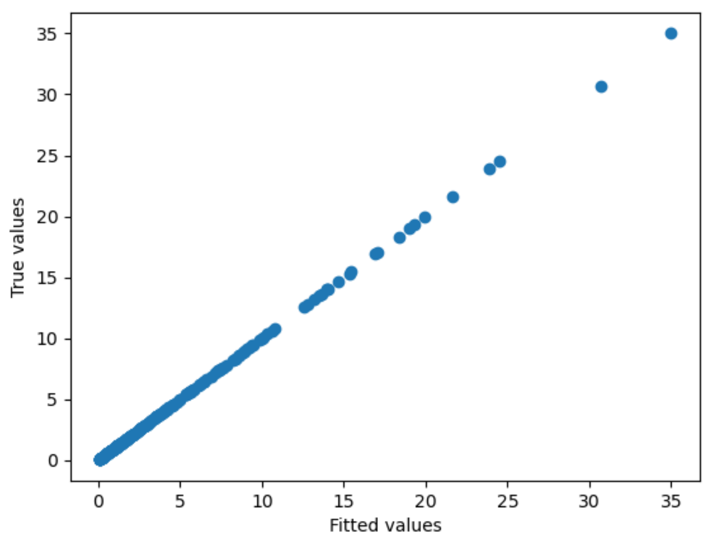


Figure 15: True vs Fitted Values, Judge-County Model

Table 32: Judge-County Model

Iteration	Beta P	Beta T
0	0.13	4.76
1	0.05	1.73
2	0.05	1.73

Judge-County Model

F.3.2 Iterative Idleness Estimation Taking Mins

Step 0: We estimate the model, $\text{Days}_j = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$.

Steps 1-n: We then use the estimates of $\beta_t^{(1)}$ and $\beta_p^{(1)}$ to estimate the expected number of days it would take each judge to complete their work. Mathematically: Expected $\text{Days}_j^{(1)} = \beta_p^{(1)} \cdot \text{Plea}_j + \beta_t^{(1)} \cdot \text{Trial}_j$. We would then set $\text{Days}_j^{(1)} = \min(\text{Days}_j, \text{Expected Days}_j^{(1)})$ We then estimate the model $\text{Days}_j^{(1)} = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$ and repeat until convergence.

F.4 Judge Model

Table 33: Judge Model

Dep. Variable:	y	**R-squared (uncentered):**	1.000
Model:	OLS	**Adj. R-squared (uncentered):**	1.000
Method:	Least Squares	**F-statistic:**	4.033e+32
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	0.00
Time:	12:00:21	**Log-Likelihood:**	1541.8
No. Observations:	50	**AIC:**	-3080.
Df Residuals:	48	**BIC:**	-3076.
Df Model:	2		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Plea	0.0715	5.92e-18	1.21e+16	0.000	0.071	0.071
Trial	3.1154	3.49e-16	8.94e+15	0.000	3.115	3.115

Omnibus:	29.779	**Durbin-Watson:**	0.611
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	60.328
Skew:	-1.784	**Prob(JB):**	7.94e-14
Kurtosis:	7.027	**Cond. No.**	85.5

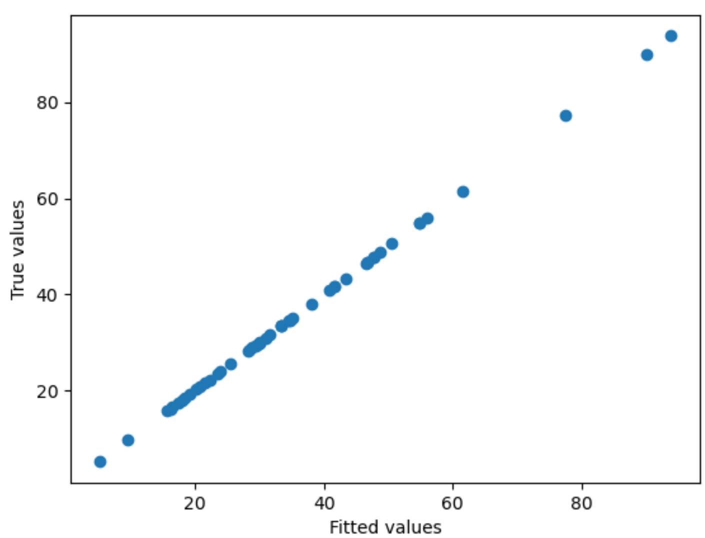


Figure 16: True vs Fitted Values, Judge Model

Table 34: Judge Model

Iteration	Beta P	Beta T
0	0.15	6.48
1	0.07	3.12
2	0.07	3.12

F.5 County Model

Table 35: County Model

Dep. Variable:	y	**R-squared (uncentered):**	0.995
Model:	OLS	**Adj. R-squared (uncentered):**	0.995
Method:	Least Squares	**F-statistic:**	4553.
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	1.01e-51
Time:	11:40:08	**Log-Likelihood:**	-157.08
No. Observations:	46	**AIC:**	318.2
Df Residuals:	44	**BIC:**	321.8
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Plea	0.1445	0.005	29.177	0.000	0.134	0.154
Trial	4.4126	0.324	13.619	0.000	3.760	5.066

Omnibus:	56.300	**Durbin-Watson:**	2.055
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	376.354
Skew:	-3.024	**Prob(JB):**	1.89e-82
Kurtosis:	15.641	**Cond. No.**	149.

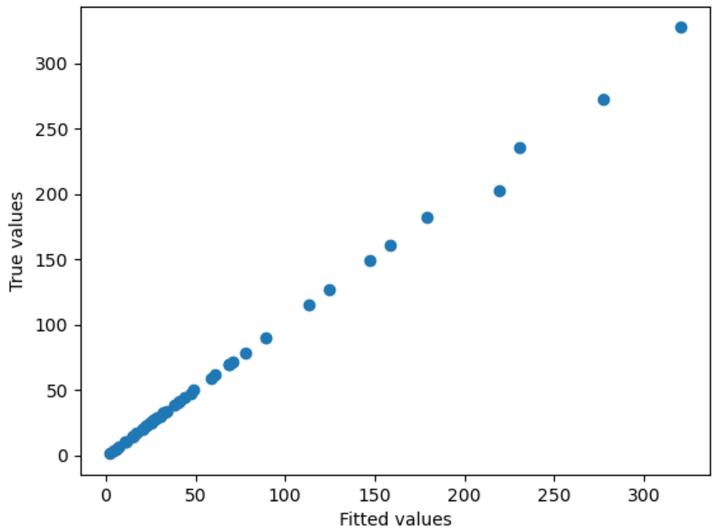


Figure 17: True vs Fitted Values, Judge-County Model

Table 36: Judge Model

Iteration	Beta P	Beta T
0	0.15	3.61
1	0.14	3.67
2	0.14	3.65

F.6 Judge-County Model

Table 37: County Model

Dep. Variable:	y	**R-squared (uncentered):**	0.986
Model:	OLS	**Adj. R-squared (uncentered):**	0.986
Method:	Least Squares	**F-statistic:**	9809.
Date:	Wed, 08 Sep 2021	**Prob (F-statistic):**	4.19e-257
Time:	11:40:09	**Log-Likelihood:**	-491.26
No. Observations:	278	**AIC:**	986.5
Df Residuals:	276	**BIC:**	993.8
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Plea	0.0668	0.001	58.241	0.000	0.065	0.069
Trial	4.2463	0.058	72.973	0.000	4.132	4.361

Omnibus:	507.684	**Durbin-Watson:**	1.990
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	235811.290
Skew:	-10.399	**Prob(JB):**	0.00
Kurtosis:	144.157	**Cond. No.**	61.2

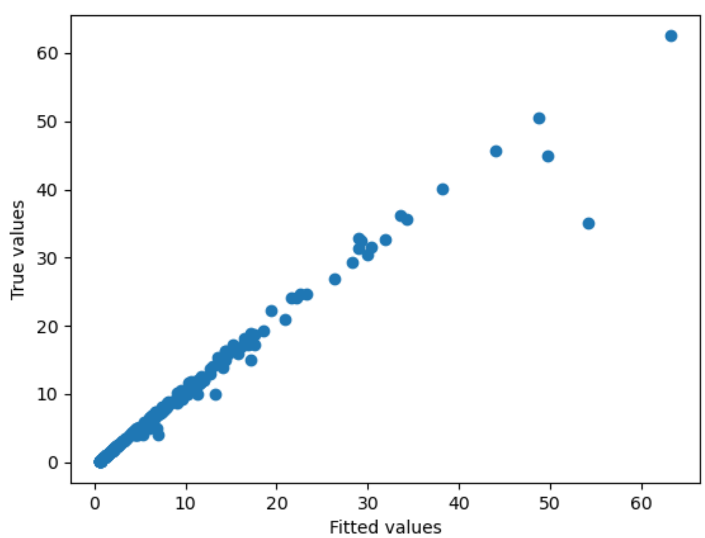


Figure 18: True vs Fitted Values, Judge-County Model

Table 38: Judge-County Model

Iteration	Beta P	Beta T
0	0.10	4.19
1	0.08	4.11
2	0.07	4.09

F.6.1 Iterative Idleness Estimation Taking All-time Mins

Step 0: We estimate the model, $\text{Days}_j = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$.

Steps 1-n: We then use the estimates of $\beta_t^{(1)}$ and $\beta_p^{(1)}$ to estimate the expected number of days it would take each judge to complete their work. Mathematically: Expected $\text{Days}_j^{(1)} = \beta_p^{(1)} \cdot \text{Plea}_j + \beta_t^{(1)} \cdot \text{Trial}_j$. We would then set $\text{Days}_j^{(n)} = \min(\text{Days}_j, \text{Expected Days}_j^{(n-11)}, ..., \text{Expected Days}_j^{(1)})$ We then estimate the model $\text{Days}_j^{(1)} = \beta_t \text{Trial}_j + \beta_p \text{Plea}_j + \epsilon_j$ and repeat until convergence.

F.7 Judge Model

Table 39: Judge Model

Dep. Variable:	y	**R-squared:**	1.000
Model:	OLS	**Adj. R-squared:**	1.000
Method:	Least Squares	**F-statistic:**	1.813e+30
Date:	Tue, 14 Sep 2021	**Prob (F-statistic):**	0.00
Time:	13:26:30	**Log-Likelihood:**	1438.6
No. Observations:	50	**AIC:**	-2871.
Df Residuals:	47	**BIC:**	-2865.
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Intercept	-1.243e-14	2.56e-14	-0.485	0.630	-6.4e-14	3.91e-14
Plea	0.0605	6.94e-17	8.72e+14	0.000	0.060	0.060
Trial	4.5182	3e-15	1.51e+15	0.000	4.518	4.518

Omnibus:	8.336	**Durbin-Watson:**	2.156
Prob(Omnibus):	0.015	**Jarque-Bera (JB):**	7.538
Skew:	-0.791	**Prob(JB):**	0.0231
Kurtosis:	4.056	**Cond. No.**	790.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

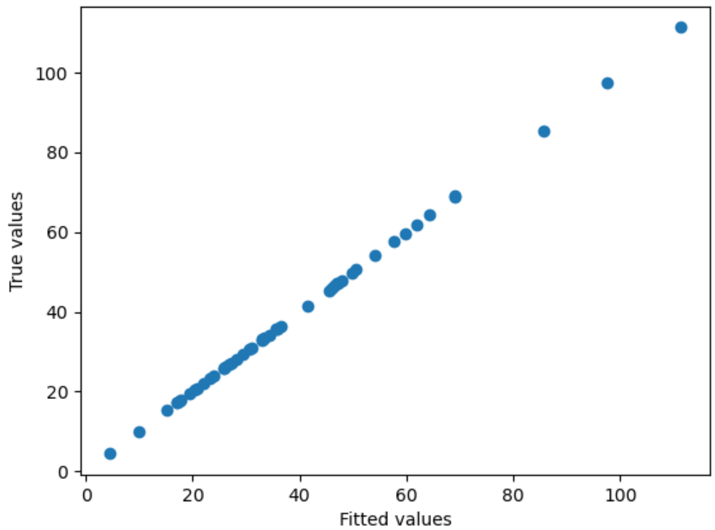


Figure 19: True vs Fitted Values, Judge Model

Table 40: Judge Model

Iteration	Beta P	Beta T
0	0.06	4.52
1	0.06	4.52

F.8 County Model

Table 41: County Model

Dep. Variable:	y	**R-squared:**	0.998
Model:	OLS	**Adj. R-squared:**	0.998
Method:	Least Squares	**F-statistic:**	1.342e+04
Date:	Tue, 14 Sep 2021	**Prob (F-statistic):**	7.67e-61
Time:	13:27:42	**Log-Likelihood:**	-115.16
No. Observations:	46	**AIC:**	236.3
Df Residuals:	43	**BIC:**	241.8
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Intercept	0.7176	0.599	1.199	0.237	-0.490	1.925
Plea	0.1403	0.002	67.351	0.000	0.136	0.144
Trial	3.6474	0.133	27.471	0.000	3.380	3.915

Omnibus:	68.690	**Durbin-Watson:**	2.097
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	840.457
Skew:	-3.638	**Prob(JB):**	3.14e-183
Kurtosis:	22.636	**Cond. No.**	679.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

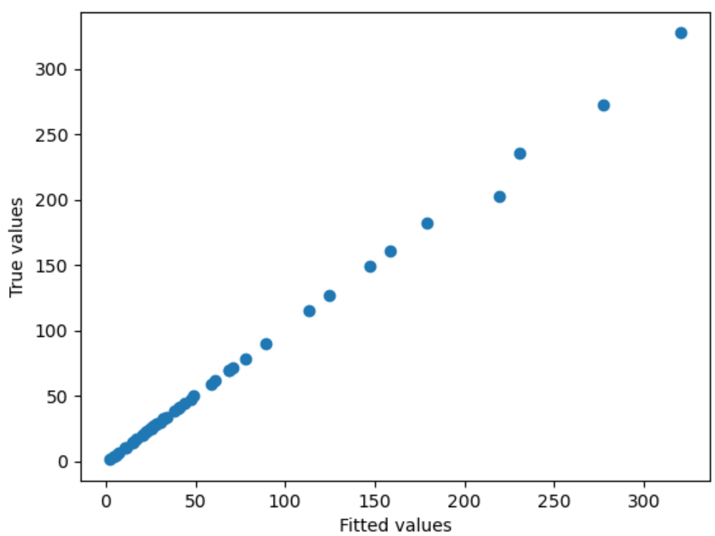


Figure 20: True vs Fitted Values, Judge-County Model

Table 42: Judge Model

Iteration	Beta P	Beta T
0	0.15	3.61
1	0.14	3.67
2	0.14	3.65

F.9 Judge-County Model

Table 43: County Model

Dep. Variable:	y	**R-squared:**	0.977
Model:	OLS	**Adj. R-squared:**	0.977
Method:	Least Squares	**F-statistic:**	5950.
Date:	Tue, 14 Sep 2021	**Prob (F-statistic):**	4.56e-227
Time:	13:28:16	**Log-Likelihood:**	-492.29
No. Observations:	278	**AIC:**	990.6
Df Residuals:	275	**BIC:**	1001.
Df Model:	2		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
Intercept	0.4634	0.110	4.203	0.000	0.246	0.681
Plea	0.0665	0.001	50.392	0.000	0.064	0.069
Trial	4.0877	0.059	68.783	0.000	3.971	4.205

Omnibus:	462.896	**Durbin-Watson:**	2.072
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	159919.551
Skew:	-8.673	**Prob(JB):**	0.00
Kurtosis:	119.212	**Cond. No.**	116.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

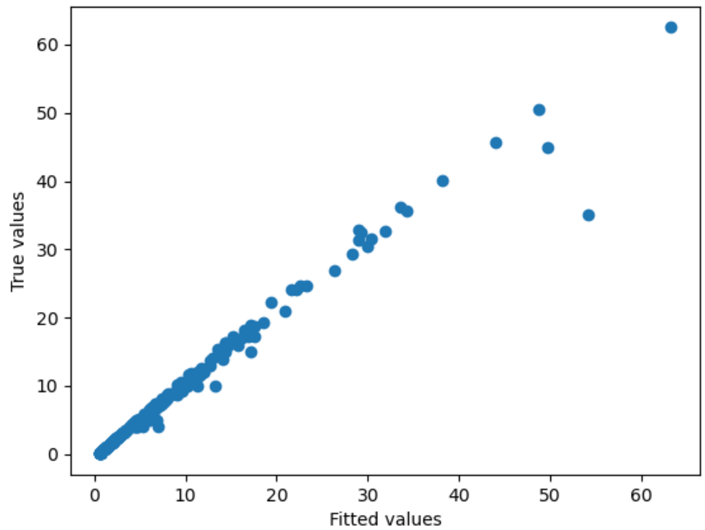


Figure 21: True vs Fitted Values, Judge-County Model

Table 44: Judge-County Model

Iteration	Beta P	Beta T
0	0.10	4.19
1	0.08	4.11
2	0.07	4.09

F.10 Ad-hoc Algorithm

Overview of Results. This approach yielded estimates of 56.5 pleas per day and about 9 days per trial. We think that the large estimate for the plea service rate is due to demand censoring. Since the amount of pleas processed can be limited by the demand, the likelihood function can always be increased by increasing the plea processing rate. An analogy we came up with was the case of a coffee shop barista claiming that if there was enough demand, they could make 100 cappuchinos a day. Since we never observe that much demand, it is impossible to test the validity of the statement. In a similar sense, suppose you know that average plea demand per day is about 6 pleas per day and there is one day in which you observe that 6 pleas were processed. In this case, it is slightly more likely that a judge that can process an average of 8 pleas per day processed the 6 pleas than a judge who can process an average of 7 pleas per day. We believe these dynamics are what increases the plea processing rate. Another reason we did not opt for this method is that it required making many decisions regarding the different samples. Each decision imposed a different assumption, and it would have made the sensitivity analysis much more difficult.

F.10.1 Samples

Plea MLE Sample This is the sample we use for the maximum likelihood estimation part of the ad-hoc algorithm. We also refer to this sample as "clean days". For this, we only consider pleas that happened on days which satisfy the following conditions:

Table 45: MLE Plea Sample Exclusion Criteria

Condition
No inconsistencies between sentencing data and calendar
Judge has at least one sentencing event that day
Judge is only assigned to one county that day
Judge only sentences in one county that day
Judge never has more than 35 sentencing events in this county
Judge calendar assignment is of type "GS"

Plea Arrival Rate Sample This is the sample we use to estimate the plea arrival rate, θ in the ad-hoc algorithm. The calculation of θ involves two quantities: the total number of pleas in the data, N_p, and d, the total number of judge days. d is meant to represent the number of days in which a judge could have been working on pleas. As a result, for d we include all days of type "GS", and we include days of other work types in which we observe sentencing events. N_p currently includes all pleas in our data.

Trial Rate Sample This is the sample we use to estimate the trial service rate, μ_t. When calculating the total number of days a judge was assigned to a county, we include all days he had a "GS" assignment to that county in the master calendar, and all days of type non-GS in which we observe a sentencing event. Note, this is the same criteria as used above for the plea arrival rate sample. We include all pleas and all trials when calculating the total number of pleas and trials the judge heard in that county. We focus on the judge-county combinations that satisfy the following conditions:

Table 46: Trial Service Rate Exclusion Criteria

Condition
Judge never has more than 35 sentencing events in one day in this county
Judge has at least 2 trial in this county

F.10.2 Estimation of μ_t

To estimate the trial service rate, we focus on the judge-county combinations that satisfy the conditions described in the Trial Rate Sample paragraph. Let K denote the number judge-county combinations satisfying these two conditions. We number these judge-county combinations $1, \ldots, K$ and define $\mathcal{K} = \{1, \ldots, K\}$. For judge-county $k \in \mathcal{K}$, we let $n_p(k)$ and $n_t(k)$ denote the total number of pleas and the total number of trials undertaken by this judge in this county. Similarly, for judge-county $k \in \mathcal{K}$, we let $T(k)$ denote the number of days this judge was assigned to this county.[15]

First, we assume the judges in judge-county combinations $k \in \mathcal{K}$ never idle. If this assumption was correct, the trial service rate of judge-county $k \in \mathcal{K}$ would be

$$\hat{\mu}_t(k) = \frac{n_t(k)}{T(k) - n_p(k)/\hat{\mu}_p}.$$

The tuples $(n_t(k), \hat{\mu}_t(k))$ for $k \in \mathcal{K}$ are depicted in Figure 22.

[15] In calculating $T(k)$ for $k \in \mathcal{K}$, we assume the judge divides his time equally among the county assignments to which he is assigned if he is assigned to multiple counties on a day.

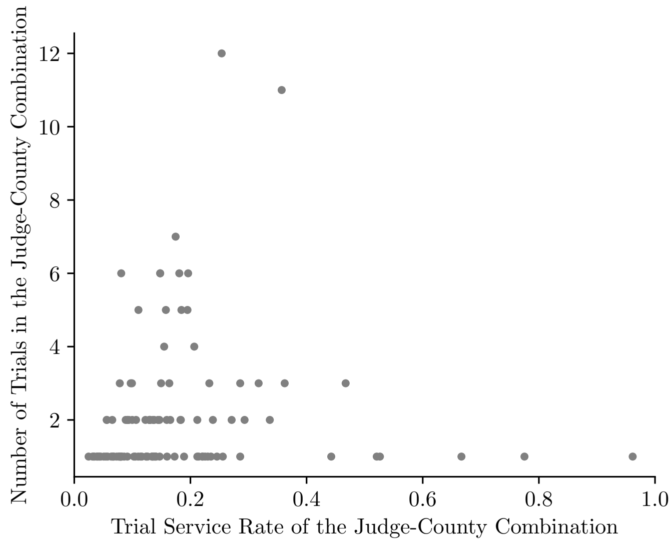


Figure 22: A graphical representation of the tuples $(n_t(k), \hat{\mu}_t(k))$ for $k \in \mathcal{K}$.

We observe that as the number of trials in a judge-county combination decreases, the variance of the $\hat{\mu}_t(k)$ estimates increase. Moreover, the $\hat{\mu}_t(k)$ estimates skew to the left as the number of trials decreases. This suggests that the judges in judge-county combinations with few trials spent some time idling. Therefore, to estimate the trail service rate, we focus on judge-county combinations for which we observe at least two trials, i.e., $k \in \tilde{\mathcal{K}} = \{k : k \in \mathcal{K}, n_t(k) \geq 2\}$. These judge-county combinations account for 72% of the trials in the dataset. The trial service rate estimate is

$$\hat{\mu}_t = \frac{\sum\limits_{k \in \tilde{\mathcal{K}}} n_t(k)}{\sum\limits_{k \in \tilde{\mathcal{K}}} T(k) - \sum\limits_{k \in \tilde{\mathcal{K}}} n_p(k)/\hat{\mu}_p}.$$

F.10.3 Ad-hoc Algorithm for Joint Estimation of μ_t, μ_p

Step 1: Let μ_p, μ_t be the current values for the plea and trial service rate. As in the estimation of μ_t, we are assuming judges only work on pleas and trials and do not idle. As a result, given the total number of trials heard, N_t, and the trial service rate, μ_t, we can calculate the expected number of days judges spent working on trials. The number of days judges spent on trials, $d_t = \frac{N_t}{\mu_t}$. We calculate the total number of days judges worked, d using the assignments from the master calendar and removing public holidays. The expected number of days judges worked on pleas is then $d_p = d - d_t$.

Step 2: Let N_p denote the total number of pleas in the data. Again, we include all pleas in our sample to calculate N_p. We set $\theta = \frac{N_p}{d_p}$. We model the plea demand for a judge as $D \sim \text{Poisson}(\theta)$, whereas the number of pleas a judge can serve in a day is denoted by X, $X \sim \text{Poisson}(\mu_p)$.

Step 3: Let $S_i = \min(D_i, X_i)$ denote the number of pleas sentenced for judge-day combination $i = 1, ..., N$. Here, we only include the judge-day combinations that satisfy our Plea MLE conditions. We have that

$$\begin{aligned}P(S_i = S) &= P(X_i = S|X_i \leq D_i)P(X_i \leq D_i) + P(D_i = S|X_i > D_i)P(X_i > D_i)\\ &= \frac{\theta^s e^{-\theta}}{s!}[1 - \sum_{k=0}^{s-1} \frac{\mu_p^k e^{-\mu_p}}{k!}] + \frac{\mu_p^s e^{-\mu_p}}{s!}[1 - \sum_{k=0}^{s} \frac{\theta^k e^{-\theta}}{k!}]\end{aligned}$$

Let $L(\mu_p) = -\sum_{i=1}^{N} \log P(S_i = s)$. We then set $\mu_p = \operatorname{argmin} L(\mu_p)$ and calculate μ_t as described in F.10.2. Again, the judge-day combinations for which we are minimizing the negative log likelihood are those that satisfy our Plea MLE conditions. We use a gradient descent algorithm with the Adam Optimizer to find the value of μ_p that minimizes the NLL. This new value of μ_p will imply a new value of μ_t as in Section F.10.2, and so we repeat Steps 1-3 until we converge.

F.11 EM Algorithm

Overview of Results. The estimates from this algorithm quickly diverged to infinity. This could be because the model is not well identified. Another shortcoming of the model is that it required deciding many samples, each of which required different assumptions. The many assumptions required for this method would have made sensitivity analysis more difficult.

EM Algorithm Sample This is the sample we use for the EM Algorithm. For this, we only consider pleas that happened on days which satisfy the following conditions:

Table 47: EM Algorithm Sample Exclusion Criteria

Condition
No inconsistencies between sentencing data and calendar
Judge has at least one sentencing event that day
Judge is only assigned to one county that day
Judge only sentences in one county that day
Judge never has more than 35 sentencing events in this county
Judge calendar assignment is of type "GS"

F.11.1 EM Algorithm under Normal Distribution Assumption

Under the normal distribution assumption, we write

$$
\begin{aligned}
f_X(x) &= \frac{1}{\sigma_1\sqrt{2\pi}} \exp -\frac{1}{2}(\frac{x-\mu_p}{\sigma_1})^2 = \phi(\frac{x-\mu_p}{\sigma_1}) \\
f_D(d) &= \frac{1}{\sigma_2\sqrt{2\pi}} \exp -\frac{1}{2}(\frac{d-\theta}{\sigma_2})^2 = \phi(\frac{d-\theta}{\sigma_2})
\end{aligned}
$$

Because $S = \min(X, D)$, we write:

$$P(S \leq s) = P(X \leq s)P(D \leq s) = \Phi(\frac{s-\mu_p}{\sigma_1})\Phi(\frac{s-\theta}{\sigma_2})$$

Thus, $f_S(s) = \frac{d}{ds}P(S \leq s)$ is given as follows:

$$f_S(s) = \frac{1}{\sigma_1}\phi(\frac{s-\mu_p}{\sigma_1})\Phi(\frac{s-\theta}{\sigma_1}) + \frac{1}{\sigma_2}\phi(\frac{s-\theta}{\sigma_2})\Phi(\frac{s-\mu_p}{\sigma_2})$$

For the E-Step, we need to calculate

$$\mathbb{E}_{X,D}[\log f(X, D, S|\theta, \mu)|S, \theta, \mu_p]$$

Note (for $s = \min(X, D)$) that

$$\log f(X, D, S) = -\log \sigma_1 - \frac{1}{2}(\frac{x-\mu_p}{\sigma_1})^2 - \log \sigma_2 - \frac{1}{2}(\frac{D-\theta}{\sigma_2})^2$$

Thus, we are interested in computing

$$-\log\sigma_1 - \log\sigma_2 - \frac{1}{2}\mathbb{E}[(\frac{X-\mu_p}{\sigma_1})^2 + (\frac{D-\theta}{\sigma_2})^2]|S = s$$

To this end, lets first derive the conditional density $f(X, D|S)$:

$$f(X,D|S) \propto \begin{cases} \frac{f(X,D)}{f(S)} & \text{if } S = \min(X,D), \\ 0 & \text{otherwise,} \end{cases}$$

In the first case, we have that

$$f(X,D|S) \propto \frac{\phi(\frac{x-\mu_p}{\sigma_1})\phi(\frac{D-\theta}{\sigma_2})}{\frac{1}{\sigma_1}\phi(\frac{S-\mu_p}{\sigma_1})\Phi(\frac{S-\theta}{\sigma_2}) + \frac{1}{\sigma_2}\phi(\frac{S-\theta}{\sigma_2})\Phi(\frac{S-\mu_p}{\sigma_1})} \tag{2}$$

Then, we have that

$$\begin{aligned} \mathbb{E}[\log f(X,D,S)|S] &= -\log\sigma_1 - \log\sigma_2 \\ -\frac{1}{2}[\int_S^\infty & f(x,S|S)[(\frac{x-\mu_p}{\sigma_1})^2 + (\frac{S-\theta}{\sigma_2})^2]dx \\ + \int_S^\infty [(\frac{S-\mu_p}{\sigma_1})^2 &+ (\frac{d-\theta}{\sigma_2})^2]f(S,d|S)dd] \end{aligned}$$

M-Step: Minimize $\sum_{i=1}^N \mathbb{E}[\log f(X_i, D_i, S_i)|S_i]$ over $\mu_p, \theta, \sigma_1, \sigma_2$.

To do so, we can use Gauss-Hermite integration (see Ken Judd's book) for Normal densities. Then we iterate until the parameters $\mu_p, \theta, \sigma_1, \sigma_2$ converge. Once that happens, we estimate μ_t as in Section F.10.2.

F.12 County Fixed Effects

This section contains the estimates of each county's fiexed effect in our final model.

Table 48: Model with County Fixed effects, table continues on next page

	Dependent variable:
	Days
Plea	0.091***
	(0.008)
Trial	4.336***
	(0.339)
CountyAbbeville	6.411*
	(3.784)
CountyAiken	3.107
	(2.909)
CountyAllendale	4.425
	(4.360)
CountyAnderson	5.237*
	(2.985)
CountyBamberg	5.443
	(4.364)
CountyBarnwell	2.608
	(3.782)
CountyBeaufort	8.232***
	(2.524)
CountyBerkeley	3.724
	(2.885)
CountyCalhoun	3.852
	(4.361)
CountyCharleston	9.165***
	(2.237)
CountyCherokee	3.112
	(3.408)
CountyChester	14.170***
	(4.365)
CountyChesterfield	4.966
	(3.093)
CountyClarendon	7.305**
	(3.387)
CountyColleton	4.991*
	(2.858)
CountyDarlington	7.293**
	(3.091)
CountyDillon	7.260**
	(3.380)
CountyDorchester	2.073
	(3.876)
CountyEdgefield	7.832*
	(4.371)
CountyFairfield	5.677
	(3.798)
CountyFlorence	9.426***
	(3.229)
CountyGeorgetown	5.565
	(3.415)
CountyGreenville	6.571**
	(2.687)
CountyGreenwood	5.803*
	(3.122)
CountyHampton	4.003
	(3.777)

Table 49: Model with County Fixed effects, continued

	Dependent variable:
	Days
CountyHorry	5.715**
	(2.482)
CountyJasper	3.514
	(3.779)
CountyKershaw	6.248**
	(2.871)
CountyLancaster	11.090***
	(3.788)
CountyLaurens	7.800**
	(3.106)
CountyLee	4.218
	(3.382)
CountyLexington	4.390**
	(2.219)
CountyMarion	5.295
	(3.790)
CountyMarlboro	10.937***
	(3.792)
CountyMcCormick	6.248
	(4.362)
CountyNewberry	6.035*
	(3.385)
CountyOconee	2.834
	(3.097)
CountyOrangeburg	6.506**
	(2.689)
CountyPickens	6.166**
	(3.101)
CountyRichland	10.717***
	(2.141)
CountySaluda	4.655
	(4.365)
CountySpartanburg	2.984
	(2.897)
CountySumter	6.194**
	(2.883)
CountyUnion	6.537*
	(3.388)
CountyWilliamsburg	5.859**
	(2.675)
CountyYork	7.085***
	(2.651)
Observations	278
R^2	0.881
Adjusted R^2	0.856
Residual Std. Error	7.552 (df = 230)
F Statistic	35.355*** (df = 48; 230)
Note:	*p<0.1; **p<0.05; ***p<0.01