Simulation Study Overview

June 30, 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 can be found in table 1.

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

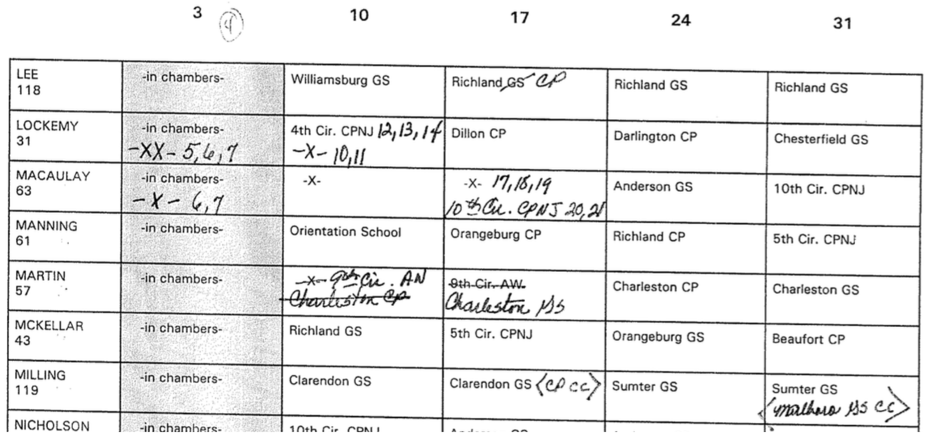
1.1.1 Sample

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. 1,551 observations are missing the date variable, so we exclued these from our sample as well. After these two exclusions, we are left with 16,034 observations.

1.2 Judge Schedule 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 1. 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 1: Snapshot of Judge Calendar



	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. AN ~~Charleston CP~~	~~9th 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	[illegible]		

1.2.1 Parsing

As can be seen from figure 1, the assignments are on a weekly basis. A judge can be assigned to one or multiple counties in a week, and we observe 5 different kinds of assignments: single assignment, single assignment with dates, multiple assignment, multiple assignment with some dates, and multiple assignment with all dates. We parse the calendar data to determine each judge's schedule for every day of the year. In the resulting data, the unit of observation is judge, day, county. Note that the only potentially ambiguous assignment types are multiple assignment and multiple assignment with some dates. For these assignment types, we set the judge to be in all counties that he is scheduled to be in. For example, if judge 2 is scheduled for both Marion and Horry in the week of March 19-March 23, for each day of that week our data would include an observation for judge 2 in Horry and an observation for judge 2 in Marion. An example of what our parsed data looks like can be seen in table 3.

Table 2: Assignment Types

Assignment	Example
Single	Marion
Single with dates	Marion 23, 24, 25
Multiple	Marion, Horry
Multiple some dates	Marion 23, 24, Horry
Multiple all dates	Marion 23, 24, Horry 25, 26, 27

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

Judge	Day	County
2	2001-03-19	Marion
3	2001-03-19	Horry
3	2001-03-19	Greenville

1.2.2 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. 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). Some assignment types, like CP (common pleas), are for civil cases. A full list of the assignment type acronyms and their meanings can be found in table 4.

Table 4: Assignment Type Acronyms

Acronym	Meaning
GS	General Session
CC	Circuit Court
SGJ	State Grand Jury
CP	Common Pleas
CPNJ	Common Pleas Non-Jury
PCR	Post Conviction Relief
Capital PCR	Capital Post Conviction Relief
AW	Administrative Week

2 Model

We use 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, $\theta\tau$ 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.

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.

3.1 Probability of Conviction at Trial - θ

We currently estimate this using l_2 regularized logistic regression with a penalty parameter of 1. We use the following variables to predict this quantity: Black, Offense Type, and Offense Seriousness. We represent all variables as binary variables. We split our sample into train and test sets using a 75/25 split to evaluate the performance of our classifier. Our classifier's performance can be seen in table 5. 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 5: Evaluation Metrics for Classifier

Metric	**Score**
AUC	0.93
F1	0.94
Accuracy	0.89

3.1.1 Sample

We only use the observations in the data that are trials and that have non missing values for the predictor variables. There are only 256 observations in the data meeting this criteria.

3.1.2 Proposed Changes

We should consider jointly estimating θ and τ using a Hurdle Regression model as in Hester and Hartman 2017.

3.2 Expected Sentence Length if Convicted - τ

We currently estimate this using Negative Binomial Regression. We use the Cameron-Trivedi test for overdispersion to choose the overdispersion parameter. As in Hester and Hartman 2017, our dependent variable is the expected minimum sentence. We use the following variables to predict this quantity: Black, Age, Offense Type, and Offense Seriousness.

3.2.1 Sample

We only use the observations in the data that are trials and that have non missing values for the predictor variables. There are only 256 observations in the data meeting this criteria.

3.2.2 Proposed Changes

We should consider jointly estimating θ and τ using a Hurdle Regression model as in Hester and Hartman 2017.

3.3 Defendant Cost of Trial - c_d

We are currently estimating this using only the first method described in the Write-Up. In this method, we use the subset of cases where the sentence, s, is less than $u_j(\theta\tau)$. In these cases, our model implies that $c_d = s - \theta\tau$.

3.3.1 Proposed Changes

We should implement the maximum likelihood estimation procedure described in Nasser's document for the other cases. Here's the discussion: 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 L_j as follows:

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

Then, we let $L = \prod_{j=1}^{J} L_j$ and $\operatorname{argmax} L$ helps us choose the parameters of the distribution F.

Remark 1. Note that the approach proposed immediately above allows c_d to be negative. We can attribute this to $c_d(i)$ being the cost of trial plus an idiosyncratic shock specific to defendant i, which encompasses all factors that are not captured explicitly in the model.

Remark 2. We envision drawing defendant profiles (along with their cases) with replacement from the dataset at a particular weekly rate in our simulation study. As we do so, we can set $c_d = c_d(i)$ whenever we draw a defendant who falls into the set $\mathcal{I}_j^1$ for some judge j. Otherwise, we can consider the following two cases:

Case i) Defendant i is in the set $\mathcal{I}_j^2$ for some judge j. Then, we draw $c_d(i)$ from the conditional distribution $F(x | x \geq u_j - \hat{\theta}_i \hat{\tau}_i)$.

Case ii) Defendant i is in the set K_j for some judge j. Then, we draw $c_d(i)$ from the conditional distribution $F(x | x \leq l_j - \hat{\theta}_i \hat{\tau}_i)$.

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

We currently estimate this using a K-nearest neighbors approach, with $K = 5$. Given a defendant's $\theta\tau$, we find the K defendants in the judge's past pleas with the most similar values of $\theta\tau$. We then pick the minimum and maximum of these pleas to determine $l_j(\theta\tau), u_j(\theta\tau)$.

3.4.1 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.

3.4.2 Convex Hull Approach

I found a function that calculates the convex hull of a finite set of points. It calculates the extreme points and the simplices of the convex hull. The simplices are given as tuples of two points $((x_1, y_1), (x_2, y_2))$. One way to implement the convex hull approach could be to iterate over the simplices of the convex hull, and for each simplex get the points that lie on the line between the simplex's two points. We could then add these points to a list that contains all the points on the boundary of the convex hull. Then, given a specific $\theta_1\tau_1$, we could find the 4 points on the boundary of the convex hull that have values of $\theta\tau$ closest to $\theta_1\tau_1$. We could then take the average of the largest (in terms of y value) two points for u_j and the average of the smallest (in terms of y value) two points for l_j. **Note:** we should think about what to do in cases where $\theta_1\tau_1$ is not in the convex hull.

3.5 County Arrival Rates - λ_c

We set each county's arrival rate to be equal 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.

3.6 Service Rates - μ_p, μ_t

3.6.1 Samples

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

Table 6: MLE Plea Sample Exclusion Criteria

Condition
No inconsistencies between sentencing data and calendar
Judge has at least 10 'clean' days
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"

Sensitivity Analysis: Add these one by one, similar to how controls are presented in Econ papers, and see how the quantity of interest changes as they are added. We could also add them in groups. Some of these correspond to outlier control and others to day cleanliness. We can run it with and without each group.

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. N_p currently includes all pleas in our data, and d currently includes all days a judge has an assignment in the master calendar, except for holidays.
Proposed Changes: None

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 was assigned to that county in the master calendar (except for public holidays). 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 7: 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

Proposed Changes: Since our explicit assumption is that judges only spend time on pleas or trials when calculating the expected trial days and expected plea days, I think it might make sense to exclude days where it is unlikely that the judge was working on either of those two tasks, like when they are hearing civil cases. **Sensitivity Analysis:** We can think of our conditions as corresponding to day cleanliness and outlier control, and group them based on those criteria. We can then add them by group and see how our estimate evolves. If any criteria significantly changes the estimate, we should argue why it is a good criteria to have.

3.6.2 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.[1]

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

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

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

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

3.6.3 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

[1] 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.

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

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

Let $L(\mu_p) = -\sum_{i=1}^{N} \log P(S_i = s)$. We then set $\mu_p = \text{argmin}\, L(\mu_p)$ and calculate μ_t as described in 3.6.2. 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, and so we repeat Steps 1-3 until we converge.

4 Simulation Plan

We simulate the system using the discrete choice model described in Section 3. Our simulation is as follows:

State Variables For each judge, we keep track of their capacity in each period, including future periods, as well as their current and future schedules. For each county, we keep track of the judges that are scheduled to appear in every time period (including future time periods), the county's backlogged defendants, and the number of defendants that choose to go to trial.

Judge Assignments We first assign each county one judge for each time period. The time periods here are discrete and we think of them as weeks. There are 50 judges and 46 counties. Each week, 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.

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. If the county has any backlogged defendants in that week, the backlogged defendants are added to that week's list of defendants, and are served first.

Trial Scheduling One potential difficulty in scheduling trials in the simulation is that trials are likely to require more than 5 days of processing time and they might need to be uninterrupted (this is something we can decide). Since our assignment of judges to counties is on a weekly schedule, we have to make sure that if a judge is scheduled to hear a trial in a county, he is there for at least as long as it will take to process the trial. We might have to decide whether to give priority to the process assigning judges to counties or to give priority to the process assigning judge trial dates. **Option 1:** One way we could assign trial dates is to first assign judges to counties as described above and only consider judges that are scheduled to be in a county for the necessary amount of time to process a trial. We could then just block the judge's capacity for these weeks in advance to represent trial times. We would have to double check that judges are assigned 'trial-length' stints sufficiently often to meet the trial demand we observe in the data. **Option 2:** Another option would be to first schedule the trials for the judges, and in those weeks deterministically assign judges to sit in a particular county (we could make this their home county). The remaining weeks in the schedule would be assigned as described above.

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. If there are no judges available in the next r weeks, the defendant is added to the county's list of backlogged defendants and processed again next week.

5 Next Steps

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

References

Hester, Rhys and Todd K Hartman (2017). "Conditional race disparities in criminal sentencing: A test of the liberation hypothesis from a non-guidelines state". In: *Journal of quantitative criminology* 33.1, pp. 77–100.

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