Machine Predictions and Contraband Hit Rates in Police Searches

Jordan Hamada
Linnet Sim

Presented to the Department of Economics at the University of Oregon, in partial fulfillment of the requirements for honors in Economics

June 8, 2018

Under the supervision of Professor Ben Hansen
University of Oregon

Abstract

This study seeks to build a predictive model that tells police officers whether or not they should search a stopped vehicle for contraband. Using data from the Stanford Open Policing Project, we identify objective variables that allow for a "yes" or "no" input before a search is conduct for use in the predictive model. With an easy-to-use model that limits subjectivity, we sought out to understand if police officers make biased decisions when searching vehicles and if our elimination of bias could make better decisions. Using a mix of traditional economic regressions and machine learning algorithms, we develop a model that is significantly more accurate than actual police officer searches but suffers from type II errors by failing to search vehicles that should have been searched. We conclude with future considerations that can make this model more robust and effective, in hopes that it can implemented and used by police officers to supplement their work.

Table of Contents

I. Introduction

Racial tension has become especially prevalent in modern society. Given recent events in America such as the Austin bombings that noted an emphasis on racial bias, we thought it would be interesting to examine how public officials were affect by this tension and if this, either intentionally or unintentionally, resulted in them adversely overemphasizing race, gender, and other demographic characteristics when performing their duties. Specifically, we wanted to see if this bias was displayed in police officers deciding to search vehicles for contraband.

The determination to search a vehicle after it has been pulled over is one that is subject to an individual officer's judgement with the general protocol being:

> *"police may search a vehicle incident to a recent occupant's arrest only if the arrestee is within reaching distance of the passenger compartment at the time of the search or it is reasonable to believe the vehicle contains evidence of the offense of the arrest."* (5)

This is true as long as there is "reasonable" suspicion or "probable cause", both of which are vague protocols which are subject to the individual officer's discretion. Though we recognize that determining how dangerous a subject is can be difficult, and decisions must be made quickly, these protocols are very ambiguous and leave room for any personal bias an officer may have whether it be intentional or unintentional.

We believe that this can be mitigated through the development of a predictive model based on police stop and search rate data that would determine if an officer should search a vehicle instead

of leaving the decision to their own discretion. This predictive model would allow the police officers to remove their subjectivity when deciding whether or not to search a vehicle as they would be able to use the model instead of relying on inadvertently allowing their personal views to affect their professional decisions. Theoretically this would also increase contraband yield rates while decreasing the number of searches conducted as it would be a more accurate way to predict and determine if a search should be conducted or not.

This paper walks through the development of this predictive model, interprets the significant factors of finding contraband during a search, and discusses the accuracy of the final model. Using machine learning to remove the potentially bias of police officers can bring its own bias, so we conclude with future steps and considerations to make this model even more equitable. Our goal is that this predictive model can be implemented and used by law enforcement in the state of Illinois to tell them whether or not they should search a vehicle for contraband once pulled over. While making it easy for the police officers to input various factors, this model can then be developed and customized in other states for widespread use across the United States.

II. Literature Review

There have been several studies to assess the racial disparities in police actions taken against drivers. For example, *A large-scale analysis of racial disparities in police stops across the United States* (2) conducted by Stanford University complied a data set of over 60 million state patrol stops and analyzed it to determine if and how there was racial discrimination. They concluded that though there was a disparity between the amount of black and white drivers stopped relative to the population, as well as a clear disparity in the likelihood of different races being searched and the likelihood of the searches finding contraband, after controlling for age, gender, time and location, there were a number of other factors that also affected the stop and search rates. They were not always necessarily due to a racial bias. For example,

> "The recent legalization of recreational marijuana in Colorado and Washington reduced the absolute gap in search rates between whites and minorities—because search rates decreased for all groups—but the relative gap remained."

Though the paper provides a unique perspective on working with large-scale policing data, there are other factors that should be considered when furthering their research and diving deeper into if there is a racial bias displayed in stop data and if so, how it affects people of different ethnicities as we try to work towards developing a way to eliminating this bias.

Another article we examined, *An Alternative Test of Racial Prejudice in Motor Vehicle Searches: Theory and Evidence* (3) by Shamena Anwar and HanMing Fang designed a simple model of trooper behaviours to determine if they exhibit relative racial prejudice in motor vehicle

searches. They cite guidelines for police on the "The Common Characteristics of Drug Couriers" which specifically list race/ethnicity as a characteristic that should affect searches as opening up a possibility for troopers to engage in racist practices against minority motorists. Though it is clear that almost all highway patrol departments have renounced the 1985 guidelines, the public still remains sceptical. The paper concluded that black and Hispanic motorist in the United States were much more likely that white motorist to be searched by troopers while

> "When applied to vehicle stop-and-search data from Florida, our tests soundly reject the hypothesis that troopers of different races are monolithic in their search behaviour, but fail to reject the hypothesis that troopers of different races do not exhibit relative racial prejudice."

However, this paper focuses only on the trooper's decision to search a vehicle, it does not look at the trooper's decision to stop the car or compare the population data. Fang and Anwar recognize that without this consideration, their results could be skewed as perhaps the decision to pull a car over could be where the racial bias is exhibited and not in the searching of the vehicle.

We also wanted to consider the "veil of darkness" theory discussed by Grogger and Ridgeway in *Testing for Racial Profiling in Traffic Stops from Behind a Veil of Darkness* (4). They attempt to develop an alternative approach to testing for racial profiling in traffic stops that does not require explicit estimates of race distribution of the population at risk of being stopped in order to find an inexpensive and efficient way to determine if a search or stop was conducted because of racial bias. The "veil of darkness" hypothesis is that police have greater difficulty observing the race

of a driver at night before they make a stop and hence would not be able to impose a racial bias when deciding whether or not to stop a car. After omitting stops that were pursuant to criminal investigation and incomplete data, they concluded that non-black drivers were disproportionately stopped during the daylight when visibility is high. This shows that police behaviour cannot be shown to display racial bias. However, this could be due to violations associated with the driver's race and darkness i.e. a headlight being broken. There are also other factors that were not taken into account such as the neighbourhoods patrolled. At night police may patrol neighbourhoods where crime is more prevalent, and these may have a disproportionate percentage of a certain race residing there, increasing the likelihood of interaction with them and hence stopping them. Grogger and Ridgeway also recognize that

> "The test is consistent, but its power is reduced by anything that reduces the correlation between visibility and darkness. In the case of two important examples, street lighting and car characteristics, additional data collection could boost the power of the test to detect racial profiling."

The test is also only designed to study the extent of racial profiling in traffic stops and not in the post-stop outcomes such as duration or search rates which are necessary to determine a comprehensive assessment of racial profiling. This data set is also specific to the Oakland area and not representative of the whole United States.

Based on the literature reviewed, we now have many factors to take into consideration when further analyzing this question and trying to determine if there is racial bias present and if so,

whether it is prevalent in vehicles being pulled over or in the outcome of a stop. Factors that affect the legitimacy of our claim and our model are also clear such as the neighbourhoods patrolled and the differences in data between states.

III. Data and Methodology

Data: Stanford Open Policing

The data used in our analysis comes from The Stanford Open Policing Project, which is a standardized data repository detailing data on pedestrian and vehicle stops from law enforcement departments from across the United States of America. The Open Policing Project has been requesting information from states since 2015 and to date has collected and standardized over 100 million traffic stop and search data from 31 states from across the country.

This standardized data includes a variety of factors such as: Stop Date, Stop Time, Stop Location, Driver Race, Driver Gender, Driver Age, Stop Reason, Search Conducted, Search Type, Contraband Found and Stop Outcome. However, different states had different factor data available. For example, while Connecticut had data for all the factors listed above, Maryland only had data for six of the factors. Moreover, different states had different categorization techniques with Connecticut including a County code for all stops, while Illinois uses a Police District code for all stops.

Based on the sizes of data sets and the number and quality of factors available, we decided to use Illinois (IL) as our sample data set to build our model. With over 4.7 million stops between 2004 and 2015 as well as being one of the few states with data for all listed factors, we thought it would provide good sample data to build and test our model with. Moreover, Illinois houses the third-largest city in the United States, Chicago, which is nationally known for the racial diversity. The data contains large distribution of races, making it easier to compare stop and search data against general state demographics.

Summary Statistics

Before developing our predictive model, we wanted to gain a better understanding of police stops, searches, and hits. In the below table, we control for gender and see that there are significantly more males stopped, searched, and found with contraband. Though the male search rate is 86.70% greater than the female search rate, the male hit rate is 4.05% lesser than the female hit rate. As such, males are searched much more than females but have a smaller hit rate. Police officers are inefficiently over-searching males, which our predictive model strives to fix.

Gender	Stops	Searches	Hits	Search rate	Hit rate
F	1,370,298	33,961	7,210	2.48%	21.23%
M	3,344,733	154,767	31,523	4.63%	20.37%

In this table, we control for race. We see that Hispanic and black drivers have a greatest search rates by far. However, black drivers have the second greatest hit rate and Hispanic drivers have the fifth greatest hit rate. Hispanic drivers, especially, are being over-searched while having the lowest hit rate.

Race	Stops	Searches	Hits	Search rate	Hit rate
Asian	111,758	2,562	339	2.29%	13.23%
Black	796,114	64,810	12,507	8.14%	19.30%
Hispanic	360,294	31,877	3,977	8.85%	12.48%
Other	9,382	313	45	3.34%	14.38%
White	3,437,483	89,166	21,865	2.59%	24.52%

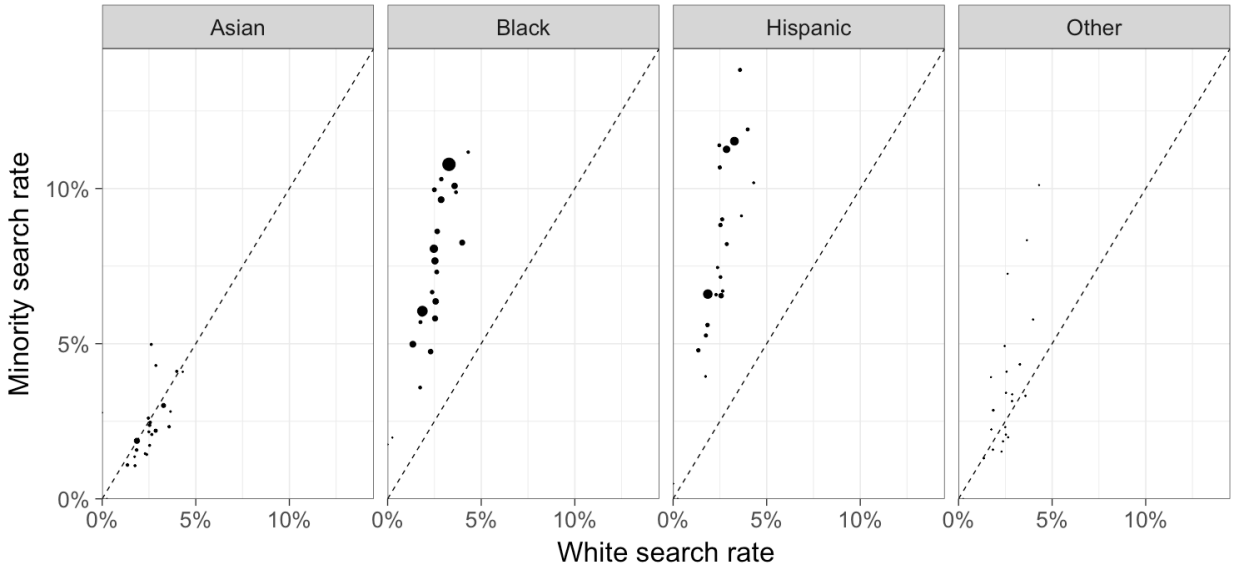
In the below table, we control for Illinois State Police district and look at the differences in race. District 3 covers Chicago. With a population of roughly 2.7 million people, Chicago is the third most populated city in the United States and is 45% white. Known for its racial diversity, we see that black drivers are stopped the most, followed by white drivers. Hispanic and black drivers have the greatest search rates, though white drivers have the greatest hit rate.

Race	District	Stops	Searches	Hits	Search rate	Hit rate
Asian	3	16,653	501	46	3.01%	9.18%
Black	3	201,600	21,731	2,674	10.78%	12.31%
Hispanic	3	74,840	8,626	830	11.53%	9.62%
Other	3	1,223	53	9	4.33%	16.98%
White	3	180,546	5,915	779	3.28%	13.17%

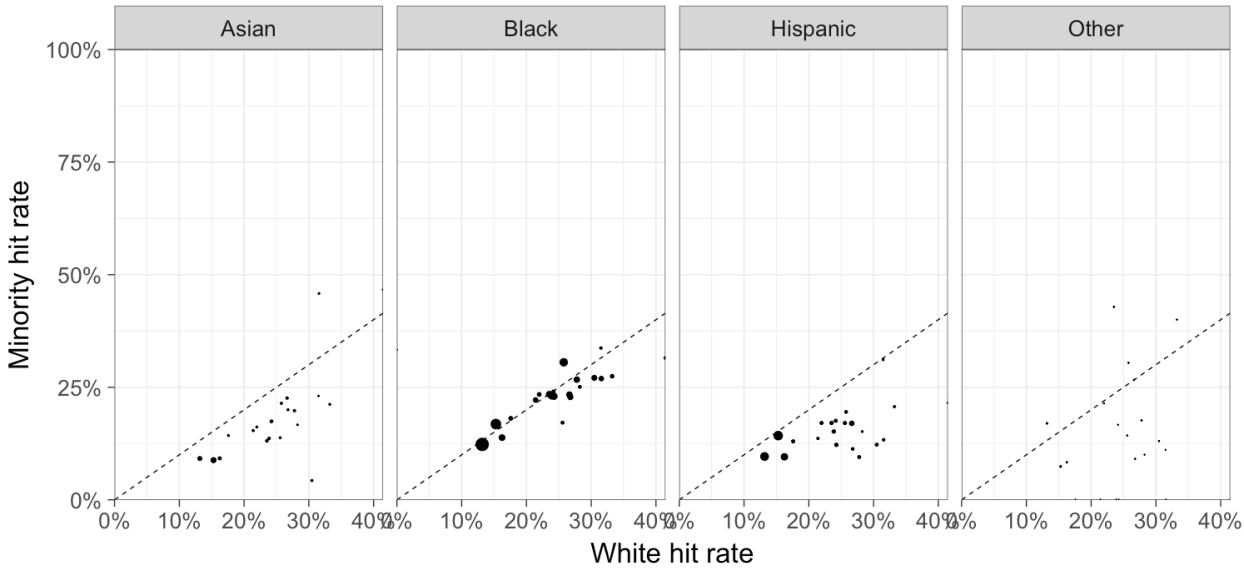
This time, we control for district 19, which covers the city Carmi. Carmi has a population of nearly 5,000 people and is 98% white. As such, white drivers are stopped significantly more than other races. Interestingly, the search rates in Carmi are much lower than those in Chicago (peaks reaching 3.94% versus 11.53%), but their hit rates are much higher (46.67% versus 16.98%). Though the sample size of Carmi is smaller due to its low population, this suggests that Carmi is acting more efficiently with their searches. Their search rate is low, and their contraband yield is high.

Race	District	Stops	Searches	Hits	Search rate	Hit rate
Asian	19	1,104	15	7	1.36%	46.67%
Black	19	6,638	238	75	3.59%	31.51%
Hispanic	19	1,648	65	14	3.94%	21.54%
Other	19	102	4	0	3.92%	0.00%
White	19	168,490	2,923	1,212	1.73%	41.46%

The below figure analyzes minority search rates on white search rates. An equal search rate to white drivers is represented by the 45-degree dotted line. The search rates are stratified by location where the points are sized proportional to the total number of stops in a given district, covering all 22 districts. While Asian drivers seem to be searched at similar rates of white drivers, black and Hispanic drivers are consistently searched more often.



While black and Hispanic drivers are searched at greater rates than white drivers, black drivers appear to have a fairly similar hit rate to white drivers and Hispanics have a consistently lower hit rate. As such, by searching black and Hispanic drivers at a greater rate but not finding contraband in their vehicles, police officers are inefficiently searching vehicles and not optimizing their contraband yield. This over-searching of some minorities is what we hope to address with our predictive model.



Data Cleaning

For the purpose of building a predictive model that could easily be used by law enforcement, we began by changing all of our variables to dummy variables. Easy to use, police officers can simply plug in “yes” (1) or “no” (0) to allow the machine to predict the likelihood of contraband in a vehicle.

We transformed *fine_grained_location* into 22 dummy variables, with possible values 0 and 1, representing the 22 Illinois State Police districts: *IL_fip01* to *IL_fip22*. *IL_fips04* was excluded since there is no police district 4. Using police districts gives us the opportunity to account for differences in location. By doing so, we could remove columns *state*, *location_raw*, *county_name*, *county_fips*, *police_department*, and *district*. Next, we turned *driver_gender* into dummy variables *IL_F* and *IL_M*. *driver_age* was transformed into 20 five-year age group dummy variables, spanning *IL_agecat1* to *IL_agecat20*. These age groups cover ages 1-100. A 21st age group was added, *IL_agecatNA*, for drivers above 100 years old. Columns *IL_agecat1*, *IL_agecat2*, and *IL_agecat3* were dropped due to them covering ages 0-5, 6-10, and 11-15. The legal driving age is 16, so *IL_agecat4* is the first age group. With the transformation of *driver_age*, *driver_age_raw* was dropped.

Our next manipulation includes changing *stop_date*, formatted YYYY-MM-DD, to days of the week. We developed seven dummy variables titled *IL_weekday1* through *IL_weekday7* to denote Sunday through Saturday. *stop_time* was then dropped from analysis. After, we transformed *driver_race* into five dummy variables named *IL_raceAsian*, *IL_raceBlack*, *IL_raceHispanic*, *IL_raceOther*, and *IL_raceWhite*. *driver_race_raw* was then dropped. Then, we changed *violation* into eight dummy variables called *IL_Equipment*, *IL_License*, *IL_Moving violation*, *IL_Other*, *IL_Registration/plates*, *IL_Safe movement*, *IL_Seat belt*, and *IL_Speeding*. *violation_raw* was dropped and the transformed *violation* factor was used as the reason for a vehicle being stopped.

Lastly, we filtered for *search_conducted* = "TRUE" so that we could train and test our predictive model on data in which we knew the outcome, that being contraband found or not found during a search. The factors that were transformed include known information from before a search is conducted. In this way, a police officer can input the information into the predictive model to produce a likelihood of finding contraband. Factors that are later excluded from analysis include information known after a search is conducted, as well as redundant variables and data that is too messy to cleanly transform. For example, *vehicle_type* is dropped since it has too many inconsistent observations (some include the year of the vehicle, others include the make of the vehicle). Our data cleaning is summarized below and transitions us into the building of the models.

Dependent factor
contraband_found

Tranformed factors	**Dummies created**
fine_grained_location	*IL_fips01, IL_fips02, IL_fips03, IL_fips05, IL_fips06, IL_fips07, IL_fips08, IL_fips09, IL_fips10, IL_fips11, IL_fips12, IL_fips13, IL_fips14, IL_fips15, IL_fips16, IL_fips17, IL_fips18, IL_fips19, IL_fips20, IL_fips21, IL_fips22*
driver_gender	*IL_F, IL_M*
driver_age	*IL_agecat4, IL_agecat5, IL_agecat6, IL_agecat7, IL_agecat8, IL_agecat9, IL_agecat10, IL_agecat11, IL_agecat12, IL_agecat13, IL_agecat14, IL_agecat15, IL_agecat16, IL_agecat17, IL_agecat18, IL_agecat19, IL_agecat20, IL_agecatNA*
stop_date	*IL_weekday1, IL_weekday2, IL_weekday3, IL_weekday4, IL_weekday5, IL_weekday6, IL_weekday7*
driver_race	*IL_raceAsian, IL_raceBlack, IL_raceHispanic, IL_raceOther, IL_raceWhite*
violation	*IL_Equipment, IL_License, IL_Moving violation, IL_Other, IL_Registration/plates, IL_Safe movement, IL_Seat belt, IL_Speeding*

Filter factor
search_conducted = "TRUE"

Dropped factors
id, state, location_raw, county_name, county_fips, police_department, district, driver_age_raw, stop_time, driver_race raw, violation_raw, search_type_raw, search_type, stop_outcome, is_arrested, stop_duration, vehicle_type, drug_related_stop

IV. Building the Models

Model preparation

The variables used in our models were the 61 dummies created in the previous section. These variables allow for objective input by police officers when deciding whether to search a vehicle or not. We used these variables and only these variables because we believe the models should remove nearly all subjectivity from decision making. These variables then became independent variables, in which *contraband_found* is the dependent variable.

After setting up our variables, we partitioned our data into 40% training data and 60% testing data. All of our model building and training is done on the training data using a random seed of 123. Once the models are finalized using the training data, their predictions are run against the testing data to analyze their accuracy.

Traditional econometric regressions

When choosing the predictive models to develop, we wanted to use a mix of traditional econometric analysis and newer machine learning algorithms. As such, it made sense to first begin our analysis with simple OLS and logit regressions. All of the analysis and output produced in this paper was done using R. R is a powerful open source statistical software that provides the greatest variety of machine learning techniques, which allowed for the most flexibility when building our models.

Below are the results of our OLS regression. All of the *violation* and *weekday* factors have significance at the 1% level. Most of the *fips* factors have significance and some of the *race* ones do. Interestingly, we do not see significance in any of the *age* categories.

OLS Regression						
	Term	**Estimate**	**Std. error**	**Statistic**	**P value**	
1	(Intercept)	0.1257	(0.0712)	1.7668	0.0773	*
2	IL_M	0.0108	(0.0038)	2.8477	0.0044	***
3	IL_Equipment	0.1193	(0.0220)	5.4348	0.0000	***
4	IL_License	0.1363	(0.0233)	5.8512	0.0000	***
5	IL_Speeding	0.1314	(0.0220)	5.9861	0.0000	***
6	IL_Moving.violation	0.1295	(0.0219)	5.9102	0.0000	***
7	IL_Registration.plates	0.0558	(0.0226)	2.4686	0.0136	***
8	IL_Safe.movement	0.1554	(0.0222)	6.9994	0.0000	***
9	IL_Seat.belt	0.1001	(0.0269)	3.7269	0.0002	***
10	IL_raceAsian	-0.0184	(0.0371)	-0.4969	0.6193	
11	IL_raceBlack	0.0596	(0.0351)	1.6984	0.0894	*
12	IL_raceHispanic	0.0004	(0.0352)	0.0118	0.9906	
13	IL_raceWhite	0.0646	(0.0351)	1.8414	0.0656	*
14	IL_fip01	-0.0622	(0.0147)	-4.2291	0.0000	***
15	IL_fip02	-0.1426	(0.0112)	-12.7499	0.0000	***
16	IL_fip03	-0.1530	(0.0104)	-14.6590	0.0000	***
17	IL_fip05	-0.0336	(0.0123)	-2.7332	0.0063	***
18	IL_fip06	-0.0535	(0.0131)	-4.0904	0.0000	***
19	IL_fip07	-0.0745	(0.0128)	-5.8073	0.0000	***
20	IL_fip08	-0.0786	(0.0126)	-6.2188	0.0000	***
21	IL_fip09	-0.0417	(0.0114)	-3.6446	0.0003	***
22	IL_fip10	-0.0733	(0.0117)	-6.2530	0.0000	***
23	IL_fip11	-0.0080	(0.0114)	-0.7055	0.4805	
24	IL_fip12	-0.0174	(0.0114)	-1.5221	0.1280	
25	IL_fip13	-0.0309	(0.0122)	-2.5439	0.0110	***
26	IL_fip14	-0.0015	(0.0134)	-0.1143	0.9090	
27	IL_fip15	-0.1250	(0.0107)	-11.7005	0.0000	***

28	IL_fip16	-0.1166	(0.0130)	-8.9864	0.0000	***
29	IL_fip17	-0.0429	(0.0131)	-3.2620	0.0011	***
30	IL_fip18	0.0092	(0.0135)	0.6818	0.4954	
31	IL_fip19	0.1120	(0.0146)	7.6534	0.0000	***
32	IL_fip20	-0.0316	(0.0127)	-2.4812	0.0131	***
33	IL_fip21	-0.0775	(0.0144)	-5.3915	0.0000	***
34	IL_weekday1	-0.0175	(0.0049)	-3.5858	0.0003	***
35	IL_weekday2	-0.0511	(0.0054)	-9.5002	0.0000	***
36	IL_weekday3	-0.0620	(0.0053)	-11.6062	0.0000	***
37	IL_weekday4	-0.0530	(0.0052)	-10.1872	0.0000	***
38	IL_weekday5	-0.0566	(0.0052)	-10.8647	0.0000	***
39	IL_weekday6	-0.0268	(0.0050)	-5.3908	0.0000	***
40	IL_agecat4	0.0794	(0.0572)	1.3884	0.1650	
41	IL_agecat5	0.0305	(0.0570)	0.5356	0.5922	
42	IL_agecat6	0.0054	(0.0570)	0.0953	0.9240	
43	IL_agecat7	-0.0042	(0.0571)	-0.0739	0.9411	
44	IL_agecat8	-0.0206	(0.0571)	-0.3614	0.7178	
45	IL_agecat9	-0.0176	(0.0572)	-0.3070	0.7588	
46	IL_agecat10	-0.0206	(0.0572)	-0.3590	0.7196	
47	IL_agecat11	-0.0296	(0.0574)	-0.5147	0.6068	
48	IL_agecat12	-0.0302	(0.0578)	-0.5221	0.6016	
49	IL_agecat13	-0.0195	(0.0585)	-0.3332	0.7389	
50	IL_agecat14	-0.0302	(0.0603)	-0.5012	0.6162	
51	IL_agecat15	-0.0929	(0.0658)	-1.4120	0.1580	
52	IL_agecat16	0.0169	(0.0764)	0.2205	0.8254	
53	IL_agecat17	0.0196	(0.0986)	0.1982	0.8429	
54	IL_agecat18	0.0123	(0.1507)	0.0816	0.9350	
55	IL_agecat19	-0.1682	(0.2348)	-0.7166	0.4736	
56	IL_agecat20	0.1321	(0.1708)	0.7733	0.4393	

In the below logit regression, we see similar coefficient significance levels to the OLS regression.

Logit Regression						
	Term	**Estimate**	**Std. error**	**Statistic**	**P value**	
1	(Intercept)	-2.4125	(0.5106)	-4.7245	0.0000	***
2	IL_M	0.0703	(0.0239)	2.9372	0.0033	***
3	IL_Equipment	1.1938	(0.2279)	5.2372	0.0000	***
4	IL_License	1.3028	(0.2336)	5.5764	0.0000	***
5	IL_Speeding	1.2744	(0.2279)	5.5919	0.0000	***
6	IL_Moving.violation	1.2606	(0.2278)	5.5349	0.0000	***
7	IL_Registration.plates	0.6560	(0.2326)	2.8204	0.0048	***
8	IL_Safe.movement	1.4192	(0.2288)	6.2043	0.0000	***
9	IL_Seat.belt	1.0646	(0.2506)	4.2486	0.0000	***
10	IL_raceAsian	-0.1889	(0.2694)	-0.7012	0.4832	
11	IL_raceBlack	0.4172	(0.2522)	1.6542	0.0981	*
12	IL_raceHispanic	-0.0417	(0.2532)	-0.1648	0.8691	
13	IL_raceWhite	0.4353	(0.2520)	1.7272	0.0841	*
14	IL_fip01	-0.3273	(0.0867)	-3.7728	0.0002	***
15	IL_fip02	-0.9110	(0.0690)	-13.1999	0.0000	***
16	IL_fip03	-1.0038	(0.0618)	-16.2329	0.0000	***
17	IL_fip05	-0.1579	(0.0708)	-2.2292	0.0258	***
18	IL_fip06	-0.2836	(0.0762)	-3.7210	0.0002	***
19	IL_fip07	-0.3997	(0.0760)	-5.2571	0.0000	***
20	IL_fip08	-0.4295	(0.0744)	-5.7708	0.0000	***
21	IL_fip09	-0.2159	(0.0656)	-3.2906	0.0010	***
22	IL_fip10	-0.3980	(0.0683)	-5.8265	0.0000	***
23	IL_fip11	-0.0288	(0.0648)	-0.4446	0.6566	
24	IL_fip12	-0.0924	(0.0650)	-1.4226	0.1549	
25	IL_fip13	-0.1597	(0.0695)	-2.2979	0.0216	***
26	IL_fip14	-0.0347	(0.0753)	-0.4608	0.6450	
27	IL_fip15	-0.7489	(0.0631)	-11.8726	0.0000	***
28	IL_fip16	-0.6754	(0.0803)	-8.4108	0.0000	***
29	IL_fip17	-0.2156	(0.0763)	-2.8263	0.0047	***
30	IL_fip18	0.0417	(0.0757)	0.5511	0.5815	
31	IL_fip19	0.4834	(0.0792)	6.1021	0.0000	***
32	IL_fip20	-0.1690	(0.0728)	-2.3213	0.0203	***
33	IL_fip21	-0.4223	(0.0864)	-4.8905	0.0000	***

34	IL_weekday1	-0.0972	(0.0299)	-3.2524	0.0011	***
35	IL_weekday2	-0.3154	(0.0348)	-9.0614	0.0000	***
36	IL_weekday3	-0.3919	(0.0350)	-11.2109	0.0000	***
37	IL_weekday4	-0.3243	(0.0333)	-9.7377	0.0000	***
38	IL_weekday5	-0.3497	(0.0335)	-10.4377	0.0000	***
39	IL_weekday6	-0.1556	(0.0307)	-5.0724	0.0000	***
40	IL_agecat4	0.4372	(0.3777)	1.1577	0.2470	
41	IL_agecat5	0.1860	(0.3768)	0.4938	0.6215	
42	IL_agecat6	0.0352	(0.3769)	0.0935	0.9255	
43	IL_agecat7	-0.0283	(0.3772)	-0.0751	0.9402	
44	IL_agecat8	-0.1442	(0.3776)	-0.3820	0.7024	
45	IL_agecat9	-0.1175	(0.3779)	-0.3110	0.7558	
46	IL_agecat10	-0.1315	(0.3784)	-0.3475	0.7282	
47	IL_agecat11	-0.1920	(0.3795)	-0.5058	0.6130	
48	IL_agecat12	-0.1954	(0.3819)	-0.5117	0.6089	
49	IL_agecat13	-0.1333	(0.3866)	-0.3448	0.7303	
50	IL_agecat14	-0.1923	(0.3988)	-0.4822	0.6297	
51	IL_agecat15	-0.7042	(0.4580)	-1.5375	0.1242	
52	IL_agecat16	0.0931	(0.4881)	0.1908	0.8487	
53	IL_agecat17	0.1239	(0.6081)	0.2037	0.8386	
54	IL_agecat18	0.0592	(0.9109)	0.0650	0.9481	
55	IL_agecat19	-8.9754	(67.4322)	-0.1331	0.8941	
56	IL_agecat20	0.7262	(0.9614)	0.7554	0.4500	

Machine learning algorithms

For the purpose of building a predictive model, we wanted to utilize more complex machine learning algorithms that specialize in predictive statistics. We chose three machine learning algorithms to focus on: ridge regression, lasso regression, and random forest. These three are commonly used and work well with categorical variables.

Ridge and lasso regression are both types of regularized linear regressions. Ridge regression works to lower variance by shrinking the coefficients towards zero through the use of a ridge constraint. Lasso (least absolute shrinkage and selection operator) regression both shrinks the parameter estimates and does variable selection, where it has the potential to set variables to zero and remove them from analysis. The random forest algorithm builds an ensemble of decision trees, which it merges together to get the most accurate prediction.

All of our R code to build and run the five models can be found in the Appendix. The code appears in the order of OLS, logit, ridge, lasso, and random forest.

V. Testing the Models

Comparing the models

To test the five models we built, we use the predictions in the training to develop predictions in the testing data. These predictions are then compared to the actuals in the testing data. For comparison purposes, we opted using the misclassification rate of each model. Misclassification rate is easy to understand, as it is the number of incorrect predictions divided by the number of total predictions. Other comparison methods include analyzing RMSE (root mean square error), though it has complex interpretation when comparing it across the five different models. The below table displays the misclassification rate of each of the models sorted from low to high. Our misclassification rates were determined based off deciding to search a vehicle if the predicted likelihood of contraband is equal to or greater than 0.50 (on scale 0 to 1).

Misclassification rates	
OLS	0.2056751
Ridge	0.2056927
Lasso	0.2057192
Logit	0.2057457
Random forest	0.2235234

The OLS regression performs best when it comes to misclassification rate, while the random forest algorithm performs worse. OLS, ridge, lasso, and logit all have similar errors rates and random forest is very different. Next, we looked at the classification table of each of the five models to understand how they made their predictions, as well as the type of errors they suffered

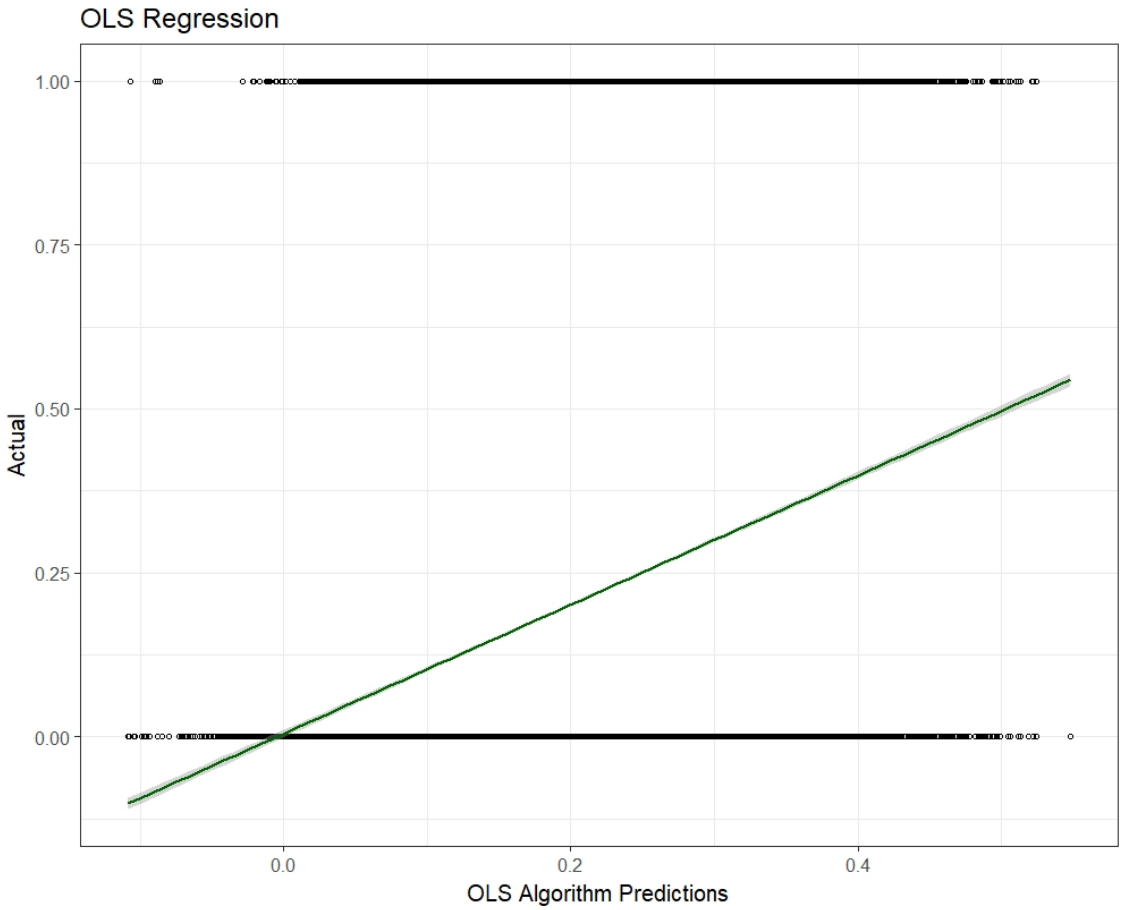
from. In the table, 1 is equivalent contraband in the vehicle, whether that be by prediction or actual. No contraband is represented by 0.

When predicted and actual both equal 0, the model correctly predicted that there was no contraband in the vehicle. When predicted and actual both equal 1, the model correctly predicted that there was contraband in the vehicle. A type I error is an incorrect rejection of the null hypothesis. In this case, it means that we predict that there is contraband (1), but there actually was not contraband (0). A type II error is a failure to reject a false null hypothesis. In this case, we predict that there is no contraband (0), but there actually is contraband in the vehicle (1).

OLS		Actual	
		0	*1*
Predicted	*0*	89,909	23,260
	1	29	34

The above table displays the classifications from the OLS regression. This regression acts very conservatively, as it rarely predicts anyone has contraband. Since our analysis was done only on vehicles that were searched, the model correctly predicted that roughly 90,000 of the 113,000 vehicles that were searched had no contraband in them. As such, the model says that these vehicles should not have been searched, though they were in reality. Furthermore, use of this model could have stopped nearly 90,000 contraband-free searches from being conducted.

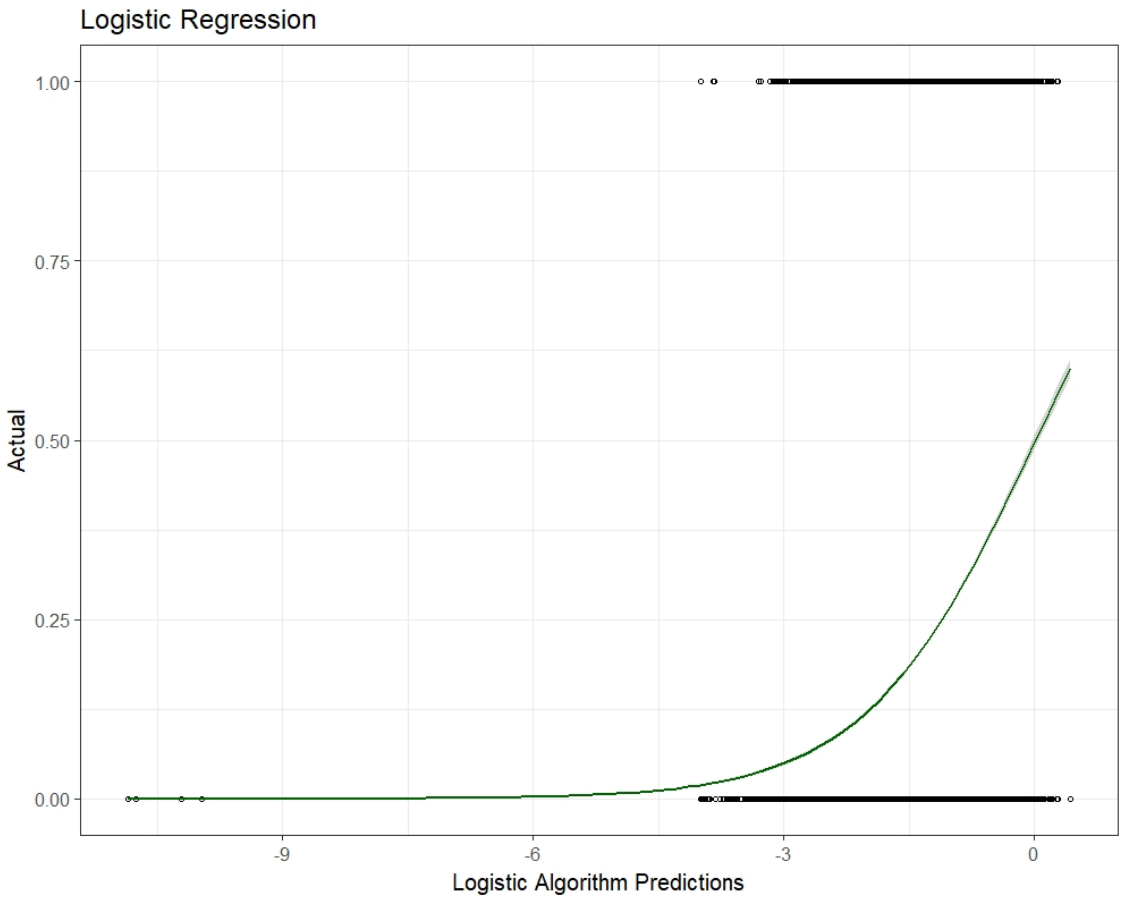
On the other hand, the model also predicted that 23,000 vehicles had no contraband in them, though they actually did. This type II error results in 23,000 people getting away with contraband who would have been stopped. There is certainly a trade-off between not searching 90,000 innocent people or letting 23,000 get away. This question should be posed to policy makers who can decide what is prioritized and the implications of such decisions. Searching 90,000 innocent people can lead to public outcry for reform and justice, as we have seen recently in the United States, as well as the potential for lawsuits. Failing to search 23,000 guilty people means that a high amount of contraband would still circulate in the state of Illinois and the greater United States.



The above plot displays the OLS predictions against the actual values. It appears the OLS predictions barely go over 0.50, which was our determinant for searching a vehicle (greater than 50% chance of contraband). In line with the misclassification table, we see that most predictions from the OLS model are near 0.

Logit		Actual	
		0	*1*
Predicted	*0*	89,844	23,203
	1	94	91

The misclassification table for the logit regression is similar to the OLS regression, but it predicts more vehicles with contraband. The increased wrongly predicted vehicles outweigh the benefit of the new correct predictions. This leads to the logistic regression having the fourth best misclassification rate.



The logit regression plot predictions reach roughly 0.60 at their peak. The plot also displays outliers that may be weighing the logit regression down.

Ridge		Actual	
		0	*1*
Predicted	*0*	89,916	23,269
	1	22	25

The ridge algorithm performs slightly worse than the OLS regression when comparing the misclassification rates. It predicts just 16 less cases of contraband.

Lasso		Actual	
		0	*1*
Predicted	*0*	89,938	23,294
	1	0	0

The lasso regression was third best by misclassification rate and interestingly predicts no cases of contraband.

Random forest		Actual	
		0	*1*
Predicted	*0*	86,283	21,655
	1	3,655	1,639

Lastly, the random forest performs worst by far. It predicts over 5,000 more cases of contraband in the vehicle, of which over 3,600 of them are incorrectly predicted.

How accurate is our model?

Our model is more accurate than actual procedures, though that does not necessarily mean it is better. Of the nearly 113,000 vehicles that police officers actually searched, only 23,000 of them contained contraband. This is a hit rate of 20% and misclassification rate of 80%.

Our most accurate model, the OLS regression, has a hit rate of nearly 0% and a misclassification rate of 20%. Though significantly more accurate than actual police searches, it fails to find the contraband that police officers are searching for. As discussed earlier, policy and decision

makers need to decide what is more important: accuracy of searches or over-searching to lower contraband circulation.

Are the police really making biased decisions?

Yes. Police officers are absolutely making biased decisions. Our model attempts to eliminate subjectively from vehicle searches by including only objective facts: vehicle violation, driver gender, driver race, driver age, location of stop, and day of the week. One of the questions we continued to ask ourselves as we developed our models and wrote this paper is, "How can we predict contraband in a vehicle *only* based off these factors?" Especially since most are demographic factors, it seemed we were missing police officer intuition and other implicit factors that appear before a search.

Some variables missing from our models include whether the police officer saw contraband in plain sight inside the vehicle, whether the driver or vehicle smelled of contraband, details of the exchange between the police officer and the driver, etc. We assume factors like these may have strong influence on a police officer's decision to search a vehicle and though they may add varying amounts of subjectivity and bias, they should be considered for inclusion into the model. Police officer bias is not necessarily bad, but their subjectively needs to be further analyzed, with certain aspects of it added to the model.

VI. Conclusion and Future Considerations

To conclude, we developed a model that predicts the likelihood of finding contraband in a stopped vehicle, thereby telling a police officer whether they should search a vehicle or not. This model was purposely built how it was so police officers can objectively input "yes" (1) or "no" (0) into it for a set of factors, thus removing any subjectivity they may have. In this way, the goal was to see if we could build an accurate and useful predictive model that did not require law enforcement intuition and bias.

In the end, the best model (OLS regression) predicts contraband findings with a misclassification rate of 20.569% using variables: vehicle violation, driver gender, driver race, driver age, location of stop, and day of the week. This is significantly lower than the actual misclassification rate of nearly 80%. Though the produced model is more accurate, it fails to select the 20% of searched drivers who actually had contraband. Put another way, this model specializes in efficiency and accuracy by searching nearly no vehicles. Policy and decision makes need to understand what is important to them, whether that is minimizing searches and making sure to not search those without contraband, or potentially over-searching and maximizing the contraband yield.

Furthermore, we conclude that our model suffers from a lack of police intuition. There are possibly unobservable factors omitted from our models that only a police's eyes and ears can catch. The model succeeds in its ease of use and should be built on with additional variables by law enforcement departments and other interested stakeholders. One major variable that should be better understood and potentially modified for future use is the *search_type* variable. If a police officer can input into the model that they have reasonable suspicion or probable cause

before a search is conducted, these inputs could have significant impacts on the likelihood of finding contraband. However, this variable, in its current use, seems to be too connected to the event of the search happening and does not act as an early predictor like we would prefer for the model. A second variable that is currently accessible for use is the *stop_time* variable. It is in HH:MM format but could be manipulated to different time of day buckets such as morning, afternoon, evening, and late night. We hypothesize that these dummies may be significant on the contraband outcome.

Beyond the constraints of the current model, a future study should interview police officers to understand their reasoning and thought process when it comes to searching vehicles. The interviews could reveal some interesting variables that are easily implementable into the model, while not adding a large amount of bias. In addition, it would be interesting to understand how the model training time period impacts the prediction results and misclassification rate. We used data from 2004 to 2015. During that period in the United States, we have witnessed the Great Recession and seen police tension with the public become more prevalent. It may make sense to train and test the model on only 2011 and beyond, as well as see how earlier years predict differently.

Lastly, a future study should use a computer with more computing power. We were restricted in the packages we could run through R and the analysis we could do. At one point, a package told us we needed 20,000 gigabytes of memory to run it. Some machine learning experts recommend grouping various combinations of observations to lower the needed computing power, so this should be considered as well. Our computer was nearly at its limit when running the random

forest algorithm, which is supposed to be an easier and faster one to run. Greater computing power would allow for more intensive and complex R packages, such as the caret package for machine learning algorithms.

Restating our conclusion, we developed a predictive model that tells police officers whether or not they should search a vehicle for contraband, based on the likelihood of finding contraband. The model is significantly more accurate than current search procedures and results in more efficiency among police officers. It addresses any over-searching problems but needs additional variables that have positive effects on contraband findings, since the model misses a large number of vehicles containing contraband. With these future considerations taken into account, a future study can develop a more robust model and go further to supplement police officers while out in the field.

VII. Appendix

Figure 1: Variable descriptions

Column name	Column meaning	Example value
id	The unique ID we assign to each stop. Contains the state and year.	VT-2011-00012
state	The two-letter code for the state in which the stop occurred.	VT
stop_date	The date of the stop, in YYYY-MM-DD format. Some states do not provide the exact stop date: for example, they only provide the year or quarter in which the stop occurred. For these states, stop_date is set to the date at the beginning of the period: for example, January 1 if only year is provided.	11/27/11
stop_time	The 24-hour time of the stop, in HH:MM format.	20:15
location_raw	The original data value from which we compute the county (or comparably granular location) in which the stop occurred. Not in a standardized format across states.	Winooski
county_name	The standardized name of the county in which the stop occurred.	Chittenden County
county_fips	The standardized 5-digit FIPS code in which the stop occurred.	50007
district	In several states (e.g., Illinois) the stop county cannot be inferred, but a comparably granular location can. This comparably granular location is stored in the district column. Most states do not have this column.	ILLINOIS STATE POLICE 01
fine_grained_location	Any higher-resolution data about where the stop occurred: e.g., milepost or address. Not standardized across states.	90400 I 89 N; EXIT 15 MM90/40
police_department	The police department or agency that made the stop. Not in a standard format across states.	WILLISTON VSP
driver_gender	The driver's gender, as recorded by the trooper. M, F, or NA.	M
driver_age_raw	The original data value from which we compute the driver's age when they were stopped. May be age, birth year, or birth date. Not in a standard format across states.	1988
driver_age	The driver's age when they were stopped. Set to NA if less than 15 or greater than or equal to 100.	23

driver_race_raw	The original data value from which the driver's standardized race is computed. Not in a standard format across states.	African American
driver_race	The standardized driver race. Possible values are White, Black, Hispanic, Asian, Other, and NA, with NA denoting values which are unknown. Asian refers to Asian, Pacific Islander, and Indian. Native Americans/American Indians are included in the "other" category. Anyone with Hispanic ethnicity is classified as Hispanic, regardless of their recorded race.	Black
violation_raw	The violation committed by the driver, in the language of the original data. Not in a standard format across states. Some stops have multiple violations.	Speeding (10–19 MPH Over Prima Facie Limit *)
violation	The violation committed by the driver, standardized into categories which are consistent across states.	Speeding
search_conducted	A TRUE/FALSE value indicating whether a search was performed.	TRUE
search_type_raw	The justification for the search, in the language of the original data. NA if no search was performed. Not in a standard format across states. Some states have multiple justifications for a search.	CONSENT SEARCH CONDUCTED
search_type	The normalized justification for the search. Where possible, this is standardized into categories which are consistent across states. For example, if something is clearly a consent search, search_type is referred to as "Consent".	Consent
contraband_found	A TRUE/FALSE value indicating whether a search was performed and contraband was found. FALSE if no search was performed.	TRUE
stop_outcome	The outcome of the stop. Many states have idiosyncratic outcomes — for example, "CHP 215" in California — so this column is not standardized across states. "Citation" and "Warning" are the values which occur most commonly across states. If the stop has multiple outcomes, the most severe outcome is used. For example, if a stop resulted in a citation and a warning, stop_outcome would be "Citation".	Citation
is_arrested	A TRUE/FALSE value indicating whether an arrest was made.	TRUE

Figure 2: R code for predictive model

```
#Set-up
#Library
library(tidyverse)
library(lubridate)
library(dummies)
library(dplyr)

# read data
IL <- read.csv("C:/Users/--------------/THESIS/IL_cleaned.csv")

#Day of the week (1 = sunday, 2 = monday, 3 = tuesday, 4 =
wednesday, 5 = thursday, 6 = friday, 7 = saturday)
IL <- cbind(IL, wday(IL$stop_date))

# Rename a column in R
colnames(IL)[colnames(IL)=="wday(IL$stop_date)"] <- "weekday"

#Continuous Var to categorical (age) (0-5=1, 5-10=2, .....95-100
= 20)
IL$Agecats<-cut(IL$driver_age, seq(0,100,5), right=FALSE,
labels=c(1:20))

#recode contraband found
IL$contraband_found <- as.numeric(IL$contraband_found)

# Dummy variable code
IL <- cbind(IL, dummy(IL$driver_gender, sep="_"))
IL <- cbind(IL, dummy(IL$violation,sep = "_"))
IL <- cbind(IL, dummy(IL$driver_race,sep = "_race"))
IL <- cbind(IL, dummy(IL$fine_grained_location,sep = "_fip"))
IL <- cbind(IL, dummy(IL$weekday,sep = "_weekday"))
IL <- cbind(IL, dummy(IL$Agecat,sep = "_agecat"))

#Filtering out Errors
IL <- IL %>%
  filter(search_conducted == "TRUE", IL_fip == "0", IL_fip00 ==
"0", IL_fip04 == "0", IL_fipZ2 == "0")
# Summarize data
summary_stats <- function(search_conducted, contraband_found) {
  n_stops     = length(search_conducted)
  n_searches  = sum(search_conducted)
  n_hits      = sum(contraband_found)
  search_rate = n_searches / n_stops
  hit_rate    = n_hits / n_searches
```

```
  return(data.frame(n_stops, n_searches, n_hits, search_rate,
hit_rate))}

# Summary stats by race
basic_summary_statistics_by_race = IL %>%
  group_by(driver_race) %>%
  do(summary_stats(.$search_conducted, .$contraband_found))
basic_summary_statistics_by_race

# Summary stats by gender
basic_summary_statistics_by_gender = IL %>%
  group_by(driver_gender) %>%
  do(summary_stats(.$search_conducted, .$contraband_found))
basic_summary_statistics_by_gender

# Summary stats by race and fips
basic_summary_statistics_by_race_and_fips = IL %>%
  filter(!is.na(fine_grained_location)) %>%
  group_by(driver_race, fine_grained_location) %>%
  do(summary_stats(.$search_conducted, .$contraband_found))
basic_summary_statistics_by_race_and_fips

# Scatter plot
data_for_plot <- basic_summary_statistics_by_race_and_fips %>%
  filter(driver_race == 'White') %>%
  right_join(basic_summary_statistics_by_race_and_fips %>%
filter(driver_race != 'White'), by='fine_grained_location')

# Plot search rates
max_val =
max(basic_summary_statistics_by_race_and_fips$search_rate) *
1.05
search_plot = ggplot(data_for_plot) +
  # Specify data we want to plot
  geom_point(aes(x = search_rate.x, y = search_rate.y, size =
n_stops.y)) +
  # Make one subplot for each minority race group
  facet_grid(.~driver_race.y) +
  # Add a diagonal line to indicate parity
  geom_abline(slope = 1, intercept = 0, linetype='dashed') +
  scale_x_continuous('White search rate', limits=c(0, max_val),
labels = scales::percent, expand=c(0,0)) +
  scale_y_continuous('Minority search rate', limits=c(0,
max_val), labels = scales::percent, expand=c(0,0)) +
  theme_bw(base_size=15)
  theme(legend.position="none") +
  scale_size_area(max_size=5)
```

```
search_plot

# Plot hit rates
max_val =
max(basic_summary_statistics_by_race_and_fips$hit_rate) * 1.05
hit_plot = ggplot(data_for_plot) +
  geom_point(aes(x = hit_rate.x, y = hit_rate.y, size =
n_stops.y)) +
  facet_grid(.~driver_race.y) +
  geom_abline(slope = 1, intercept = 0) +
  scale_x_continuous('White hit rate', limits=c(0, max_val),
labels = scales::percent, expand=c(0,0)) +
  scale_y_continuous('Minority hit rate', limits=c(0, max_val),
labels = scales::percent, expand=c(0,0)) +
  theme_bw(base_size=15) +
  theme(legend.position="none") +
  scale_size_area(max_size=5)
hit_plot
#Creating Test and Training Groups From Final Data Set
set.seed(123)
smp_size <- floor(0.4 * nrow(ILregMaster))
train_ind <- sample(seq_len(nrow(ILregMaster)), size = smp_size)

train <- ILregMaster[train_ind, ]
test <-ILregMaster[-train_ind, ]
#OLS Regression
OLS <- lm(contraband_found ~ (IL_M + IL_Equipment + IL_License +
IL_Speeding + IL_Moving.violation + IL_Registration.plates +
IL_Safe.movement + IL_Seat.belt +  IL_Other+ IL_raceAsian +
IL_raceBlack + IL_raceHispanic + IL_raceWhite + IL_raceOther +
IL_fip + IL_fip00 + IL_fip01 + IL_fip02 + IL_fip03 + IL_fip04 +
IL_fip05 + IL_fip06 + IL_fip07 + IL_fip08 + IL_fip09 + IL_fip10
+ IL_fip11 + IL_fip12 + IL_fip13 + IL_fip14 + IL_fip15 +
IL_fip16 + IL_fip17 + IL_fip18 + IL_fip19 + IL_fip20 + IL_fip21
+ IL_fip22 + IL_fipZ2 + IL_weekday1 + IL_weekday2 + IL_weekday3
+ IL_weekday4 + IL_weekday5 + IL_weekday6 + IL_weekday7 +
IL_agecat4 + IL_agecat5 + IL_agecat6 + IL_agecat7 + IL_agecat8 +
IL_agecat9 + IL_agecat10 + IL_agecat11 + IL_agecat12 +
IL_agecat13 + IL_agecat14 + IL_agecat15 + IL_agecat16 +
IL_agecat17 + IL_agecat18 + IL_agecat19 + IL_agecat20 +
IL_agecatNA), data=train)
summary (OLS)
#Predicted values based on OLS Regression
OLS.pred <- predict.lm(OLS,test)
#Making Predictions based on 50% Probability
OLS.pred[OLS.pred >= 0.5] <- 1
OLS.pred[OLS.pred < 0.5] <- 0
```

```
#Misclassification Rate
OLS.class <- mean(OLS.pred != test$contraband_found)
#Sorting Type I and Type II Errors
table(OLS.pred, test$contraband_found)

#Logistic Regression
logit <- glm(contraband_found ~ (IL_M + IL_Equipment +
IL_License + IL_Speeding + IL_Moving.violation +
IL_Registration.plates + IL_Safe.movement + IL_Seat.belt +
IL_Other+ IL_raceAsian + IL_raceBlack + IL_raceHispanic +
IL_raceWhite + IL_raceOther + IL_fip + IL_fip00 + IL_fip01 +
IL_fip02 + IL_fip03 + IL_fip04 + IL_fip05 + IL_fip06 + IL_fip07
+ IL_fip08 + IL_fip09 + IL_fip10 + IL_fip11 + IL_fip12 +
IL_fip13 + IL_fip14 + IL_fip15 + IL_fip16 + IL_fip17 + IL_fip18
+ IL_fip19 + IL_fip20 + IL_fip21 + IL_fip22 + IL_fipZ2 +
IL_weekday1 + IL_weekday2 + IL_weekday3 + IL_weekday4 +
IL_weekday5 + IL_weekday6 + IL_weekday7 + IL_agecat4 +
IL_agecat5 + IL_agecat6 + IL_agecat7 + IL_agecat8 + IL_agecat9 +
IL_agecat10 + IL_agecat11 + IL_agecat12 + IL_agecat13 +
IL_agecat14 + IL_agecat15 + IL_agecat16 + IL_agecat17 +
IL_agecat18 + IL_agecat19 + IL_agecat20 + IL_agecatNA),
data=train, family = binomial())
#making Predictions based on LOGIT
logit.pred <- predict.glm(logit,test)
#Making Predictions based on 50% Probability
logit.pred[logit.pred >= 0.5] <- 1
logit.pred[logit.pred < 0.5] <- 0
#Misclassification Rate
logit.class <- mean(logit.pred != test$contraband_found)
#Sorting Type I and Type II Errors
table(logit.pred, test$contraband_found)
#Set-up for Machine Learning Algorithms
library(glmnet)
library(randomForest)
x <- model.matrix(contraband_found ~ (IL_M + IL_Equipment +
IL_License + IL_Speeding + IL_Moving.violation +
IL_Registration.plates + IL_Safe.movement + IL_Seat.belt +
IL_Other+ IL_raceAsian + IL_raceBlack + IL_raceHispanic +
IL_raceWhite + IL_raceOther + IL_fip + IL_fip00 + IL_fip01 +
IL_fip02 + IL_fip03 + IL_fip04 + IL_fip05 + IL_fip06 + IL_fip07
+ IL_fip08 + IL_fip09 + IL_fip10 + IL_fip11 + IL_fip12 +
IL_fip13 + IL_fip14 + IL_fip15 + IL_fip16 + IL_fip17 + IL_fip18
+ IL_fip19 + IL_fip20 + IL_fip21 + IL_fip22 + IL_fipZ2 +
IL_weekday1 + IL_weekday2 + IL_weekday3 + IL_weekday4 +
IL_weekday5 + IL_weekday6 + IL_weekday7 + IL_agecat4 +
IL_agecat5 + IL_agecat6 + IL_agecat7 + IL_agecat8 + IL_agecat9 +
IL_agecat10 + IL_agecat11 + IL_agecat12 + IL_agecat13 +
```

```
IL_agecat14 + IL_agecat15 + IL_agecat16 + IL_agecat17 +
IL_agecat18 + IL_agecat19 + IL_agecat20 + IL_agecatNA),
train)[,-1]
xtest <- model.matrix(contraband_found ~ (IL_M + IL_Equipment +
IL_License + IL_Speeding + IL_Moving.violation +
IL_Registration.plates + IL_Safe.movement + IL_Seat.belt +
IL_Other+ IL_raceAsian + IL_raceBlack + IL_raceHispanic +
IL_raceWhite + IL_raceOther + IL_fip + IL_fip00 + IL_fip01 +
IL_fip02 + IL_fip03 + IL_fip04 + IL_fip05 + IL_fip06 + IL_fip07
+ IL_fip08 + IL_fip09 + IL_fip10 + IL_fip11 + IL_fip12 +
IL_fip13 + IL_fip14 + IL_fip15 + IL_fip16 + IL_fip17 + IL_fip18
+ IL_fip19 + IL_fip20 + IL_fip21 + IL_fip22 + IL_fipZ2 +
IL_weekday1 + IL_weekday2 + IL_weekday3 + IL_weekday4 +
IL_weekday5 + IL_weekday6 + IL_weekday7 + IL_agecat4 +
IL_agecat5 + IL_agecat6 + IL_agecat7 + IL_agecat8 + IL_agecat9 +
IL_agecat10 + IL_agecat11 + IL_agecat12 + IL_agecat13 +
IL_agecat14 + IL_agecat15 + IL_agecat16 + IL_agecat17 +
IL_agecat18 + IL_agecat19 + IL_agecat20 + IL_agecatNA), test)[,-
1]
y <- train$contraband_found
lambda <- 10^seq(10, -2, length = 100)

#Ridge Regression
ridge <- glmnet(x, y, alph = 0, lambda = lambda)
#find the best lambda from our list via cross-validation
blambda <- cv.glmnet(x, y, alpha = 0)
bestlam <- blambda$lambda.min
#make predictions based on RIDGE
ridge.pred <- predict.glmnet(ridge, se.fit = FALSE, s = bestlam,
newx = xtest)
#Making Predictions based on 50% Probability
ridge.pred[ridge.pred >= 0.5] <- 1
ridge.pred[ridge.pred < 0.5] <- 0
#Misclassification Rate
ridge.class <- mean(ridge.pred != test$contraband_found)
#Sorting Type I and Type II Errors
table(ridge.pred, test$contraband_found)

#Lasso Regression
lasso <- glmnet(x, y, alpha = 1, lambda = lambda)
#make predictions based on LASSO
lasso.pred <- predict(lasso, s = bestlam, newx = xtest,
interval='confidence')
#Making Predictions based on 50% Probability
lasso.pred[lasso.pred >= 0.5] <- 1
lasso.pred[lasso.pred < 0.5] <- 0
#Misclassification Rate
```

```
lasso.class <- mean(lasso.pred != test$contraband_found)
#Sorting Type I and Type II Errors
table(lasso.pred, test$contraband_found)

#Random Forest Algorithm
rf <- randomForest(x, y)
rf.pred<-predict(rf, newdata = xtest, predict.all = TRUE)
#make predictions based on Random Forest
rf.pred.binary <- rf.pred$aggregate
#Making Predictions based on 50% Probability
rf.pred.binary[rf.pred.binary >= 0.5] <- 1
rf.pred.binary[rf.pred.binary < 0.5] <- 0
#Misclassification Rate
rf.class <- mean(rf.pred.binary != test$contraband_found)
#Sorting Type I and Type II Errors
table(rf.pred.binary, test$contraband_found)
#Summarizing Findings - Graphs
#Set-up
library(ggplot2)
library(RcolorBrewer)

ggplot(test, aes(OLS.pred, test$contraband_found)) +
  geom_point(shape=1) +
  geom_smooth(method=lm , color="dark green", se=TRUE)+
  labs(title ="OLS Regression", x = "OLS Algorithm Predictions",
y = "Actual")+
  theme_bw(base_size=15)

ggplot(test, aes(logit.pred, test$contraband_found)) +
  geom_point(shape=1) +
  geom_smooth(method = "glm", color="dark green",
              method.args = list(family = "binomial"),
              se = TRUE) +
  labs(title ="Logistic Regression", x = "Logistic Algorithm
Predictions", y = "Actual")+
  theme_bw(base_size=15)
```

VIII. Works Cited

1. Rosenberg, E. (2018, March 15). Exploding packages tap into simmering tensions over Austin's racial segregation. Retrieved March 25, 2018, from https://www.washingtonpost.com/national/exploding-packages-tap-into-simmering-tensions-over-austins-racial-segregation/2018/03/15/595a7b24-28a4-11e8-874b-d517e912f125_story.html?utm_term=.c6495248b7ea/

2. Pierson, E., Simoiu, C., Overgoor, J., Corbett-Davies, S., Ramachandran, V., Phillips, C., & Goel, S. (2017). A large-scale analysis of racial disparities in police stops across the United States. Retrieved March 25, 2018.

3. Shamena Anwar and Hanming Fang. An alternative test of racial prejudice in motor vehicle searches: Theory and evidence. The American Economic Review, 2006

4. GROGGER, J., & RIDGEWAY, G. (2006). Testing for Racial Profiling in Traffic Stops From Behind a Veil of Darkness. Retrieved March 25, 2018, from https://www.rand.org/content/dam/rand/pubs/reprints/2007/RAND_RP1253.pdf

5. No. 07–542. Arizona v. Gant, 556 U.S. 332 (2009). no. No. 07–542, 21 Apr. 2009. Justia, supreme.justia.com/cases/federal/us/556/332/.

6. Hadley Wickham (2017). tidyverse: Easily Install and Load the 'Tidyverse'. R package version 1.2.1. https://CRAN.R-project.org/package=tidyverse

7. Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL http://www.jstatsoft.org/v40/i03/.

8. Christopher Brown (2012). dummies: Create dummy/indicator variables flexibly and efficiently. R package version 1.5.6. https://CRAN.R-project.org/package=dummies

9. Hadley Wickham, Romain Francois, Lionel Henry and Kirill Müller (2017). dplyr: A Grammar of Data Manipulation. R package version 0.7.4. https://CRAN.R-project.org/package=dplyr

10. Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, the R Core Team, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, Can Candan and Tyler Hunt. (2018). caret: Classification and Regression Training. R package version 6.0-79. https://CRAN.R-project.org/package=caret

11. David Robinson (2018). broom: Convert Statistical Analysis Objects into Tidy Data Frames. R package version 0.4.4. https://CRAN.R-project.org/package=broom

12. Jerome Friedman, Trevor Hastie, Robert Tibshirani (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software, 33(1), 1-22. URL http://www.jstatsoft.org/v33/i01/.

13. A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. R News 2(3), 18--22

14. H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2009.

15. R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

16. 5harad. "5harad/Openpolicing." GitHub, github.com/5harad/openpolicing/blob/master/DATA-README.md.