MS&E 226: Mini-Project Part I

By: Olamide Oladeji, Abuzar Royesh

Data Investigation and Exploration

Description:

The data is a curation of all police arrests in New Orleans, Louisiana from 2010 to 2018. We retrieved this data from the Stanford Open Policing Project who filed public records requests to retrieve them[1]. The project provides opportunities to undertake predictive analysis and inferencing on issues around arrests and law enforcement bias.

In this part of the project, we are answering these two research questions:

1. How accurately can we predict whether someone is arrested given their demographic information, location data, and other variables related to why and how they were stopped?
2. Can we predict a person's age based on their other demographic information, location data, and other variables related to why and how they were stopped?

In addition to these questions, we can also learn about the frisk policies in the United States and association between certain variables such as gender and race in how police make their decision to arrest individuals that are stopped.

The raw data, as downloaded from the database, contains 512,092 rows and 32 features. Depending on the nature of the task i.e. classification or regression, we selected two of these as the response variables. The covariates include those on location, subject age, race and sex, details about the arresting police officer (ID, age, race, sex, unit), and covariates related to the nature of the incident, that is, the action undertaken, the reasons for stop or search, and the free form notes entered by the Police Officer.

Data "Cleaning" / Preprocessing:

Before building our models, we conducted several preprocessing/data cleaning tasks. First, based on the fraction of missing values and the usefulness of certain covariates we decided to eliminate certain covariates from the data. In particular, we deleted extraneous location covariates such as latitude ("lat"), longitude ("long"), "zone" and "location", leaving only the district as our location covariate. We also deleted covariates columns related to the vehicle details such as "vehicle_year", "vehicle_type", "vehicle_color", and "vehicle_name", since these do not have values for pedestrian stop situations and are the least complete.

We removed the binary column "search_conducted", since the "search_person" and "search_vehicle" are dependent on it and provide more granular information on what person was searched.

Based on the data dictionary, we assume that "NA" in the "raw_actions_taken" covariate refer to incidents for which no action was taken by the police. We re-coded these NA's to "Not searched." Subsequently, we removed all observations that had at least one missing variable.

We also converted the raw date-time covariates into a "time_of_day" categorical variable which takes values of "morning", "afternoon", "evening" and "night", depending on the actual time of the day of the arrests. We also used the raw date column to generate a new covariate for month of the year.

Our cleaned dataset included 204,794 observations and 16 variables: "arrest_made", "time_of_day" , "district", "subject_age", "subject_race", "subject_sex", "officer_assignment", "type", "contraband_found",

[1] E. Pierson, C. Simoiu, J. Overgoor, S. Corbett-Davies, D. Jenson, A. Shoemaker, V. Ramachandran, P. Barghouty, C. Phillips, R. Shroff, and S. Goel. (2019) "*A large-scale analysis of racial disparities in police stops across the United States*"

"frisk_performed", "search_person", "search_vehicle", "search_basis", "reason_for_stop", "month", "weekday". Finally, we split this cleaned dataset into training and validation datasets of size 80% and 20%, respectively.

Response Variables:

For our regression model, we selected the subject's age ("subject_age") as the continuous response variable. We chose this for multiple reasons: it was one of the only continuous variables present in the data, and, since we had all these other covariates about an individual, we thought it would be interesting to see if they could reasonably predict a person's age. We also discussed this with the Teaching Assistants of the course.

For the classification model, we settled on the "arrest_made", a binary response on if the stopped/searched person was eventually arrested. We chose this because an arrest is the worst-case culmination of a stop/search and we wanted to examine if there were any significant relationships between other variables like a person's gender, race, time of the day, etc. and whether they are ultimately arrested or not (i.e. simply warned, cited or just let go).

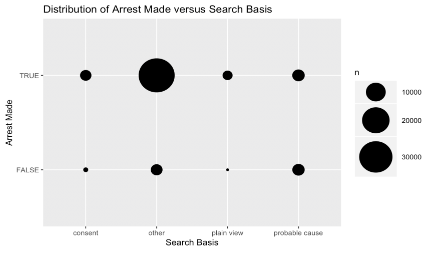
Correlations:

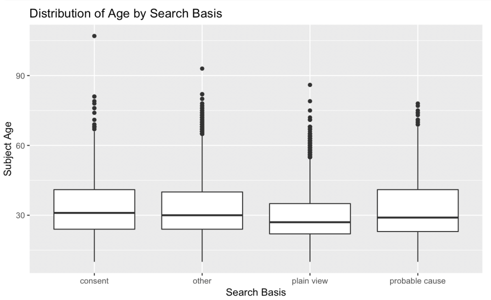
We also undertook some correlation analyses on the data. In particular, we tried to identify which correlations between the covariates and our response variable and other covariates. As most of the covariates are categorical, the correlation output of the R 'ggpairs' did not often apply so we used the Pearson chi test as a proxy for 'correlation' in categorical covariates. Using this, we identified the following pairs as having the most significant 'correlation' with the proposed binary response variable ("arrests_made"). Note that higher 'v' scores imply higher 'correlation' with the response variable (Table 1).

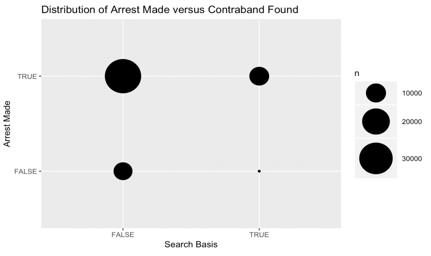
Number	variable	p_value	v
1	time_of_day	0	0.07
2	district	0	0.05
3	subject_race	0	0.09
4	subject_sex	0	0.07
5	officer_assignment	0	0.1
6	type	0	0.17
7	contraband_found	0	0.68
8	frisk_performed	0	0.52
9	search_person	0	0.66
10	search_vehicle	0	0.43
11	search_basis	0	0.7
12	reason_for_stop	0	0.17

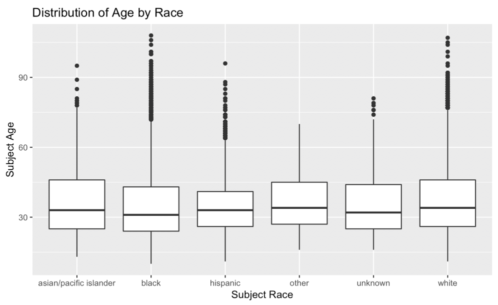
Table 1: Results of Pearson Chi test

For exploring correlations between the continuous response of subject age and the categorical covariates we had, we used a **one-way ANOVA test**, the results of which are shown below (Table 1 Appendix). We also generated visualizations to better understand the relationships between covariates and responses, some of which are shown below.









Predictive Modeling

As required, we explore two categories of prediction; regression, and classification.

Regression

We sought to build a regression model to predict the **"subject_age",** given the rest of the covariates in the dataset. We first focused on simple interpretable models like the Ordinary Least Squares (OLS) regression. For all our models, we trained the model on the training data and then predicted the outcomes for the validation data. After setting aside 20% of the data for testing (as instructed), we used the 80-20 train-valid split method on the rest of the data rather than cross validation (CV) to benchmark our models, because we believed we had a relatively substantial data set to train on. In addition to the 80-20 valid split, we also explored and compared 10-fold cross-validation for some of these methods as well (it proved to be computationally challenging for others).

We also defined a baseline model as one in which we build an OLS model to predict the sample mean subject's age from all the untransformed covariates as the baseline model. Based on our data, we think that a satisfactory model's RMSE would be at least 0.05 lower than the RMSE for the baseline model. Using the OLS, we first used all covariates to build our first model to predict the subject's age. The resulting RMSE for this model was **12.9842**

Next, we focused on exploring how interaction terms covariates might improve model performance (as measured by the validation RMSE). We began by exploring all interaction terms and report the RMSE obtained for this in **Table 2.** This decreased the baseline OLS model's RMSE by 0.5%. We then examined the mutual covariate 'correlations' values (from the Pearson chi test) and based on this as well as our subjective reasoning, we decided to explore other interaction terms. For example, we added interaction terms on 'subject_sex' and other covariates and we obtained lower RMSE than the Base OLS which we report in **Table 2**.

We then proceeded to examine regularized models to control for overfitting, and started with the **Ridge and Lasso Regression models.** We standardized the covariates as required for the Ridge and Lasso Regression model, trained and tested it on the 20% validation set. We report the best RMSE (we used functions that can grid search the best lambda value for Ridge and Lasso) obtained using ridge and lasso models in **Table 2.** We also train an Elastic Net Regression model on all covariates and interactions with auto-optimized hyperparameters, with the Elastic Net + all interaction term model giving us a model with 0.5% decrease in RMSE compared to the baseline OLS model's RMSE. Finally, we explored two other regression models: **CART Decision trees** with grid searched auto-optimized hyperparameters (max-depth, max_leaf) and the K-Nearest Neighbors with automatically-tuned k. We report the RMSE of these in **Table 2.**

Number	**Model**	**RMSE**
1	Elastic Net Standardized with all interaction terms	12.91732
2	Base OLS with all interaction terms	12.93224
3	Base OLS with gender interaction	12.97679
4	Ridge Standardized	12.98436
5	Base OLS	12.98442
6	Lasso Standardized	12.98512
7	CART Decision Tree	13.2996
8	k-Nearest Neighbors	13.30513

Table 2: Performance of Models for the Regression Task

Table 2 summarizes the performance of all the models, ranked by lowest RMSE first.

Classification

For classification, our response variable was **"arrest_made"** i.e. whether an arrest was made or not. We defined our baseline model here as a logistic regression model using all the covariates without any transformation or interaction terms. We ran this model using the same 80-20 train-valid split procedure we used for the regression. We defined performance based on three metrics: **Accuracy, Sensitivity,** and **Specificity**. We chose to consider the

Sensitivity and Specificity because our original data was unbalanced in terms of distribution of classes of "arrest_made". Using these, our baseline model had an accuracy of 0.9242, sensitivity of 0.9778, and specificity of 0.6318.

Next, we tried to improve the baseline model by experimenting with different models and using interaction terms. As we did with regression, we first added all interaction terms to our logistic regression model and report the RMSE in **Table 3.** We also again explore a K-Nearest Neighbor classifier with auto-optimization of k, yielded an accuracy of 0.9231, which was lower than the base logistic regression. We did plot the ROCs for all our models but chose not to include them in this report due to space constraints. We present in the table 3 below, a summary of the performance of these models in comparison to the baseline model.

Number	Model	Accuracy	Sensitivity	Specificity
1	Base Logistic Regression	0.9242	0.9778	0.6318
2	k-Nearest Neighbors	0.9231	0.9770	0.6296
3	Logistic Regression with Interaction Terms	0.9185	0.9708	0.6332
4	CART Decision Tree	0.845	1.0	0.0[2]

Table 3: Performance of Models for the Classification Task

Our best regression model, based on the RMSE, is the Elastic Net model with all interaction terms and auto-optimized hyperparameters.

In trying to estimate the error of this model on our previously held out test set, we considered two approaches:

1- We could run leave-one-out cross validation on the entire training data using our chosen model approach, and use the averaged fold error ErrCV as a (nearly) unbiased estimate of the generalization/test error Err. We then report the model parameters as trained on the entire training data, and this ErrCV. In practice, we did not do this because of how computationally intensive it was.

2- Since our data was fairly large in size (>200,000 data points after cleaning) and we already divided the 80% training data into a train subset and validation subset, the error on validation set could be seen as a reasonable estimate of the generalization / test error obtained from using model as trained on the training subset.

Using option (2), our best estimate of the Root Mean Squared Error = **12.91732.**

For the classification, we choose the Base Logistic Regression as our best model. We chose this because, besides having the highest specificity and accuracy on the validation set, the specificity obtained was approximately the same as the model with the actual highest specificity on the test set. The test set accuracy, sensitivity, and specificity from our best estimates are those obtained from the validation set and are 0.9242, 0.9778, 0.6318.

There are a number of factors that could lead to these values being biased. For instance, we might have biased results due to omitted variables that were not included in the dataset initially, or due to variables that we manually removed because they did not have complete data. We also discarded rows with missing values, which could impact our analysis. Another factor that could skew our findings could be biases in the data collection and reporting process by the police departments.

[2] We noticed that the auto-optimized CART decision tree classifier gave a specificity of 0, and while we debated whether to report this, we ultimately decided to.

Appendix:

Results of ANOVA test

Variable	DF	Sum Sq	Mean Sq	F Value	Pr(>F)
arrest_made	1	293775	293775	1760.66	< 2e-16 ***
time_of_day	3	493385	164462	985.657	< 2e-16 ***
district	7	211118	30160	180.754	< 2e-16 ***
subject_race	5	166635	33327	199.737	< 2e-16 ***
subject_sex	1	238609	238609	1430.04	< 2e-16 ***
officer_assignment	9	44456	4940	29.604	< 2e-16 ***
type	1	925	925	5.546	0.0185 *
contraband_found	2	138881	69440	416.172	< 2e-16 ***
frisk_performed	1	11828	11828	70.890	< 2e-16 ***
search_person	1	1004	1004	6.015	0.0142 *
search_vehicle	1	2	2	0.014	0.9073
search_basis	3	7485	2495	14.953	9.94e-10 ***
reason_for_stop	1	7550	7550	45.250	1.74e-11 ***
month	11	22939	2085	12.498	< 2e-16 ***
weekday	1	4136	4136	24.790	6.40e-07 ***
Residuals	204745	3.4E+07	167		

Table 1: Results of one-way ANOVA test