Making Our Cities Safer:
A Study In Neighbhorhood Crime Patterns

Aly Kane
alykane@stanford.edu

Ariel Sagalovsky
asagalov@stanford.edu

Abstract

Equipped with an understanding of the factors that influence crime rates, regional governments have the power to allocate resources to high-risk areas and reduce the frequency of crimes altogether. The purpose of this project is to understand the demographic factors that contribute to crime, in order to take a preventative approach to tackling them. Previous studies have identified how particular circumstances may lead an individual to commit a criminal act, however, there are few resources or statistical studies that show how crime is distributed across varying demographics. In our analysis, we implement supervised learning techniques to predict crime rates from publicly-available demographic data and learn the key variables that contribute to the predictive accuracy of our models. We describe a series of models, both linear and non-linear, as well as additional improvements that were made to draw conclusions about how to predict crime rates at the neighborhood level across large cities. We hope that our findings will help regional planners and police departments understand where to allocate resources and make the cities they serve safer.

1 Introduction

America's largest cities are changing very rapidly, and we seek to make them safer by developing insights on crime patterns. Our project will attempt to understand which demographic factors may influence crime rates. We wish to reason if key findings for specific neighborhoods in a particular city can be extrapolated to others, or if crime patterns vary widely across different cities.

Ultimately, the results of this project can be used for policy and planning purposes, helping cities understand if regulatory policies should be implemented at neighborhood level (police station) or at a city level (police department). Figure 1 illustrates the distribution of crime rate by Census tract in New York, with dark shades denoting safer neighborhoods.

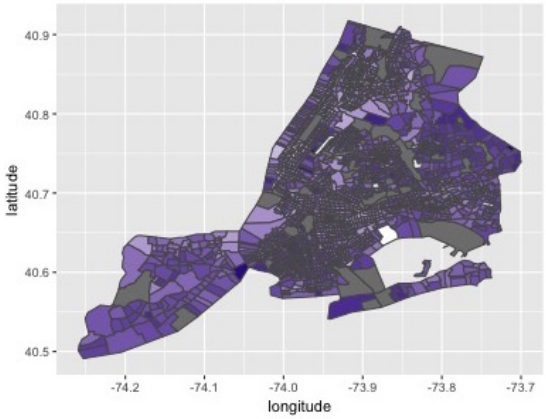

Figure 1: New York City

2 Data

2.1 Sources

There is no easily accessible dataset containing crime totals and demographics for the specific neighborhood in which each crime occurred. We chose to create our own by merging two data sources: (1) 2010 incident-level crime data, downloaded from individual cities' government OpenData portals, and (2) Census data containing demographic information at the tract level from the 2010 100% American Community Survey (ACS) Census. Both sources were compiled for six cities across the U.S.: New York City, Chicago, San Francisco, Detroit, Philadelphia, and Washington, D.C.

2.2 Features

The crime dataset contains every incidence of reported crime for the major cities in our study, with one entry per row describing the date and time of occurrence, crime description, and geographic identifier of location (latitude/longitude).

Census data includes demographic information for each Census tract. The relevant fields pertain to age, gender, race, ethnicity, and household relationships. At a stage of iterative refinement in this project, we went back to the source and augmented our feature set by adding fields pertaining to income levels, poverty, property value, and educational attainment.

2.3 Pre-Processing

Tremendous effort in this project was allocated to collecting and aggregating data. Because location of crime incidents are generally reported with latitude/longitude coordinate identifiers, we had to map locations to their respective Census tracts.

To accomplish this task, we downloaded all the necessary shape files for each county in our study. These files contain exact latitude/longitude coordinates of the polygons defining each Census tract in the county. We used the `sp` package in R to overlay the borders of the Census tracts on top of the map of the locations of each crime incident. In doing so, we were able to tag each observation in our crimes dataset with the appropriate Census tract. Finally, we aggregated the counts of crime incidents at the tract level, as we ultimately hope to predict crime rates at this level.

The Census dataset also required pre-processing. For each tract, we scrubbed the raw dataset to retain only relevant fields, reducing number of fields from around 500 to 80. To allow for better comparisons across tracts, fields with raw numbers were dropped and only fields with percentages were kept. Identifiers for Census tract were not consistent across counties and needed to be converted to a common ID. For instance, individual tracts were tagged with an 11-digit numeric ID, which we converted to the appropriate value, generally between one to four digits. We used a very systematic approach in verifying that each city had all tracts correctly tagged and in the same format, to allow for a seamless merge across datasets later on.

Once the individual city-wide crime datasets were cleaned and aggregated, we stacked them into a single data frame before merging with the Census data. We created our response variable, crime rate, by dividing the total number of crimes in a given tract by the tract population. In a few instances, crimes occurred in tracts where population was zero. We investigated tracts where these situations happened, and learned that this was mostly cases of crimes in city parks. Such tracts were left out of analysis, as no demographic data exists.

Prior to modeling, we plotted a histogram of the response variable to motivate the choice of particular models in our implementation. We noticed that the distribution of crime rate exhibited a heavy right tail. We postulated that the response was either Poisson or log-normally distributed. As a consequence, we chose to transform the response variable by taking logarithms of the crime rate for each Census tract and using the log-crime rate as the new response variable. This distribution more closely resembles a Gaussian one, so the standard assumptions of the linear models are more likely to hold.

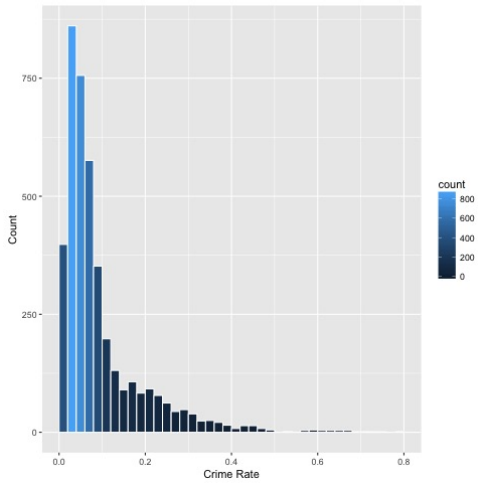

Figure 2: Crime Rate

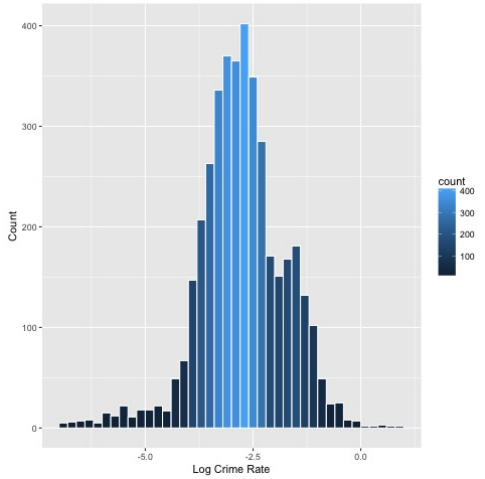

Figure 3: Log Crime Rate

Additionally, we centered and scaled each numeric predictors to take on mean zero and variance one for stability of numerical estimates.

3 Methods

To solve the regression problem, a combination of linear and non-linear models were used. Response variable, log-crime rate, is denoted $y^{(i)}$ and feature vector is denoted $x^{(i)}$. Each $(x^{(i)}, y^{(i)})$ pair corresponds to a Census tract for which we seek to predict crime rate using a specific hypothesis, $h_\theta(x^{(i)})$.

3.1 Linear

We began initial analysis of our data using Ordinary Least Squares Linear Regression. Linear regression is the hypothesis class $h_\theta(x) = \theta^T x$, which seeks to minimize cost function:

$$\sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 \tag{1}$$

By convexity of the cost function, the training error will be reduced with the addition of each new variable. However, we are ultimately interested in decreasing generalization error. Our dataset has a large number of predictors and, as such, is prone to overfitting. We are able to reduce the likelihood of our model having high variance through feature selection and regularization.

Feature Selection
Feature Selection is the process by which we can reduce the number of features in the model. To avoid testing 2^n unique models across all combinations of features, we implement a forward selection search strategy to find the best subset.

The forward selection algorithm adds one feature to the model at each iteration, choosing the feature which leads to the largest decrease in cost function at each step. From these *n* models, we used cross-validation to estimate test error and chose the model which minimized CV error.

Penalized Regression
An alternative approach to reduce the dimensionality of our problem is through regularization. This method of preventing overfitting adds a penalty term on the magnitude of the coefficients on each feature in our model. We implemented three forms of regularized linear regression with the following cost functions:

- Ridge

$$\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda\|\theta\|_2^2 \tag{2}$$

- Lasso

$$\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda\|\theta\|_1^2 \tag{3}$$

- Elastic Net

$$\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda_1\|\theta\|_1^2 + \lambda_2\|\theta\|_2^2 \tag{4}$$

Each of these methods will shrink coefficients toward zero, with Lasso allowing coefficients to be exactly zero.

3.2 Non-Linear

Given the complexity and lack of structure in our parameter set, we also attempted non-linear models.

Random Forests
Random Forests are an ensemble method of decision trees. At each node, a recursive binary split is performed where all variables and split points are considered, choosing the split which leads to the greatest reduction of the loss function. Decision trees divide the feature space into distinct, non-overlapping regions. Values are predicted by taking the mean of all observations which fall into a region.

Random Forests build many decision trees from bootstrapped samples to reduce variance, limiting the number of features considered at each split to reduce correlation between trees. Both number of trees and features considered are parameters of a Random Forest which must be tuned.

Gradient Boosting Machines
Applied to this regression problem, gradient boosting makes use of an ensemble of decision trees to produce a predictive model. The model is analagous to that of other boosting methods, such as decision stumps, however, it generalizes to any differentiable loss function, rather than squared-error or absolute-error loss.

Generalized Additive Models
Generalized Additive Models, or GAMs, allow for different non-linear relationships between each feature and response variable. GAMs are the hypothesis class such that

$$h_\theta(x) = \theta_0 + \sum_{j=1}^{n} f_{j\theta}(x_j) \tag{5}$$

GAMs allow us to fit very flexible models, however models can only capture additive effects while interactions may be missed.

3.3 Metrics

The process by which we constructed and evaluated each of the models was the same for all methods. First, we randomly allocated 80% of the dataset to the training set and left the remaining 20% for testing. The learning curve below indicates that test error is minimized using an 80-20 split.

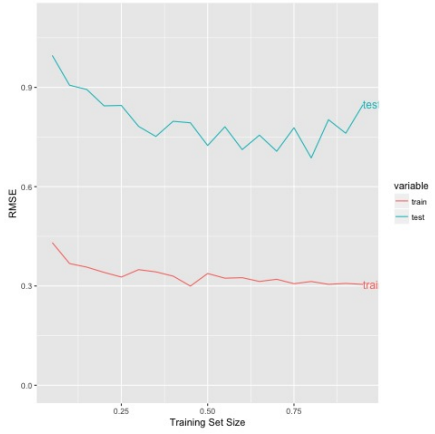

Figure 4: Learning Curve

We built all models on the training set, and optimized the parameters using cross-validation. Finally, we generated a series of predictions using the same features in the test set and computed the root mean-squared error (RMSE) on the response variable for the same test set.

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2} \qquad (6)$$

4 Experimental Results

4.1 Multi-City Model

As described in Section 2.2, we initially experimented using a dataset containing only race, gender, age, ethnicity, and housing relationship demographics. The performance of each model can be seen in the table below:

Model Type	RMSE
Ordinary Least Squares	1.124
Ridge Regression	0.908
Lasso	0.899
Elastic Net	0.897
Random Forests	0.693
Gradient Boosting Machines	0.893
Generalized Additive Models	0.905

Penalized linear regression models reduced prediction error over OLS a small amount, indicating that the full feature set is not necessary for predicting crime rates. The large decrease in RMSE between the Elastic Net model and Random Forests indicates that more flexible methods are needed to fit our dataset.

Random Forests and Gradient Boosting Machines were run on the full feature set, as both methods select splitting points on a subset of the feature space (this is a parameter which needs to be tuned). Before running GAMs, we ran a forward selection algorithm on a linear regression model. Using 10-fold cross-validation, the best model was a subset of 19 features, which we tried to fit data using a GAM with smoothing to allow for a non-linear fit.

In addition to predicting crime rate at high levels of accuracy, we were interested in learning which features are most relevant in predicting crime. Using our best model to-date, the Random Forest, we plotted the importance of its variables. See Figure 5 below for details.

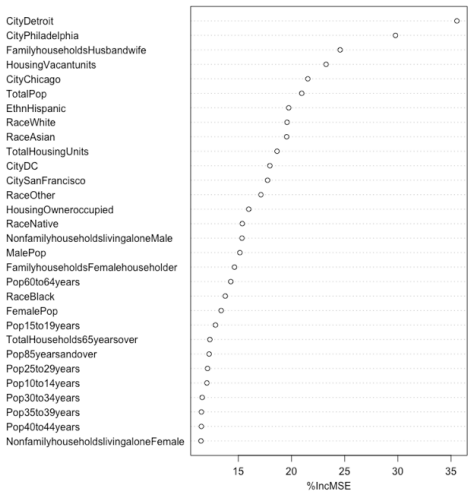

Figure 5: Variable Importance Plot

The most relevant predictors in our model are the dummy variables denoting the cities in which a crime occurred. We were surprised to learn that our previous assumption that we would be able to accurately predict crime rates based on demographics alone, with no information about the city itself, was incorrect.

Using this knowledge, we believed that we were likely to achieve better predictive accuracy by modeling each city independently of the rest. We chose to test this hypothesis on the largest city in our dataset, New York City, for proof-of-concept.

4.2 Single-City Model

Our assumption that a single-city model would outperform a multi-city model was confirmed immediately, as we achieved a RMSE of 0.491 on the New York dataset. To further improve upon this model, we augmented the New York City dataset in two ways.

First, we believed that neighborhoods where violent and non-violent crimes occurred would have different demographics. We classified all crimes into these two categories and calculated a violent and non-violent crime rate and modeled to these responses separately. Unfortunately, the predictive accuracy of our model worsened with these labels, as our new RMSE was 0.549 on the test set.

Next, we enriched our dataset with additional demographic data. Housing value, income, poverty and education levels were added for each census tract in New York City. These fields proved to be useful, as we achieved our best RMSE of 0.394.

Parameter tuning is an important method to find the best model in each hypothesis class. Note that parameter tuning was performed for all models where necessary, however we will illustrate one example here. In a Random

Forest model, the number of bootstrapped trees, n, and features considered at each split, p, are parameters that can be tuned to achieve a better fit. We used the entire training dataset to perform a grid search over multiple values of parameters n and p. A heatmap of the CV error for the combinations of parameter values tested is shown below. We found that n=250 and p=40 achieved lowest 10-fold cross-validation error.

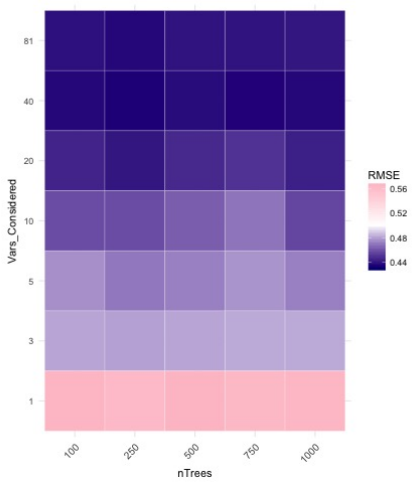

Figure 6: Parameter Tuning

Using our best model, we generated the variable importance plot once again to learn the most relevant features.

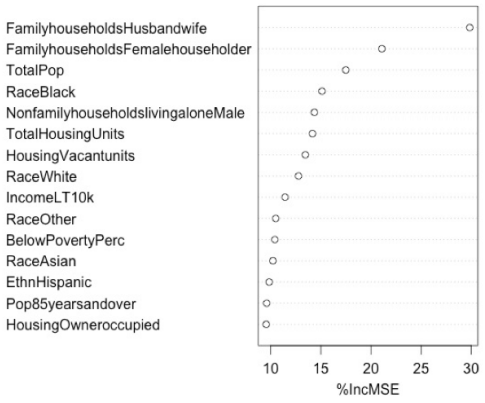

Figure 7: Variable Importance Plot

Notice that the two most relevant features are "FamilyHouseholdsHusbandWife" and "FamilyHouseholdsFemaleHouseholder". We can draw inferences about these features to claim that neighborhoods with a large proportion of two-parent households generally see lower crime rates, while areas with larger proportion of single-parent households experience higher crime rates.

5 Conclusion and Future Work

In this project, we implemented a handful of supervised learning methods and iterated to achieve the lowest generalization error. As expected, Random Forests outperformed all other models. From this model, we also learned the most relevant features correlated with crime rates.

For further work, we are interested in applying unsupervised learning methods to cluster neighborhoods across different cities and see if distinct patterns in crime level can be found based on demographic factors. In a first attempt, we performed principal component analysis (PCA) on the New York City dataset and plotted each entry in the training set along the first two principal components. We also ran a k-means clustering algorithm to tag each point as a "high" or "low" crime area based on log-crime rate. These results proved consistent with our earlier findings from the Random Forest model, namely that two-parent households are strongly aligned with low-crime neighborhoods.

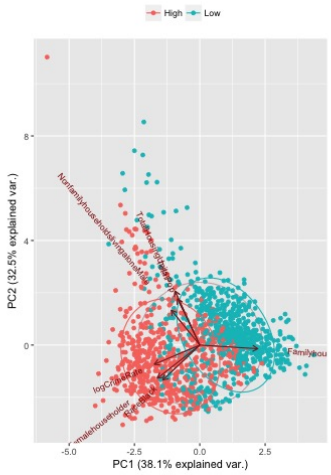

Figure 8: Principal Components Plot

Additionally, we limited ourselves to demographic and crime data for 2010 only, as this is the most recent and complete population dataset. The U.S. Census conducts its population studies every 10 years, but estimates are available every year. Although these datasets may present more variance in their reported fields, they could still provide useful information about neighborhood breakdowns. Using this data, alongside the data from other years, would help in building a better model.

Using this data, a potential approach is to model the crime rates for specific neighborhoods over time using a time series model and forecast which areas are most likely to see higher occurrences of crime due to changing demographics, based on historical patterns. This type of model would incorporate both demographic data, as well as any historical trend, possibly pertaining to increased or decreased police presence.

This type of model could be used to predict which areas in a particular city are most likely to become safer or more dangerous. This be an excellent proxy for gentrification or worsening of a particular neighborhood.

References

Bureau, US Census. *"American Community Survey (ACS)."* US Census Bureau. N.p., n.d. 2016. Web.

Gareth, James, Daniela Witten, Trevor Hastie, and Robert Tibshirani *An Introduction to Statistical Learning: With Applications in R.* 2013: Springer, New York. Print.

Hastie, Trevor, Robert Tibshirani, and J. H. Friedman *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2001: Springer, New York. Print.

Ng, Andew. *"Advice for Applying Machine Learning."* http://cs229.stanford.edu/materials/ML-advice.pdf. Stanford University, n.d. 2016. Web.

"NYC Open Data." NYC Open Data. N.p., n.d. 2016. Web.