Introduction

Los Angeles is the second-largest American city, encompassing a population of 3.8 million residents across a land area of about 500 square miles. Like any urban agglomeration, different forms of crime can disrupt residents' daily lives and place strain on the city's infrastructure, local economy, and government. By analyzing crimes by location and breaking them down by neighborhood, type, and time, we can uncover patterns that enable stakeholders, policymakers, and law enforcement personnel to better understand and predict criminal activity.

Using crime data across four years of LAPD police reports, I found that crime tends to be concentrated in a few highly populated areas, particularly Central LA, and spikes around January. Property crimes dwarf violent, white-collar, and other crimes, and are similarly concentrated in a few neighborhoods including West LA, Devonshire, and Pacific. I also developed different predictive models including pruned and bagged decision trees and LASSO, ridge, and elastic net models that predict crime and victims' ages based on characteristics like area, age, sex, weapon, and others with varying levels of accuracy. The best decision tree produced 76.4% accuracy when predicting crime categories and the best regularization technique produced a mean squared error of 14.36 when predicting victim age.

Data

I analyzed a large CSV file using Python with information concerning reported criminal activity in the city of Los Angeles from 2020 to 2024. Each crime report had a corresponding date and time, location by neighborhood and coordinates, descriptors of the incident and victim when applicable, and classification of the offense. To visualize the data, I first cleaned the 'Date Rptd' and 'DATE OCC' columns to identical formats and extracted the months, converting them from numbers 1 through 12 to their month names. I categorized crimes based on whether they fell under violent, property, white-collar, or other crimes — assault, battery, robbery, kidnapping, and homicide; burglary, theft, vehicle theft, arson, and vandalism; fraud, forgery, embezzlement, and identity theft; and drugs, weapons, and disorderly conduct, respectively, among others. I normalized the counts of each crime type across neighborhoods so I could compare their proportions directly. Finally, for mapping, I dropped rows for which the latitude or longitude was undefined.

Methodology

I began my analysis using a variety of data visualizations to understand crime patterns in geography and seasonality. My first plot was a bar graph of crime count by area (Table 1) that showed the number of crimes reported in each neighborhood, color-coded based on the quantity of crimes. This is good for visualizing hotspots of crime, but they don't adjust for population or land area, so it's hard to compare the neighborhoods directly. The second was another bar plot (Table 2) that shows the crime count by month of the year, also color-coded by number of crimes in each month across the four-year data collection period, with a taller bar indicating more crimes reported during the corresponding month. The heatmap (Table 3) provides a more nuanced perspective on the data described in Table 1, this time normalizing the data and providing proportions for each type of crime in each neighborhood. A scatterplot (Table 4) superimposed over a map of LA shows the geospatial distribution of the data, with each scatter point representing one reported crime and warmer colors showing clusters of crime.

I used the decision tree method to predict different types of crime. Decision trees are a form of supervised machine learning that display chance event outcomes as an interconnected

flowchart composed of branches. Simple decision trees can become more robust through the application of pruning and bagging, techniques that remove unnecessary branches to prevent overfitting and reduce the variance of noisy datasets. As explanatory variables, I selected columns that intuitively could impact the victim — demographic identifiers like their age and ethnicity, geographic information like the LA neighborhood, and incident-specific descriptors of the crime and weapon involved. At first, I used the specific crime classification as the response variable, but with over a hundred possible values, it overcomplicated the tree and produced a poor accuracy around 40%. To remedy this, I used the crime category variable I generated earlier for the visualizations and applied pruning and bagging to the model.

I also constructed LASSO, Ridge, and elastic net models to predict victims' ages given the characteristics of a hypothetical crime. I used the same explanatory variables as the ones I used in the decision tree. All three methods involved splitting the data into training and test groups, first teaching a model how to predict the target variable using 70% of the data and testing it on the remaining 30%. The three models differ in their shrinkage methods, limiting the number of features and other factors to prevent over or underfitting.

Results

Using the month bar plot (Table 2), we see that January has the highest number of reported crimes. The remainder of the months are mostly similar, with smaller spikes in March, July, August, and October. There doesn't seem to be much of a seasonal pattern apart from January — this makes sense since other cities see spikes of crime in the summer when the weather is unusually warm, but LA is mostly temperate year-round, moderating this effect. Spatially, in terms of pure number of crimes (Table 1), Central, 77th Street, Pacific, Southwest, and Hollywood see the most crime of any LA neighborhoods. When we break it down by crime type (Table 3), we see that West LA has the highest rate of property crime, Southwest, Harbor, and Hollaback have the highest rate of other crimes (including drugs), Southeast and 77th Street have the highest rates of violent crimes, and white-collar crimes (while uncommon), are highest in Devonshire, Topanga, Van Nuys, West Valley, and West LA. A heat map (Table 4) superimposed over a map of LA shows that the crime is mostly evenly spread throughout the city, mostly following major streets and denser residential areas, with a major hotspot in Downtown LA and smaller hotspots west and southwest of Downtown.

The simple decision tree (Table 6) using the crime category as the target variable produced an accuracy of 74.8%. Pruning reduced the accuracy to 73.1%, while bagging increased it to 76.4%. This result made the bagged decision tree the most accurate method for predicting crime categories at a broad level, but the predictive power of all three tree types became poorer as the specificity of those crime predictions increased – i.e. predicting arson precisely rather than predicting property crime broadly.

Of the three predictive models I tested to predict victim age, the Ridge model produced the smallest mean squared error at 14.36 (Table 5) compared to the LASSO's 14.54 and the elastic net's 14.73, signifying the highest accuracy. This means using the Ridge model and an array of characteristics of a crime including the area, victim's age, and other key factors, we could reasonably predict the age of the victim implicated in the reported crime.

Appendix

Table 1: Bar graph showing crime count by area of LA

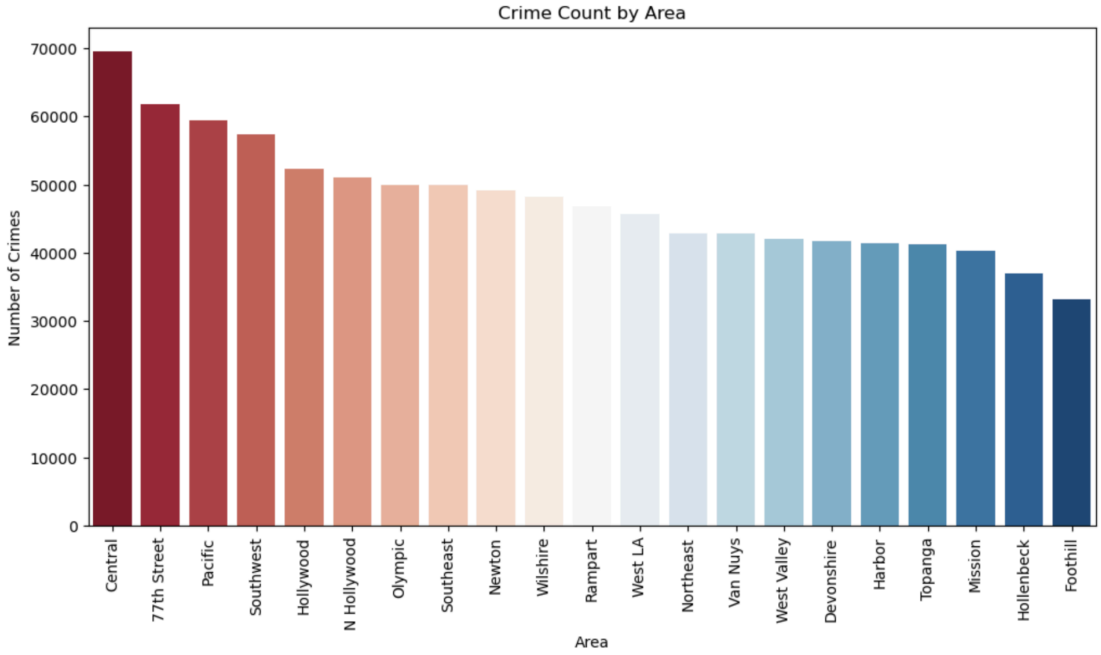


Table 2: Bar graph showing distribution of crime count across months

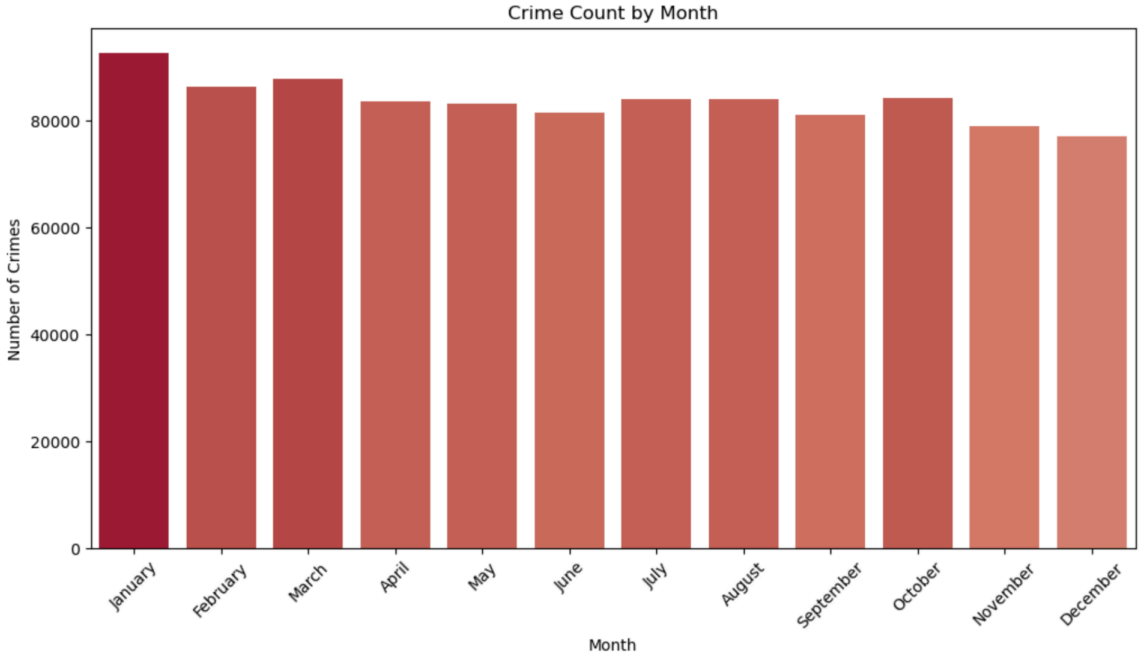


Table 3: Heat map of crime category by LA area

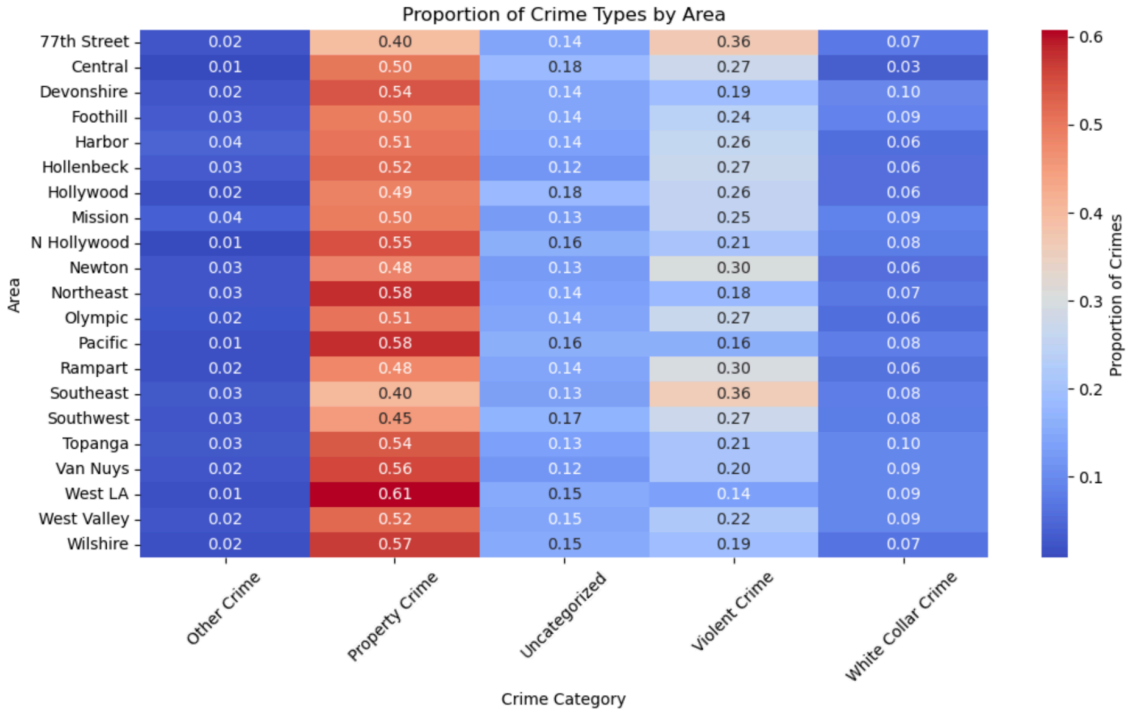


Table 4: Heat map of crimes on map of LA

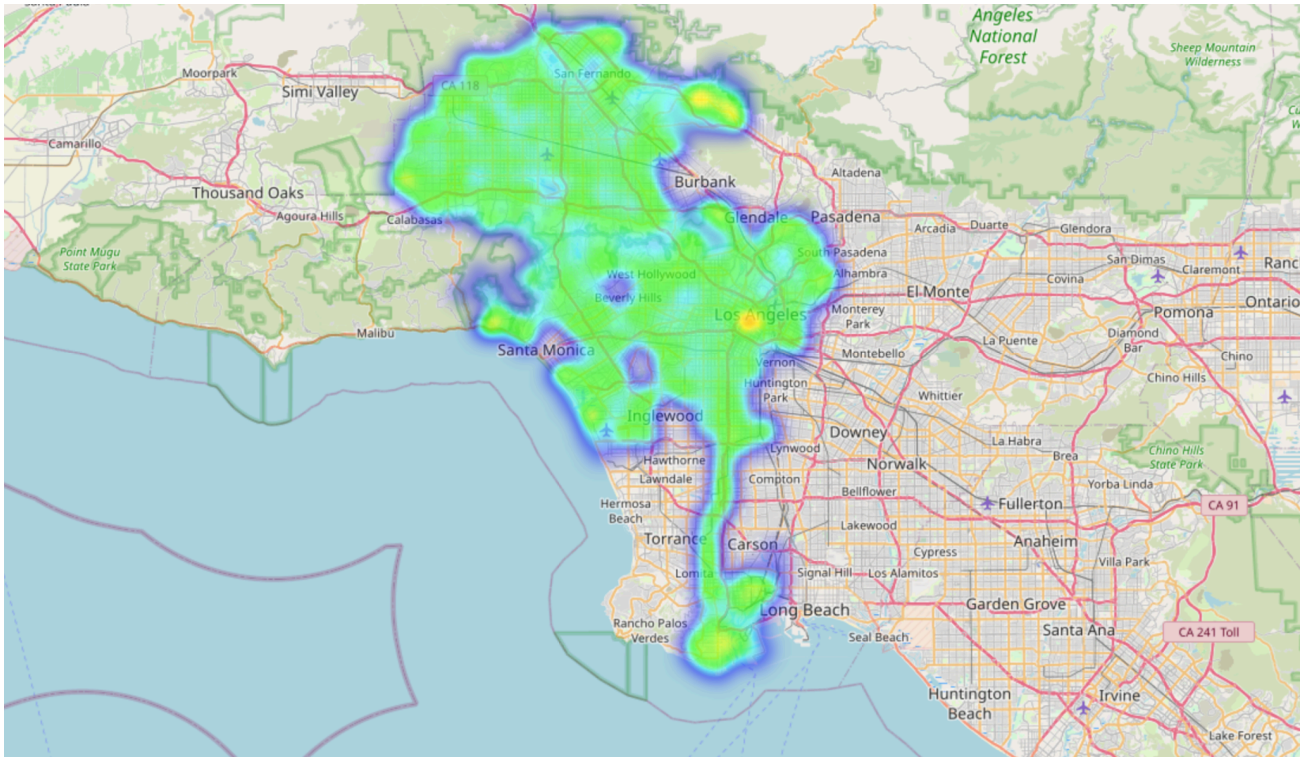
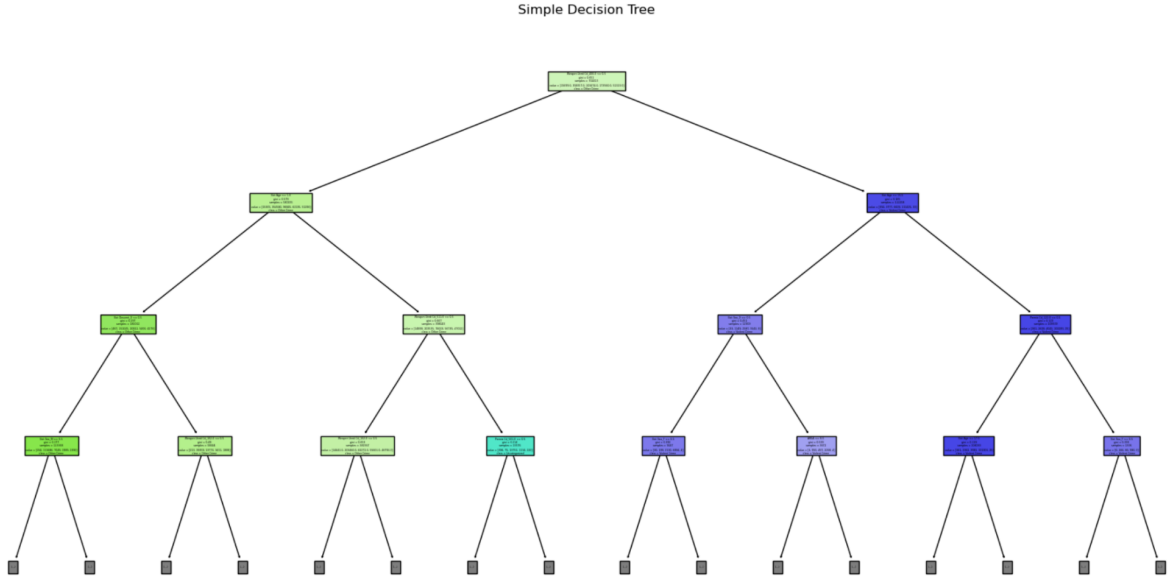


Table 5: Output of LASSO, Ridge, and Elastic Net regressions with mean squared errors

```
/opt/anaconda3/lib/python3.12/site-packages/sklearn/metrics/_regression.py:483: FutureWarning: 'squared' is deprecated in version 1.4 and w
ill be removed in 1.6. To calculate the root mean squared error, use the function'root_mean_squared_error'.
  warnings.warn(
LASSO RMSE: 14.541052912795667
/opt/anaconda3/lib/python3.12/site-packages/sklearn/metrics/_regression.py:483: FutureWarning: 'squared' is deprecated in version 1.4 and w
ill be removed in 1.6. To calculate the root mean squared error, use the function'root_mean_squared_error'.
  warnings.warn(
Ridge RMSE: 14.364452713880912
Elastic Net RMSE: 14.734085235540118
/opt/anaconda3/lib/python3.12/site-packages/sklearn/metrics/_regression.py:483: FutureWarning: 'squared' is deprecated in version 1.4 and w
ill be removed in 1.6. To calculate the root mean squared error, use the function'root_mean_squared_error'.
  warnings.warn(
```

Table 6: Decision tree with accuracies for simple, pruned, and bagged models

```
Simple Decision Tree Accuracy: 0.7478166990549272
```




```
Pruned Decision Tree Accuracy: 0.7309116880096999
Bagged Tree Accuracy: 0.7641270948560799
```