Predicting Monthly Violent Crime by ZIP Code in Los Angeles, California

Kadin Wilkins (kwilkins@berkeley.edu), Matilda Orona (matilda_orona@berkeley.edu), Vikram Magal (vikram_magal@berkeley.edu), Anushka Vazirani (avazirani@berkeley.edu)

Abstract

Violent crime presents a complex challenge for public safety, demanding proactive resource allocation strategies rooted in data-driven insights rather than reactive measures. This project investigates the feasibility of forecasting violent crime frequencies at the zip code level by analyzing spatiotemporal patterns and socioeconomic data. We aggregated historical crime data from LAPD (2020–2023) with zip code level population statistics, median household income, poverty rates, and unemployment rates to train and evaluate three machine learning models, Linear Regression, Random Forest, and XGBoost. Our experiments revealed that XGBoost using only lagged crime counts and temporal encodings achieved the best generalization with a Test RMSE of 9.79, outperforming models that included socioeconomic features. This counterintuitive finding suggests that for short-term forecasting, the immediate history of criminal activity is a more potent predictor than static demographic indicators, though performance degraded substantially in high-density "hotspot" zip codes, pointing to limitations in capturing volatile, event-driven spikes.

Introduction

Violent crime remains a critical public safety challenge, necessitating proactive strategies rather than purely reactive measures. The ability to anticipate crime trends at a granular level is essential for modern policing and public safety. This project addresses this need by developing machine learning models designed to forecast violent crime frequencies based on spatiotemporal and socioeconomic indicators. The input to our algorithm is a combination of historical monthly crime counts, zip code population counts, and socioeconomic data including median household income, poverty rates, and unemployment rates, and we use Linear Regression, Random Forest, and XGBoost models to predict the monthly violent crime count for a specific zip code.

Understanding these patterns is crucial as it enables law enforcement agencies to better allocate resources and respond more effectively and safely. Accurate predictions could guide decisions on where to station personnel, ensuring officers are adequately prepared for the specific risks of an area, which can facilitate measured de-escalation tactics. Additionally, uncovering these trends can support community policing efforts by distinguishing between areas prone to violent crime and those primarily experiencing non-violent offenses. Ultimately, our research seeks to enhance public safety by providing actionable insights that help law enforcement agencies strategically prioritize their limited resources.

Related Work

Our first related work, *Predicting the Probability of Crime Related Danger in Los Angeles* (Okpako, 2025), covered the very same topic we focus on. It investigates how different blended models perform in predicting crime danger. This acted as a core foundation for our work, as their work also utilized a Random Forest and an XGBoost. It was quite effective in its predictive accuracy based on past data, in a similar manner to our findings, but in contrast to our models, this work focused most on predicting crime danger in the entirety of the city from a victim-centric point of view and with a labelling approach.

Another work that we found related to ours was *Prevention is Better Than Cure: Predicting Violent Crime in US Counties Using Machine Learning Methods* (Bouwhuis, 2024). This work focused on the use of socioeconomic factors in predicting crime. It used Linear Regression among other models to attempt to predict violent crime within US counties. What they found was similar to our findings as can be seen below, when applying similar metrics at a zipcode level, closer to where resource allocation decisions are made. We used this work as inspiration to add socioeconomic features into our models, to see if there was predictive accuracy in these metrics. The exploration of these features within this work assisted in guiding our featureset in our final work.

In our third related work *Using Machine Learning Algorithms to Analyze Crime Data* (McClendon & Meghanathan, 2015), we took some inspiration in their crime categorization. This work classifies crime in a similar fashion to how we eventually did in our pre-processing. Taking crime labels from within a dataset, and classifying them in a binary violent or non violent based on the FBI Uniform Crime Reporting standards. This work utilizes Linear Regression to predict the number of violent crimes per 100,000 people, and also uses population as a metric similar to our work. They also used MAE and RMSE as a measure for the accuracy of their total crime reporting, which served as a strong indicator of model predictive strength for them, and was utilized in our measurements as well.

Dataset

We utilize a dataset published by the Los Angeles Police Department (LAPD) containing all reported crime incidents in Los Angeles from January 1, 2020 to May 29, 2025 (1,004,991 rows, 28 columns; City of Los Angeles, 2025). Each incident record includes the date, time, coordinate location, victim demographics, crime description, and indication of weapon use. To incorporate population counts and socioeconomic indicators such as median household income, poverty rate, unemployment rate, and the Gini index, we merged the 2023 American Community Survey 5-year estimates at the ZIP-code level (U.S. Census Bureau ACS, 2023). Yearly ACS estimates are only available for geographic areas with total populations of at least 65,000, so do not cover all LA zip codes. While our data spans multiple years, population and socioeconomic characteristics tend to change slowly over time, so the 2023 data serves as an appropriate approximation for these variables over our time period of crime incidents. We leveraged the Zip Code Tabulation Area (ZCTA) boundaries from the US Census Bureau's 2023 TIGER shapefiles (U.S. Census Bureau, 2023) to map each crime incident to a ZIP code via a spatial join on latitude and longitude coordinates. This mapping enabled us to map crime incidents to 149 zip codes in the LAPD jurisdiction.

During preprocessing, we dropped incidents with missing or invalid latitude or longitude coordinates (*n=2,240*) or that could not be mapped to a ZIP code (*n=406*). We also limited the data to incidents prior to January 2024 due to a significant decrease in incident reporting caused by LAPD's transition from the former FBI mandated Uniform Crime Reporting (UCR) system to a new National Incident-Based Reporting System (NIBRS), to comply with a nationwide FBI-mandate (FBI, 2024). Excluding this data prevents the transition from affecting our models and since we already have four years of complete data, we have sufficient data to build and evaluate models. Violent crimes were identified using a two-step rule. Incidents were labeled violent if they had a populated weapon_code or if the text crime description (crime_desc) matched federal UCR and COMPSTAT violent offense definitions. UCR and COMPSTAT define violent crime as offenses involving force or threat of force, specifically murder, rape, robbery, and aggravated assault. Crime descriptions were manually reviewed where weapon_code was not filled in to determine if the description matched that definition. We also manually went through all unique crime descriptions without a weapon code to ensure none would qualify despite the lack of a weapon. Next, we aggregated violent incidents to the ZIP code-month level. Missing ZIP-month pairs between January 2020 and December 2023 were added and filled with zeros. We engineered sine/cosine transformations of the month to capture cyclical patterns and three lagged features to enable temporal modeling. The lagged features represent the number of violent crimes that occurred in the same zip code one month (Violent Lag 1), two months (Violent Lag 2), and three months (Violent Lag 3) prior to the current month. These lags introduce information about recent trends that could strongly influence the next month. After creating the lags, we filtered to months where a full 3-month history was available. Missing ACS values were imputed using the median to preserve the shape of the distribution without adding noise.

We used a sequential, time-based method to split the data into training, validation, and test sets. Because we are trying to predict violent crime counts based on historical trends, the time-based split makes sense to preserve the forecasting nature of the problem and prevent leakage, ensuring that our model only learns patterns from historical months to predict future months. The data splits are as follows and each contain monthly records for each of the 149 zip codes that cover the LAPD jurisdiction:

- Train (April 2020 - June 2022): 4,023 records x 14 columns
- Validation (July 2022 - March 2023): 1,341 records x 14 columns
- Test (April 2023 - December 2023): 1,341 records x 14 columns

EDA uncovered many patterns that inform our modeling approach. First, violent crime is uneven across zip codes. Some zip codes report significantly higher concentrations of violent incidents, indicating clear geographic disparities and validating our decision to model crime counts at the ZIP-code level instead of the larger LAPD district areas (Figure 1). Second, the target monthly violent crime counts and lagged features demonstrate pronounced right-skewed distributions – most ZIP-month pairs reporting relatively few incidents, while a small subset experience consistently high volumes (Figure 2). This skew suggests that model algorithm and error metric choices should handle skewed distributions well. Correlation analysis showed that all three lagged violent crime features are almost perfectly correlated with the current month's counts (r = 0.97), indicating strong temporal trends (Figure 3). These lags are also highly correlated with each other, meaning linear regression models that are sensitive to multicollinearity will require limiting feature selection. Socioeconomic variables show much weaker relationships with our target. Total population, median household income, and poverty rate exhibit moderate correlation (|r| = 0.54-0.58), while the unemployment rate and Gini index show even weaker linear relationships. These findings indicate that historical crime patterns are our most important predictors, and socioeconomic factors may provide limited improvements in predictive power.

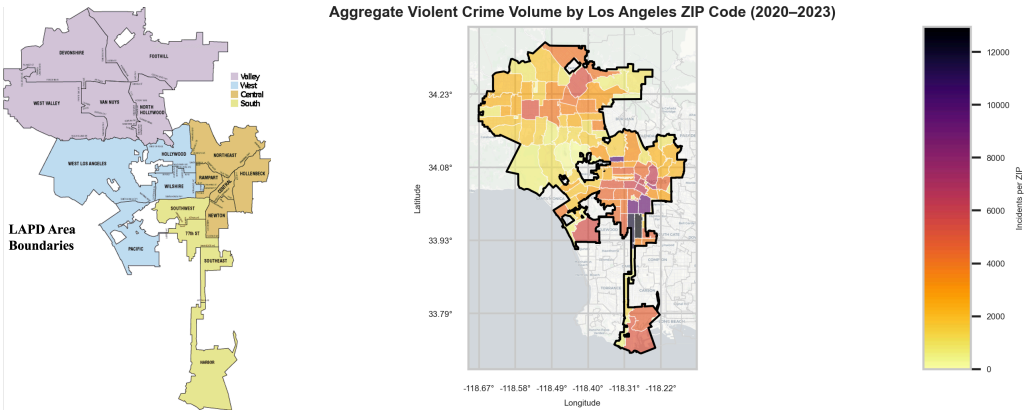


Figure 1. Left: Map of LAPD's four regions in our dataset. Right: Mapped Region-to-ZIP-code violent crime density (GeoHub, 2021).

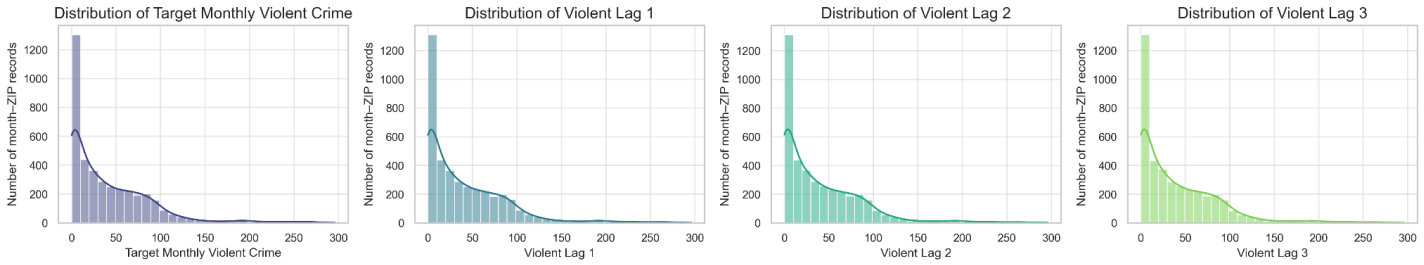


Figure 2. Left to right: Distributions of target (monthly violent incidents) and lagged features (violent incidents lag 1, 2, 3 month(s))

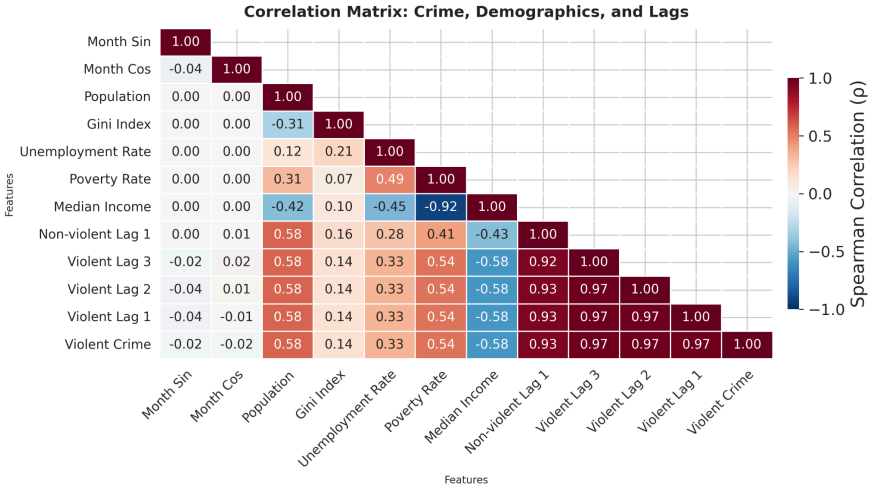


Figure 3. Correlation heatmap between target and features.

Methods

For our baseline model, we chose a linear regression using only one feature, the average number of violent crimes in the previous three months. Linear regression is a strong baseline because it provides us a simple and interpretable way to capture temporal trends in crime data seen in our correlation analysis, crime counts in each month are strongly related to those in recent months. However, these lagged features are also highly correlated with each other, which introduces multicollinearity and violates a key assumption of linear regression (that the predictors are not highly correlated with each other). By averaging them to create one feature, we eliminate the multicollinearity issue while preserving our main predictive signal.

Linear regression works by learning coefficients for the feature and bias term that will minimize the mean squared error (MSE) between the predicted and true violent crime counts. We trained our linear regression model in Tensorflow using stochastic gradient descent on standardized versions of our feature and target crime counts and evaluated its performance on the training, validation and test sets. This resulted in a simple yet accurate and generalizable baseline model which we can compare more complex approaches against. Since we ran additional models that do not require standardization, we applied inverse transformations and reverted our predictions to the original scale of the data and calculated root mean squared error (RMSE) and mean absolute error (MAE) between the predicted and actual violent crime counts to allow for comparisons against subsequent model experiments.

To improve upon our baseline Linear Regression model, we decided to implement a Random Forest. This model is particularly well suited for our crime prediction task since we are trying to capture the effects of the combinations of socioeconomic factors, spatial characteristics, and temporal trends. Unlike Linear Regression which assumes linearity and requires careful treatment of correlated features, Random Forests can handle the non-linear relationships between our features while not requiring extensive feature engineering. Random Forest's bagging approach averages predictions across independent trees, which reduces variance and provides robust performance even with correlated predictors.

Given our dataset, Random Forest was appropriate for several reasons. Firstly, we included lag-based time indicators as a part of our time series modeling (Violent Lag 1, Violent Lag 2, and Violent Lag 3) which encoded temporal relationships that varied amongst the different locations. Secondly, the dataset also included nonlinear locational and socioeconomic relationships. For example, crime could rise after unemployment reaches a certain threshold, or really low median household income in an area could lead to a higher rate in violent crimes. These correlated predictors are randomly spread across the trees, which makes it easier to capture the relationships between them. Additionally, to address missing observations in demographic variables, we imputed these fields using the median, which is robust to skewed distributions and prevents the loss of valuable data.

The third model we implemented is Extreme Gradient Boosting (XGBoost). While Random Forest relies on bagging which entails building independent trees and averaging them to reduce variance, XGBoost utilizes boosting. The latter builds the trees sequentially, where each new tree attempts to correct the errors (residuals) made by the previous ensemble. XGBoost works by optimizing a

regularized objective function using gradient descent. In each iteration, the algorithm adds a new shallow decision tree that predicts the gradient of the loss function with respect to the previous prediction. Unlike standard Gradient Boosting, XGBoost utilizes a second order Taylor expansion of the loss function to optimize the tree structure more accurately. This allows the model to capture complex, non-linear dependencies between our lag-based features and future crime rates.

XGBoost is particularly appropriate for our crime prediction task since the data inherently contains specific, hard to predict temporal dynamics, like spikes in violent crime, which a standard averaging method might smooth over. Another benefit of choosing XGBoost is that it has built-in Lasso and Ridge regularization which will help prevent the model from overfitting to noise during hyperparameter tuning.

Experiments, Results and Discussion

Linear Regression: Because our linear regression baseline only uses a single feature, the average violent crime count in the previous three months, our experimentation primarily involved selecting an appropriate learning rate for gradient descent. We implemented our baseline in Tensorflow with a single dense layer, including a bias term and initializing all the weights to ones. The model was trained using five epochs and stochastic gradient descent which, in each epoch, iteratively calculates the gradient and updates the model weights using a randomly selected subset, or batch, of 32 examples from the training data in each iteration until it gets through the whole dataset. We set the loss function as and optimized on MSE during training. Before training, we standardized both the predictor and target variable. After training, we transformed the predictions back to the original scale to assess RMSE and MAE on the original scale of our crime count data. Comparing results across learning rates of 0.0001, 0.001, 0.01, and 0.1, a learning rate of 0.01 yielded the best results with strong generalizability and no signs of overfitting (train RMSE = 10.78, validation RMSE = 10.07, test RMSE = 11.16, train MAE = 6.85, validation MAE = 6.54, test MAE = 7.30).

Random Forest: For the first Random Forest model, we conducted a set of experiments to evaluate how well the nonlinear tree based models capture the patterns of relationships between demographic (population_total, median_household_income, gini_index, unemployment_rate, and poverty_rate) and temporal variables (violent_lag_1, violent_lag_2, violent_lag_3, month_sin, month_cos) that lead to violent crime. After performing a series of hyperparameter tuning, the best validation performance was achieved with a moderately deep forest which helped limit overfitting. Performance was measured primarily using the RMSE across all splits. The best performing model achieved Train RMSE of 6.933, Validation RMSE of 9.544, and Test RMSE of 9.885.

To test if the socioeconomic features were introducing unnecessary noise, we trained a second Random Forest on a simplified feature set. This model omitted all census data (poverty, unemployment, etc.) along with seasons, and relied strictly on time-series features (Violent Lags 1-3) and Sine/Cosine month encodings. It uses n estimators 200, 400, 600 for our range of trees to balance performance and cost. The range for max depth is 10, 20, None so it can capture more complex patterns in our data. Our range for min samples is 2, 5, 10 for some smoothing of the model. Min samples were set to 1, 2, 4 so we can provide a range to work with in our GridSearchCV. After 243 fits, the best hyperparameters were found to be ‘max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 10, 'n_estimators': 600. It resulted in a MAE at 6.140, 6.523 and RMSE at 9.677, 10.218 on val, test respectively.

XGBoost: For the XGBoost model, we implemented a comprehensive hyperparameter tuning strategy using 500 iterations of Randomized Search over tree complexity (max_depth: 3–10), ensemble size (n_estimators: 1000–8000), and learning rate (0.001–0.05). Given that our target variable represents crime counts, we utilized the poisson objective function, which is statistically superior for non-negative for predicting count data (non-negative integers) compared to standard squared error. To strictly prevent data leakage and respect the temporal order of our dataset, training only on past data and validating on future data during the tuning process, we used TimeSeriesSplit (3 splits) for cross-validation rather than standard K-Fold.

Our initial XGBoost experiments revealed some overfitting, as shown in Table 1, the XGBoost configuration containing the full socioeconomic data achieved a low training error of 9.34 RMSE, but failed to generalize with a Test RMSE of 10.13. These results indicated that the complex socioeconomic variables, like poverty_rate and gini_index, while theoretically relevant, introduced noise that caused the gradient boosting algorithm to memorize the training data rather than learn generalizable patterns. To mitigate this we restricted the feature set to lag and time encodings only, which achieved our overall best model performance based on RMSE, with a Train RMSE of 8.95 and Test RMSE of 9.79. Based on these results, we selected XGBoost (Lags & Time Only) as our final model. It provided the most robust generalization with the lowest Test RMSE and effectively balanced model complexity and predictive accuracy (see Table 1 for results summary).

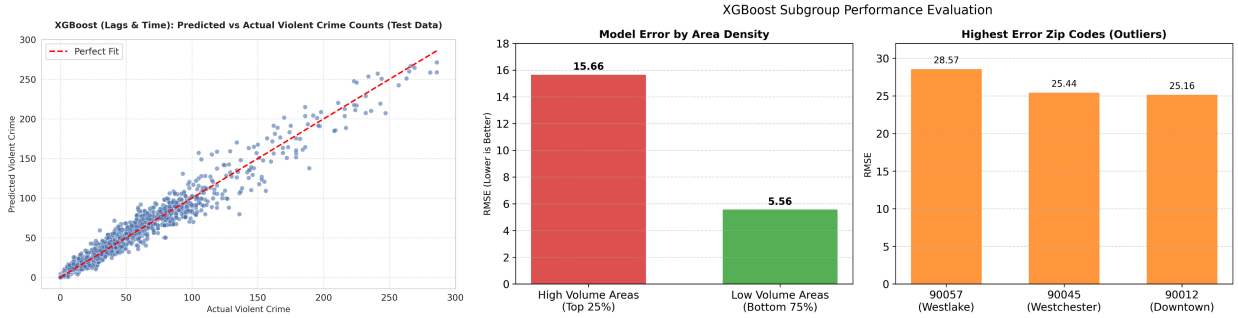


Figure 4. Left: XGBoost violent-crime actual vs. predicted. Right: RMSE by crime-volume group & for the highest RMSE ZIP codes.

To ensure our model performs equitably across different neighborhoods, we evaluated the final model performance at the Zip Code level. We categorized Zip Codes into "High Volume" and "Low Volume" buckets (see Figure 4). The model achieved an average RMSE of 15.66 across high volume areas. The error is naturally higher due to the larger variance and magnitude of crime counts in these busy districts. In low volume areas it achieved an average RMSE of 5.56, indicating high precision in quieter neighborhoods where crime patterns are likely more stable. The model struggled most in Zip Code 90057, where it yielded the highest RMSE of 28.57. This suggests that specific high-density areas contain irregular spikes in activity that simple lag features could not fully capture.

	MAE			RMSE		
Model	**Train**	**Val**	**Test**	**Train**	**Val**	**Test**
Linear Regression	6.85	6.54	7.30	10.78	10.07	11.16
Random Forest (Socioeconomic)	4.35	6.08	6.33	6.99	9.55	9.85
Random Forest (Lags & Time)	4.51	6.14	6.52	7.07	9.68	10.22
XGBoost (Socioeconomic)	6.00	6.07	6.58	9.34	9.27	10.13
XGBoost (Lags & Time)	5.93	6.22	6.36	8.95	9.74	**9.79**

Table 1. Model Performance Results

Overfitting was a concern across all models, as evidenced by the gap between training and test RMSE. Linear Regression showed minimal overfitting with a gap of 0.38 across train and test RMSE, likely due to its single-feature simplicity. Random Forest with socioeconomic features exhibited the largest gap of 2.86, suggesting the census variables introduced noise. For the final model the built-in L1 & L2 regularization helped control its gap of 0.84. We mitigated overfitting through hyperparameter tuning, particularly limiting tree depth and increasing minimum samples per leaf for Random Forest, and using early stopping with the Poisson objective for XGBoost.

Conclusion

Our research demonstrated that machine learning models can effectively forecast violent crime volume at the zip code level with a test RMSE of 9.79, equivalent to predicting within approximately 10 incidents of actual monthly counts. Our best-performing algorithms were Random Forest and XGBoost with Lags & Time features. The XGBoost showed high generalization when predicting against our validation and test data, with RMSE at 9.74 and 9.79. A key finding of our study was the lags and time feature only phenomenon showing that while socioeconomic factors are theoretically strong drivers of crime, our experiments revealed that including static census estimates alongside high-frequency time-series data introduced noise that degraded model generalization. The superior performance of the lag-based model confirms that, for short-term forecasting, the immediate history of criminal activity in a specific locale is the most potent predictor of future incidents.

Despite the model's overall success, subgroup analysis highlighted significant limitations in high-density "hotspots" like Westlake, where error rates were substantially higher than in lower volume neighborhoods. This disparity suggests that high-crime zones experience volatile spikes driven by transient factors that historical lags alone cannot capture. Future improvements should therefore focus on incorporating dynamic leading indicators such as local event schedules, weather patterns, or real-time 911 and 311 service calls, rather than relying solely on lagging indicators. Furthermore, increasing spatial granularity from zip codes to census blocks could help isolate specific high-risk street segments that are currently washed out within larger boundary areas.

Contributions (GitHub Repository)

Matilda Orona: XGBoost modeling with lags and socioeconomic features for comparison. Created the final visualizations for model performance and feature importance found in model_training_matilda.ipynb. EDA on LAPD dataset along with crime density plots are in crime_data_eda_MO.ipynb. Eda_matilda.ipynb contains EDA of aggregated dataset with feature engineering. Wrote XGBoost section, LAPD data section, conclusion, and helped refine other sections.
Anushka Vazirani: Jupyter notebooks for data cleaning, preprocessing, EDA, modeling (download_ACS.ipynb, preprocessing_AV.ipynb, eda_AV.ipynb, model_experiments_AV.ipynb). Wrote dataset and linear regression sections of report and slide deck, helped refine other sections.
Kadin - Model_KW Random Forest jupyter notebook, EDA_KW jupyter notebook, related works, random forest in report, slides in report sections/conclusion, Dataset feature selection.
Vikram Magal: Random Forest(socioeconomic+temporal) and EDA_VM, Models_VM jupyter notebooks. Random forest, Methods, and Abstract in the report. Methods and Random Forest in the slides.

References

City of Los Angeles. (2025). *Crime data from 2020 to present* [Data set]. Data.lacity.org. https://catalog.data.gov/dataset/crime-data-from-2020-to-present

City of Los Angeles GeoHub. (2021). *Zip Codes (LA County)* [Data set]. https://geohub.lacity.org/datasets/zip-codes-la-county/explore

Bouwhuis, N. (2024). *Prevention is better than cure: Predicting violent crime in U.S. counties using machine-learning methods – A comparative study* (Master's thesis). Tilburg University. https://arno.uvt.nl/show.cgi?fid=180309

Federal Bureau of Investigation. (2024). *National Incident-Based Reporting System (NIBRS)*. FBI. https://www.fbi.gov/how-we-can-help-you/more-fbi-services-and-information/ucr/nibrs

McClendon, L., & Meghanathan, N. (2015). Using machine learning algorithms to analyze crime data. *Machine Learning and Applications: An International Journal, 2*(1). https://doi.org/10.5121/mlaij.2015.2101

Okpako, J. O. (2025). *Predicting the probability of crime related danger in Los Angeles* (Master's thesis, Rochester Institute of Technology, RIT Dubai). https://repository.rit.edu/theses/12161

U.S. Census Bureau. (2023). *American Community Survey (ACS) data* [Data set]. data.census.gov. https://data.census.gov/all

U.S. Census Bureau. (2023). *TIGER/Line Shapefiles* [Data set]. https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.2023.html

Appendix

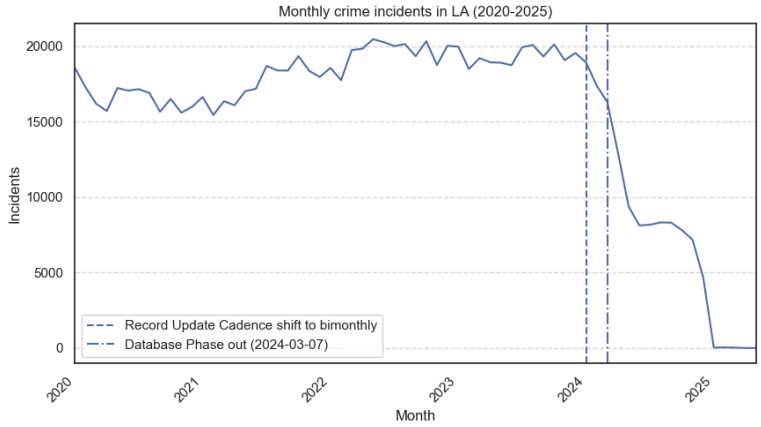


Figure 5. Monthly crime incidents in raw LAPD data download, showing sharp decline after reporting standards changed.

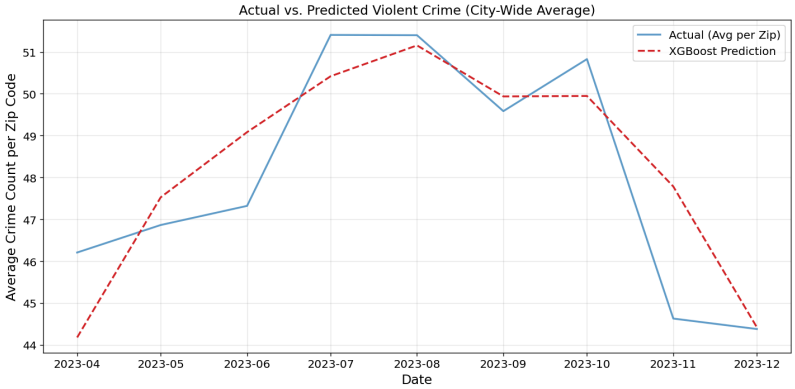


Figure 6. Performance of final XGBoost model on Test dataset.

Predicted vs Actual of all the models.

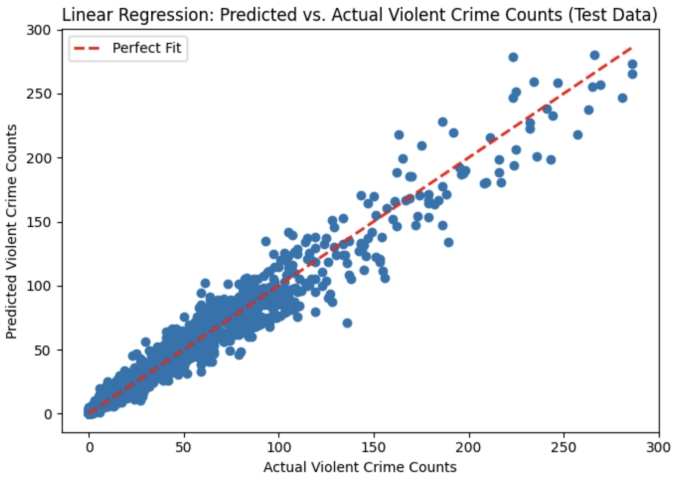
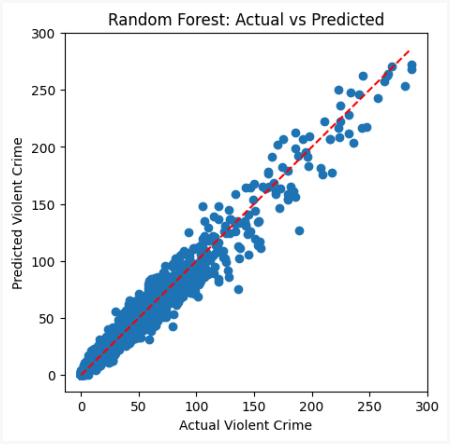
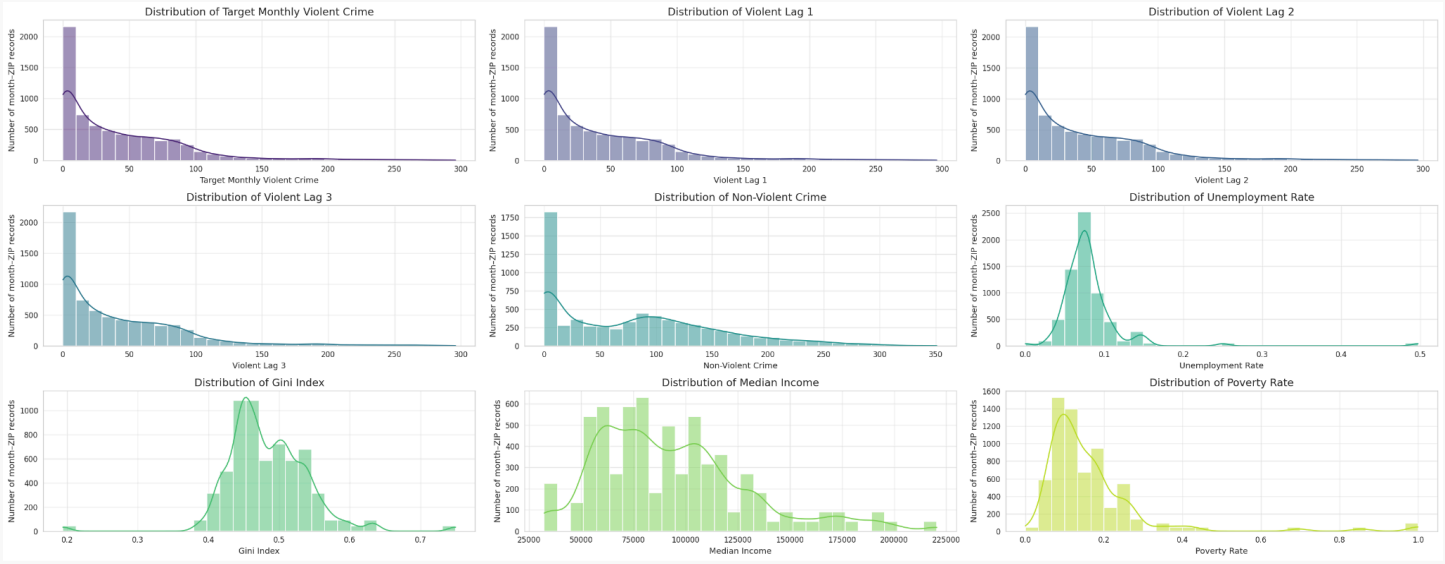


Figure 7. Linear Regression Baseline Model: Predicted vs. Actuals.



Predicted vs Actual of Random Forest Model(Socioeconomic+Temporal).



Distribution plots