Time-Series Forecasting of Crime in Los Angeles

Aryan Patidar
Data Science Institute, Brown University
December 7, 2023

GitHub Repository: github.com/apatidar/data-1030-project

Introduction

Problem

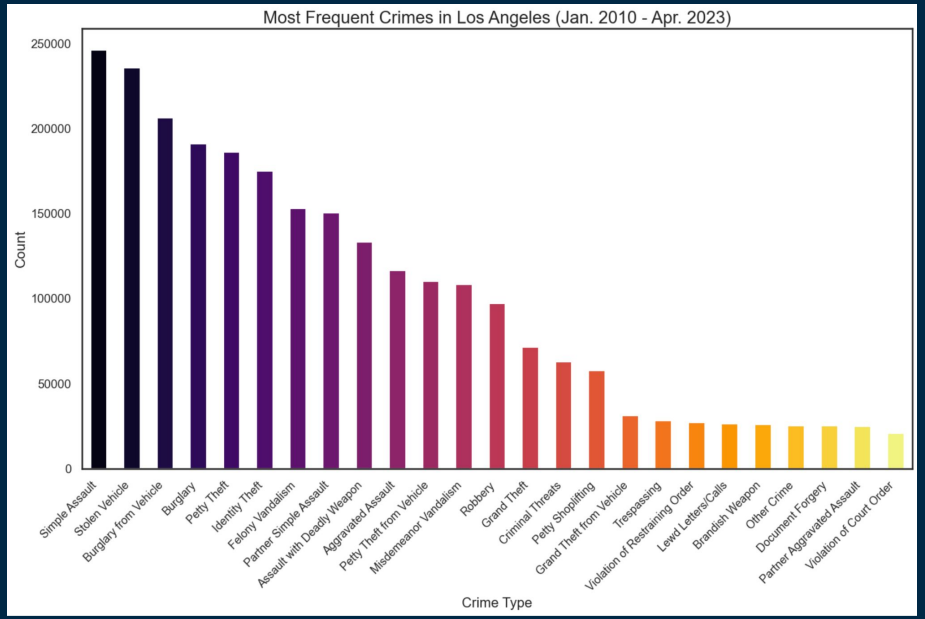
- Los Angeles is one of the most dangerous cities and experiences a wide range of crime
- **Aim**: use historical data and regression techniques to forecast future criminal activity
- **Importance**: improves understanding of crime patterns and trends, guiding resource allocation by law enforcement to enhance public safety

Source of Dataset: Kaggle

- From manually-typed crime reports, resulting in several missing values
- Spans January 2010 to April 2023 (2 million points)

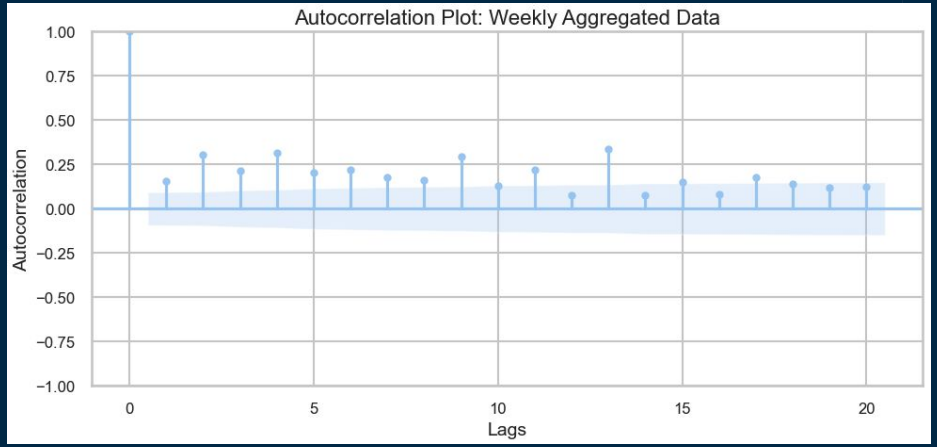
EDA and Preprocessing

- Out of 144 crime types, 15 account for the majority
- **Categorical Features**: 'Area Name', 'Victim Sex', 'Victim Race', 'Weapon Used Code'; applied one-hot encoding
- **Numerical Features**: 'Vict Age'; standardized using standard scaling
- After preprocessing, the feature count rose to 46



Lag Counts

- **Goal**: aggregating points to forecast future crime totals from past data via regression
  - Ex. predicting next month's number of crimes based on the current month
- Utilized AutoCorrelation and AutoRegression, chose weekly over monthly and quarterly
  - More statistically significant lag features
  - Higher plot variability: more capability to understand model complexity



	coef	std err	z	P>|z|	[0.025	0.975]
const	747.6042	190.514	3.924	0.000	374.204	1121.004
Total Crimes.L1	−0.0375	0.060	−0.630	0.529	−0.154	0.079
Total Crimes.L2	0.2486	0.057	4.342	0.000	0.136	0.361
Total Crimes.L3	0.1359	0.059	2.310	0.021	0.021	0.251
Total Crimes.L4	0.2764	0.057	4.834	0.000	0.164	0.389
Total Crimes.L5	0.0937	0.059	1.585	0.113	−0.022	0.210

Data Splitting and Algorithms

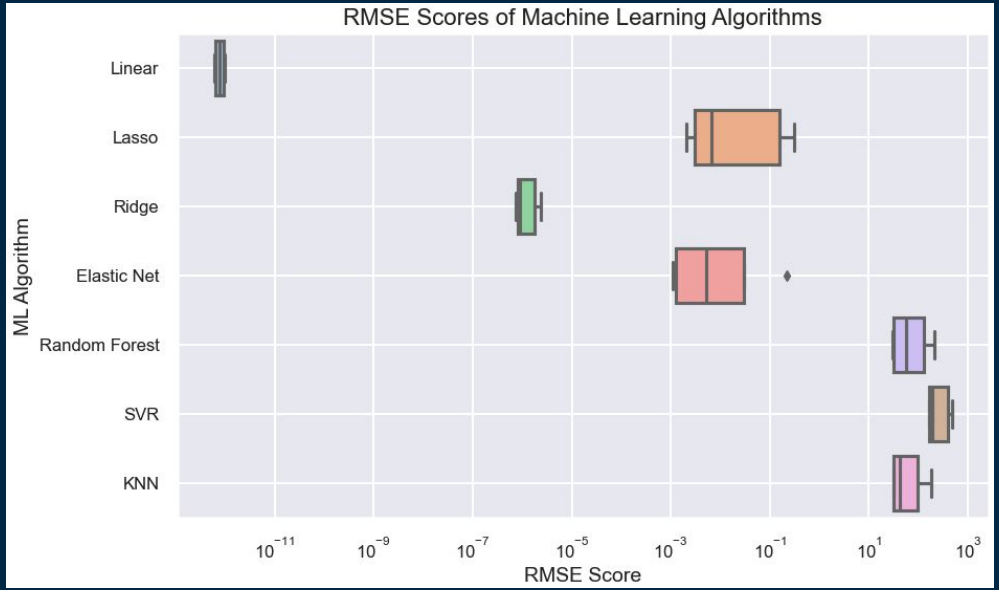
- Split using TimeSeriesSplit with 5 splits due to nature of the dataset
- **Overall Split**: ~70% training, ~15% validation, ~15% testing

- **Cross-Validation**: used GridSearchCV to find optimal combination of hyperparameters
- **Pipeline**: included the preprocessor and each ML algorithm

ML Algorithm	**Hyperparameters**
Linear Regression	*Baseline*
Lasso Regression	Alpha L1 Regularization
Ridge Regression	Alpha L2 Regularization
Elastic Net	Alpha Regularization L1 Ratio
Random Forest Regression	Max Depth (of each tree) Max Features (to consider when splitting)
Support Vector Regression	C Regularization Gamma Kernel
K-Nearest Neighbors Regression	Number of Neighbors range Weighting of Neighbors

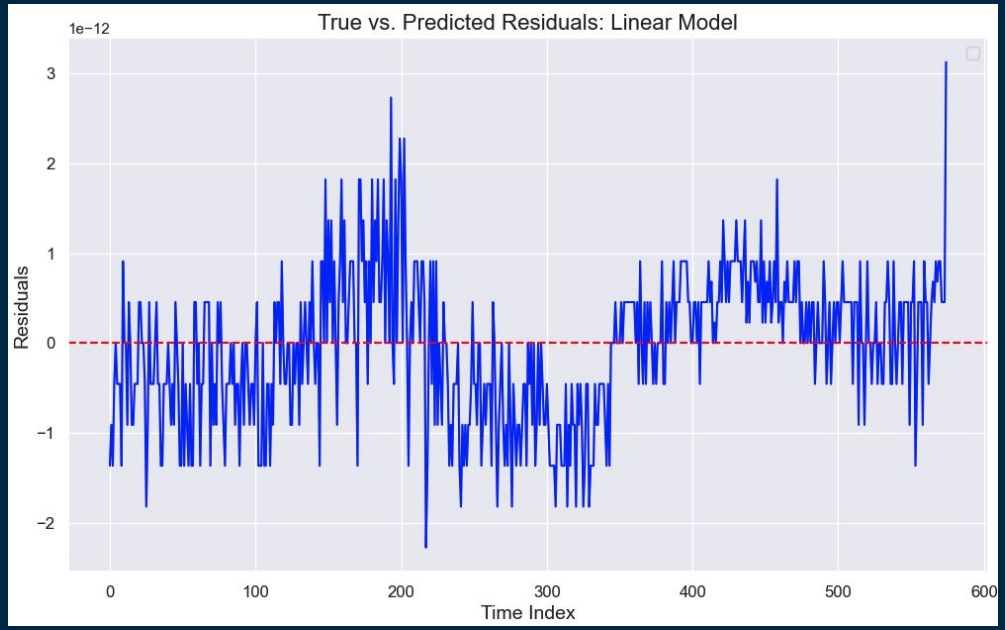
Results: RMSE Scores

ML Algorithm	Mean of Test Scores	Standard Deviation of Test Scores
Linear	7.8839×10^{-13}	1.5797×10^{-13}
Lasso	0.0953	0.1210
Ridge	1.3247×10^{-6}	6.4927×10^{-7}
Elastic Net	0.0508	0.0837
Random Forest	90.7564	68.6874
SVR	278.5141	130.0304
KNN Regression	77.9039	58.8649



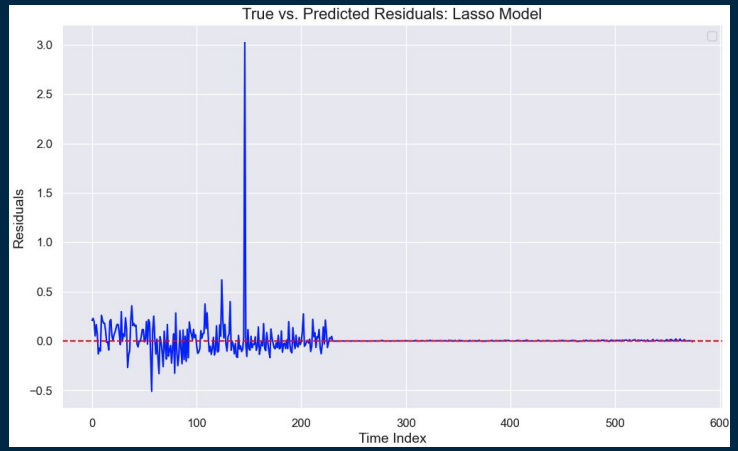
- Significant variance observed across the models
- Linear models generally display superior performance compared to the nonlinear models, suggesting linearity in the data

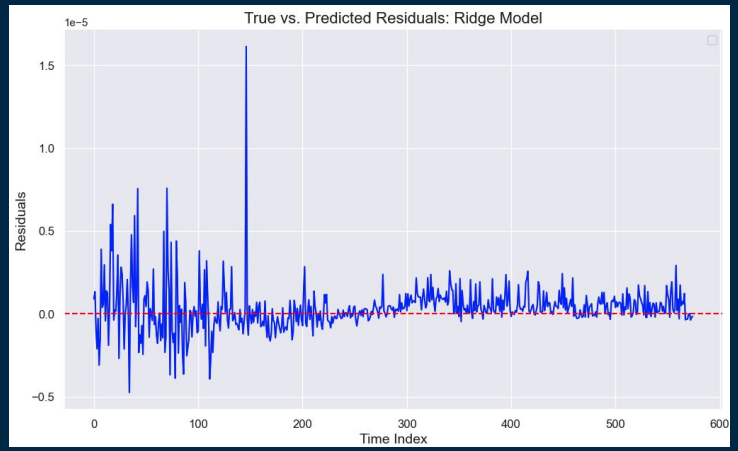
Results: Inspection of Linear Algorithms

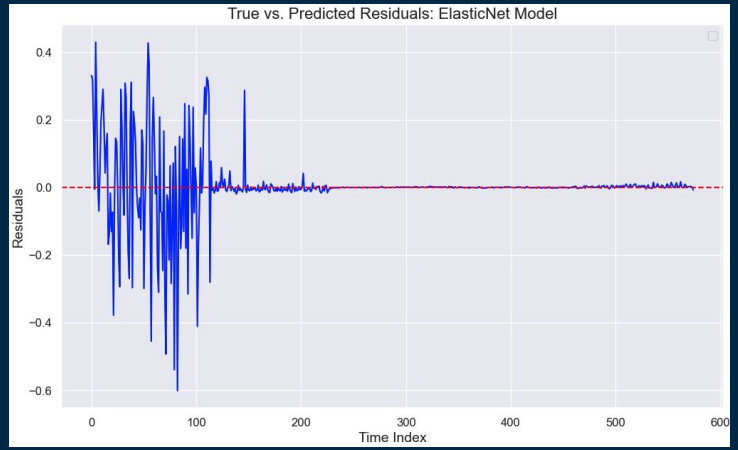


- Represents the baseline model
- Residuals exhibit consistent distribution across time, resulting in low RMSE score
- Demonstrates almost perfect prediction on test data, which is concerning
- Data leakage from test set to training set was limited by carefully selecting features and organizing the validation set after the training set, so score may be due to overfitting

Results: Inspection of Linear Algorithms

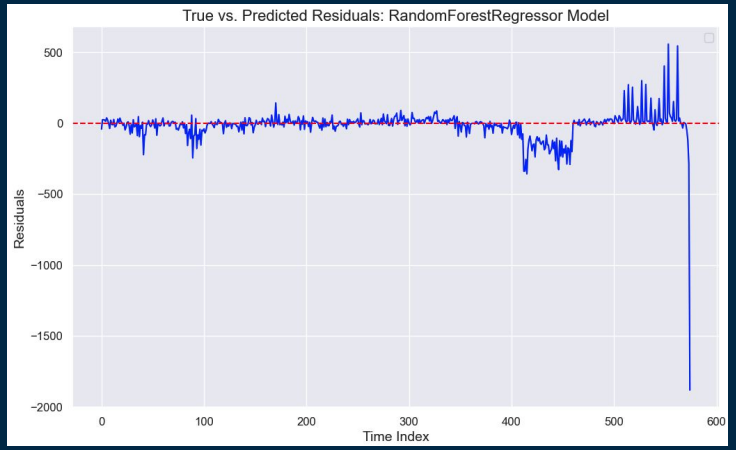


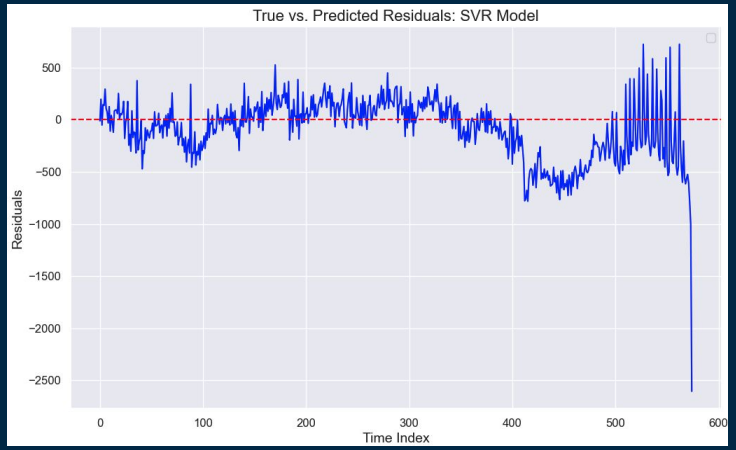


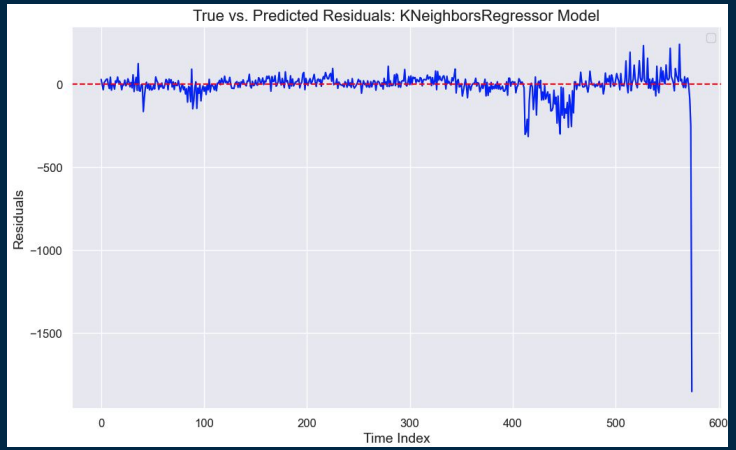


- Lasso and Elastic Net outperformed Ridge, indicating that regularization by shrinking coefficients to 0 was effective
- Smaller residuals at end of time series may indicate improved capture of data structure, but might also hint at potential overfitting or data leakage

Results: Inspection of Nonlinear Algorithms







- Conversely, nonlinear models exhibited rising residual variance at end of time series, limiting their practical use to generate predictions over the next few weeks
- Presence of few large residuals implies that models struggled with some poorly-fitted points, hinting at potential overfitting or unexpected shifts in distribution of the data

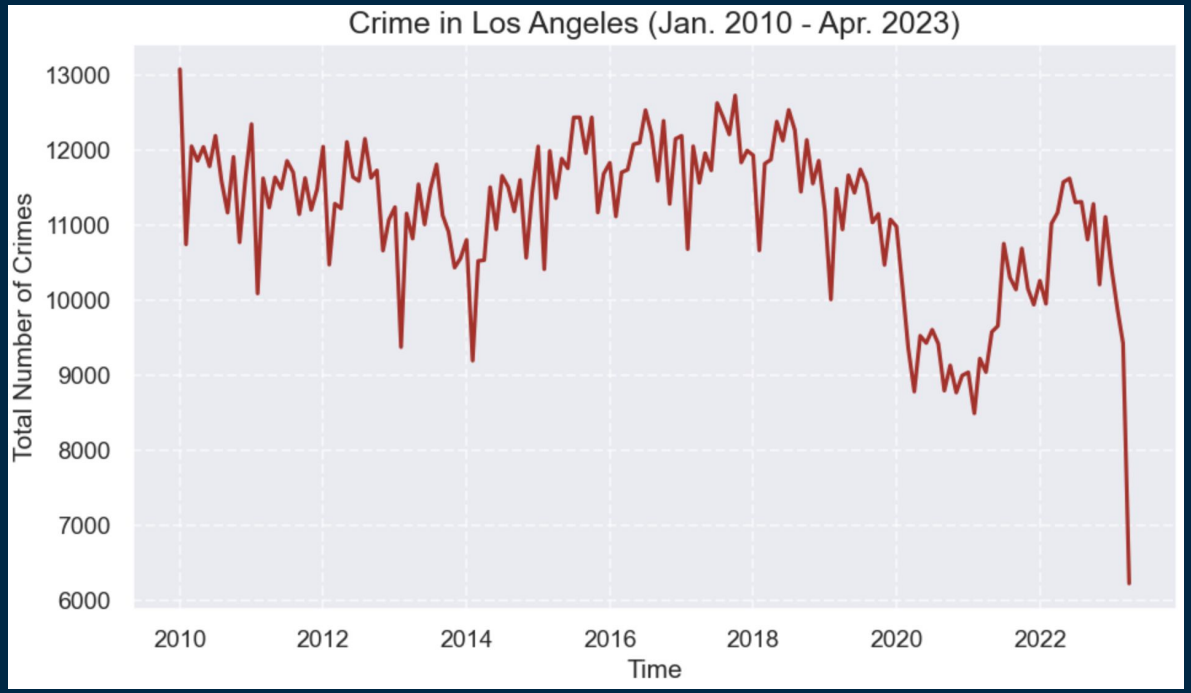
Results: Interpretability

- **SHAP Plot**: highlighted the features that were the most important, and therefore contributed the most significantly to the prediction
    - 'Area Name': generally minimal influence, but more dangerous regions positively impacted the prediction (ex. 77th Street, Southwest, and Pacific)
    - 'Victim Race': certain races contributed to a higher prediction
    - 'Weapon Used Code': most influential positive predictor, as use of a weapon strongly influenced the prediction
    - 'Victim Sex' and 'Victim Age': not as significant, randomly impacting predictions in both positive and negative directions

Outlook

- **General Takeaway**: strong performance of the Lasso and Elastic Net models instills confidence in their utility by law enforcement
- To improve predictive power, the following enhancements could be made:
  1. Overfitting: lower complexity of model (ex. reduce depth of trees in the Random Forest algorithm) and more rigorously split the data
  2. Investigate Outliers: evaluate outliers that potentially affect performance
  3. Additional Features: incorporate new features to better capture any underlying trends and seasonality in the data
  4. Feature Refinement: conversely, remove noisy or irrelevant features

Appendix: Total Crime



Appendix: Geographic Distribution of Crime