**Classifying Parking Violations utilizing Spatiotemporal Data**

**Harrison Cho - Brown Data Science Initiative**
https://github.com/hcho1111

## Introduction

To reduce traffic in large metropolitan areas, a road's accessibility, mobility, and openness are top priorities amongst city officials. The pitfalls of lax traffic enforcement extend beyond motorists; long-term negative externalities can affect the flow of commercial goods, air quality, and even pedestrian wellbeing. As such, complex regulatory schemes have arisen to minimize the externalities of poor driving behavior and general traffic-related inefficiencies.While a problem of significant scale, addressing traffic violations in large cities carries significant monetary and logistical benefits. I propose a data-centric approach to classify traffic violations utilizing vehicle, temporal, and location-based features.

I am working with parking violation data collected by New York City's Department of Finance from the years 2020 and 2021. The target variable is the parking violation code of a given vehicle.[1] For simplicity's sake, the roughly 100 violation codes are grouped into 17 different categories based on violation similarities.[2] The current dataset has 266,510 observations and 20 features. These features range between vehicle, location, and temporal characteristics of individual violations.

Previous analyses have utilized this data to predict various aspects of on-street parking. One study by Gao et al. utilized location-based data to determine violation patterns and measure the legality of on-street parking.[3] The study utilized six machine learning classifiers to predict parking legality. The most successful classifier, i.e. the lowest total root-mean squared error, employed random forest classification methods. Overall, their results generalized that commercial, healthcare, and food-service locations were positively associated with increased parking violations. An additional study by Li et al. predicted the availability of street parking within metropolitan areas.[4] Again, random forest models best approximated the availability and timing of parking. The use of spatiotemporal features from the parking violations dataset was coupled with human mobility data and point of interest data. This was done to suggest policy solutions for parking regulators and city managers.

## Exploratory Data Analysis

The data was extensively cleaned to conserve space and ease computational time; a number of columns were dropped before preprocessing began. First, columns whose data had 90%-100% missing values were dropped. Second, I dropped several broad, location-based measures due to the increased threat of multicollinearity. I replaced these measures with fine-grain latitude and longitude coordinate-data generated through a geolocation API. Third, if external information relating to feature interpretation could not be found, uninterpretable features were dropped from the dataset. A significant part of this analysis stems from classifying parking violations. A dictionary for the 17 classifications, along with definitions for other categorical variables, is in the references section.[5] Below are several graphics that provide general insights on the structure of the dataset.

---

[1] Data dictionary listed in the references

[2] Violation codes provided by the NYC department of finance. Link to descriptions and fine amounts provided in references

[3] Gao et al.. "Spatiotemporal Legality of On-Street Parking". 299-312

[4] Li et al.. "Understanding the Spatiotemporal Availability of Street Parking". Sigspatial'19. doi.org/10.1145/3347146.3359366

[5] See Appendix Figure A

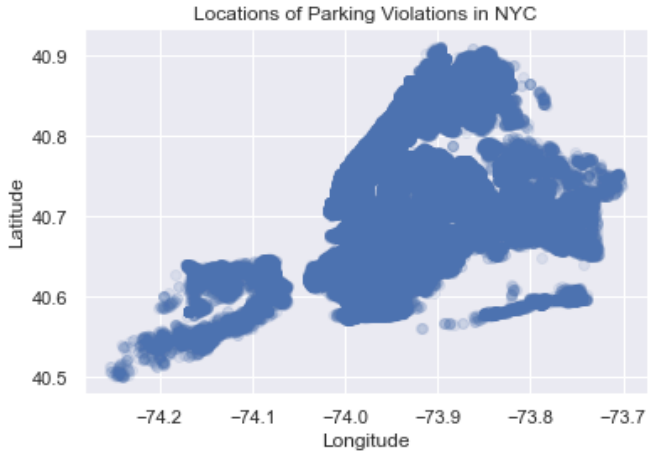


**Figure One:** Coordinate locations of parking violations generated via New York City's Geoservice API.

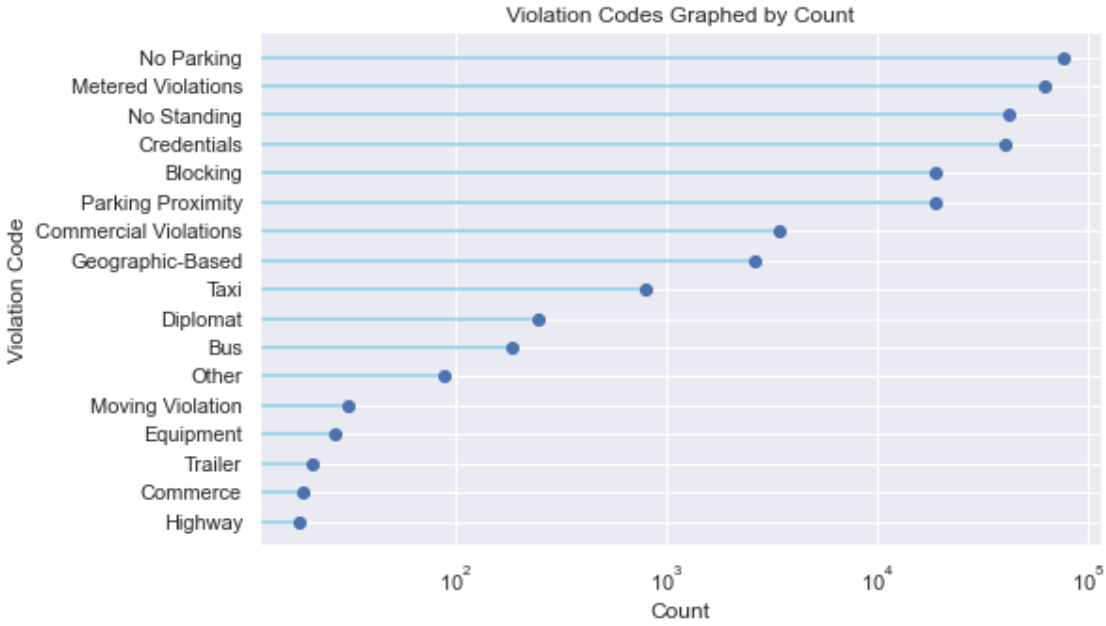


**Figure Two:** The target variable, its groups, and general imbalances between these groups is illustrated. Compared to the largest group, the smallest category is smaller by several orders of magnitude. Data is normalized when possible to account for this imbalance.

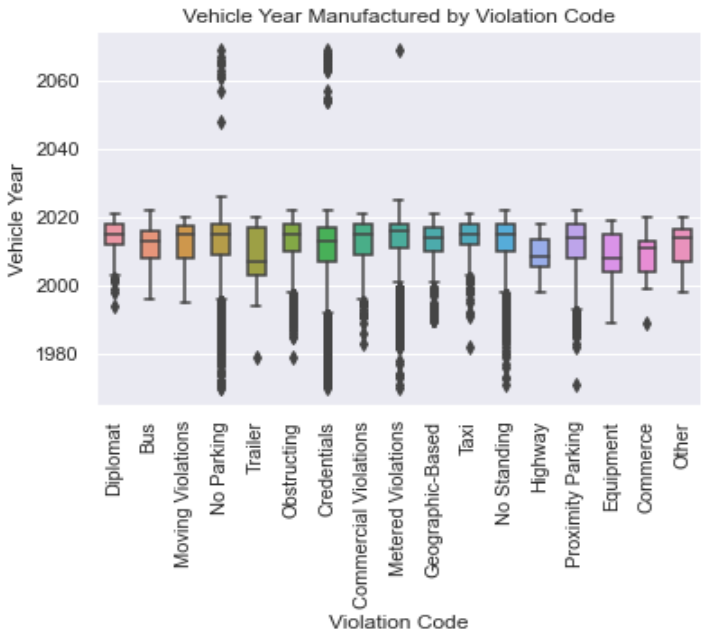


**Figure Three:** This figure maps the distribution of a vehicle's year to its corresponding violation code. There appears to be many outliers in each distribution, skewing the true group mean. Despite this, it appears that the mean ages of vehicles appear to be similar across violation codes. Significant mean disparities arise for trailer, highway, equipment, and commerce groups.

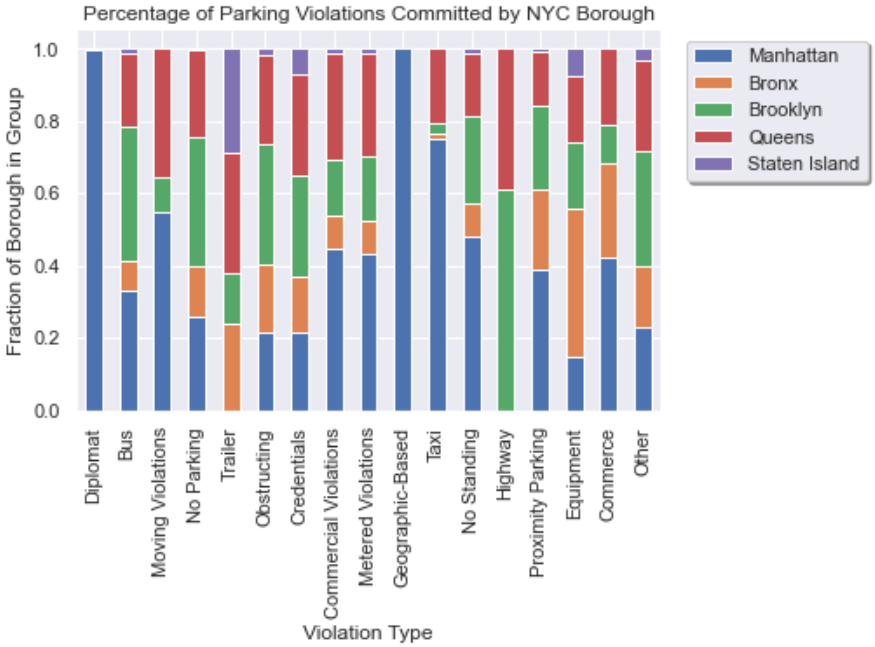


**Figure Four:** The stacked bar plot illustrates the percentage each NYC borough contributes to a total violation group. Interestingly, both diplomat and geographic-based violations occur exclusively in Manhattan. Overall, it appears that a significant portion of total violations occur in Manhattan.

**Methods**

Splitting and Preprocessing

Data readable to the API was utilized in the model. This accounted for approximately 220k observations. All categorical features were reengineered to increase calculation speed.[6] Figure two illustrates the imbalances which exist by violation code within the data. To account for violation code inequality, the data is split through the StratifiedShuffleSplit method. It is important to stratify the data such that roughly equal proportions of each violation code appear in each data split. Three splits were incorporated in the model to ensure that the training, validation, and testing splits (80%, 10%, and 10%, respectively) were representative of the entire dataset, increasing model generalizability. I assume that individual parking violations are IID. Intuitively, parking violations are independent of one another, and do not affect the inclusion of other violations. Because individual vehicles are represented, the data lacks group structure. Worth noting are the data's temporal elements. Despite these elements, the data is not a time series because observations are uncorrelated with previous observations.

The data contains 19 preprocessed categorical[7] and numeric[8] features. All categorical features in the dataset do not have a natural order associated with their classification. These categorical features are preprocessed with the OneHotEncoder. Per results from the EDA, all numerical features are unevenly distributed. To address these imbalances, the StandardScaler encoder is employed to normalize counts. After preprocessing, 85 variables are present.

Metrics, Models, and Cross Validation

The question at hand is a multiclass classification problem with imbalanced data. Prioritizing metrics that were robust to imbalance, yet easily interpretable in a large multiclass problem, I selected balanced accuracy score as my overall evaluation metric. Model selection was based on overall interpretability, complexity, and computational efficiency. Below is a table summarizing models utilized in this paper. Over three random states and 3 splits, optimal parameters for the models were found through exhaustive grid search.

**Table One:** Summary of Models and Selected Hyperparameters

| **Model** | **Hyperparameters** |
|---|---|
| Naive Bayes | $\alpha$: 30 logarithmically spaced values between $1e^{-4}$ and 10 |
| Ridge Classifier | $\alpha$: 30 logarithmically spaced values between $1e^{-4}$ and 10<br>**solver**: auto<br>**class_weights**: balanced |
| Random Forests | **max_depth**: 1 through 15 features<br>**n_estimators**: 10 |
| Gradient Boosted Trees | **n_estimators**: 1, 3, 10, 30, 50, 100, 150, 300<br>**learning_rate**: 0.3 |

[6] See Appendix D through I for more details
[7] See Appendix Figure B
[8] See Appendix Figure C

Naive Bayes served as a baseline model due to its ease of use and inherent multiclass structure. A ridge classifier was implemented to control for correlation between features. Two ensemble methods, random forests and gradient boosted trees, were implemented to decrease potential overfitting seen in baseline models. Additionally, these ensemble methods were selected due to strong performances in high-dimensional datasets.

Hyperparemeter ranges were generated from a series of sources. A general understanding of the problem's structure, advice from mentors, and various online guidelines informed this process.[9] After splitting and preprocessing, hyperparameter values were found via validation splits. Subsequently, optimal hyperparameters were fit to a testing set to calculate balanced accuracy scores. Figure five summarizes this process, illustrating variation in each model's performance across randomly generated splits.

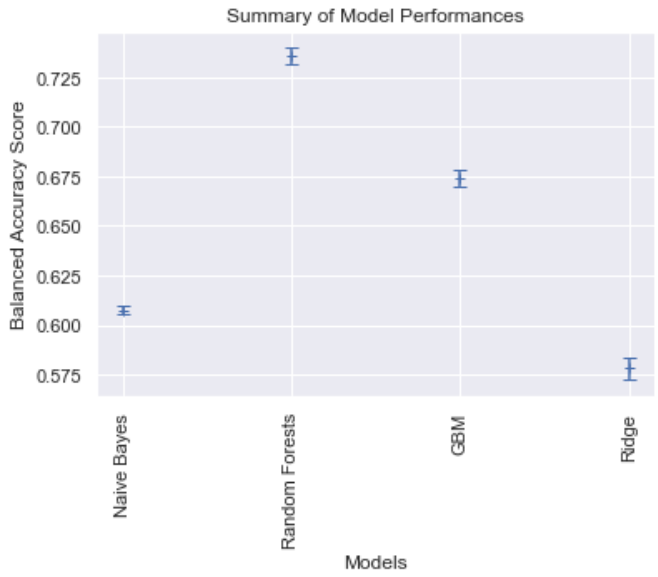


**Figure Five:** Testing performance across each model's random state and subsequent split. Confidence bands denote standard deviations for balanced accuracy score.

**Results**

General Results

I generate a baseline for the balanced accuracy score, the ratio between 1 and the number of classes, approximately 0.0588. We can also utilize accuracy as a baseline; we take the proportion of the most populace class in the training set and the size of the training set, approximately 0.2849. After hyperparameter tuning, it appears that all models perform above both baseline values. Table two summarizes the results of my models. As denoted by the standard deviations below, test scores have a relatively small spread.

[9] Brownlee, Jason. "Tune Hyperparameters for Classification Machine Learning Algorithms." Machine Learning Mastery, August 27, 2020.

Ridge classification is the least predictive of all models, and has the highest variation in test score. Naive Bayes slightly outperforms ridge regression and has half the spread of the ridge classifier. Both ensemble methods demonstrate higher predictive accuracy and have similar variability in test scores across random states. Overall, random forest methods boast the highest predictive accuracy of all models.

**Table Two: Summary Statistics across Machine Learning Models**

| | **Mean Test Scores** | **Standard Deviation of Test Scores** |
|---|---|---|
| Naive Bayes | 0.607573 | 0.002216 |
| Ridge Classifier | 0.578026 | 0.005678 |
| Gradient Boosted Trees | 0.674046 | 0.004513 |
| Random Forests | 0.735547 | 0.004227 |

Following from previous empirical studies, random forest classification boasts the highest prediction rate. Figure six illustrates the confusion matrix for the random forest classification model. Although relatively accurate across most classes, it appears that the random forest model was relatively ineffective at predicting several classes. False positives within diplomat, bus, proximity parking, and equipment violations suggest that class imbalances may be a contributing factor.

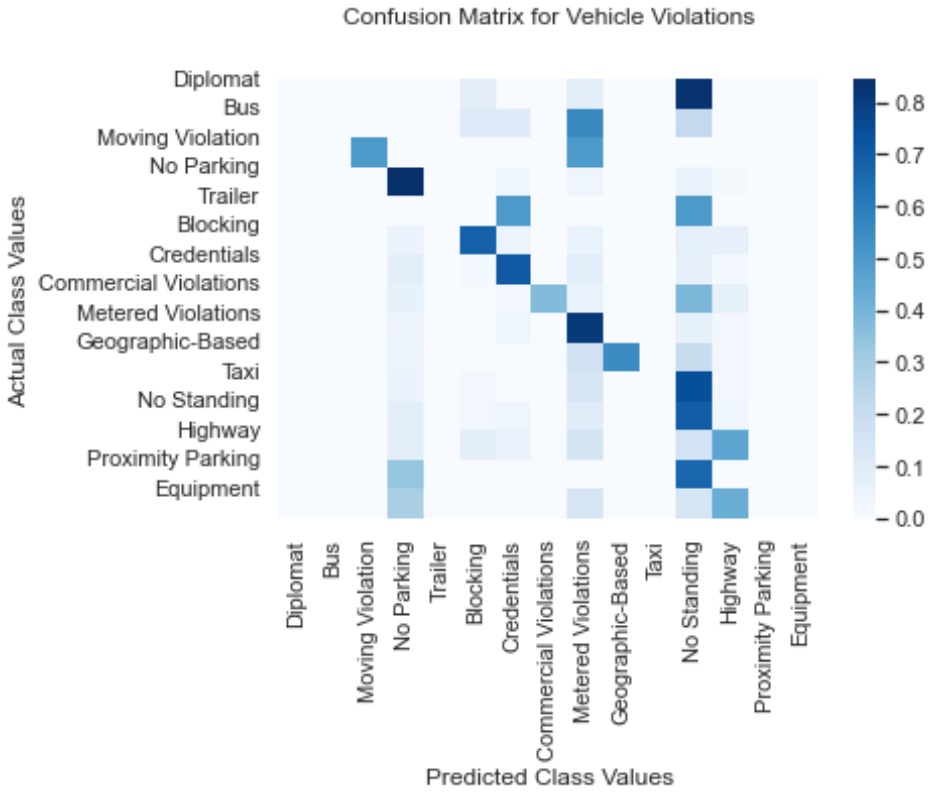


**Figure Six:** Confusion matrix for vehicle classifications, normalized for ease of interpretation

Feature Importance

While evaluating random forests's performance, we aggregate the Gini importance, or mean decrease in impurity (MDI), to evaluate global feature influence. Figure seven sums these results. Intuitively, features with greater decreases in impurity indicate improvement to the splitting criterion along every tree. Impactful features include building subdivisions (D, J, and H are the top three),[10] violation time, and vehicle distance from the curb. Interestingly, location-based features are not ranked within the top ten for MDI. Features are permuted, and SHAP values are computed to extract alternative global and local importance measures. Figures eight, nine, and ten exhibit significant overlap in feature importance; permutation and SHAP measures corroborate the significance of top features in MDI measures.

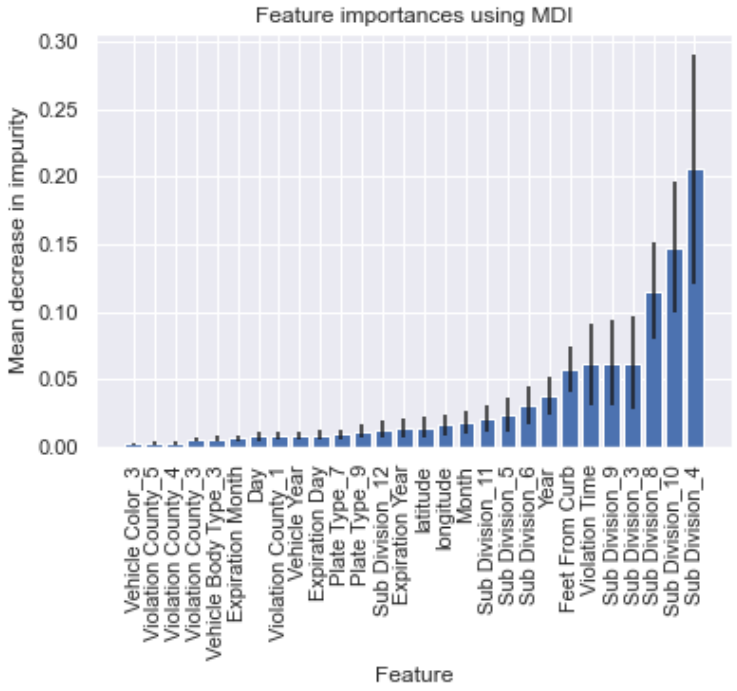


**Figure Seven:** Mean decrease in impurity for top 28 model features.

[10] Building codes given by NYC Department of Finance. See appendix and **https://www1.nyc.gov/assets/finance/jump/hlpbldgcode.html#Z** for more details

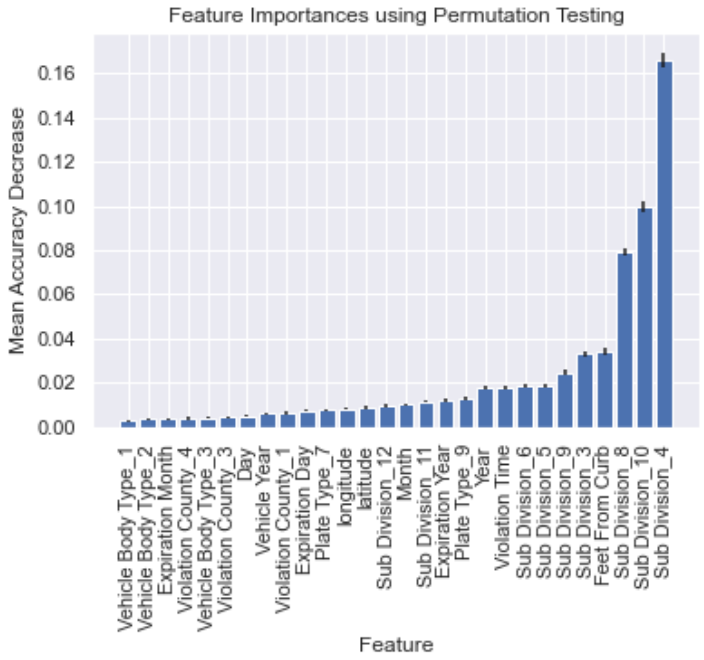


**Figure Eight:** Summary of feature importance based on permutation. Feature importance by magnitude is nearly universal compared to the Gini impurity measure.

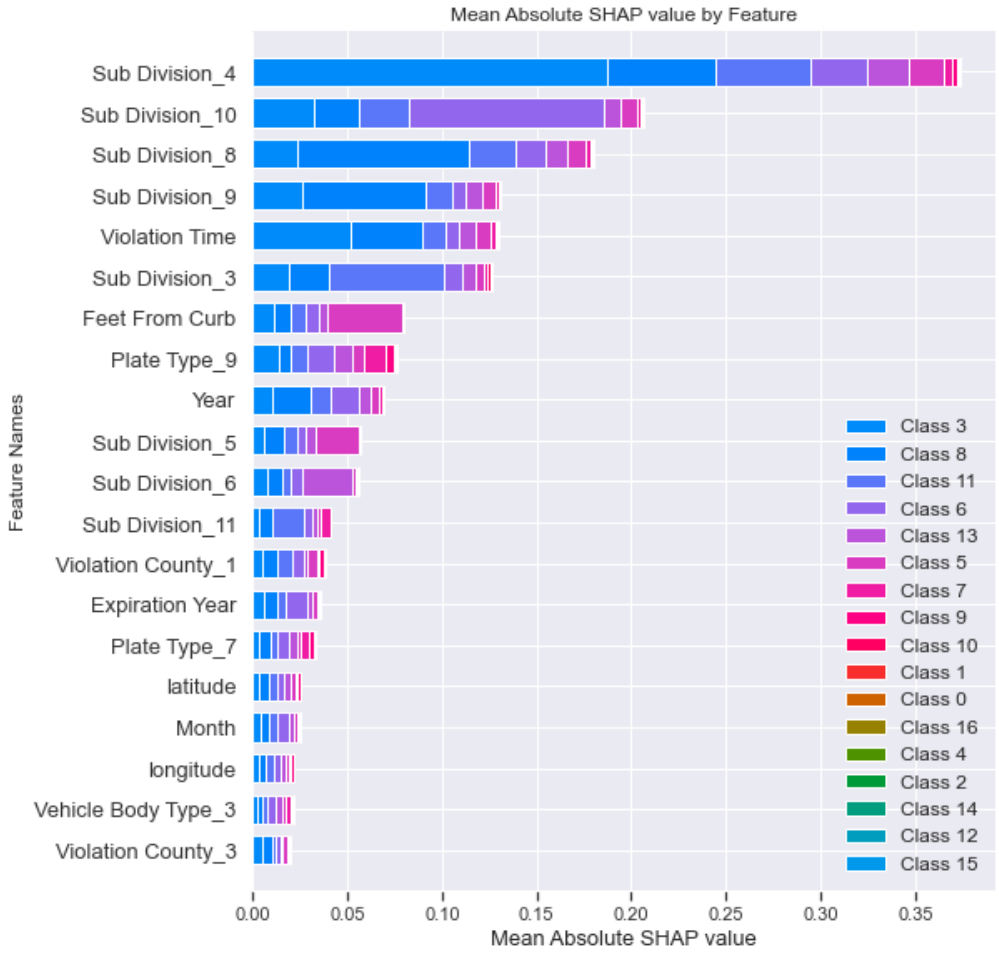


**Figure Nine**: Top 20 determinants of vehicle violation class by SHAP values. Outside of plate type 9 (trailers), subdivision codes, violation time, and feet from curb remain top violation determinants.


**Figure Ten:** Local feature importance for moving violation (class 3) at point 46.

The importance of subdivision four (apartments) stems from the city's high population density. Interestingly, subdivision ten (theaters and studios) remains high up on the feature importance list. We could attribute this to the density of sets, theaters, and studios that dot Manhattan, Brooklyn, and Queens and the equipment they utilize. Nonetheless, the elevated importance of building codes within our model is an unexpected result. In fact, anecdotal determinants of violations (i.e. red sports cars are more likely to be ticketed) do not seem to be as significant. Foreign vehicles and more 'distinct' vehicle colors rank among the least important features. This is not to preclude other relevant characteristics; a vehicle's age and location are still very relevant factors (top 25 features). Coupled with the 'seasonality' of a vehicle's violation time (tickets occur more frequently at some points in the day versus others), it appears that

ticketing rates occur in accordance with traffic flow. Features that hinder this flow, i.e. feet from curb, help the random forest model better determine violation classes.

**Outlook**

Future analyses can modify the structure of the dataset to improve efficacy. First, more sophisticated sampling methods could be used. The original dataset had approximately 12 million observations; a small, random sample was selected via a random seed and then fed into an API. Additionally, a larger sample could improve the model's generalizability. This analysis only evaluated a small subset. If additional outside information is found, the inclusion of previously removed uninterpretable features could improve model sophistication. In particular, policing-related features could enhance predictive accuracy.

Another potential source of improvement: feature engineering. In the raw dataset, many categorical features did not have standard entries. Reengineering efforts, with the aid of some outside materials, were based on intuition. A standard manner for feature engineering could be established to aid future analyses.

Finally, alternate modeling techniques could improve the overall results. For example, this analysis did not include XGBoost due to computational constraints.XGBoost, empirically, is an alternative that can also handle missing values efficiently. The general structure of the data, i.e. nonstandard entries and missing values, pairs well with this more complex model. If computational power permits, kernelized SVM could also be implemented to leverage geolocation data. Alternative multiclass models should be selected while accounting for the noise and misentry error underlying this data.

## Works Cited


Brownlee, Jason. “Tune Hyperparameters for Classification Machine Learning Algorithms.” Machine Learning Mastery, August 27, 2020. https://machinelearningmastery.com/hyperparameters-for-classification-machine-learning-algorithms/.

NYC Department of Finance. “Building Classification | City of New York” NYC Department of Finance. Accessed December 7, 2021. **https://www1.nyc.gov/assets/finance/jump/hlpbldgcode.html#Z**

NYC Department of Finance. “DOF Parking Codes.” NYC Open Data. Accessed October 10, 2021. **https://data.cityofnewyork.us/widgets/ncbg-6agr.**

NYC Department of Finance. “Violation Codes, Fines, Rules & Regulations.” NYC Department of Finance. Accessed October 10, 2021. https://www1.nyc.gov/site/finance/vehicles/services-violation-codes.page**.**

Mingxiao Li et al.. “A Data-Driven Approach to Understanding and Predicting the Spatiotemporal Availability of Street Parking.” SIGSPATIAL '19: Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. 2019. doi.org/10.1145/3347146.3359366.

Song Gao et al.. “Predicting the Spatiotemporal Legality of On-Street Parking Using Open Data and Machine Learning.” Annals of GIS. 25:4. 299-312

## Appendix

### A. Violation Group Code

| Original Coding | Renamed | Short Description |
|---|---|---|
| 100 | Diplomat | Parking violations committed by diplomats |
| 200 | Bus | Parking violations committed by busses |
| 300 | Moving Violation | Violations where person was moving |
| 400 | No Parking (Gen) | General No Parking signs present |
| 500 | Trailer | Commercial trailers or hauling involved |
| 600 | Blocking | Motorist obstructed path |
| 700 | Credentials | Improper credentials, plates, or tags |
| 800 | Commercial Violations | Violations committed by commercial vehicles2 |
| 900 | Metered Violations | Parking meters involved in violation |
| 1000 | Geographic-Based | Specific geographies with specific rules on parking |
| 1100 | Taxi | Violations caused by taxis or other ride-services |
| 1200 | No Standing | General no standing violation |
| 1300 | Highway | Violations committed on a highway |
| 1400 | Proximity Parking | Vehicle proximity violates regulations |
| 1500 | Equipment | Improper equipment for vehicle |
| 1600 | Commerce | Business/Commerce related violations |
| 1700 | Other | Miscellaneous |

### B. Categorical Features (Involved in Preprocessing Only)

| Feature Name | Encoder | Short Description |
|---|---|---|
| Registration State | OneHot | Vehicle's home state denoted by license plate |
| Plate Type | OneHot | Plate class for vehicle |
| Vehicle Body Type | OneHot | Vehicle body style |
| Vehicle Make | OneHot | Vehicle manufacturer |
| Issuing Agency | OneHot | NYC Administrative agency involved in administering violation |
| Violation County | OneHot | NYC borough where violation occurred |
| Violation in Front of Or Opposite | OneHot | Violation occurred in front of or opposite building |
| Sub Division | OneHot | Code for building vehicle parked in front of |
| Vehicle Color | OneHot | Vehicle's color |

### C. Numerical Features (Involved in Preprocessing Only)

| Feature Name | Encoder | Short Description |
|---|---|---|
| Vehicle Year | StandardScaler | Vehicle's manufacture year |
| Distance from Curb | StandardScaler | Vehicle distance from curb |
| Year | StandardScaler | Year in which parking violation occurred |
| Month | StandardScaler | Month in which parking violation occurred |
| Day | StandardScaler | Day in which parking violation occurred |
| Violation Time | StandardScaler | Scaled value for time vehicle violation recorded |
| Longitude | StandardScaler | Longitude of vehicle violation's location |
| Latitude | StandardScaler | Latitude of vehicle violation's location |
| Expiration Year | StandardScaler | Vehicle registration expiration year |
| Expiration Month | StandardScaler | Vehicle registration expiration month |
| Expiration Day | StandardScaler | Vehicle registration expiration day |

### D. Vehicle Body Type - Categorizations

| Category Value | Definition |
|---|---|
| 1 | SUV |
| 2 | Sedan |
| 3 | Van |
| 4 | Truck |
| 5 | Bike |
| 6 | Trailers |
| 7 | Public Transit |
| 8 | Miscellaneous |

### E. Vehicle Make - Categorizations

| Category Value | Definition |
|---|---|
| 1 | Japanese |
| 2 | American |
| 3 | German |
| 4 | Italian |
| 5 | Korean |
| 6 | British |
| 7 | Swedish |

| | |
|---|---|
| 8 | Canadian |

### F. Vehicle Color- Categorizations

| Category Value | Definition |
|---|---|
| 1 | Grey |
| 2 | Purple |
| 3 | White |
| 4 | Black |
| 5 | Red |
| 6 | Brown |
| 7 | Orange |
| 8 | Blue |
| 9 | Yellow |
| 10 | Green |
| 11 | Tan |
| 12 | Miscellaneous |

### G. Vehicle Registration State - Categorizations

| Category Value | Definition |
|---|---|
| 1 | Northeast |
| 2 | Midwest |
| 3 | South |

| | |
|---|---|
| 4 | West |
| 5 | International |
| 6 | Government |

## H. Vehicle Plate Type- Categorizations

| Category Value | Definition |
|---|---|
| 1 | Emergency |
| 2 | Military |
| 3 | Veteran |
| 4 | Government |
| 5 | Agricultural |
| 6 | Motorcycle |
| 7 | Passenger |
| 8 | Dealer |
| 9 | Commercial |
| 10 | Trailer |
| 11 | Public |
| 12 | Sports |
| 13 | Non-NYC registered |
| 14 | Miscellaneous |
| 15 | Carrier |

| | |
|---|---|
| 16 | Unknown |

## I. Building Code Subdivisions - Categorizations

| Category Value | Building Categories, |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | D |
| 5 | E |
| 6 | F |
| 7 | G |
| 8 | H |
| 9 | I |
| 10 | J |
| 11 | K |
| 12 | L |
| 13 | M |
| 14 | N |
| 15 | O |

| | |
|---|---|
| 16 | P |
| 17 | Q |
| 18 | R |
| 19 | S |
| 20 | T |
| 21 | U |
| 22 | V |
| 23 | W |
| 24 | X |
| 25 | Y |
| 26 | Z |
| 27 | Unknown |