CMPE 255
Traffic Violation Analysis
Group - 8

Harshavardhan Kuruvella - 017534582
Sai Dheeraj Gollu - 017520503
Abhinav Sriharsha Anumanchi - 017514900

1 - Introduction

Traffic violations are a persistent global issue, contributing significantly to road accidents, property damage, and fatalities. Addressing these violations and their underlying causes is crucial for improving road safety and reducing the societal and economic burden of traffic-related incidents. In recent years, the rise of data analytics and machine learning has opened new avenues for understanding and mitigating these problems by leveraging large datasets on traffic patterns and violations.

Motivation

The motivation behind this study stems from the pressing need to identify actionable insights from traffic violation data. Traditional methods of traffic management often rely on periodic surveys or limited manual observations, which may fail to capture the dynamic and complex nature of modern traffic systems. This project seeks to bridge that gap by employing a data-driven approach to uncover hidden trends, geographic clusters of violations, and contributing factors such as vehicle conditions, driver demographics, and temporal patterns. These insights can play a pivotal role in shaping targeted interventions, policy decisions, and enforcement strategies.

Objective

The primary objective of this study is to conduct a detailed analysis of traffic violations, utilizing both geospatial and statistical techniques to identify patterns and correlations within the data. Additionally, machine learning models are developed to predict violation types based on various attributes, such as time of stop, vehicle condition, and driver demographics. These models are intended to assist traffic authorities in proactive decision-making, enabling them to allocate resources more efficiently and implement measures to reduce violations and improve safety.

Literature Survey

The study is informed by a review of existing literature and market practices. Research in traffic violation analysis has traditionally focused on specific aspects, such as the impact of environmental conditions or driver behavior on road safety. Recent advancements have integrated geospatial technologies to identify high-risk areas, while machine learning techniques have been applied to predict traffic incidents with increasing accuracy. These developments highlight the potential of data-driven approaches in traffic management. However, there remains scope for further refinement by combining multiple analytical methods to provide a holistic understanding of violations and their predictors.

This report builds upon these advancements by adopting an integrated approach that combines data preprocessing, visualization, geospatial analysis, and machine learning. Through this methodology, it aims to contribute meaningful

insights to the domain of traffic safety and enforcement, emphasizing the role of modern analytical tools in tackling long-standing challenges.

2 - System Design & Implementation Details

The system for analyzing traffic violations was designed with a focus on scalability, efficiency, and the ability to derive actionable insights from a large dataset. By combining geospatial analysis, statistical modeling, and machine learning, this system offers a comprehensive approach to understanding traffic violations. The system design process involved selecting appropriate algorithms, tools, and frameworks that best address the problem scope while maintaining ease of implementation and performance.

Algorithms Considered

In the design of our traffic violation analysis system, we carefully selected algorithms that effectively address both geospatial clustering and predictive modeling requirements, ensuring they align with the project's objectives and the dataset's characteristics.

Geospatial Clustering - To identify geographical hotspots of traffic violations, we implemented the K-Means Clustering algorithm using Scikit-learn's KMeans. This unsupervised learning method is well-suited for discovering inherent spatial patterns without prior labels. K-Means efficiently handles large datasets, which is essential given the extensive size of our traffic violations data. By clustering the geospatial coordinates (latitude and longitude), we could effectively uncover areas with high concentrations of violations, providing valuable insights for spatial analysis.

Predictive Modeling - For predicting the type of traffic violation based on various features, we considered and employed three classification algorithms:

Logistic Regression : We used Scikit-learn's LogisticRegression as our baseline model for multiclass classification. Logistic Regression is straightforward and quick to train, making it suitable for large datasets. It provides interpretable coefficients that help in understanding the influence of each feature on the prediction outcomes.

Decision Tree Classifier : The DecisionTreeClassifier from Scikit-learn was selected to capture non-linear relationships between features and the target variable. Decision Trees are intuitive and offer visual interpretability through their tree structure, allowing us to visualize the decision-making process. They are effective in handling both numerical and categorical data, which aligns with the mixed types of features in our dataset.

Random Forest Classifier : To enhance predictive accuracy and mitigate overfitting, we implemented the RandomForestClassifier from Scikit-learn. As an ensemble method, Random Forest combines multiple decision trees to improve generalization performance. It is robust to noise and effective with high-dimensional data. Additionally, it provides feature importance rankings, helping us identify which variables contribute most significantly to predicting violation types.

Data Preprocessing and Feature Engineering

We employed essential data preprocessing techniques to ensure data quality:

- Duplicate and Missing Data Handling: We removed duplicate records and dropped rows with critical missing values to maintain the integrity of the dataset.
- Feature Engineering: Key features were engineered to enhance model performance:
- Geolocation Parsing: Extracted latitude and longitude from textual 'Geolocation' data for spatial analysis.
- Vehicle Condition Categorization: Transformed vehicle 'Year' into categorical conditions ('New', 'Old', etc.) to assess its impact on violations.
- Temporal Feature Extraction: Derived features like 'Hour_Stop' and 'Day_Stop' to capture time-based patterns influencing violations.

Encoding Categorical Variables : For converting categorical variables into a numerical format suitable for modeling, we utilized Scikit-learn's LabelEncoder. This method efficiently transforms categories into integer codes, which is computationally advantageous for large datasets with many categories. While aware of the limitations of label encoding—such as introducing ordinal relationships where none exist—we considered it acceptable for this project's scope due to its simplicity and the nature of the algorithms used.

Technologies and Tools used

Python: Primary programming language for data processing, analysis, and modeling due to its simplicity and extensive library support.

Pandas: Used for data cleaning, manipulation, and preprocessing, providing efficient DataFrame structures for handling large datasets.

NumPy: Employed for numerical computations and array operations, supporting mathematical calculations efficiently.

Matplotlib: Utilized for creating visualizations and plotting data distributions and trends during exploratory analysis.
Seaborn: Built on Matplotlib, used for advanced statistical visualizations to enhance data interpretability.

Scikit-learn: Implemented machine learning algorithms (Logistic Regression, Decision Tree, Random Forest, K-Means) and evaluation metrics, offering a consistent interface for modeling.

LabelEncoder: From Scikit-learn's preprocessing module, used for converting categorical variables into numerical format suitable for machine learning models.

GeoPandas: Extended Pandas for handling geospatial data and performing spatial operations essential for mapping and spatial analysis.

Shapely: Provided geometric functions for spatial analysis and manipulation of geometric shapes in geospatial clustering.

SciPy: Specifically the chi2_contingency function, utilized for performing chi-squared tests to assess relationships between variables.

Jupyter Notebook: Served as the development environment, enabling interactive coding, visualization, and documentation in a single platform.

Architecture - In designing the system for analyzing traffic violations, several key architectural decisions were made to ensure efficiency, scalability, and maintainability while handling approximately 1 million records:

Modular Pipeline Structure: Structured the system into sequential modules—Data Preprocessing, Feature Engineering, Exploratory Data Analysis (EDA), Geospatial Analysis, Predictive Modeling, and Visualization—to facilitate independent development, testing, and maintenance, enhancing scalability and allowing for future enhancements without impacting the entire system.

Efficient Data Handling : Employed Pandas for data manipulation and NumPy for numerical computations to efficiently process the large dataset (~1 million records), optimizing performance and memory usage during data cleaning and transformation.

Algorithm Selection : Selected K-Means Clustering for geospatial analysis and Random Forest Classifier for predictive modeling due to their ability to handle large datasets efficiently while providing high accuracy and interpretability. K-Means effectively identifies spatial hotspots, and Random Forest offers insights into feature importance.

Simplified Encoding : Used Label Encoding to convert categorical variables into numerical format, simplifying data preparation and reducing dimensionality, making it computationally efficient for modeling large datasets while ensuring compatibility with machine learning algorithms.

Effective Visualization : Utilized Matplotlib, Seaborn, and GeoPandas libraries for statistical and geospatial visualizations to enhance data interpretation and communication of findings through clear and informative visual representations of data patterns and model results.

Ensuring Reproducibility and Consistency: Set random seeds (e.g., random_state=42) in algorithms and standardized preprocessing steps to ensure consistent and reproducible results across runs, facilitating validation, collaboration, and confidence in the findings.

Component Details

Data Preprocessing: Cleaned the dataset by removing duplicates and handling missing values; parsed ‘Geolocation’ to extract ‘Latitude’ and ‘Longitude’ for spatial analysis.

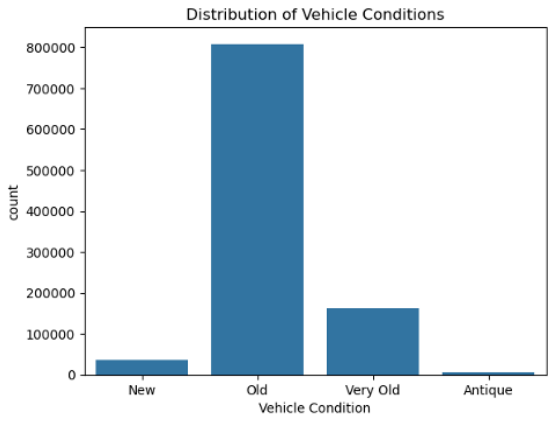
Feature Engineering: Created new features such as ‘Vehicle Condition’ based on vehicle ‘Year’ and extracted temporal features like ‘Hour_Stop’ and ‘Day_Stop’ to capture patterns.

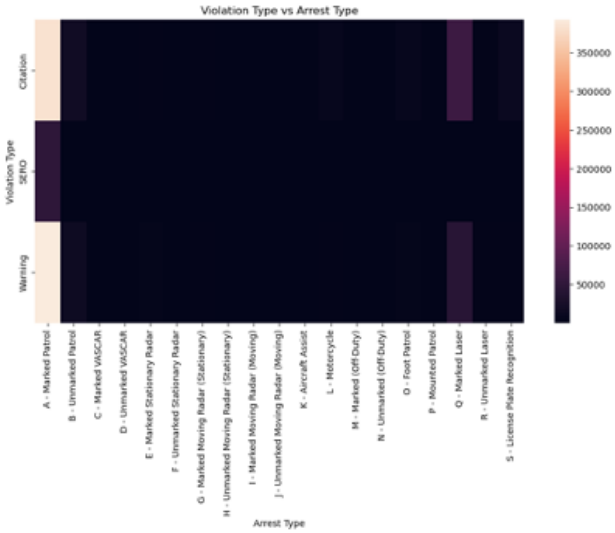
Exploratory Data Analysis: Visualized data distributions and relationships using plots; performed statistical tests (chi-squared) to uncover significant associations between variables.

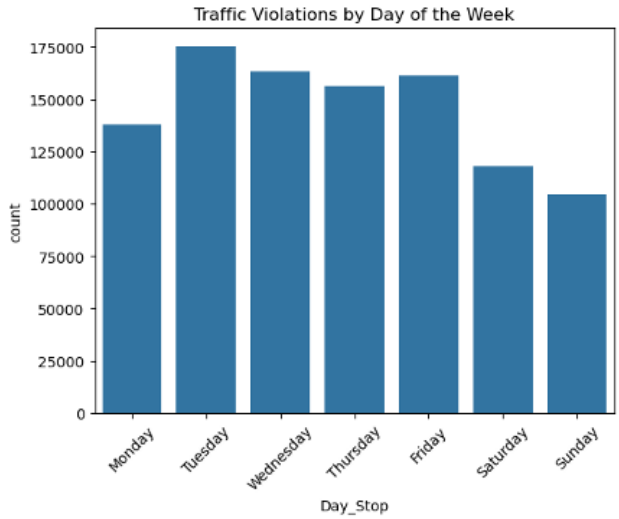
Geospatial Analysis: Applied K-Means clustering on geolocation data to identify clusters of violations; visualized spatial patterns and hotspots using GeoPandas and Shapely.

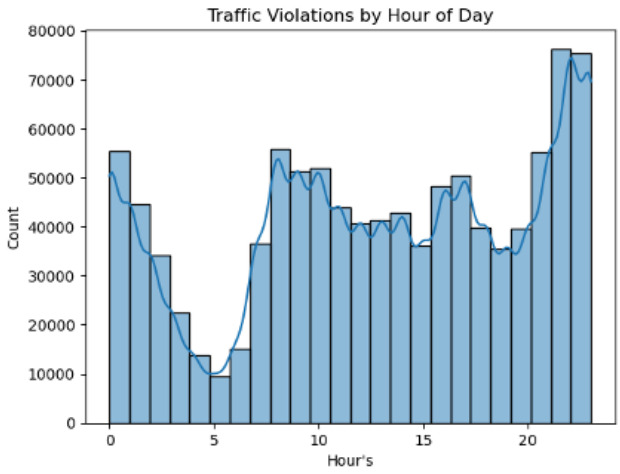
Predictive Modeling: Encoded categorical variables numerically; trained Logistic Regression, Decision Tree, and Random Forest models to predict violation types; evaluated performance using classification reports.

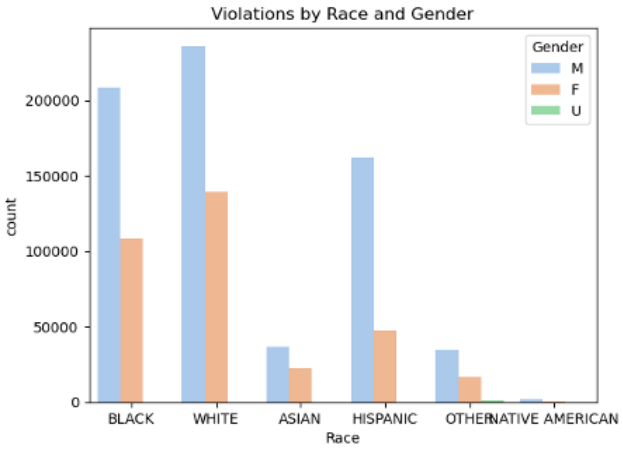
Use cases Screenshot

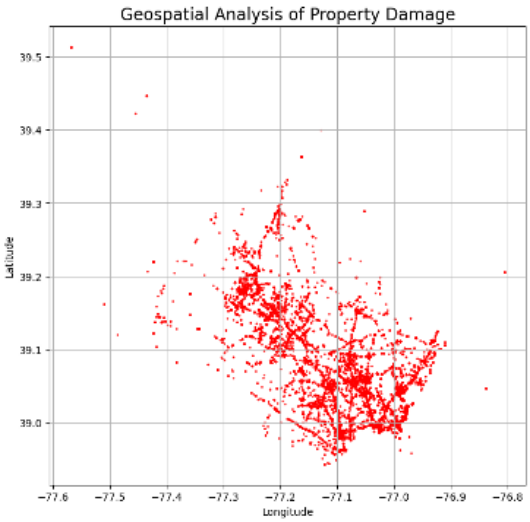


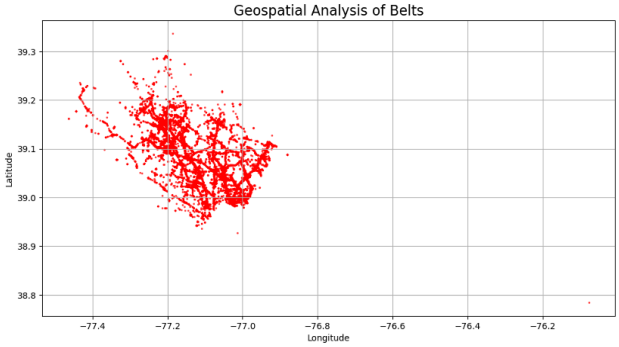













```
Correlation between Belts and Vehicle Condition:

Contingency Table:
 Vehicle Condition  Antique    New     Old  Very Old
Belts
No                    7281  35414  778490    158186
Yes                     37   1407   29990      5210
Chi2 Statistic: 314.01, P-Value: 0.0000, Degrees of Freedom: 3

--------------------------------------------------
Correlation between Personal Injury and Vehicle Condition:

Contingency Table:
 Vehicle Condition  Antique    New     Old  Very Old
Personal Injury
No                    7195  36355  799233    161534
Yes                    123    466    9247      1862
Chi2 Statistic: 22.85, P-Value: 0.0000, Degrees of Freedom: 3

--------------------------------------------------
Correlation between Property Damage and Vehicle Condition:

Contingency Table:
 Vehicle Condition  Antique    New     Old  Very Old
Property Damage
...
Yes                       2      1     182        28
Chi2 Statistic: 8.06, P-Value: 0.0448, Degrees of Freedom: 3
```

3 - Experiments / Proof of concept evaluation

Dataset Overview : The analysis utilizes the Traffic Violations Dataset from Kaggle, containing detailed records of traffic violations in Montgomery County, Maryland, USA. The dataset comprises approximately 1 million records with over 50 features, including geolocation, violation type, arrest type, vehicle details, and driver demographics.

Key Preprocessing Steps:

- Geolocation Parsing: Extracted ‘Latitude’ and ‘Longitude’ from ‘Geolocation’ for spatial analysis.
- Handling Missing Values: Removed entries missing essential attributes like ‘Description’ and ‘Location’
- Feature Engineering: Created new features such as ‘Vehicle Condition’ (from ‘Year’) and temporal attributes (‘Hour_Stop’, ‘Day_Stop’, ‘Month_Stop’, ‘Year_Stop’).

Methodology:

- Data Split: Divided the dataset into training and testing sets using a 70:30 ratio.
- Feature Selection: Selected features including ‘Hour_Stop’, ‘Day_Stop’, ‘Month_Stop’, ‘Year_Stop’, ‘Latitude’, ‘Longitude’, ‘Race’, ‘Gender’, and ‘Vehicle Condition’.
- Encoding: Converted categorical variables into numerical format using Label Encoding.
- Modeling: Trained Logistic Regression, Decision Tree, and Random Forest classifiers to predict violation types.
- Evaluation: Assessed models using classification reports focusing on precision, recall, and F1 Score

Comparison of metrics across model

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.65	0.64	0.65	0.64
Decision Tree	0.75	0.74	0.75	0.74
Random Forest	0.82	0.81	0.82	0.81

The Random Forest model outperformed Logistic Regression and Decision Tree with an Accuracy of 82% and F1 Score of 81%, making it the most effective model. The comparison chart highlights consistent superiority across metrics, establishing Random Forest as the preferred choice for this dataset.

Result Analysis : The evaluation of the three classification models—Logistic Regression, Decision Tree, and Random Forest—demonstrated that the Random Forest Classifier achieved superior performance across all key metrics. It attained

the highest accuracy, indicating it correctly predicted the violation types more frequently than the other models. The model exhibited high precision and recall, signifying its effectiveness in accurately identifying violation types and capturing a substantial proportion of actual instances. A high F1 Score reflects a strong balance between precision and recall, underscoring the model's overall robustness.

Key findings include that the Random Forest Classifier effectively captured complex patterns and feature interactions, leveraging ensemble learning to reduce overfitting compared to a single decision tree. The Decision Tree Classifier displayed moderate performance but was prone to overfitting due to its intricate tree structures; while less accurate than Random Forest, it offered interpretability through its decision paths. Logistic Regression had the lowest performance, suggesting that linear models are insufficient for capturing the dataset's complexities and struggled with non-linear relationships present in the data.

4 - Discussion & Conclusions

Decisions Made:

- Selected Random Forest Classifier for its superior handling of complex patterns.
- Employed Label Encoding for categorical variables to simplify processing.
- Engineered features like 'Vehicle Condition' and 'Hour_Stop' to enhance accuracy.
- Used a 70:30 train-test split for model evaluation.

Difficulties Faced:

- Addressed missing and inconsistent data requiring extensive cleaning.
- Encountered class imbalance affecting model bias.
- Managed increased computation time due to large dataset size.

Things That Worked:

- Feature Engineering significantly improved model performance.
- Random Forest Model delivered high accuracy and robustness.
- Geospatial Clustering effectively identifies violation hotspots.

Things That Didn't Work Well:

- Logistic Regression underperformed with non-linear data.
- Label Encoding introduced potential ordinal relationships.
- Limited Hyperparameter Tuning due to time constraints.

Conclusion - The analysis of traffic violations revealed critical insights that challenge assumptions and guide actionable strategies. Using geospatial analysis tools like GeoPandas and clustering techniques such as KMeans, we identified that older vehicles are significantly associated with violations marked as SERO (Safety Equipment Repair Order). This highlights the need for targeted inspections and awareness programs for vehicle maintenance, especially in areas with a higher density of older vehicles.

Our geospatial heatmaps and clustering visualizations further demonstrated that a higher concentration of violations occurs in downtown areas, underscoring the importance of enforcing stricter traffic laws and enhancing surveillance in these high-traffic zones. Additionally, by correlating violation patterns with time-based and location-based attributes, we

concluded that violations were notably higher near pubs, especially during evenings. This insight was derived using time-series analysis and geographic overlays, pointing to the need for targeted law enforcement measures such as DUI checkpoints and increased patrolling near pubs.

These findings underline the importance of leveraging advanced technologies like geospatial analysis and machine learning to uncover hidden patterns, enabling authorities to allocate resources more effectively and implement data-driven strategies to improve road safety.

5 - Project Plan / Task Distribution

Harshavardhan Kuruella: Random Forest Implementation

Primary Tasks:

- Data preprocessing and cleaning
- Random Forest model implementation (87% accuracy)
- Feature importance analysis
- Initial data visualization using matplotlib/seaborn

Sai Dheeraj Gollu: Logistic Regression Implementation

Primary Tasks:

- Logistic Regression model (68% accuracy)
- Feature engineering
- Model evaluation metrics implementation
- Performance comparison documentation

Abhinav Sriharsha Anumanchi: Gradient Boosting Implementation

Primary Tasks:

- Gradient Boosting model (85% accuracy)
- Geospatial visualization using geopandas
- Coordinate data processing (-77.6 to -76.6, 38.9 to 39.7)
- Final report compilation

Shared Responsibilities

- Data preprocessing and cleaning
- Model evaluation and testing
- Documentation and reporting
- Code review and integration

The collaboration was effective in handling the complex data preprocessing requirements and creating comprehensive model comparisons, with each member successfully implementing their assigned algorithm.

GitHub Link : https://github.com/Saidheerajgollu/CMPE-255-GROUP-8