CENTER FOR DEVELOPMENT OF ADVANCED COMPUTING

Report on
NYC Parking Tickets Analysis

PG-DBDA September 2023

Submitted by:

Project Team: 7

Niraj Wadhe **Srushti Vinchurkar**

Dhanashree Bhor **Udayraj Kothavade**

Premkumar Jadhav

Project Guide-

Mr. Sarvesh Raut

1. INTRODUCTION:

In the vibrant cityscape of New York City, a common dilemma shared by countless other bustling metropolises arises the struggle with finding adequate parking spaces. The usual situation of having many cars and not enough space in the city leads to a lot of parking tickets. To understand why this happens, the NYC Police Department has carefully collected data on parking tickets. These collections of information cover many years and are available to everyone through NYC Open Data, making it easier for us to study and learn more about why parking violations occur.

For our project, we aim to analyze this data using Azure and PySpark. PySpark will enable us to process the large data files quickly, while Azure provides the infrastructure for our analysis. Additionally, we will use Power BI to visualize the insights we uncover. Our focus will be on parking tickets issued from the year 2020 to March 2024. Through this analysis, we will gain valuable insights into New York City's parking situation.

Dataset:

The NYC Department of Finance gathers information about each parking ticket given out in the city, which adds up to around 10 million tickets every year. They share this information with the public to help people deal with their tickets and to help officials make decisions about policies. This data is made publicly available on the NYC Open Data website.

These files include five sets of data, spanning from June 2020 to March 2024, obtained from NYC Open Data. They contain records of parking violations issued each fiscal year. Each column in the dataset provides details about the violation, like the vehicle involved, the type of ticket given, where it happened, and when. Overall, the dataset contains approximately 23 million rows and takes up 12 GB of space.

2. PROBLEM STATEMENT:

NYC Parking Violations Detection: Investigate New York City, USA's parking ticket data from 2020 to 2024, gathered by the Finance Department. Utilize exploratory data analysis (EDA) and machine learning to uncover trends and make predictions. The goal is to devise strategies for minimizing parking infractions and enhancing traffic management, supported by data visualization techniques for comprehensive insights and forecasting of high-violation areas.

Objective:

This project primarily focuses on exploring the use of Spark, Machine Learning Algorithms and Microsoft services such as Azure Cloud, Azure Data Factory, Blob Storage, and Databricks. One of the main goals of this study is to become familiar with analyzing data using PySpark instead of regular Python. Additionally, we aim to grasp the basic concepts of Databricks as an ETL (Extract, Transform, Load) tool and Power BI as a business intelligence tool.

The objective of this project is to conduct a detailed exploration of the data, aiming to understand its patterns and uncover insights using machine learning algorithms**.**

3.FUNCTIONAL REQUIREMENTS:

3.1 Modules Used:

1. PySpark
2. SparkSQL
3. Azure Data Factory
4. Azure Blob Storage
5. Azure Databricks
6. Python
7. Machine Learning Algorithm
8. Power BI

3.1.1 PySpark:

- PySpark serves as the Python interface for Apache Spark, a widely used open-source distributed computing framework.
- This Python API enables seamless interaction with Spark Data Frames, facilitating data manipulation and analysis.
- Using PySpark, we can leverage Python APIs to perform ETL (Extract, Transform, Load) transformations on our data. Additionally, PySpark provides a convenient interface for conducting interactive data analysis in a distributed environment through the PySpark Shell.
-

3.1.2 SparkSQL:

- SparkSQL is the component of Apache Spark that allows users to work with structured data using SQL queries.

- This feature provides a familiar SQL interface for querying data stored in Spark Data Frames, making it easy to analyze structured data.
- With SparkSQL, we can execute SQL queries directly on their Spark Data Frames, enabling seamless integration of SQL-based operations into their Spark applications.
- SparkSQL provides advanced optimization techniques to improve query performance, making it an efficient tool for processing large-scale datasets.

3.1.3 Azure Data Factory:

- Azure Data Factory is a cloud-based service provided by Microsoft Azure. It assists in creating, scheduling, and managing data integration pipelines.
- These pipelines automate data movement and transformation tasks between various sources and destinations, including on-premises and cloud environments.
- Azure Data Factory offers monitoring and logging features for pipeline management, ensuring smooth execution and troubleshooting.
- Additionally, it seamlessly integrates with other Azure services, enhancing data integration capabilities with advanced analytics and storage solutions within the Azure ecosystem.

3.1.4 Azure Blob Storage:

- Azure Blob Storage provides scalable cloud storage for large amounts of unstructured data, such as images, videos, documents, and log files.
- Users only pay for the storage they use, with no upfront costs.
- Data is replicated across multiple data centers for high durability and resilience.
- It seamlessly integrates with other Azure services for building comprehensive solutions.
- Azure Blob Storage provides robust security features, including encryption and access controls

3.1.5 Azure Databricks:

- Azure Databricks provides a unified analytics platform that combines data engineering, data science, and business analytics capabilities in one integrated environment.
- It offers scalable data processing using Apache Spark, enabling users to process large volumes of data efficiently and cost-effectively.
- Azure Databricks provides a collaborative workspace where data engineers, data scientists,

and analysts can work together on data-driven projects, share code, and collaborate in real time.

- It seamlessly integrates with other Azure services such as Azure Data Lake Storage, Azure SQL Database, and Azure Machine Learning, enabling users to build end-to-end analytics pipelines.
- Azure Databricks is a fully managed service, meaning that Azure takes care of infrastructure provisioning, maintenance, and security, allowing users to focus on data analysis and innovation rather than managing infrastructure.

3.1.6 Python:

In our project, we utilized several Python libraries to facilitate data processing, analysis, and visualization:

- **Pandas**:

  Pandas is a powerful data manipulation and analysis library. It provides data structures and functions to efficiently manipulate large datasets, offering tools for data cleaning, transformation, and exploration. With Pandas, tasks such as data filtering, grouping, and merging become streamlined, making it a fundamental tool for data wrangling.

- **NumPy**:

  NumPy is a fundamental library for numerical computing in Python. It offers support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently. NumPy's capabilities are essential for various numerical computations, including linear algebra, statistical analysis, and mathematical operations on arrays.

- **Scikit-learn**:

  Scikit-learn is a versatile machine-learning library that provides a wide range of algorithms and tools for data mining and analysis. It offers easy-to-use interfaces for implementing supervised and unsupervised learning algorithms, including classification, regression, clustering, dimensionality reduction, and model evaluation. Scikit-learn simplifies the process of building and deploying machine learning models, making it suitable for both beginners and experienced practitioners.

- **Matplotlib:**

  Matplotlib is a comprehensive plotting library in Python that enables the creation of high-quality visualizations for data analysis and presentation. It offers a wide variety of plotting functions and customization options to create diverse types of plots, including line plots, scatter plots, bar charts, histograms, and more. Matplotlib's flexibility and extensive functionality make it a go-to choose for generating insightful and visually appealing plots to communicate data insights effectively.

3.1.7 Machine Learning Algorithm:

In our project, we have employed various machine learning algorithms to analyze NYC parking ticket data. These algorithms include:

- **Support Vector Machine (SVM):**

  In our project, we utilized a Support Vector Machine (SVM), a powerful algorithm for classification tasks. SVM works by finding the best possible boundary (hyperplane)

between different classes in the dataset. It's effective in handling both linear and non-linear data, making it suitable for a wide range of applications.

- **K-Nearest Neighbors (KNN):**
  We also utilized K-Nearest Neighbors (KNN) in our project. KNN is a simple yet effective algorithm for classification and regression tasks. It classifies data points based on the majority vote of their nearest neighbors. It's particularly useful when the underlying data distribution is not well understood or when there are no clear boundaries between classes.

- **Gradient Boosting:**
  Additionally, Gradient Boosting was employed in our project. Gradient Boosting is an ensemble learning technique that builds multiple weak learners sequentially, with each one correcting the errors of its predecessor. It's known for its high predictive accuracy and ability to capture complex relationships in the data.

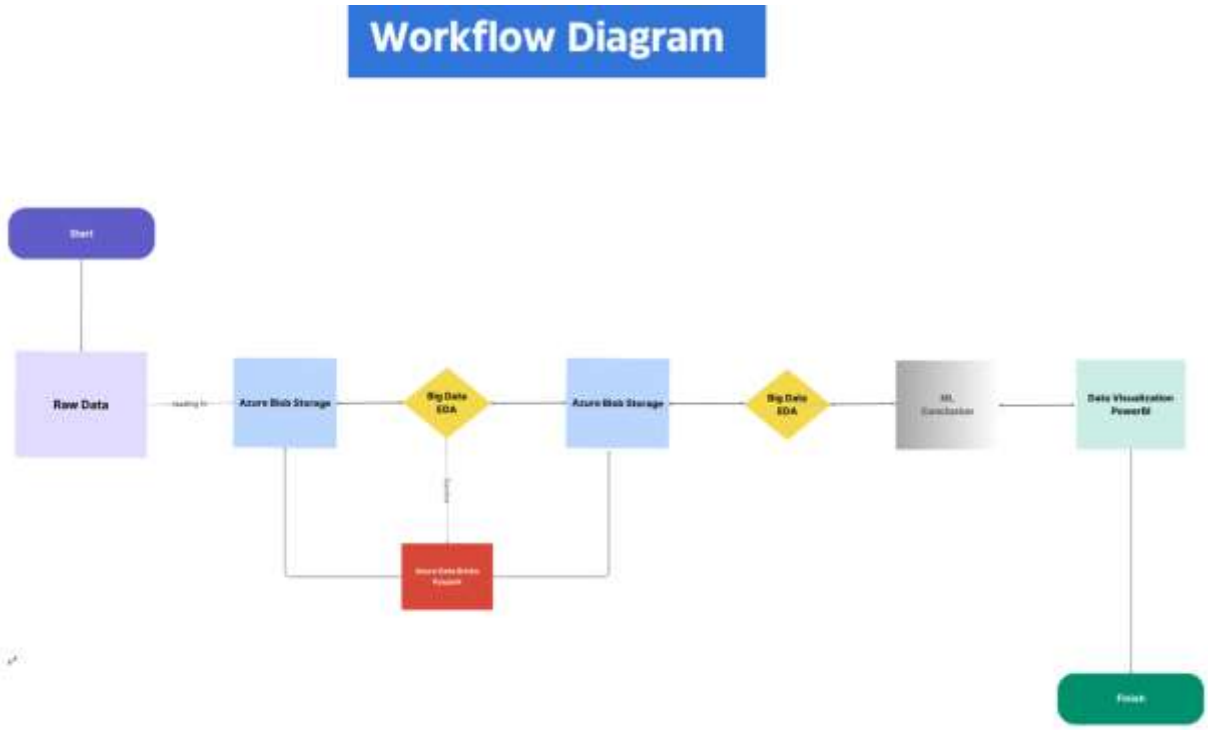
These algorithms were instrumental in analyzing the NYC parking ticket data and extracting valuable insights for our project

3.1.8 Power BI:

- Power BI enables users to create interactive and visually appealing dashboards and reports to visualize data.
- It offers connectivity to a wide range of data sources, allowing users to import and analyze data from multiple sources

- Power BI empowers users to perform self-service business intelligence tasks, such as data modeling, analysis, and visualization, without extensive technical expertise.

4. PROJECT FLOW



5. METHODOLOGY

5.1 Data Used:

The dataset utilized in this project encompasses parking violations recorded in New York City throughout the year 2024, presented in CSV format. Comprising 43 distinct features for each ticket, it encompasses details regarding the vehicle, registration information, violation type, geographical location, borough, precinct, and street details, as well as the rank and division of the ticket issuer. The dataset's richness in information makes it particularly promising for in-depth analysis and exploration.

5.2 ETL Process Implementation:

5.2.1 Data Extraction:

In preparation for data integration, the first stage in the ETL process is to extract or pull data from relevant sources and amalgamate it. This process involves the necessary groundwork for facilitating the subsequent integration of data.

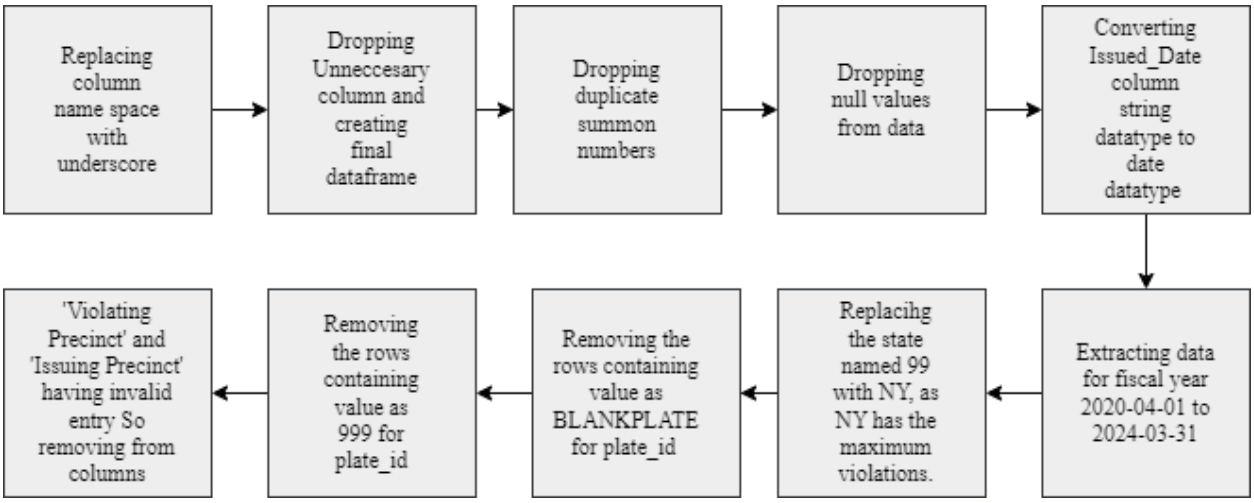
To commence the data organization phase, the primary ETL step entails the extraction or retrieval of data from pertinent sources, followed by its consolidation. This process includes the essential groundwork to set the stage for effective data integration.

Raw data was dumped into Azure Blob Storage. It is an Object Storage i.e., Azure Blob Container was used for storing the large amount of data.

5.2.2 Data Transformation:

After completing the initial ETL deployment, the next step in the ETL process is Data Transformation. In this phase, the extracted data undergoes various operations such as compilation, conversion, reformatting, and cleansing within the staging area. The refined data is then prepared for transfer into the target database in the subsequent step.

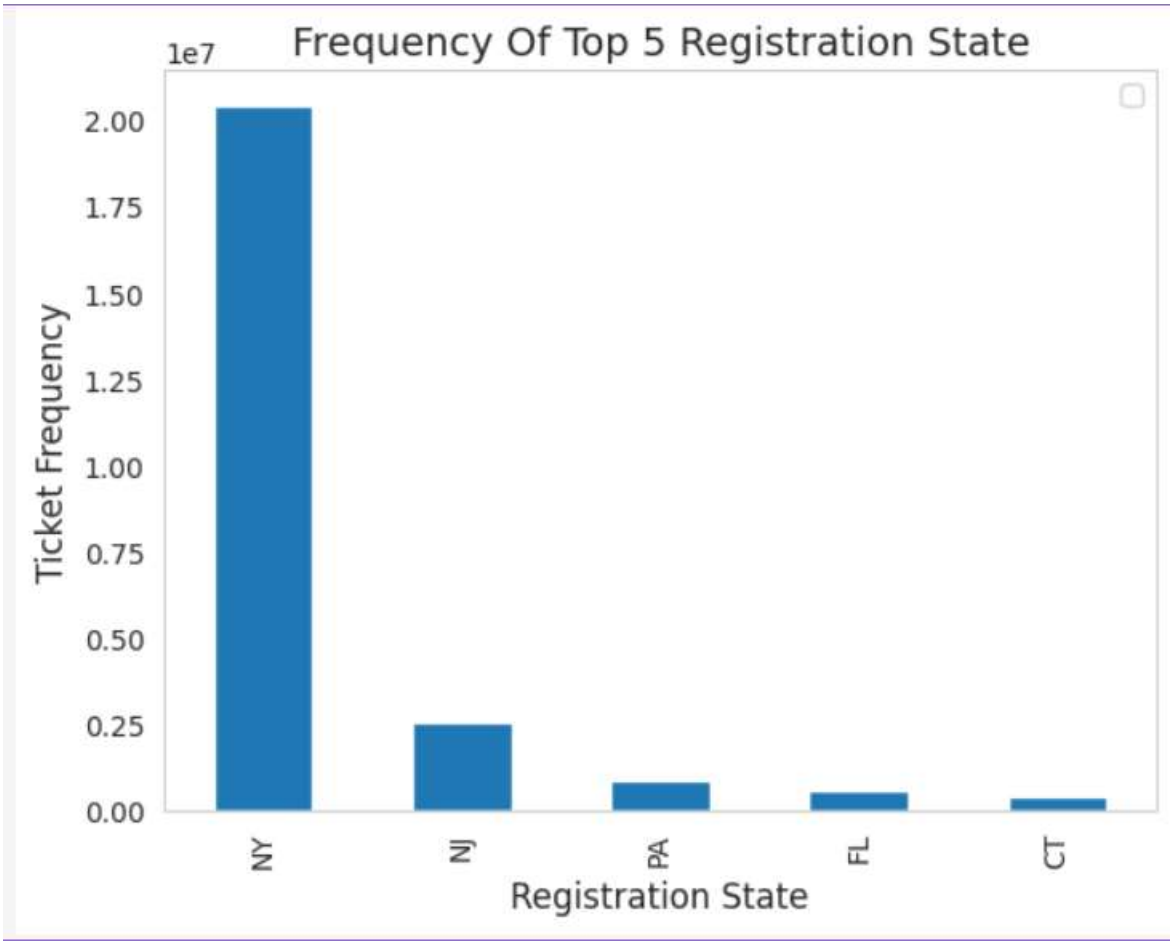
Transformations:



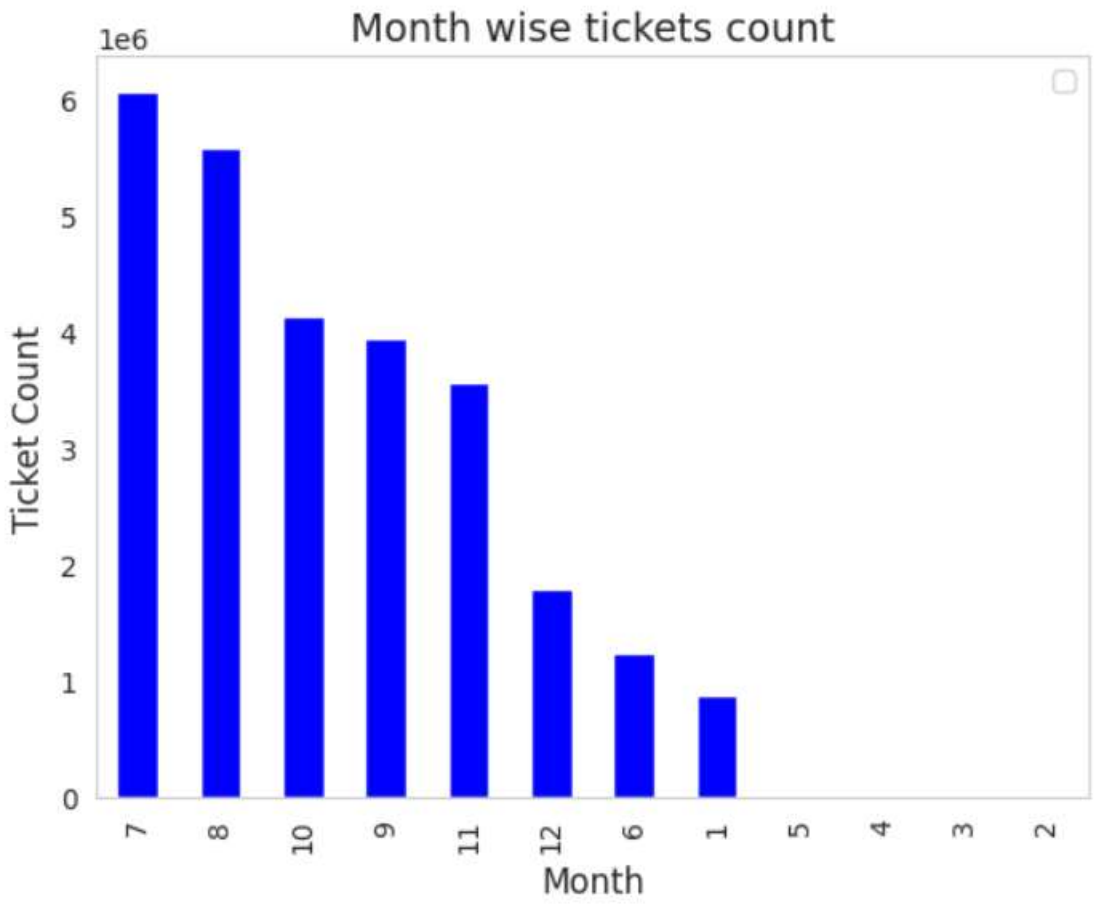
5.2.3 Data Loading

The concluding step in the three-step data ETL process is loading the datasets that have been extracted and transformed earlier into the target database. We load transformed data into Azure Blob Container.

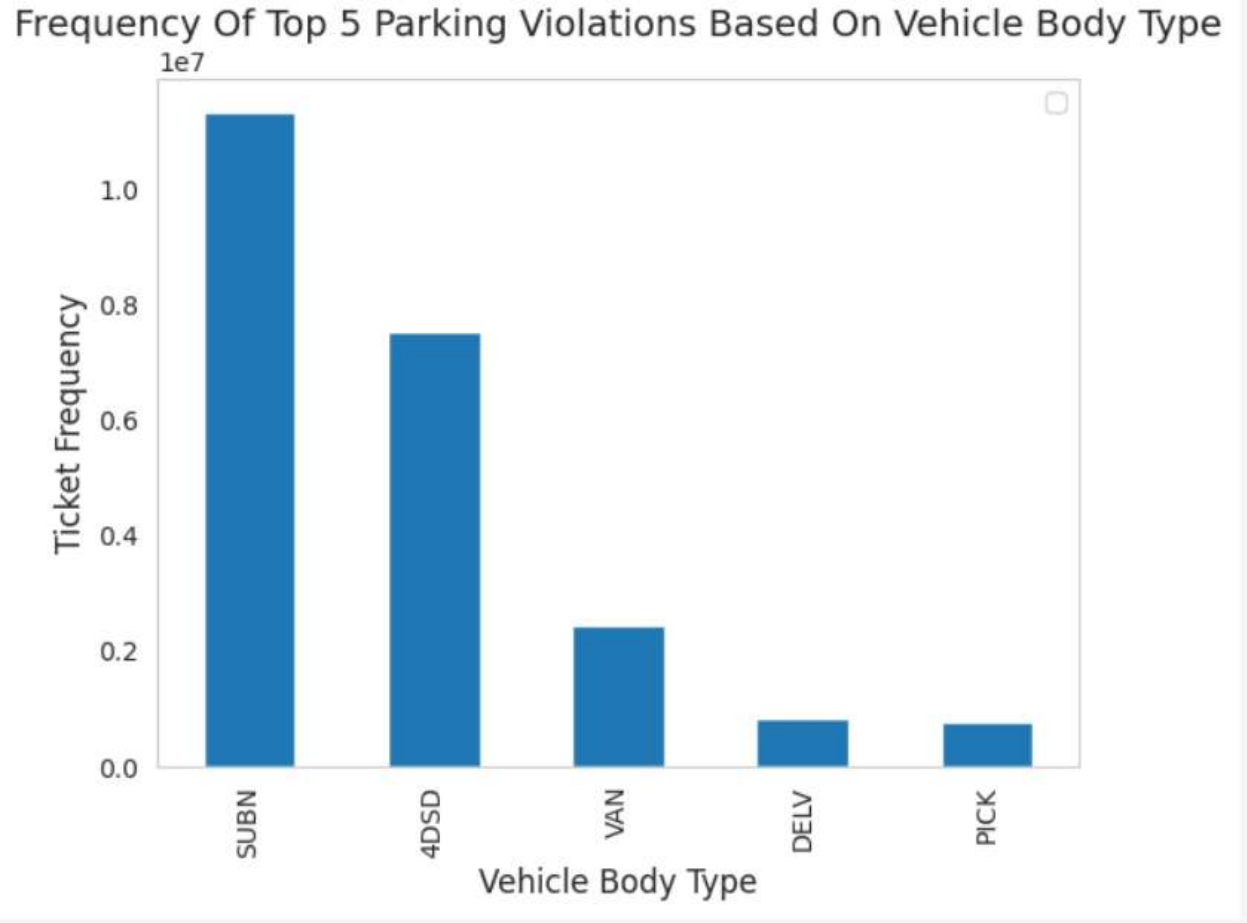
6. Observations:



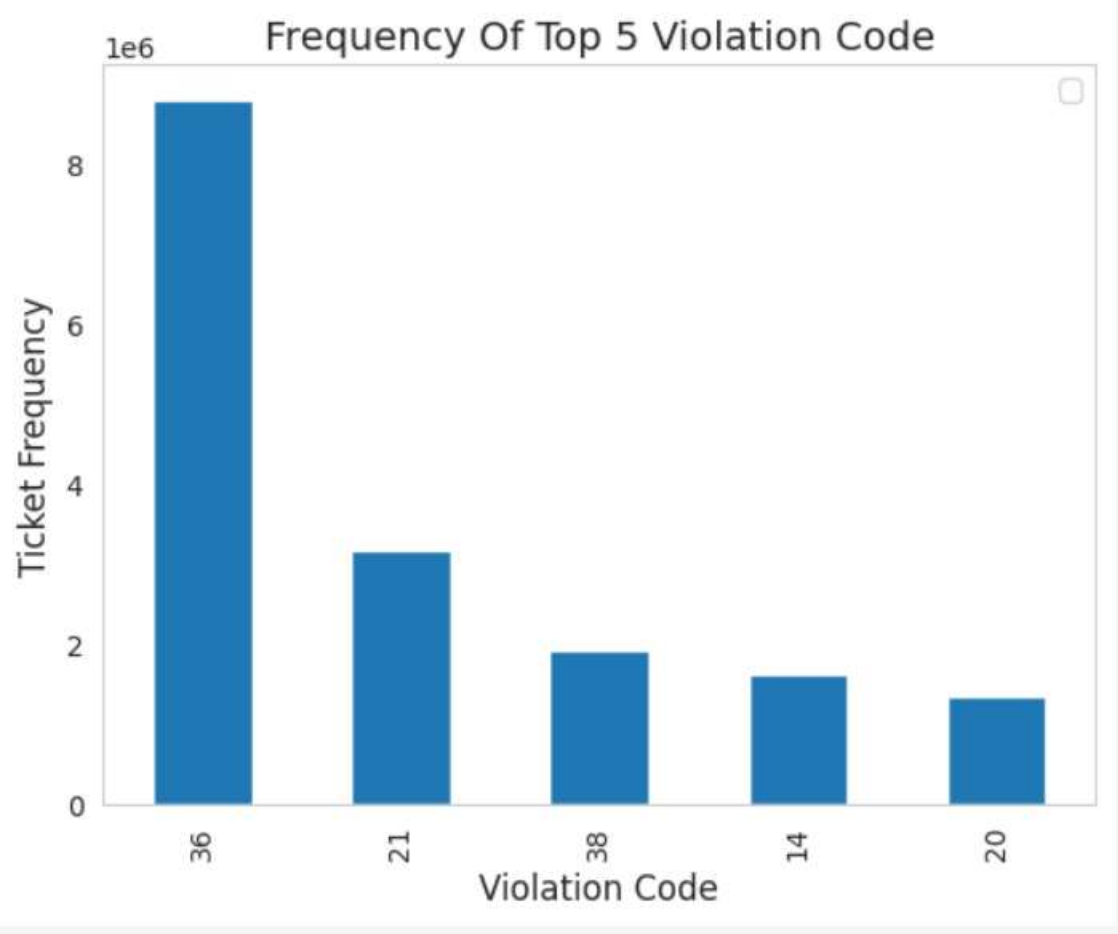
- There are 64 unique states where the cars that got parking tickets came from.
- New York has the highest parking violations followed by New Jersey and Pennsylvania.



- The number of parking tickets sold varies considerably from month to month.
- The highest number of tickets were sold in July followed by August and October.
- There is a general upward trend in the number of tickets sold from February to July, followed by a downward trend from July to December.



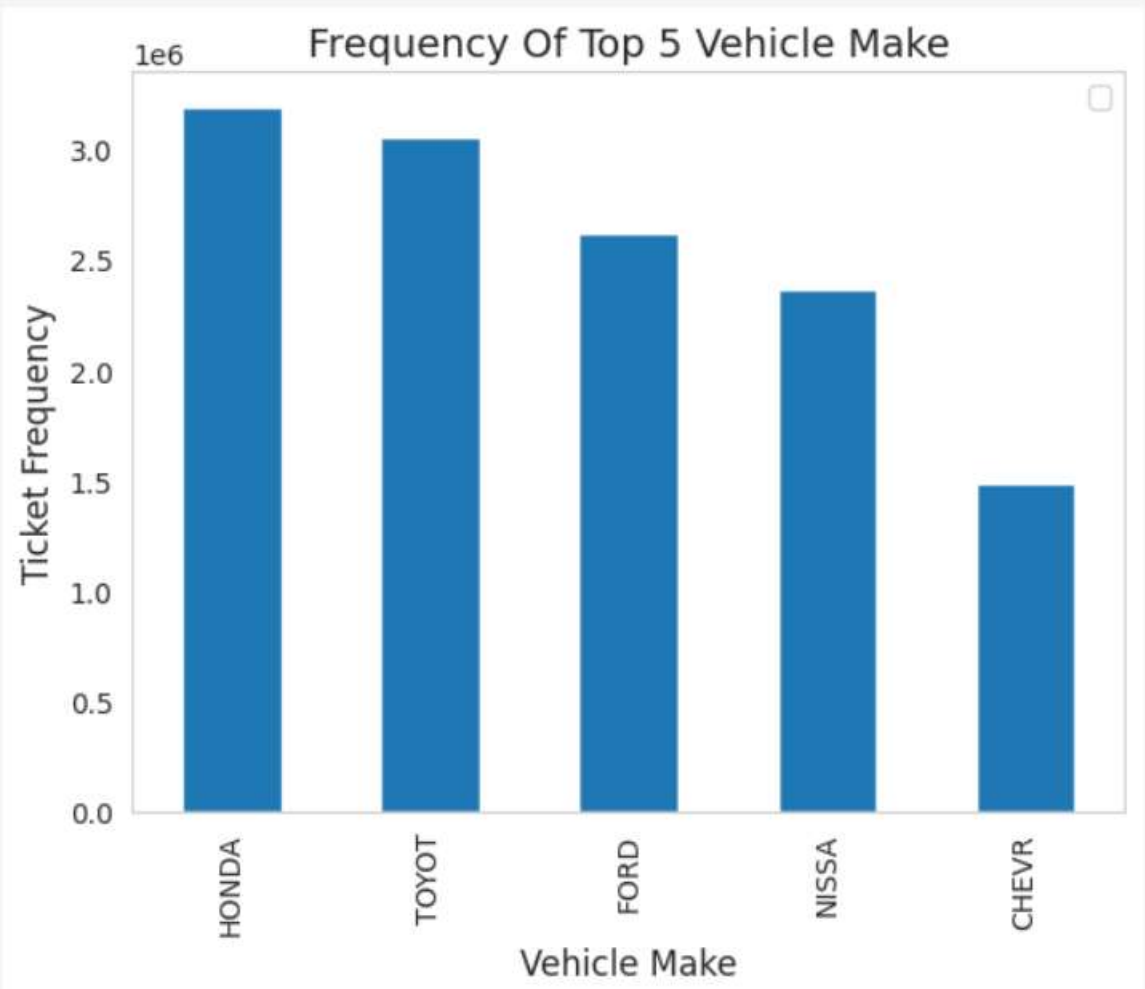
- For Vehicle Body Type, maximum parking violations happen for Suburban (SUBN) followed by four door sedan(4DSD), Van, Delv and Pick



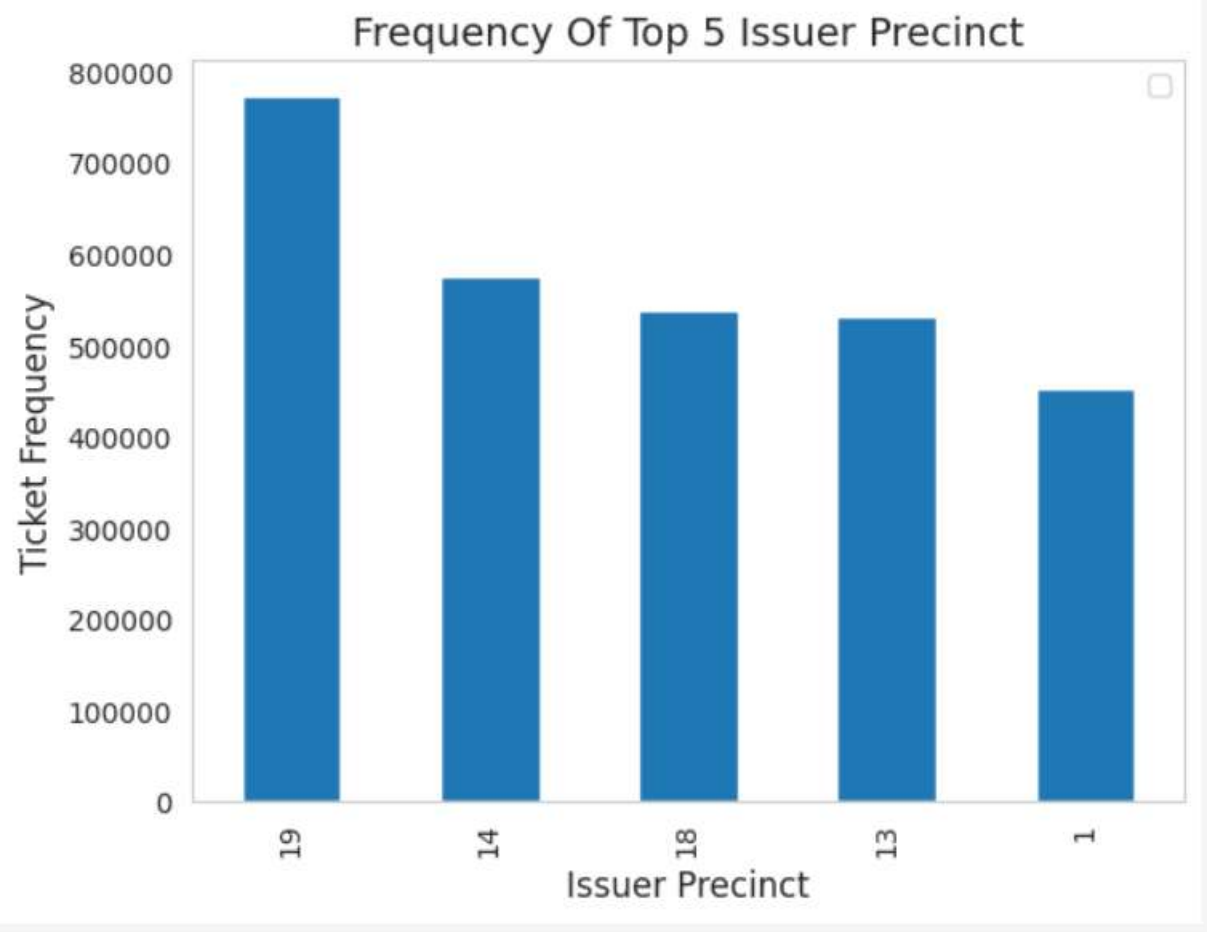
Observations:

The top 5 violation codes are 36,21,38,14,20.

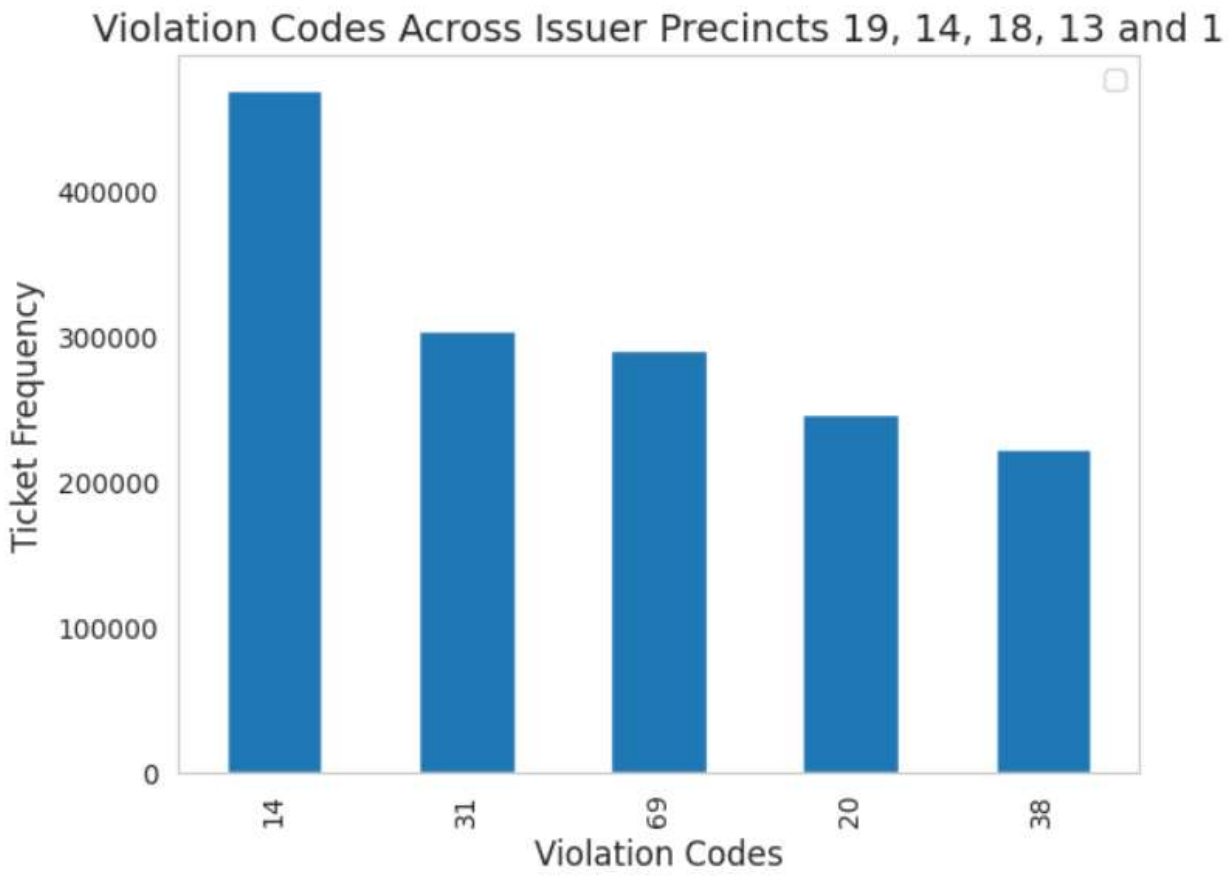
- Code 36: Exceeding the posted speed limit in or near a designated school zone.
- Code 21: No parking where parking is not allowed by sign, street marking or traffic control device.
- Code 38: Failing to show a receipt or tag in the windshield.
- Code 14: General No Standing: Standing or parking where standing is not allowed by sign, street marking, or; traffic control device.
- Code 20: General No Parking: No parking where parking is not allowed by sign, street marking or traffic control device.



- For Vehicle Make, maximum parking violations happened for Honda followed by Toyota, Ford, Nissa, Chevr.



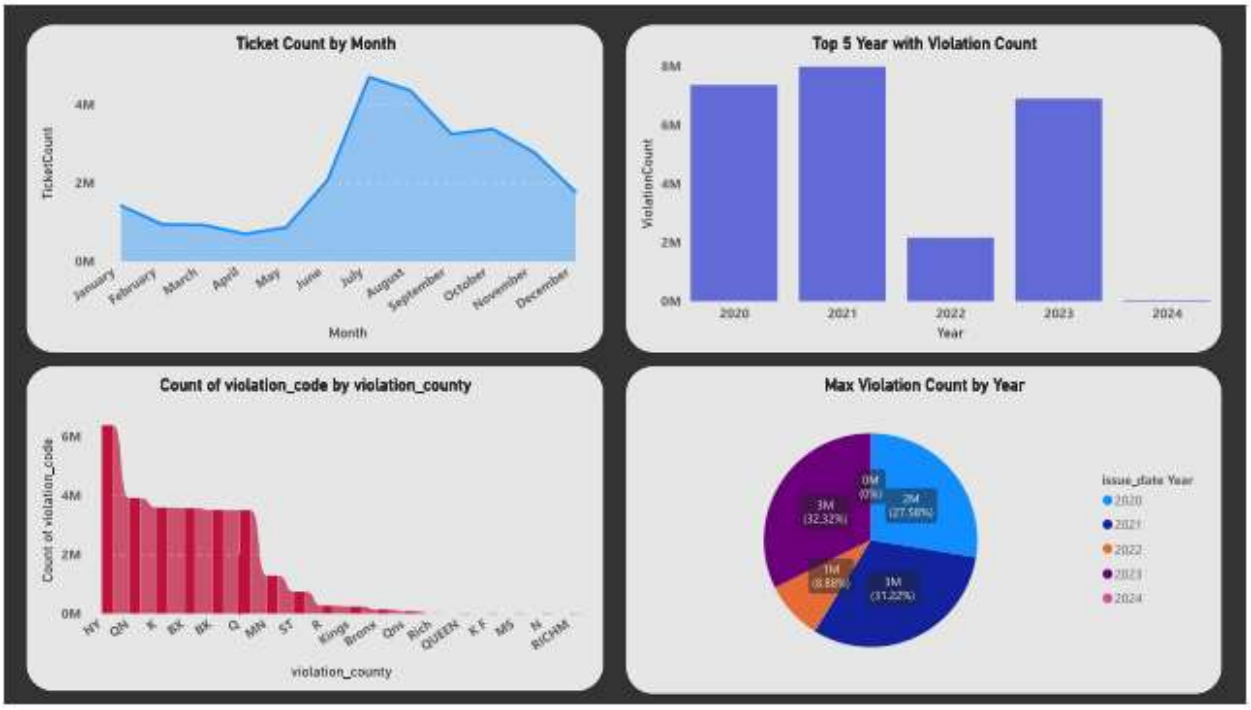
- Issuer precinct 19 issued the most tickets, with over 800,000 tickets issued.
- Issuer precincts 14, 18, and 13 also issued a large number of tickets, with between 500,000 and 6,00,000 tickets issued.

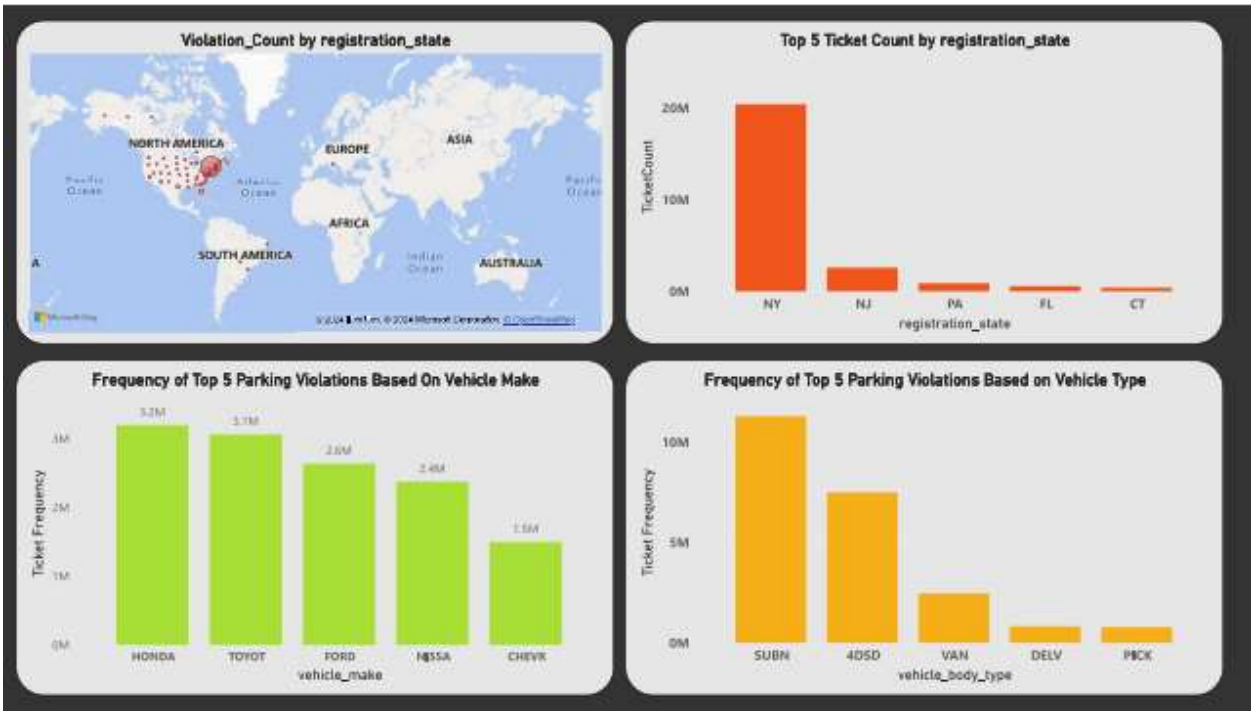


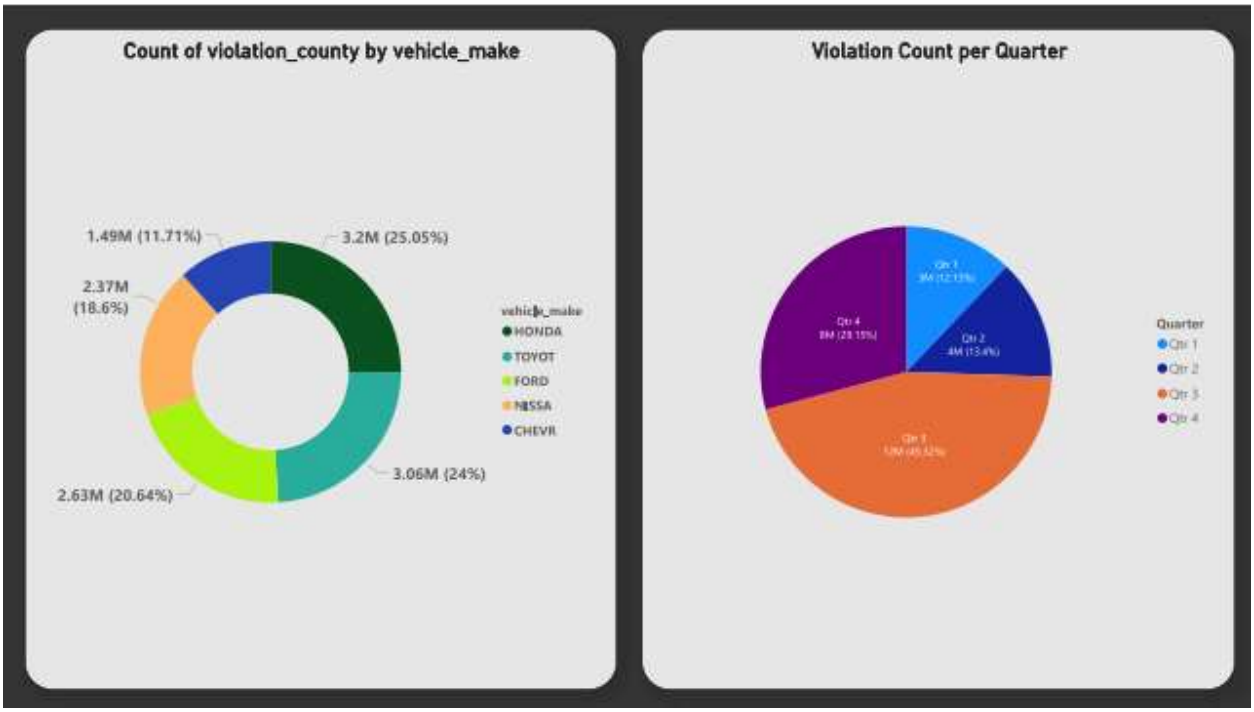
- **The top 3 violation precincts and Issuer Precincts where maximum parking violations happen are 19, 18,**

7. DATA ANALYSIS:

- The processed data was leveraged for sentiment analysis, and efforts were made to interpret meaningful insights from the analysis results.
- To facilitate data visualization, we opted for Power BI, a dynamic business intelligence platform renowned for its rapid delivery of insights across organizations.
- Utilizing Power BI, we seamlessly integrated data sourced from Azure Blob Storage, enhancing our ability to explore and showcase key findings effectively.


Ticket Count by Month
TicketCount
4M
2M
0M
January
February
March
April
May
June
July
August
September
October
November
December
Month
Top 5 Year with Violation Count
ViolationCount
8M
6M
4M
2M
0M
2020
2021
2022
2023
2024
Year
Count of violation_code by violation_county
Count of violation_code
6M
4M
2M
0M
HY
QN
K
BX
BK
Q
MN
ST
R
Kings
Bronx
Qns
Rich
QUEEN
K F
MS
N
RICHM
violation_county
Max Violation Count by Year
0M
(0%)
2M
(27.58%)
3M
(32.32%)
1M
(8.88%)
3M
(31.22%)
issue_state Year
2020
2021
2022
2023
2024


Violation_Count by registration_state
NORTH AMERICA
EUROPE
ASIA
AFRICA
SOUTH AMERICA
AUSTRALIA
Pacific Ocean
Atlantic Ocean
Indian Ocean
Top 5 Ticket Count by registration_state
TicketCount
20M
10M
0M
NY
NJ
PA
FL
CT
registration_state
Frequency of Top 5 Parking Violations Based On Vehicle Make
Ticket Frequency
3M
2M
1M
0M
HONDA
TOYOT
FORD
NISSA
CHEVR
vehicle_make
Frequency of Top 5 Parking Violations Based on Vehicle Type
Ticket Frequency
10M
5M
0M
SUBN
4DSD
VAN
DELV
PICK
vehicle_body_type


Count of violation_county by vehicle_make
1.49M (11.71%)
3.2M (25.05%)
2.37M (18.6%)
2.63M (20.64%)
3.06M (24%)
vehicle_make
HONDA
TOYOT
FORD
NISSA
CHEVR
Violation Count per Quarter
Quarter
Qtr 1
Qtr 2
Qtr 3
Qtr 4

8. CONCLUSION:

- Total Number of tickets issued for the year

Year	Violation Count
2020	7.3M
2021	7.9M
2022	2.1M
2023	6.8M
2024	282

- The top 5 violation codes are 21, 38, 14, 20,40

  Code 21 - No parking where parking is not allowed by sign, street marking or traffic control device.

  Code 38 - Parking Meter - Failing to show a receipt or tag in the windshield.

  Drivers get a 5-minute grace period past the expired time on parking meter receipts.

Code 14 - General No Standing: Standing or parking where standing is not allowed by sign, street marking or; traffic control device.

Code 20 - General No Parking: No parking where parking is not allowed by sign, street marking or traffic control device.

Code 40 - Stopping, standing or parking closer than 15 feet of a fire hydrant. Between sunrise and sunset, a passenger vehicle may stand alongside a fire hydrant as long as a driver remains behind the wheel and is ready to move the vehicle if required to do so.

Code 14 - General No Standing: Standing or parking where standing is not allowed by sign, street marking or; traffic control device.

Code 20 - General No Parking: No parking where parking is not allowed by sign, street marking or traffic control device.

Code 40 - Stopping, standing or parking closer than 15 feet of a fire hydrant. Between sunrise and sunset, a passenger vehicle may stand alongside a fire hydrant as long as a driver remains behind the wheel and is ready to move the vehicle if required to do so.

9 Reference

"NYC Parking Tickets" [Online] Available:

https://www.kaggle.com/datasets/new-york-city/nyc-parking-tickets

"Amazon Learner Lab Guide" [Online] Available:

https://awsacademy.instructure.com/courses/24007/modules/items/1969629

Nick Cox "Learn Python Data Analytics by Example: NYC Parking Violations" [Online]

Available: https://towardsdatascience.com/learn-python-data-analytics-by-example-nyparking-violations e1ce1847fa2

"Violation Codes, Fines, Rules & Regulations" [Online] Available:

https://www1.nyc.gov/site/finance/vehicles/services-violation-codes.page

QuickSight Manifest files

https://docs.aws.amazon.com/quicksight/latest/user/supported-manifest-fileformat.html