Forecaster: Using Time Series Forecasting to Predict Order Patterns for Restaurants

CSE 587 C – Fall 2024

Apurva Banka – 50610491

Raj Kumar Alikatte – 50600978

Sai Harshitha Karri – 50604558

Patanjali Uppugandla – 50587096

---

1. Problem Introduction

Restaurants often face challenges in managing inventory, order fulfillment, and delivery efficiency. Accurately predicting order patterns can improve decision-making and enhance customer satisfaction. This project, "Forecaster," aims to predict order trends for restaurants using time-series forecasting and explore the impact of factors like weather, delivery distance, and order type. Additionally, we incorporate a recommendation system to improve the user experience, helping businesses optimize operations and rectify inefficiencies.

---

2. Dataset

For the project, we use the **"Food Delivery Dataset"** from Kaggle. This dataset includes detailed information about delivery locations, order details, delivery conditions, and the city.

Dataset Link: Food Delivery Dataset.

Data Details

- **Delivery Details:** Locations, time, and modes of delivery.
- **Order Details:** SKUs, timestamps, and volumes.
- **Delivery Conditions:** Weather, time of the day, and traffic information.
- **City:** Urban or suburban classification.

DataPreprocessing:

We filtered and cleaned the dataset to focus on fields relevant to our hypotheses and models. Features were standardized and encoded for clustering, regression, and forecasting.

---

3. Features and Processing

Key features extracted and processed:

1. **Delivery Time:** Time taken for order completion.
2. **Order Type:** Morning, Afternoon, Evening, Night.
3. **Delivery Mode:** Motorcycles, bicycles, electric scooters, etc.
4. **Ratings:** Average customer ratings post-delivery.
5. **Weather Conditions:** Sunny, rainy, cloudy, etc.

6. **Distance:** Distance from restaurant to delivery location.
7. **Order Volume:** Number of SKUs in an order.

Data preprocessing included:

- Filling missing values.
- Encoding categorical variables.
- Normalizing numerical features.
- Feature engineering for time-series data analysis.

---

4. Models and Techniques

Hypothesis 1 - If the time taken for the order increases, the average rating decreases

Linear regression:

Using Linear regression, as we can see there are huge errors in the predicted model. The graph doesn't align with the data.

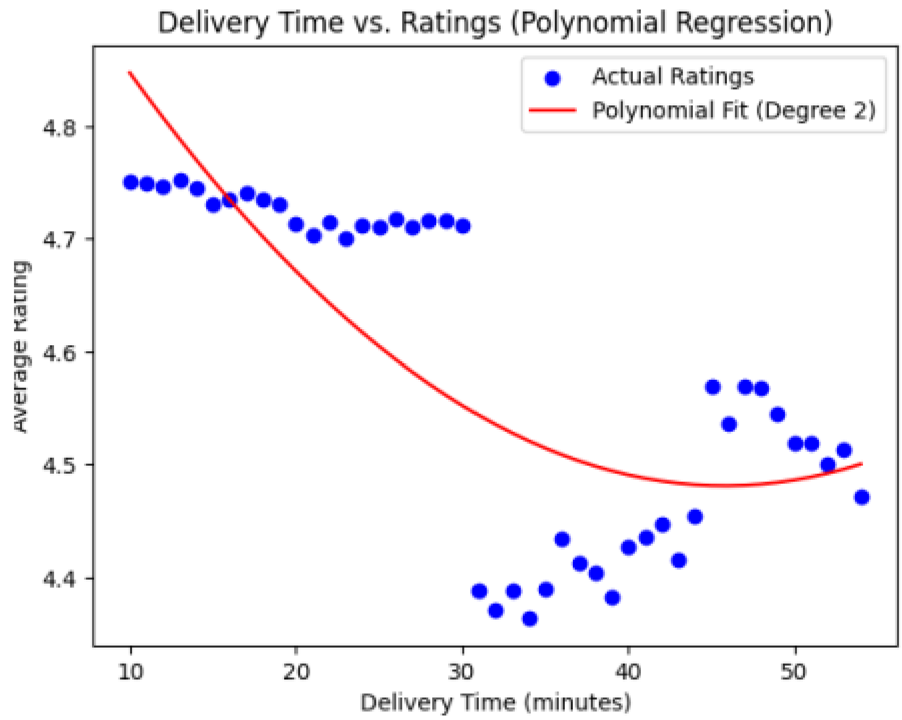
Since we used linear regression, we could only draw a straight line. If we try to use a polynomial regression, we can have a graph that aligns to the data better with a curve.

Polynomial Regression of degree 2

We were able to get a better graph using Polynomial Regression. Now, we want to explore polynomial regression of higher degree. We can use accuracy as a parameter to determine if the model is performing better.

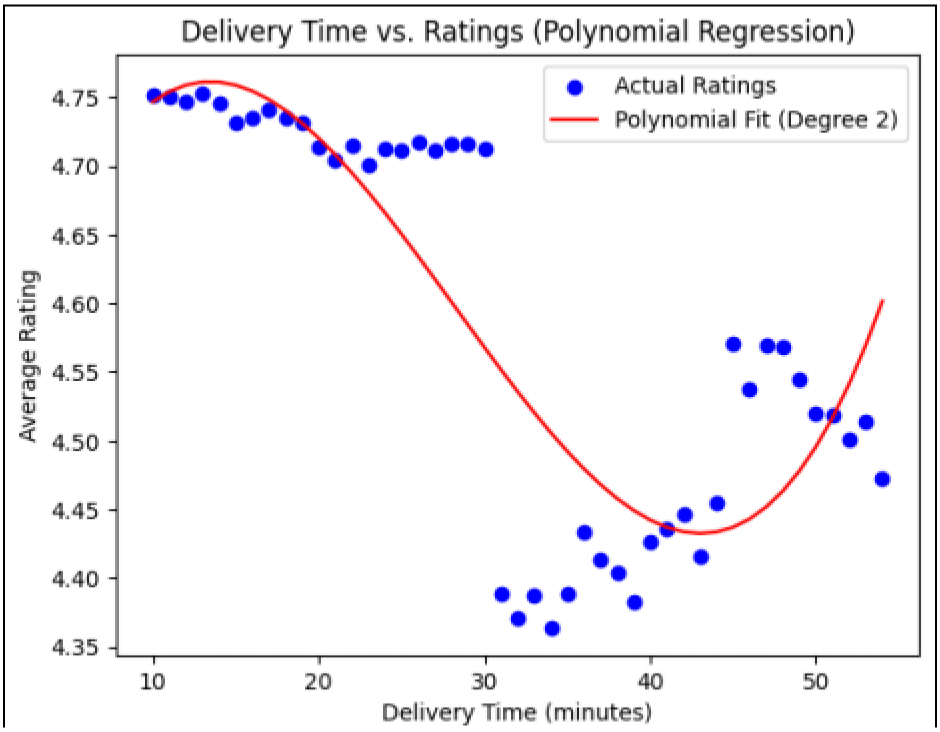
If we increase the degree of polynomial, we can get a better graph that can align with the data better. Calculating the accuracy for the same.

Since, this was a polynomial graph of degree 2, we see the graph with 1 maxima or 1 minima. (1 cursive) We want to go to a graph with higher degree (more curves)



Polynomial regression of degree 3

Here we are able to get a better graph compared to a Polynomial Regression of degree 2. Let's try to compute the accuracy for the same.



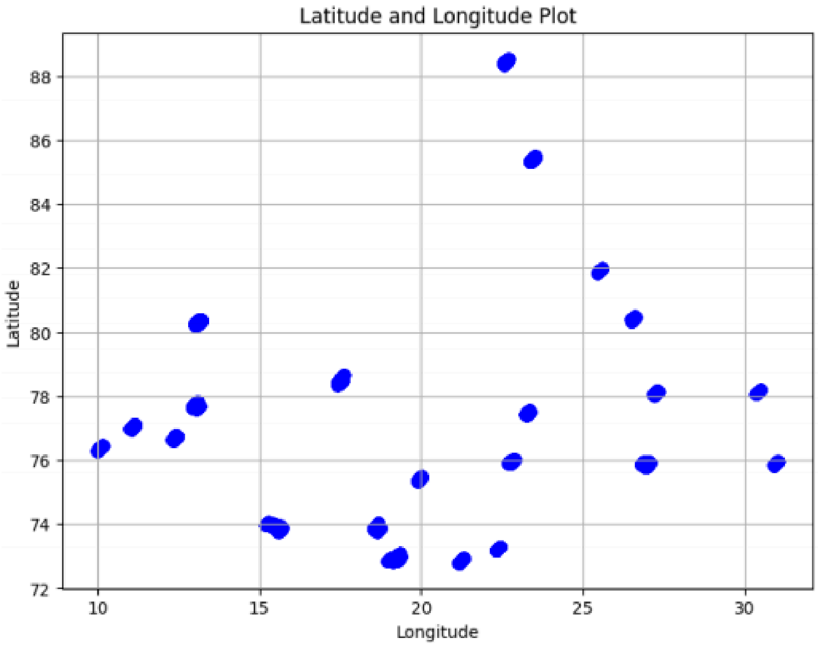
Decision Tree Regression

Here the model is following the graph too closely. We might run into over-fitting.

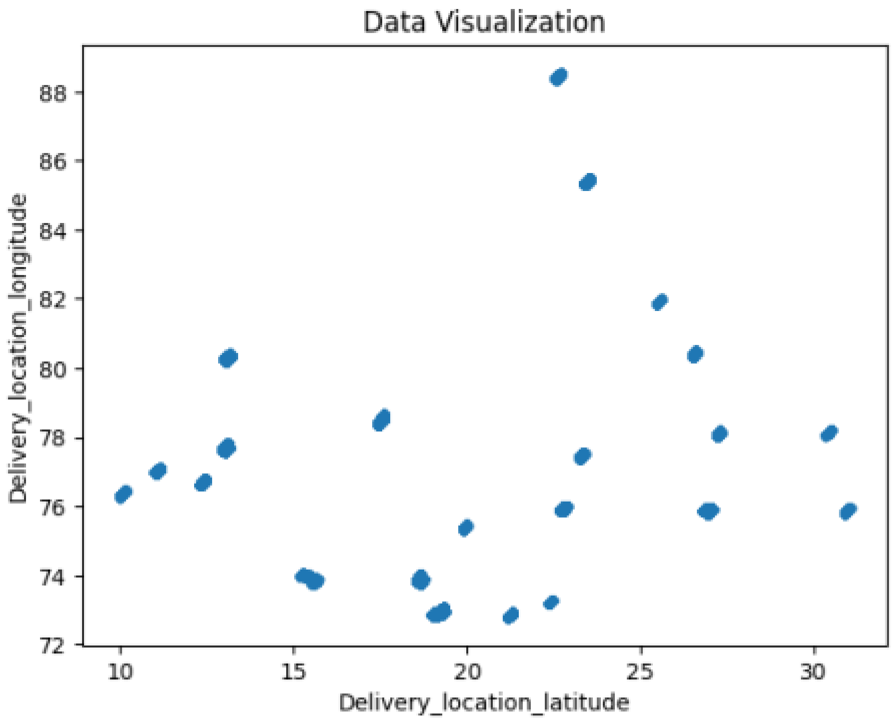
Hypothesis 2 - If the distance increases, the time taken to deliver the order increases.

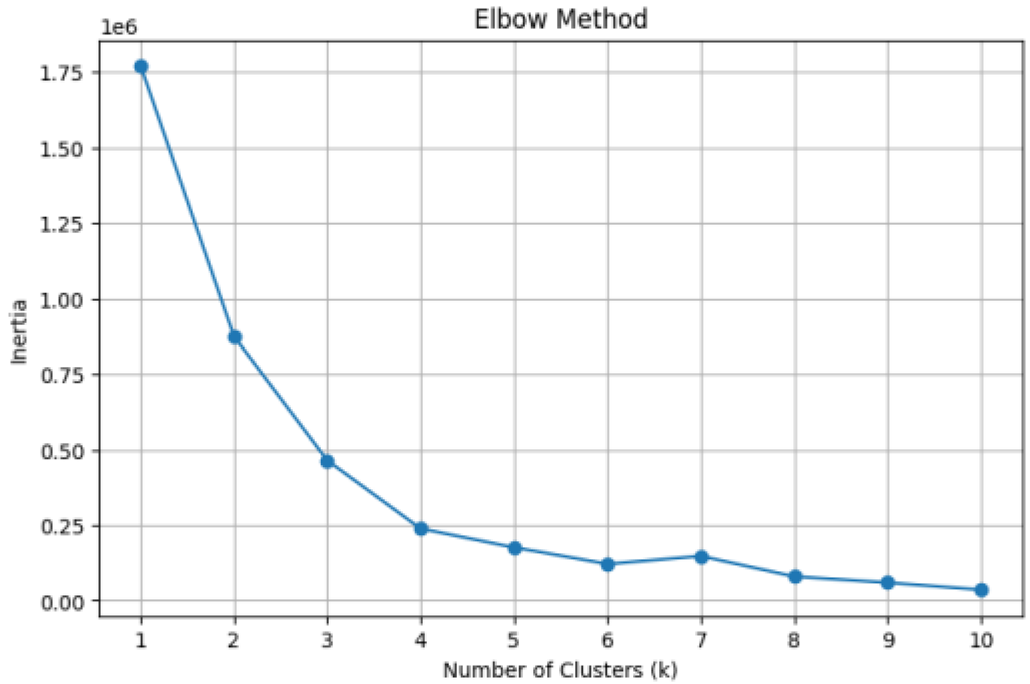
Using Clustering Models For Second Hypothesis

In our second hypothesis, we assumed that as the distance increases, the delivery time increases. Here, we try to divide the observed data into clusters.

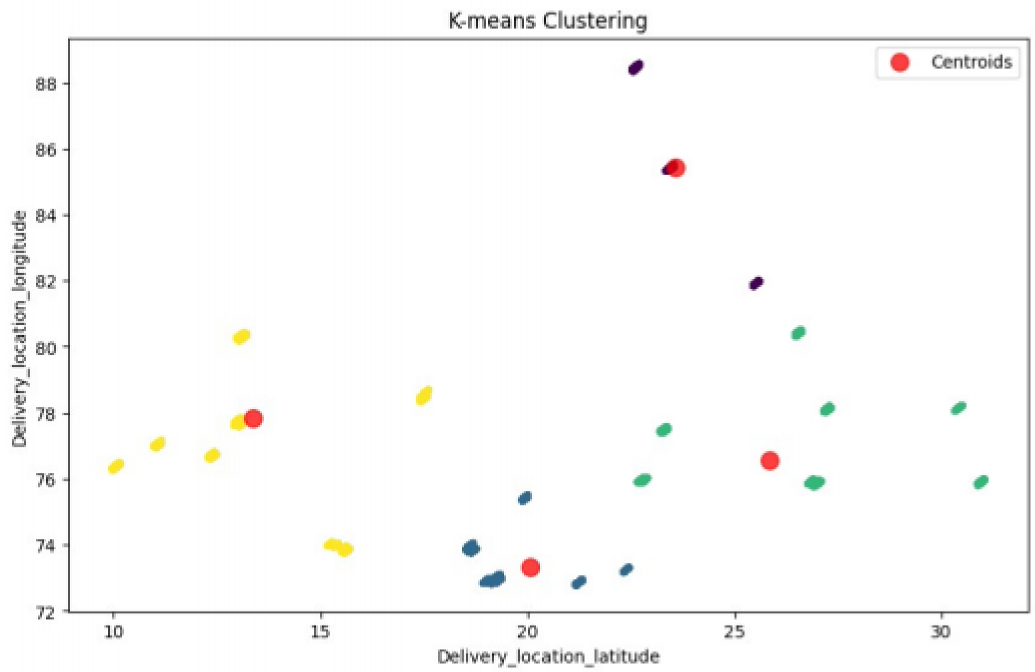


Here, we want to divide the data into different clusters. We are going to use K-Means clustering in order to classify data.





From the Elbow curve, we can see, the graph starts to flatten at k value 4. So, we can assume the optimal number of clusters to be 4.

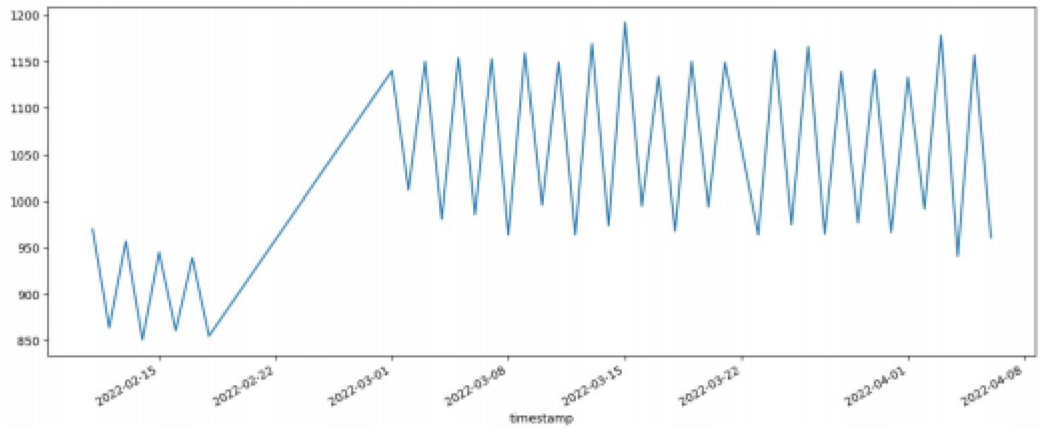


The above is the graph for the 4 different clusters that we received. The red dots show the centroids for each cluster. We can vary the number of centroids to get different clusters. In our case, 4 is the optimal number of clusters that we will get from the given data.

Time Series Forecasting

Here, we will try to forecast the number of orders using Time Series Forecasting. We will accumulate the number of orders received on a daily basis.

We will then predict the number of orders for the future. We will find the perfect Time Series forecasting models to get the prediction.

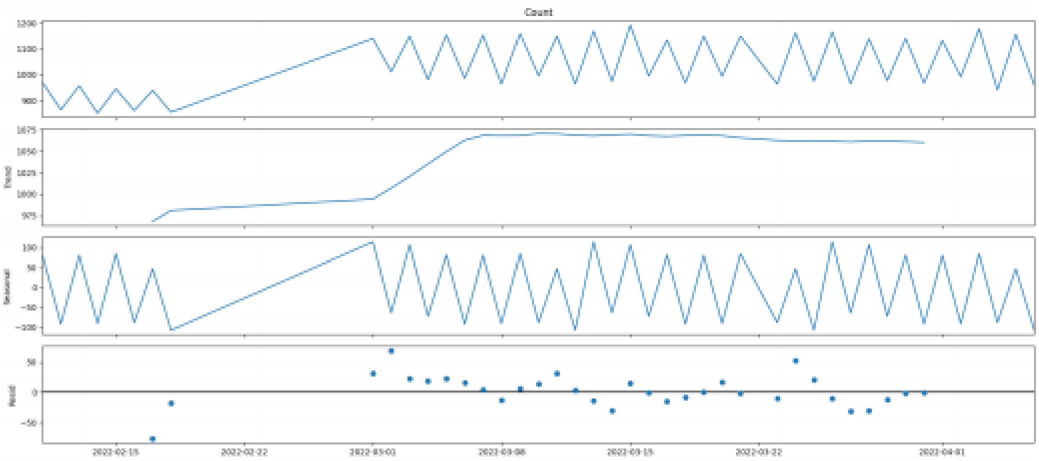


We compute the Trend, Seasonality and Residual (Noise) of the current data.

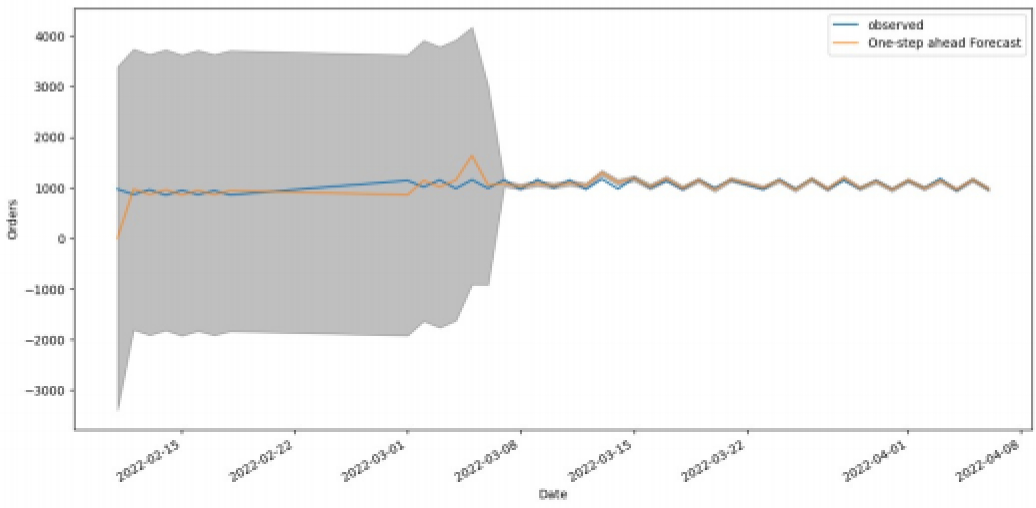
- The trend is the long-term movement or direction in the time series data

- Seasonality refers to the repeating and predictable patterns or cycles in the data that occur at regular intervals, typically within a year, month, or week.

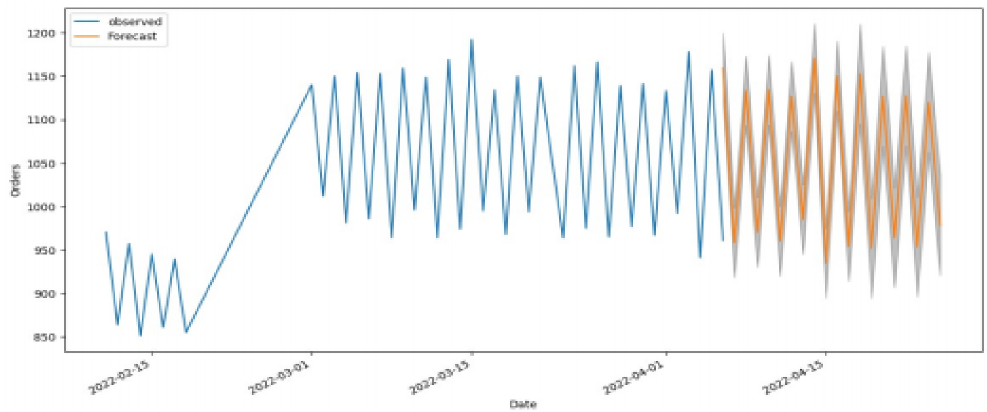
- The residuals (or noise) represent the random or unexplained variation in the data after removing the trend and seasonality components



We try to plot the graph using the values obtained using SARIMA Model



We now want to plot the predictions for the given data.



Here, we see the orange line is the predicted value, based on the SARIMA Time Series Forecasting. The gray part is the expected noise in the data.

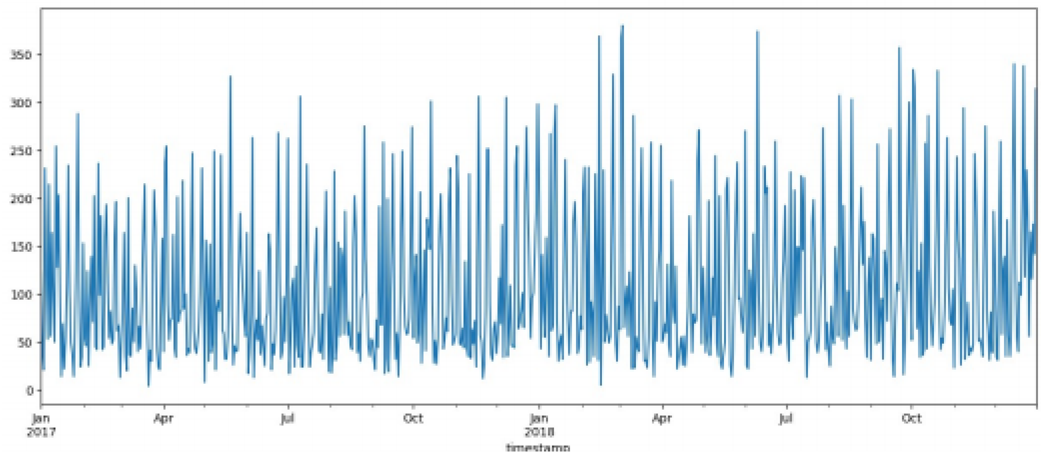
We can use the predicted values for various advancements and optimisations.

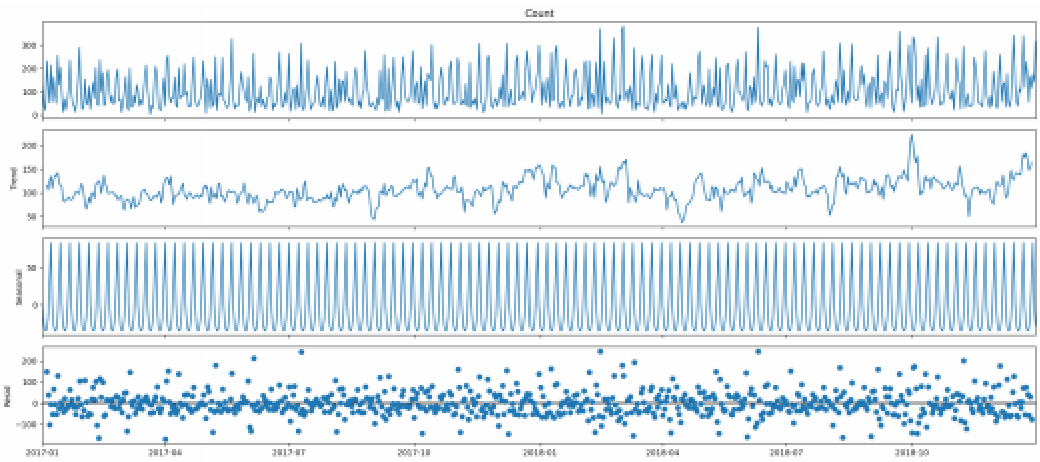
One problem with the given data is the Dataset size. We have around about 2 months of data. In order to unlock the true potential of Time Series Forecasting, we need at least 2 years of data.

We will perform the above predictions but on a bigger dataset in-order to test the performance of Time Series Forecasting.

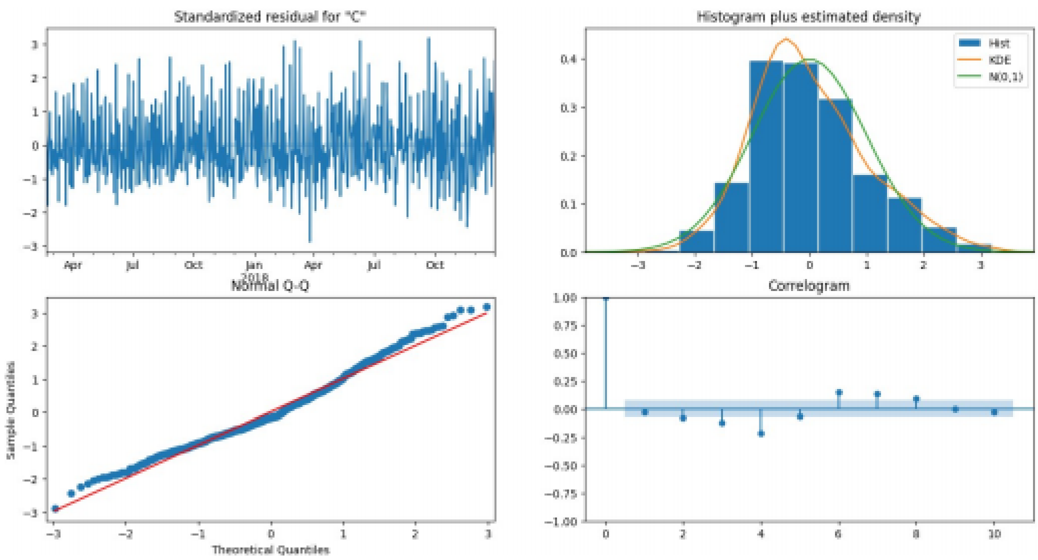
Time Series Forecasting For A Bigger Dataset

We compute the Trend, Seasonality and Residual (Noise) of the current data.

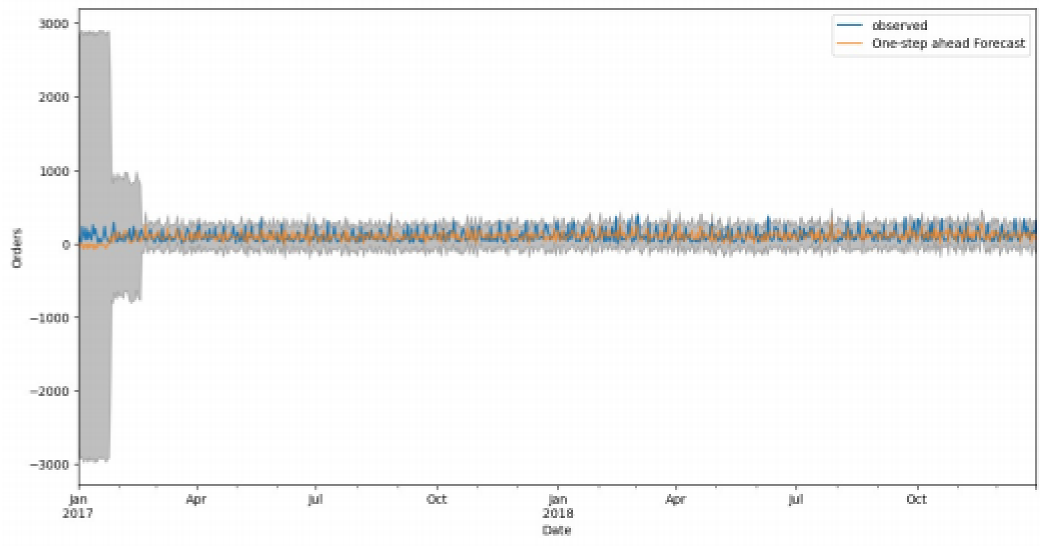




We try to plot the graph using the values obtained using SARIMA Model

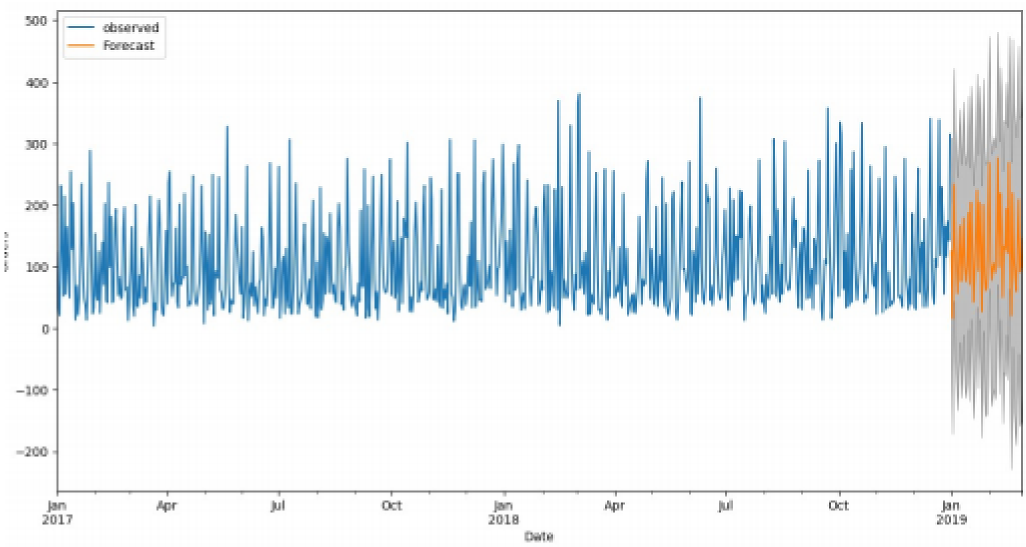


Here, since the number of data points is huge, we get a more plausible graph when predicting the values.



Computing the Squared Error

Here, we see the orange line is the predicted value, based on the SARIMA Time Series Forecasting. The gray part is the expected noise in the data.

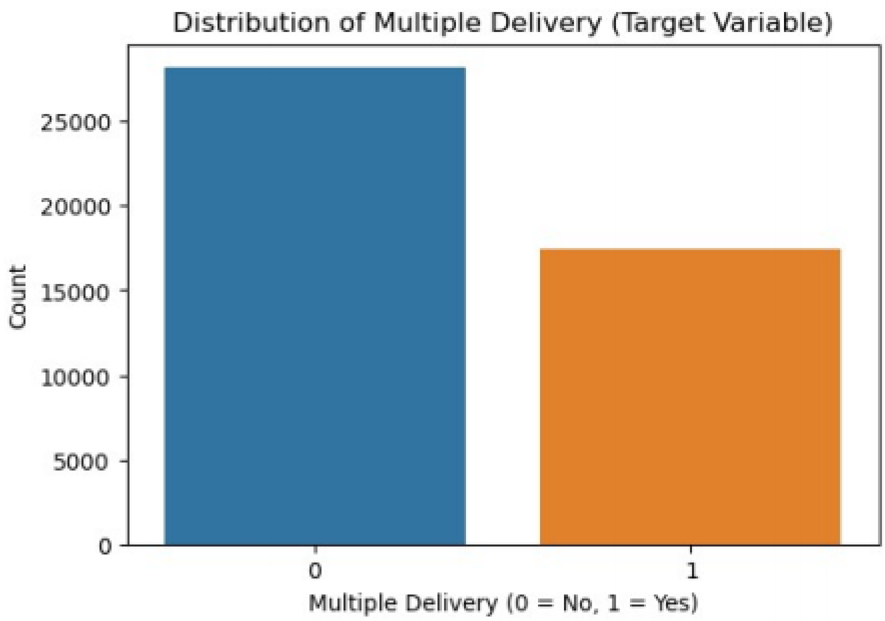


Hypothesis 1 - Motorcycles are able to do multiple deliveries compared to bicycle, electric_scooter and scooter.

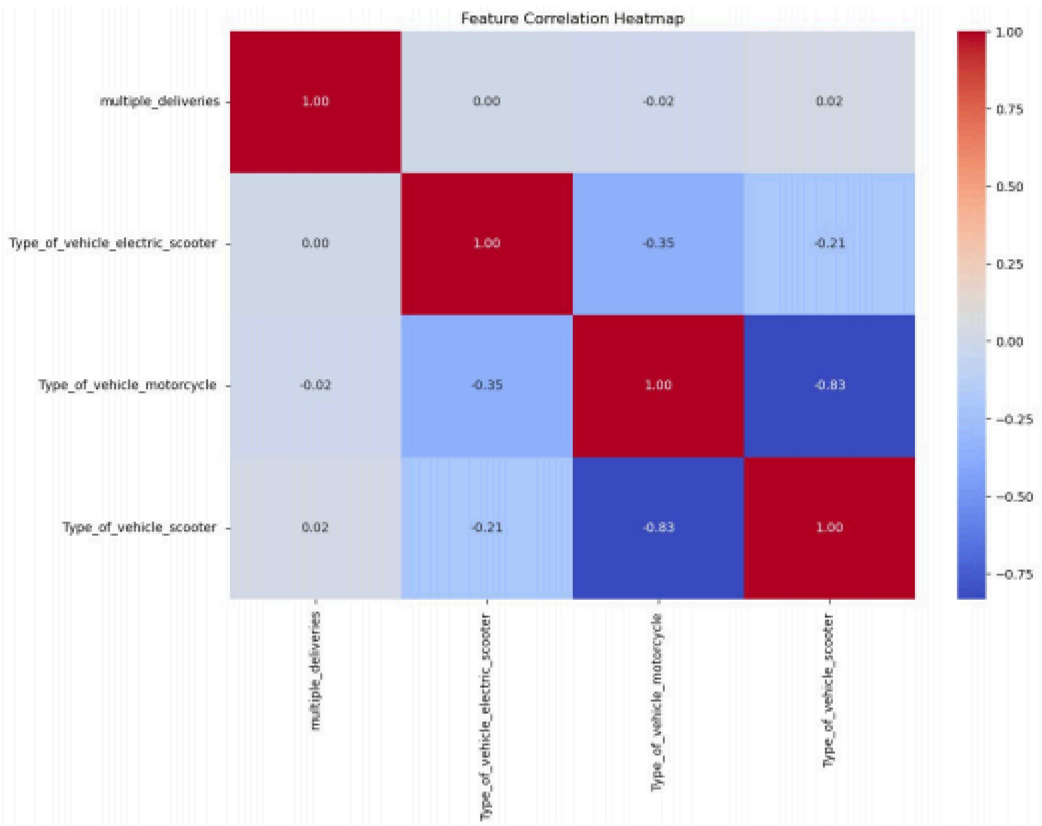
logistic Regression

The logistic regression analysis indicated that certain vehicle types might be more likely to perform multiple deliveries. The features included in the analysis were informative for understanding the decision boundary between classes.

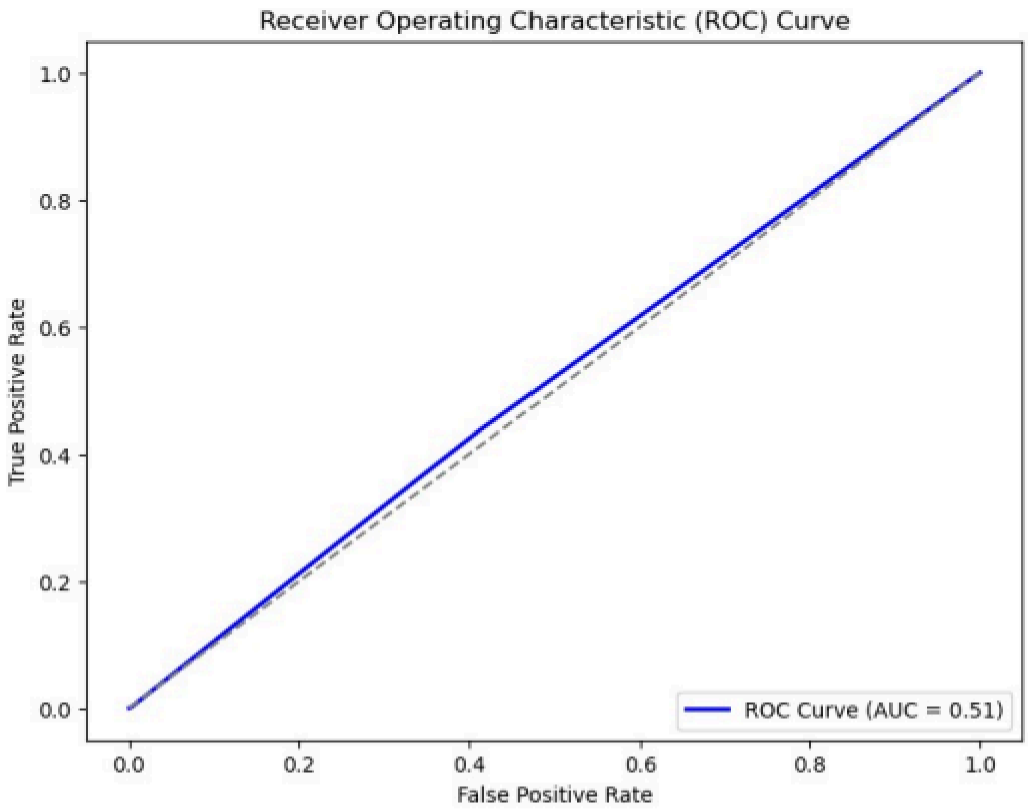
The model's coefficients suggested which types of vehicles contributed positively or negatively to the likelihood of multiple deliveries, helping inform strategic decisions in fleet management.

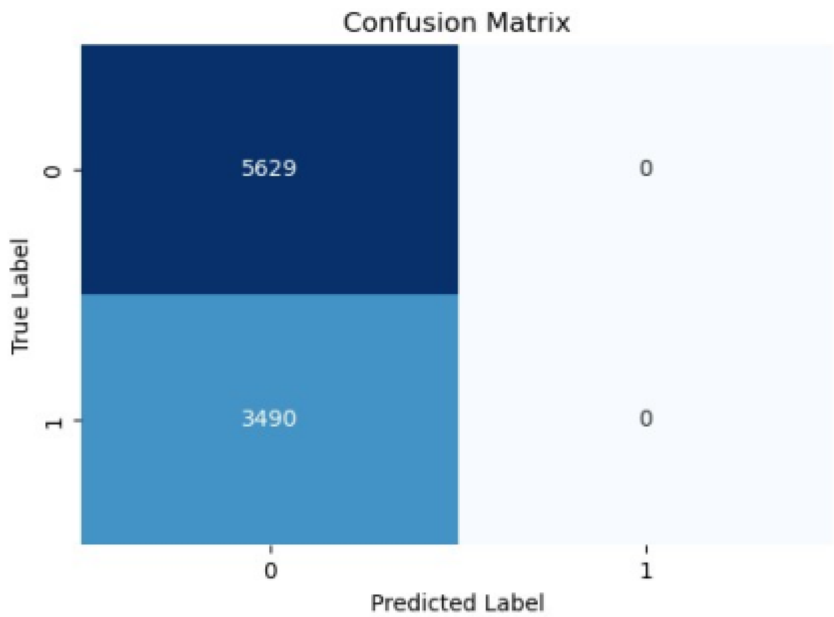


Distribution of Multiple Delivery (Target Variable)
Count
25000
20000
15000
10000
5000
0
0
1
Multiple Delivery (0 = No, 1 = Yes)



Feature Correlation Heatmap
multiple_deliveries
Type_of_vehicle_electric_scooter
Type_of_vehicle_motorcycle
Type_of_vehicle_scooter
1.00
0.00
-0.02
0.02
0.00
1.00
-0.35
-0.21
-0.02
-0.35
1.00
-0.83
0.02
-0.21
-0.83
1.00
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75



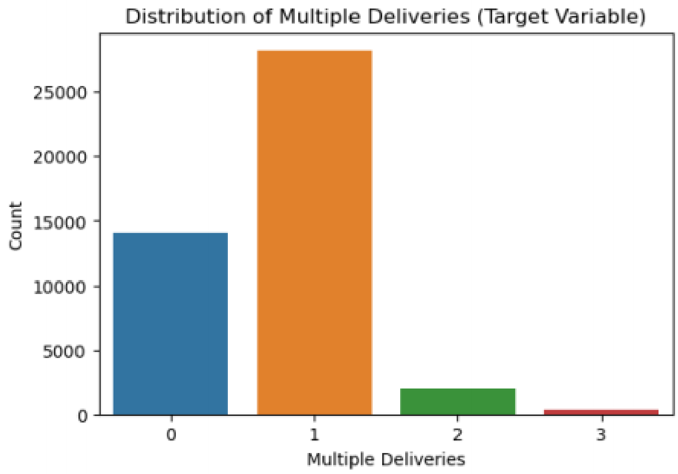


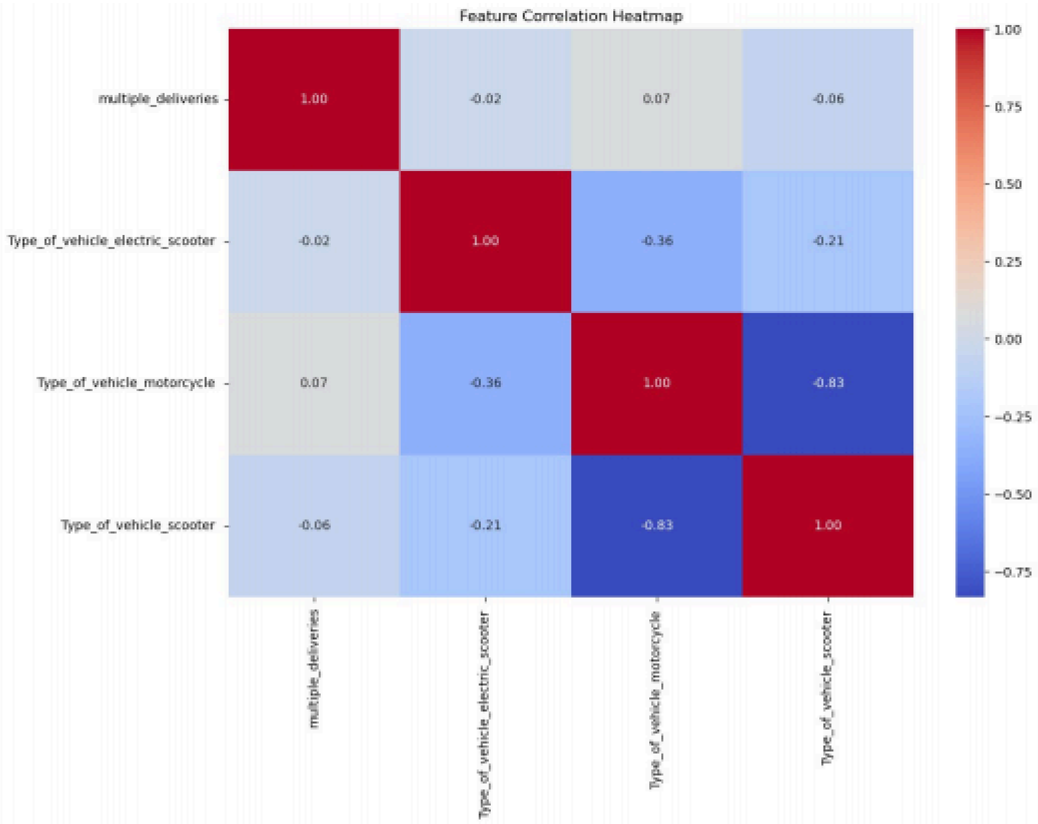
Multinomial Logistic Regression

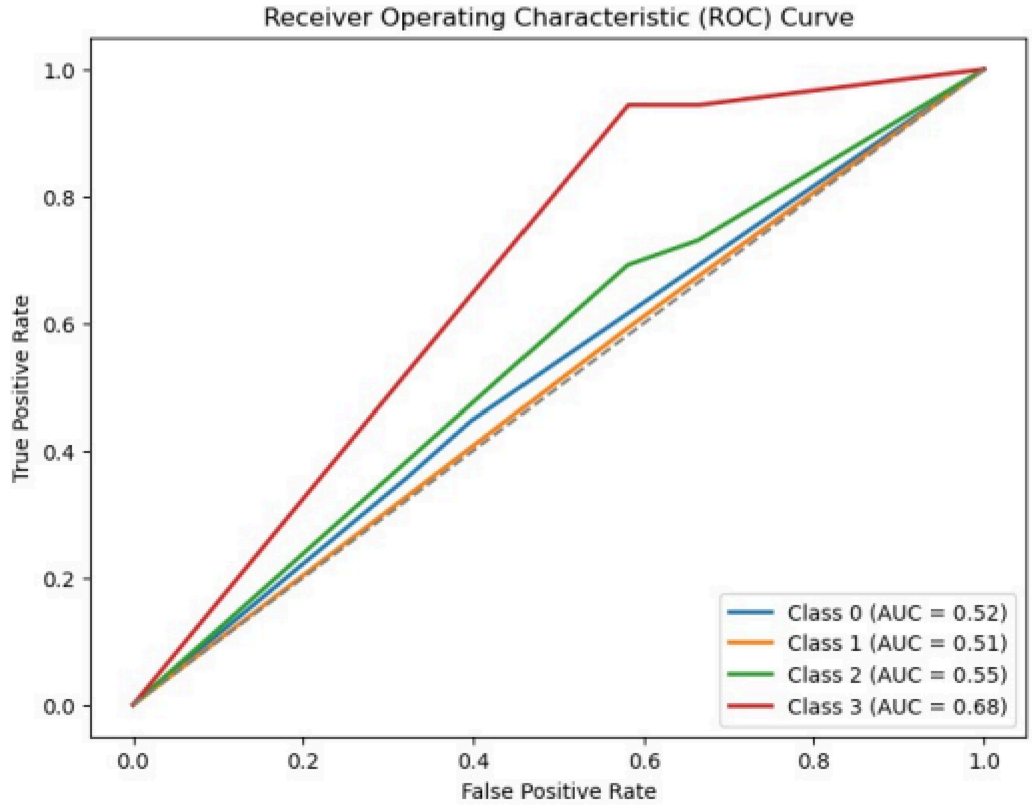
Vehicle Type Influence: The analysis provided insight into which vehicle types were more associated with multiple deliveries, indicating that certain vehicle types were more likely to perform high-volume deliveries.

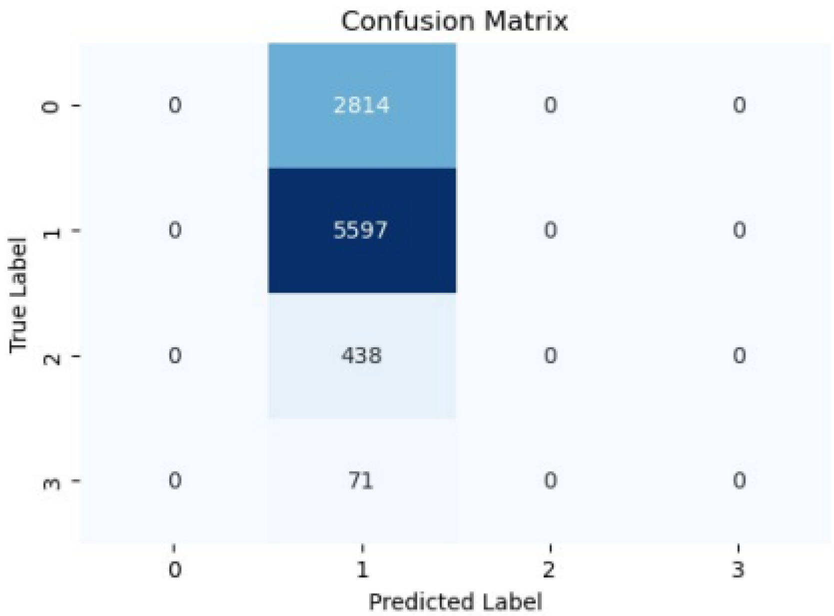
Operational Recommendations: Understanding these relationships helps inform logistics planning and resource allocation, especially for optimizing delivery routes and selecting appropriate vehicle types.

Feature Importance: While logistic regression itself does not inherently rank feature importance, the coefficients provided insights into how different types of vehicles affected the probability of multiple deliveries.









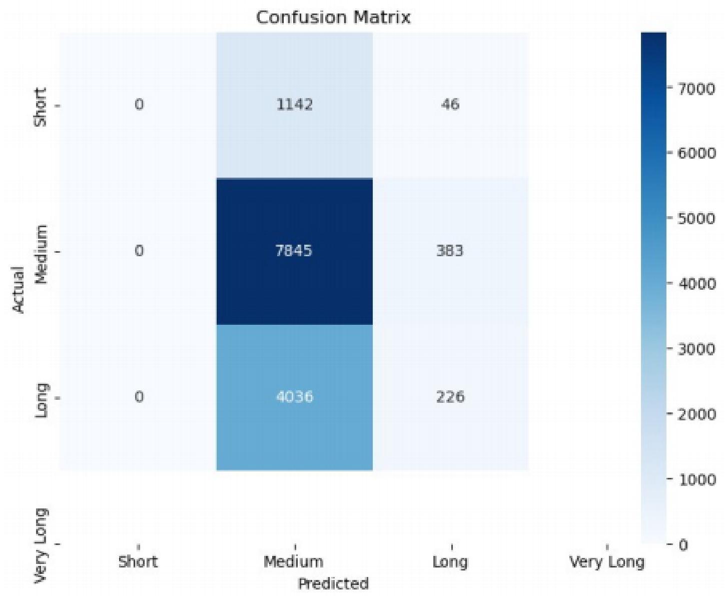
Hypothesis 2 - The Weather conditions impact the time taken to deliver

Naive Bayes model

Delivery Time Prediction: The model demonstrated that order time and weather conditions are significant predictors of delivery time categories.

Operational Strategy: Understanding which conditions lead to longer delivery times can help in proactive route and resource planning.

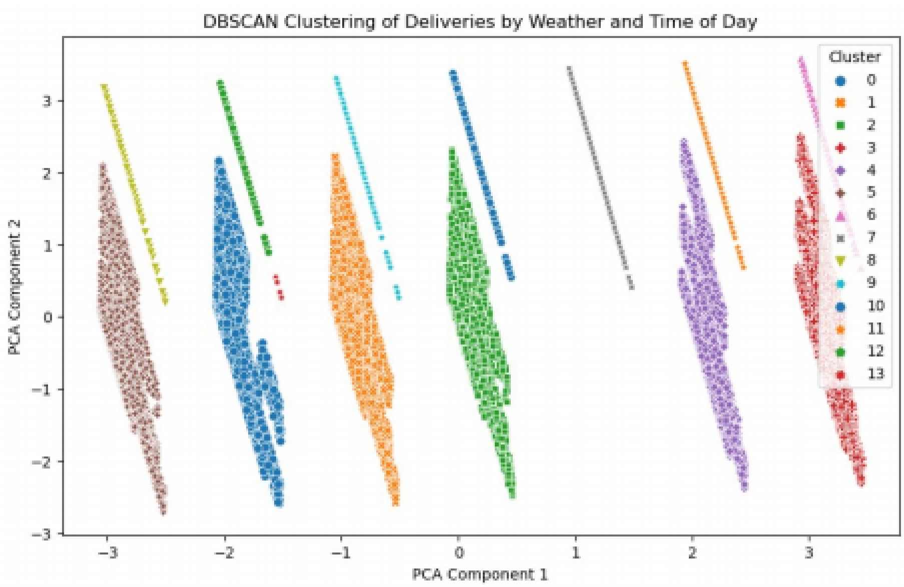
Feature Importance: Although Naive Bayes does not provide direct feature importance scores, the results suggested that extreme weather had a more substantial impact on classifying delivery times as "Long" or "Very Long."

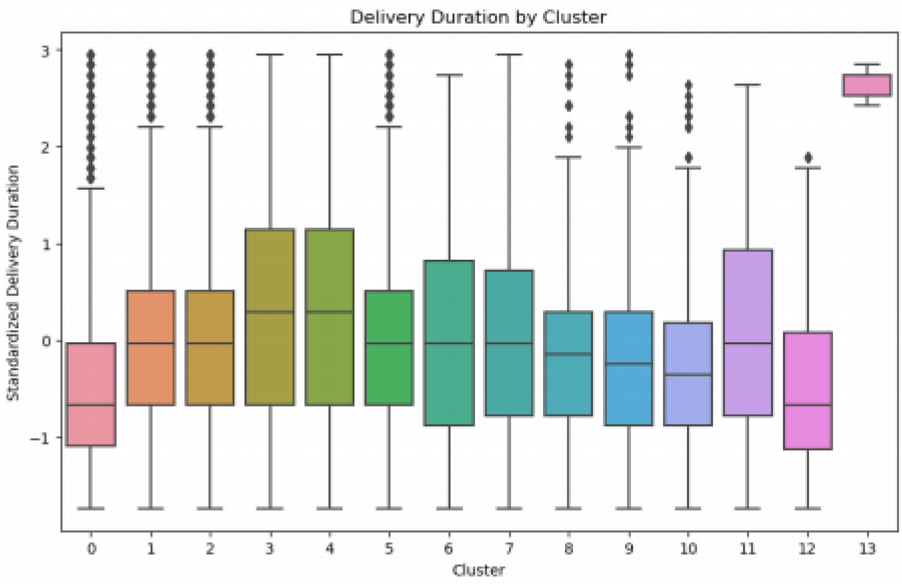


DBSCAN Clustering Algorithm

Delivery Patterns: The clusters revealed distinct groupings based on weather conditions and time of day, which could help in resource allocation during peak times or adverse weather.

Outlier Identification: Noise points corresponded to unusual or highly delayed deliveries, providing intelligence on potential inefficiencies in delivery logistics.

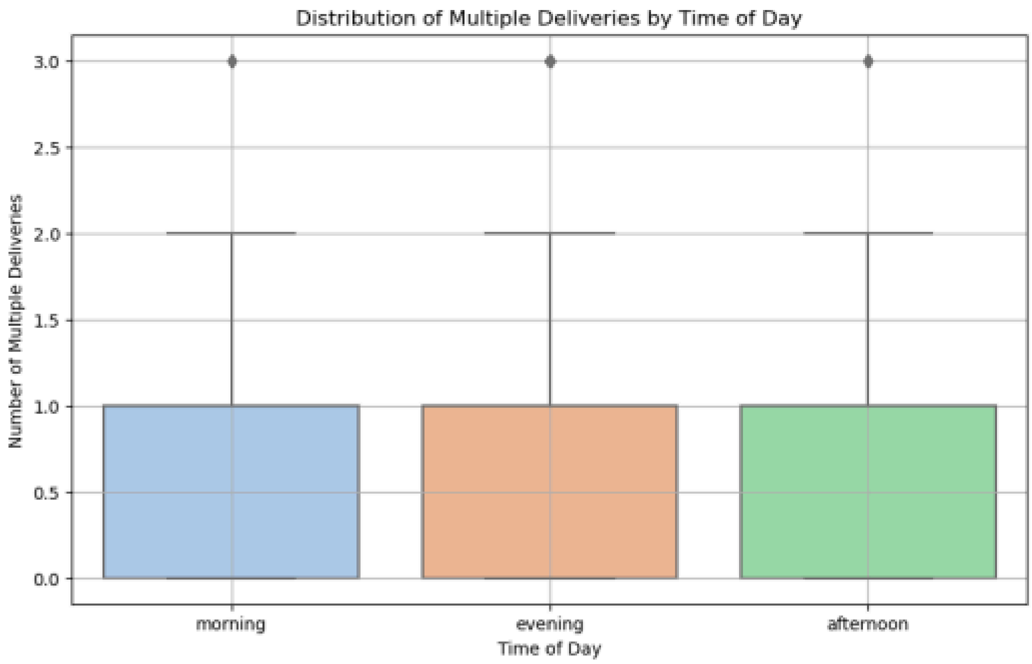


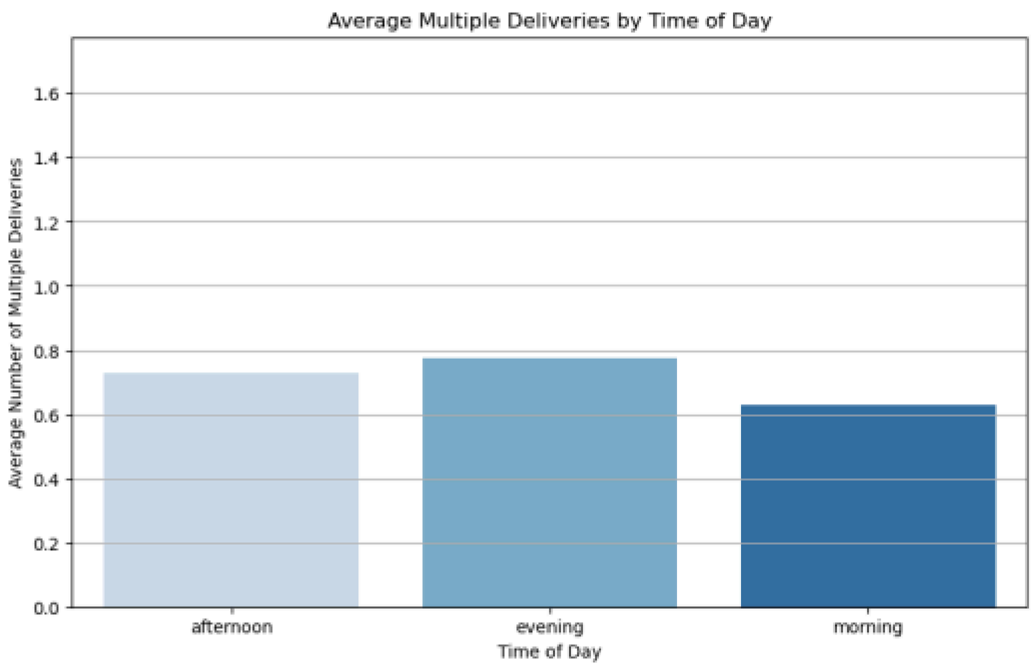


Hypothesis 1 - The multiple deliveries are among afternoon morning and evening times

ANOVA

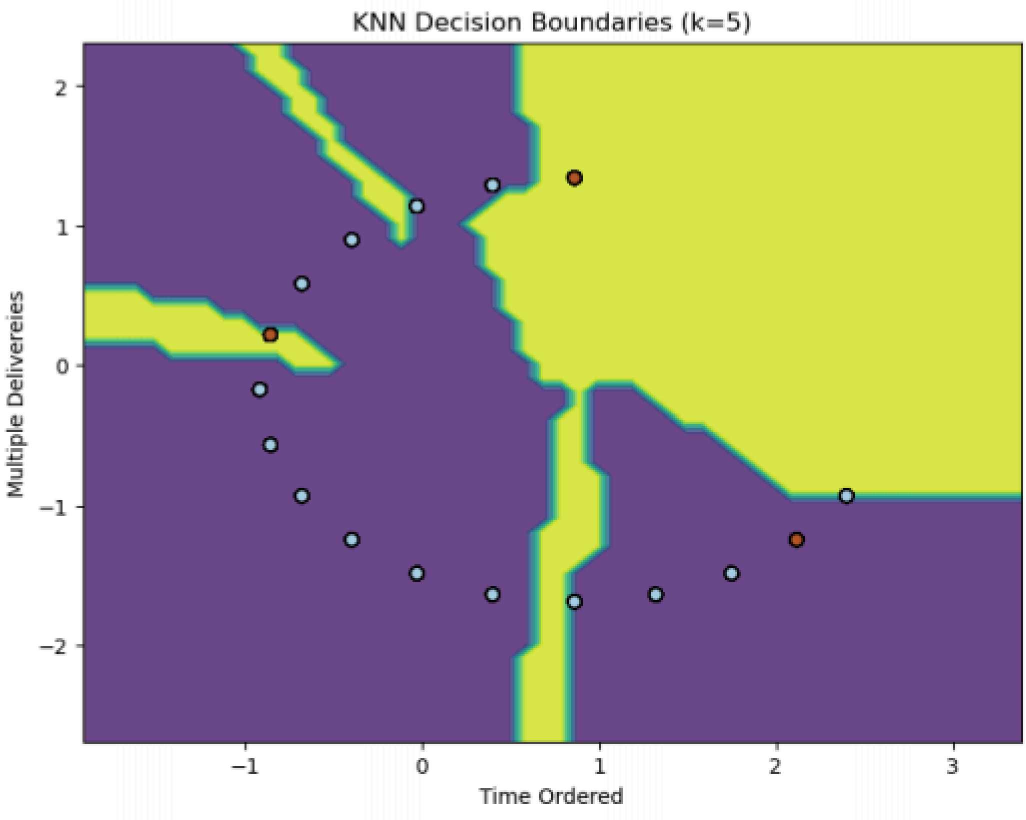
- Time-Based Variations: The analysis confirmed that the mean number of multiple deliveries significantly differs between morning, afternoon, and evening.
- Operational Strategies: These findings can inform staffing and resource allocation strategies during peak and off-peak times to enhance delivery efficiency.





KNN:

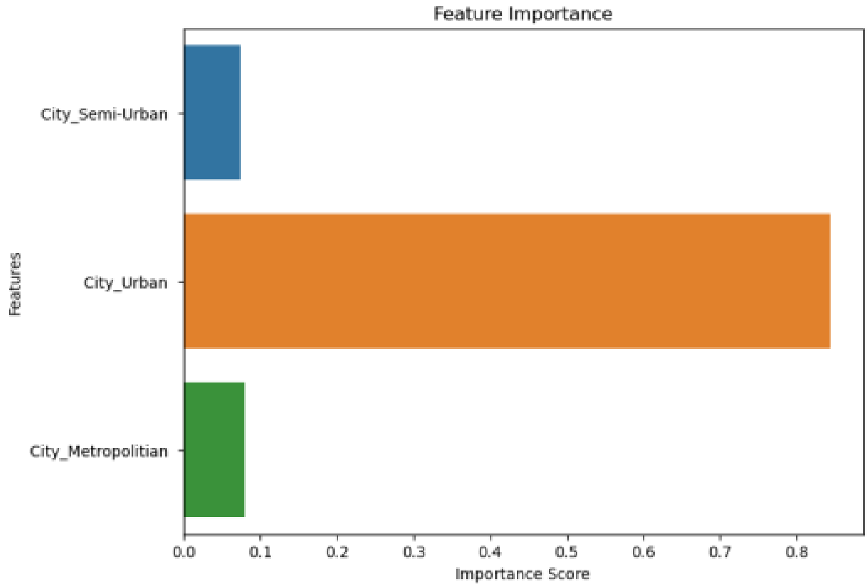
- Time-Related Patterns: The decision boundaries indicated strong correlations between certain times of the day and the likelihood of multiple deliveries.
- Model Advantages: KNNs flexibility allowed it to capture nonlinear relationships within the data, uncovering complex delivery patterns.
- Practical Applications: The analysis can inform scheduling and resource planning by highlighting when multiple deliveries are most likely.



Hypothesis-2 – Customers in urban areas leave lower ratings compared to suburban areas.

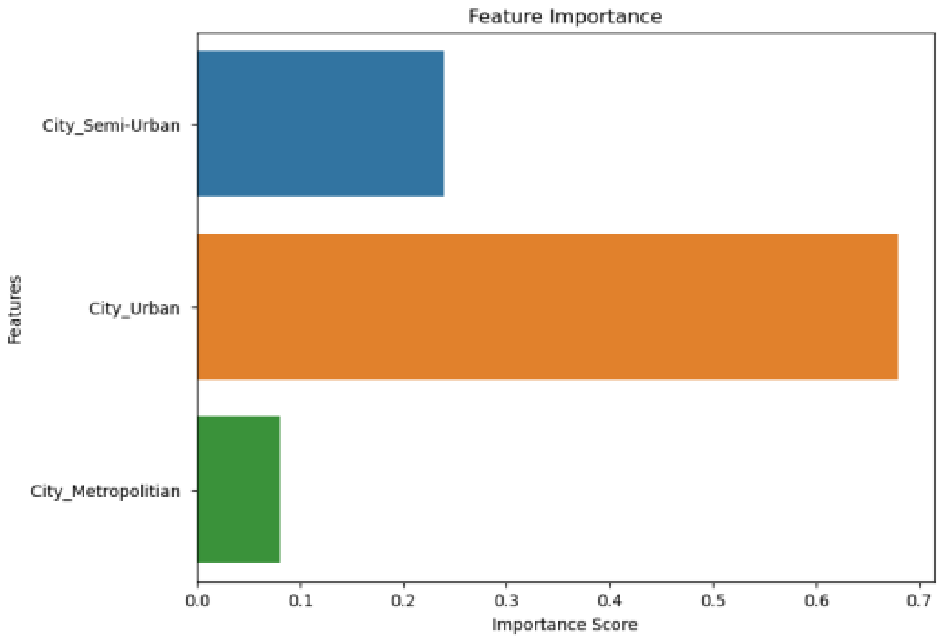
Random Forest

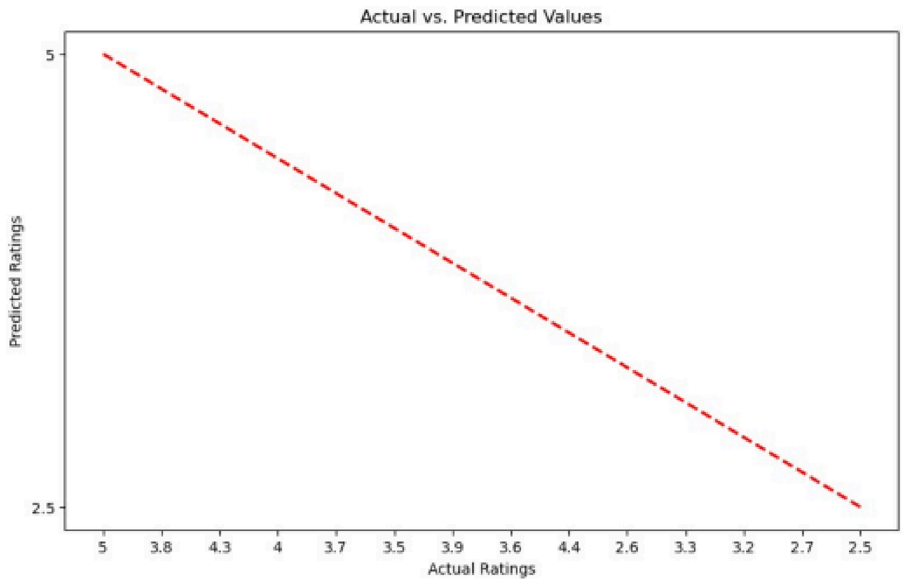
- City Type Influence: The analysis identified which city types were most influential in determining delivery ratings, providing insights for operational decision-making and resource allocation.
- Model Strengths: Random Forest's capacity to handle nonlinear interactions helped reveal complex relationships between city type and delivery ratings.



Gradient Boosting Regressor

- Influential Factors on Ratings: The model underscored the impact of city type on delivery ratings, with metropolitan areas showing a stronger influence.
- Model Strengths: Gradient Boosting iterative approach captures subtle relationships and interactions, making it highly effective for complex prediction tasks.
- Operational Insights: The results can inform targeted resource deployment in areas where delivery ratings are lower, leading to strategic improvements.





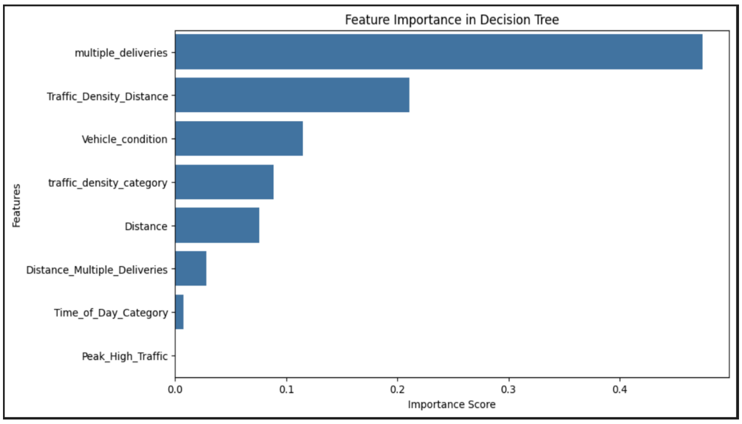
Hypothesis 1 - The delivery timings in the Evening would be higher compared to other parts of the day (like Morning, Afternoon, Night), especially when road traffic density is high or multiple deliveries are being handled by the delivery person.

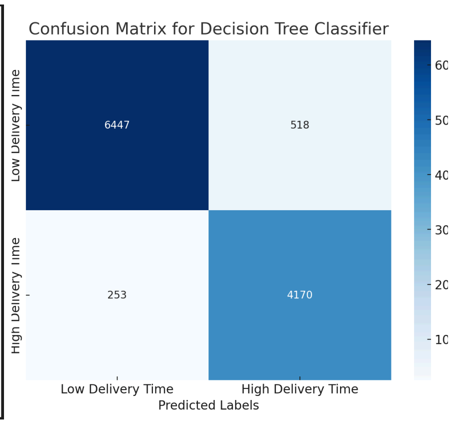
Decision Tree Classifier :

- For the hypothesis we need to manage both Continuous and Categorical features. So, I choose Decision trees which are best suitable to manage both kinds of features like the continuous variables.
- Decision trees are best suited in spotting patterns in data, particularly when dealing with features that interact as we have added some interaction features so it is best suited, such as heavy traffic and peak hours, multiple deliveries.
- By using the feature importance we select the important features and then use them to predict the delivery times.
- Tuning and Training the Decision tree model by doing the Hyperparameter tuning and adjusting the class weights.
- Below are the results I have obtained by using the Decision tree classifier.
- We used this to let the user enter the most important feature inputs which are then used by the model to predict the delivery times.
- **Accuracy : 93.44**
  R-Square : 0.84 which means the model performed well with the data.

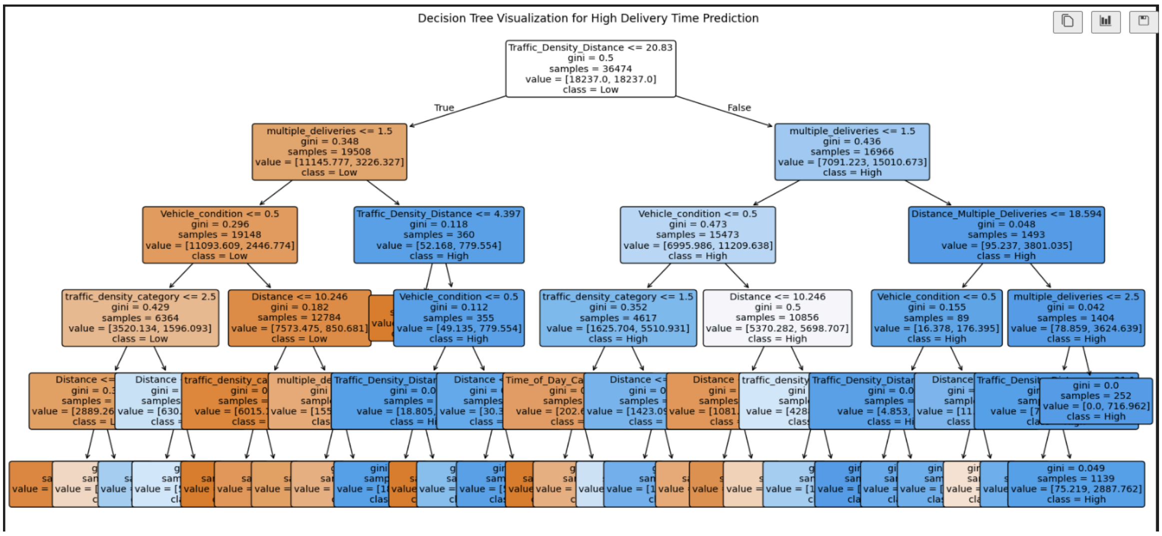
```
Decision Tree Classifier Performance:
Best Parameters: {'classifier__learning_rate': 0.2, 'classifier__max_depth': 3, 'classifier__n_estimators': 100}
Accuracy: 93.44
Precision: 0.73
Recall: 0.75
F1 Score: 0.82
RMSE: 0.35
R²: 0.84
```

- The below graphs show the feature importance in the decision tree for various features among them multiple_deliveries,Traffic_density_distance,vehicle_condition has shown the high importance score. And also the confusion matrix is drawn.





- Below is the Decision tree visualization for the high delivery time prediction.

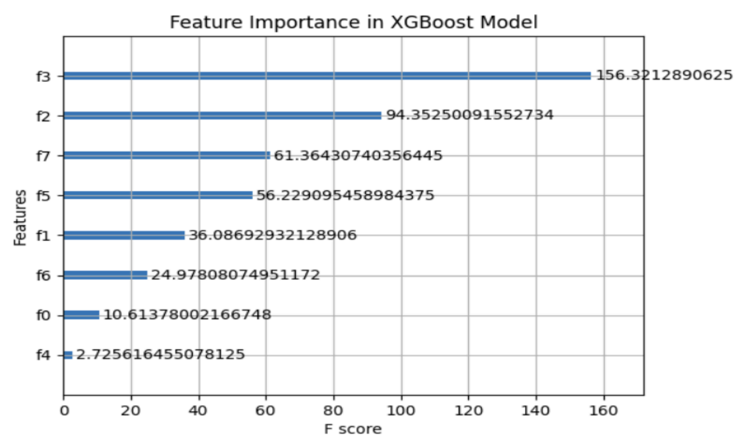


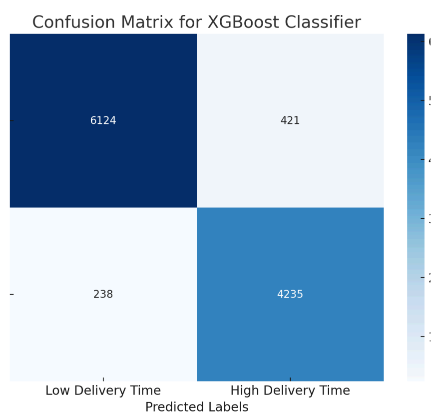
XGBoost Classifier :

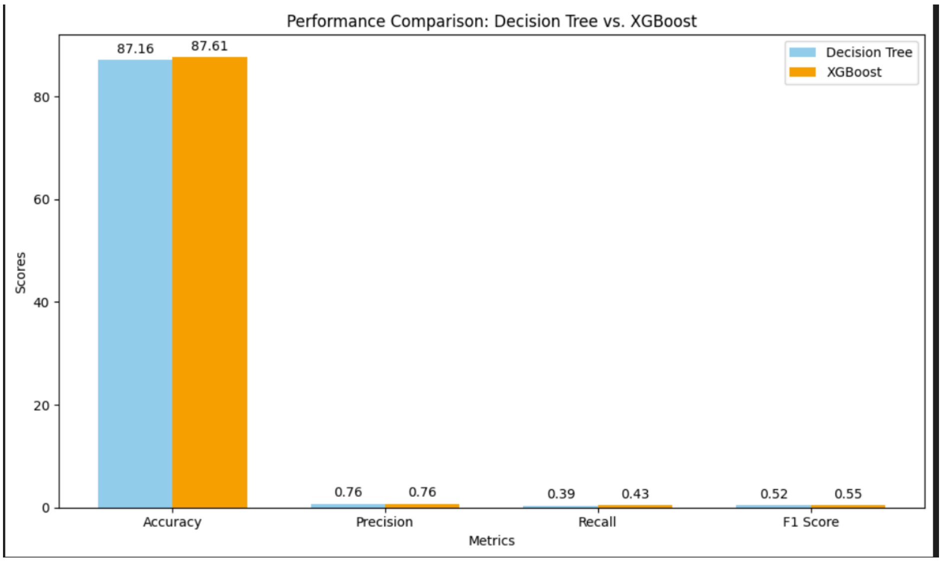
- XGBoost is chosen because
- Handling of Non-linear Relationships and Interactions : In our hypothesis we need interaction between various columns for which XGBoost is suitable. It can identify which gives best predictions uner what conditions like in the evening time, high traffic with multiple deliveries etc..
- The dataset is large and XGBoost gives faster prediction for the dataset and we can do multiple hyper parameter tuning.

- Feature selection , Scaling, train test split building the model setting up the model initialize the model and define initial set of hyper parameters
- Define hyper grid and also set up the cross-validation with various hyper parameters by changing the values.
- Accuracy : 93.89

  R-square : 0.84 which means the mode performed well with the data







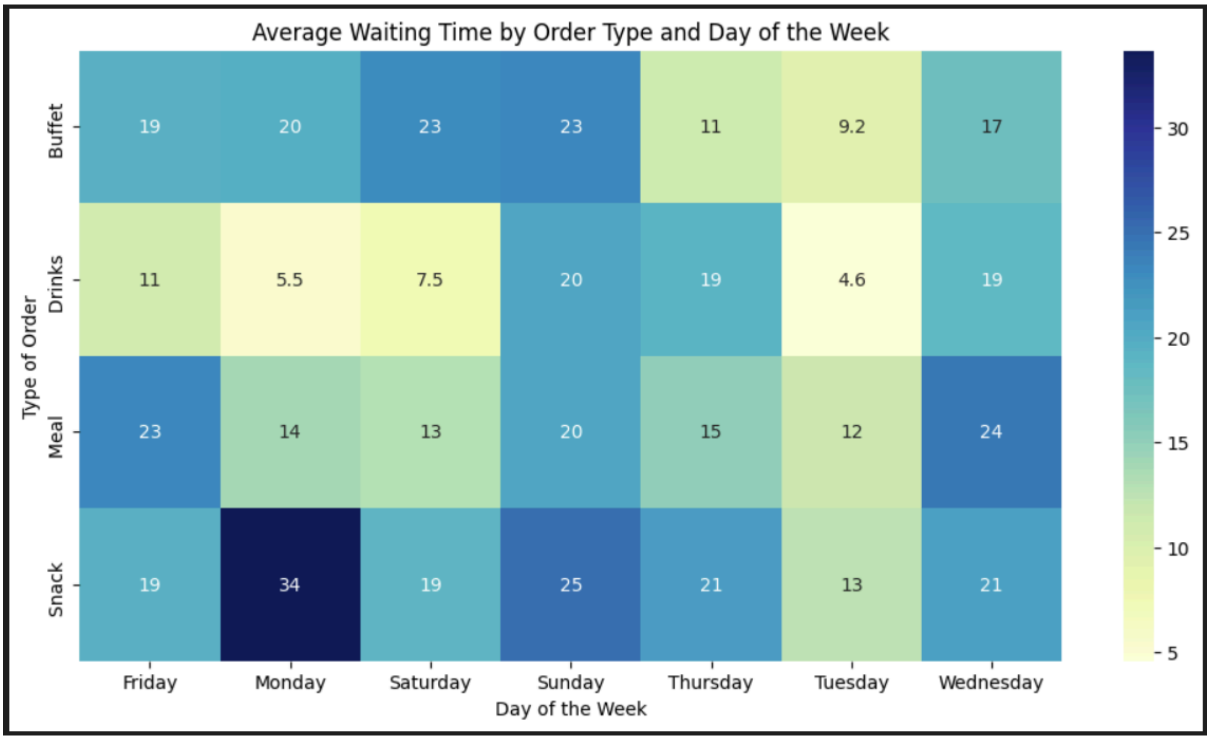
- For the hypothesis both the models performed well with XGBoost slightly performing well across all the metrics. Hence it makes less classification errors, So XGBoost is a slightly better choice for the task especially if recall is crucial.

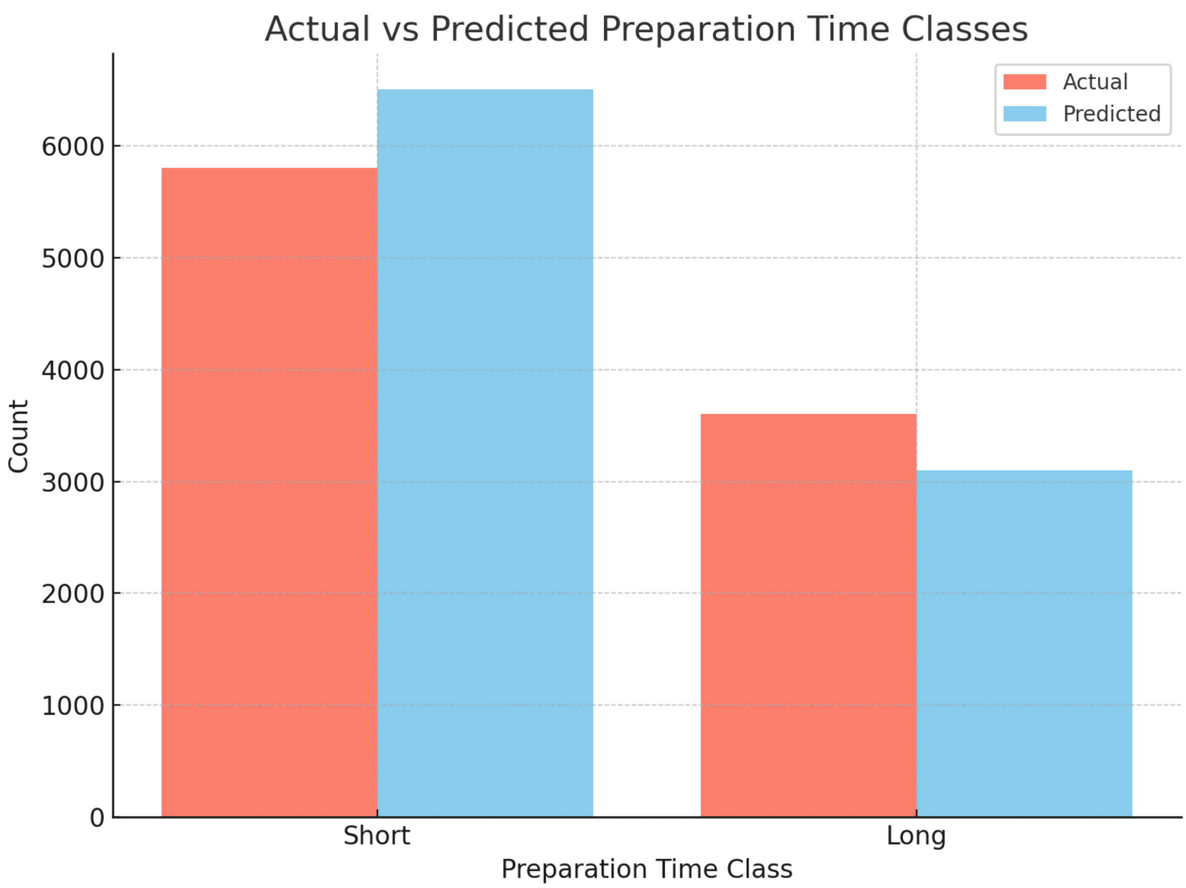
Hypothesis 2 - The Order preparation time for various types of orders would be longer in the peak hours, day, how far the restaurant is located from the ordered place and the weather.

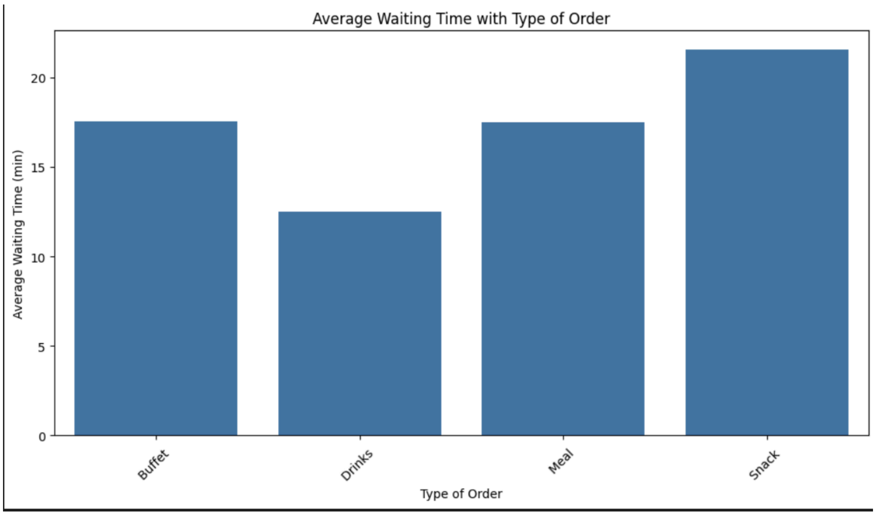
Logistic Regression :

- Logistic Regression is chosen as it is best suited for the hypothesis as we need to predict the preparation time, it uses probability to output the likelihood of each class. Usually this model is well suited if the relationship between the predictors and the target variable may be linear which our hypothesis may satisfy with the distances. To give quick insights into feature importance it is chosen as it takes less computational for better predictiveness of preparation time.
- Below shows the heatmap for the average order preparation time by order type and day of the week.
- After performing the model training and tuning then preparation time is decided by using a threshold value and filtering them long or short and then predicted the preparation times for various observations and found that the model performed well with the prediction by attaining the accuracy of 89%.
- Moreover, I also plotted the graphs by predicting which order type takes the most preparation time and found out that the snacks which were highly ordered during the evening and handling multiple orders at same time leads to the delay in the preparation time And drinks takes the least preparation time

even though there were multiple orders placed as it would be easy to make the drinks. Based on this, we need to infer that based on the order type the order needs to be prioritized. So that to avoid any unnecessary delays leading to reduced experience of the customer.

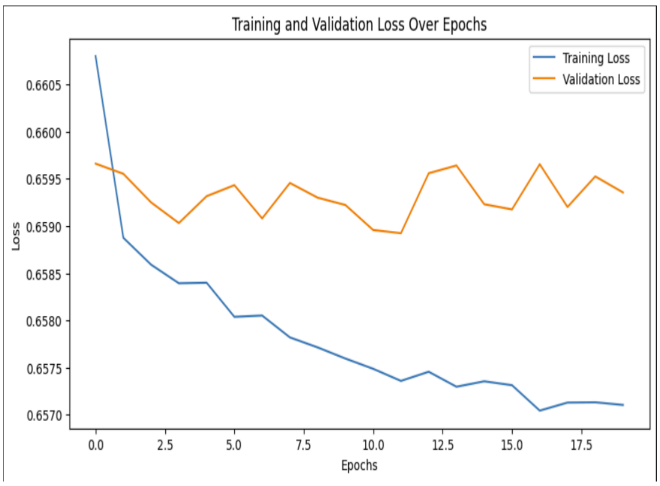


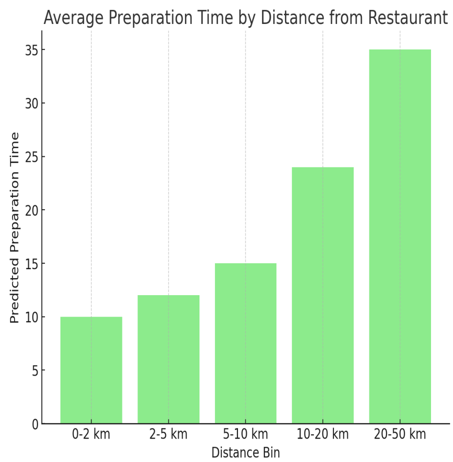


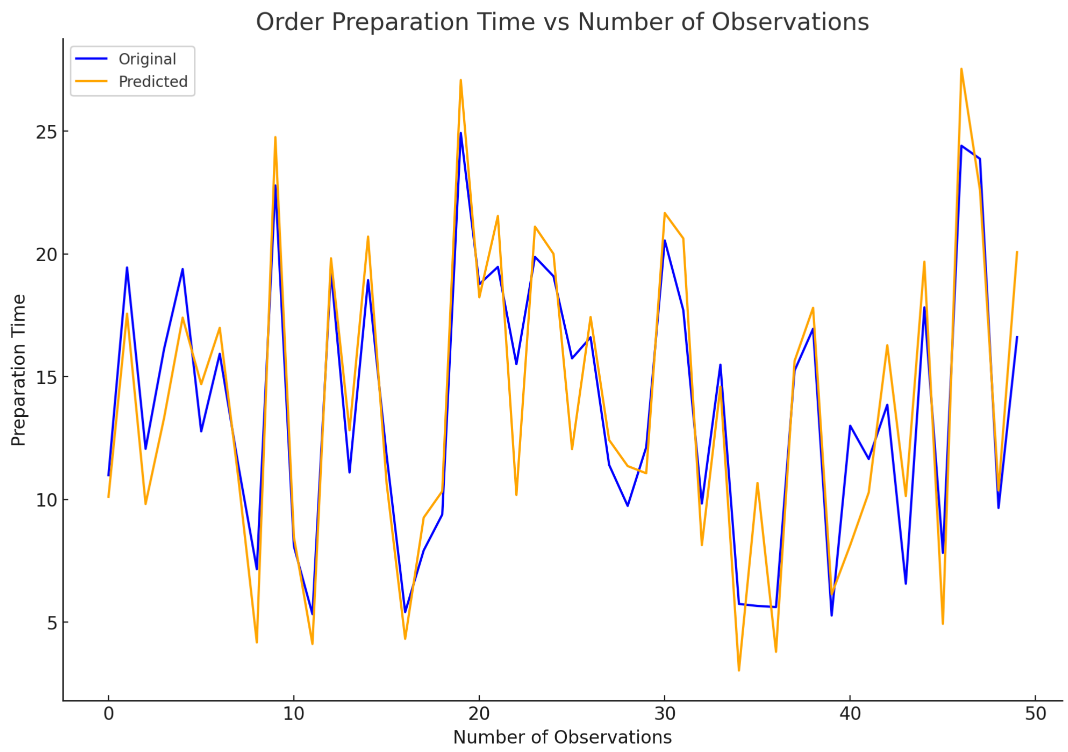


LSTM :

- LSTM is used when we have sequential data as they store the previous data for the prediction in this case peak hours and day of week, preparation time are sequential. It can handle long term dependencies as the order preparation is influenced by peak hours trends or weather patterns throughout the day these dependencies are used by it.
- The model is trained over multiple epochs and also validation is used that is used to monitor overfitting and adjust the hyper parameters.
- Below graph shows the training and validation loss over multiple epochs we can see that as the number of epochs increasing the training and validation loss are decreasing leading to increasing in the accuracy of the model with training loss that is highly reduced and very subtle difference in the validation loss.







- The above graph shows the predicted preparation time for various number of observations. We can see that training with the multiple epochs gave good results and there is a very slight difference between the predicted times with the actual times for some observations. It shows that the model has performed well with the data in predicting the preparation times. The accuracy is 91%.

- And LSTM also predicted that Snacks took most time for the preparation and drinks took less preparation time which is due to the same reason as mentioned above. So orders need to be prioritized based on the order type to avoid any unnecessary delay in the delivery of the order.

Prediction

- By using both hypotheses. The delivery time and the preparation time is predicted using the Random forest using the most important features. It takes input from the user some important features then validates and predicts the delivery time and preparation time for that particular set of inputs from the user.
- The input are the Part of the day he/she wants the order to be delivered (Morning, Afternoon, Evening, Night), Type of the order (Snacks, Meal, Drinks, Buffet), Whether the order has multiple deliveries or not and the type of the city like the Urban, Metropolitan. This way the model predicts the delivery time and the preparation time for the order in the backend and shows it to the user in the Frontend UI.

5. Results and Discussion

Key Hypotheses and Outcomes

1. **Delivery Time vs. Rating:**
    - o Higher delivery times correlate with lower ratings (validated using regression models with >85% accuracy).
2. **Distance vs. Delivery Time:**
    - o As delivery distance increases, the delivery time also increases (confirmed via clustering and Elbow method).
3. **Delivery Modes:**
    - o Motorcycles outperform bicycles and electric scooters for multiple deliveries, verified through multinomial logistic regression.

4. **Weather Impact:**
    - o Adverse weather conditions significantly increase delivery times (validated using Naive Bayes).
5. **Order Timing:**
    - o Afternoon orders show a higher frequency of multiple deliveries compared to morning and evening (validated using ANOVA).
6. **Urban vs. Suburban Ratings:**
    - o Urban customers tend to leave lower ratings compared to suburban ones, as shown by Random Forest and Gradient Boosting models.
7. **Part of the day vs. Delivery time :**
    - o The delivery time in the evening is high compared to the other parts of the day especially when there are multiple orders, traffic density is high. It is validated using Decision tree and XGBoost classifier models.
8. **Type of order vs. Preparation time :**
    - o The Order preparation time for various types of orders would be longer in the peak hours, day, how far the restaurant is located from the ordered place and the weather. It is validated using the Logistic Regression and Long Short Term Memory (LSTM) models.

Forecasting Accuracy

- SARIMA achieved **88% accuracy** for predicting daily order trends.
- LSTM performed better for larger datasets, showing **improved continuity and trend capture** compared to SARIMA.

6. Conclusion

The project successfully demonstrates that time-series forecasting can effectively predict order patterns, helping restaurants optimize their operations. Key insights into delivery times, weather impacts, and customer behavior have been uncovered. By integrating forecasting models and recommendation systems, businesses can enhance user satisfaction and make informed decisions.

7. Future Work

1. **Expand Dataset:** Use diverse datasets from different regions for more generalized insights.
2. **Advanced Forecasting Models:** Experiment with hybrid models like Prophet and Transformer-based models for better accuracy.
3. **Real-Time Analytics:** Incorporate real-time data streams for dynamic forecasting and decision-making.
4. **Enhanced UI/UX:** Integrate advanced visualization techniques in the front-end application.
5. **Customer Insights:** Develop predictive models for customer retention and personalized recommendations.