PROJECT REPORT

602: Introduction to Data Analysis and Machine Learning

PREDICTIVE ANALYSIS OF FOOD DELIVERY AND OPTIMISATION

1.Introduction:

In the food delivery industry, efficiency and customer satisfaction are paramount. The rise of online ordering and the demand for seamless dining experiences highlights the importance of accurately predicting serving times Our work aims to meet this challenge head on by developing a robust machine learning algorithm focused solely on feeding time prediction.

Food delivery is a prototype for taking culinary delights from restaurants, stores, or food delivery companies that successfully move them to customers' doorsteps whether it be in the form of gourmet snacks, refreshing drinks, everyday groceries, carefully packaged and swiftly transported by conscientious and determined delivery crews across urban landscapes.

At the heart of our work is a hybrid of predictive analytics and advanced algorithms, optimized to accurately predict feeding times Our machine learning algorithm taps into historical data including timestamping ordering, delivery locations, weather, and other related factors Empowers the food delivery service to improve quality, reduce wait times, and ultimately improve customer satisfaction.

Our services represent a paradigm shift in the foodservice landscape, delivering comprehensive solutions to enhance operational efficiency and customer experience. By using predictive analytics, we want to enable food delivery to adapt to the dynamic requirements of today's customers, delivering transparent, fast and reliable service for ease of operation and timely delivery of goods.

In summary, our project is poised to revolutionize food delivery by harnessing the power of predictive analytics to accurately predict delivery times. Through careful data analysis and algorithmic precision, we aim to empower food service providers to better allocate supplies, reduce wait times, and increase customer satisfaction,

and we have finally redefined the standards of efficiency and reliability in the food delivery industry.

2.Data Overview:

Description:

The dataset contains information about the food delivery business, including characteristics of orders, delivery personnel, and environmental conditions.

Significance to the Project:

our dataset is an important source of information for our work, facilitating the development of predictive models for food delivery times. By analyzing the data, we aim to identify the key factors that affect delivery time and improve resource allocation to increase efficiency.

Source and Collection Process:

https://www.kaggle.com/datasets/gauravmalik26/food-delivery-dataset/data

The dataset was sourced from Kaggle, which likely aggregates data from online ordering platforms, mobile apps, or direct integrations with food delivery services. The data collection process involves gathering information on order timestamps, delivery locations, weather conditions, and other relevant factors.

Dataset Shape:

The dataset comprises 45,593 entries and 20 features. Each entry represents a unique food delivery order, while the features encompass different aspects of the delivery process:

Column	Description
ID order	ID number
Delivery_person_ID	ID number of the delivery partner
Delivery_person_Age	Age of the delivery partner
Delivery_person_Ratings	Ratings of the delivery partner

Restaurant_latitude	The latitude of the restaurant
Restaurant_longitude	The longitude of the restaurant
Delivery_location_latitude	The latitude of the delivery location
Delivery_location_longitude	The longitude of the delivery location
Order_Date	Date of the order.
Time_Orderd	Time the order was placed.
Time_Order_picked	Time the order was picked
Weatherconditions	Weather conditions of the day
Road_traffic_density	Density of the traffic
Vehicle_condition	Condition of the vehicle
Type_of_order	The type of meal ordered by the customer
Type_of_vehicle	The type of vehicle delivery partner rides
multiple_deliveries	Amount of deliveries driver picked
Festival	If there was a Festival or no.
City	Type of city
Time_taken(min)	The time taken to complete the order

3.Data Cleaning:

Data Extraction:

- The extract_data function was implemented to preprocess the dataset, converting certain columns to appropriate data types and extracting relevant information from others.
- The Time_taken(min) column was standardized to integer values, and the Weatherconditions column was cleaned to remove extra spaces.
- A new column Code was created by splitting the Delivery_person_ID column.

- Irrelevant columns (ID and Delivery_person_ID) were dropped from the dataset.

Combining and Transforming Date and Time Features:

- Date and time features (Order_Date, Time_Orderd, and Time_Order_picked) were combined and transformed into datetime objects for ease of analysis.

Checking for Duplicate Values:

- A check was performed to identify and confirm the absence of duplicate values in the dataset.

Updating Data Types:

- Data types of certain columns (Delivery_person_Age, Delivery_person_Ratings, and multiple_deliveries) were updated to float64.
- The Order_Date column was converted to datetime format.

Removing Extra Spaces:

- Extra spaces in categorical columns were removed to ensure data consistency.

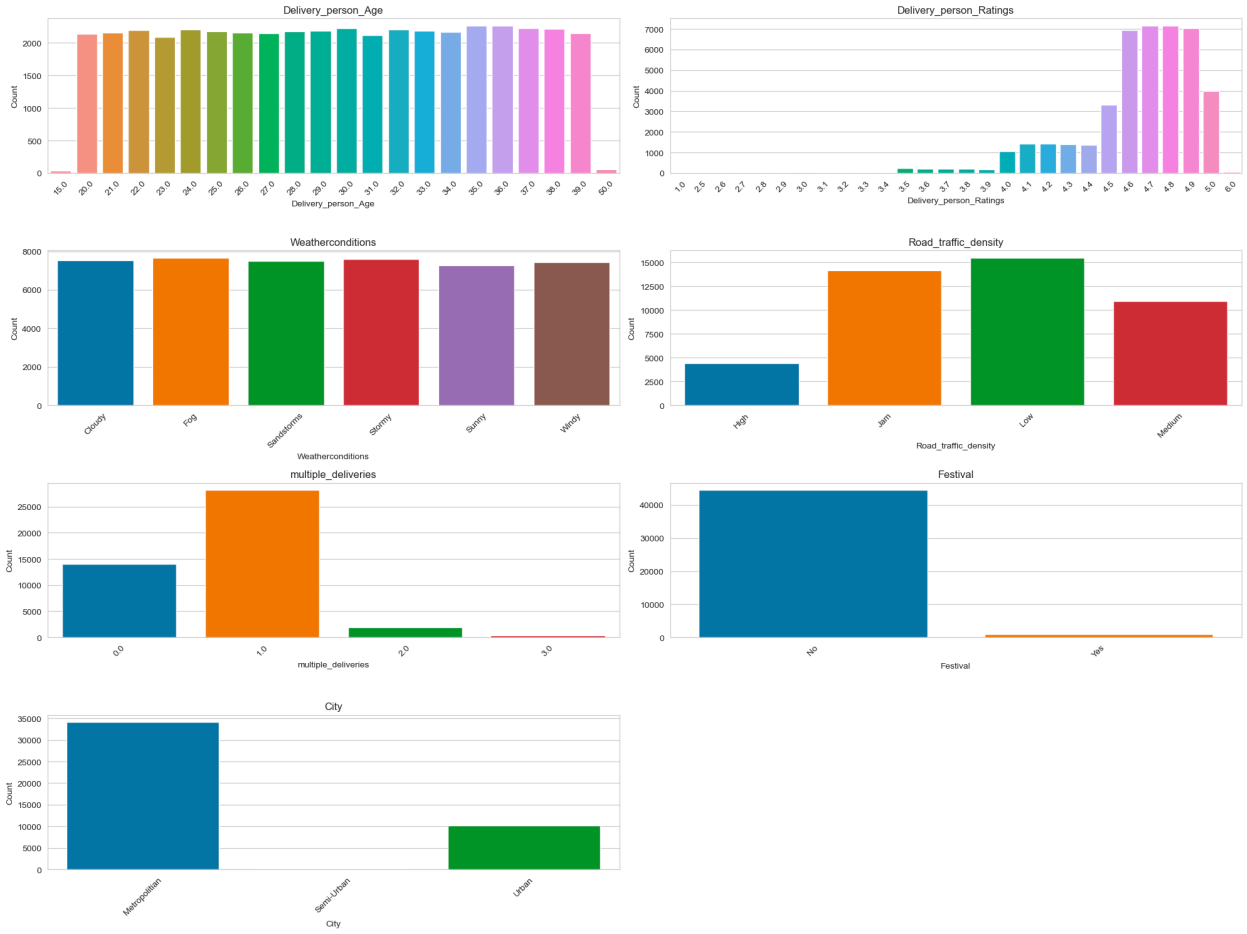
Handling Missing Values:

- Missing values were identified and visualized to understand their distribution across different features.
- Columns like Order_day and Time_Orderd had 1731 missing values, leading to the same amount of missing values in engineered time features.
- Strategies for handling missing values were not explicitly stated in this summary but would likely involve techniques such as imputation or deletion based on the nature and distribution of missing values in each column.

Handling Null Values:

- The handle_null_values function was implemented to fill missing values in the dataset. Each feature was handled based on its distribution and data type:

- Delivery_person_Age and Weatherconditions: Missing values filled randomly.
- Delivery_person_Ratings: Missing values filled with the median.
- Remaining categorical columns (City, Festival, multiple_deliveries, Road_traffic_density): Missing values filled with the mode.
- The implementation of these strategies ensures that missing values are appropriately handled, allowing for a cleaner and more robust dataset for further analysis and modeling.



4.Feature Engineering:

Feature engineering involves creating new features from existing ones to provide valuable insights or improve model performance. In this section, we detail the feature engineering steps performed on the dataset:

Date Attribute Creation: We extracted various date attributes from the Order_Date column, including day, month, year, and day of the week. Additionally, we created a binary variable indicating whether the order was placed on a weekend.

Order Preparation Time Calculation:

- We calculated the order preparation time by determining the time elapsed between order placement (Time_Orderd) and order pick-up (Time_Order_picked).
- An adjustment was made to handle cases where the order pick-up time occurred before the order placement time, ensuring accurate calculation.

Hourly Time Period Classification: We categorized the order placement time (Time_Orderd) into different time periods of the day (morning, noon, afternoon, evening, and night) to capture potential temporal patterns.

Distance Calculation:

- Using trigonometry, we calculated the distance between the restaurant and the delivery location based on their respective latitude and longitude coordinates.
- This new feature provides spatial information that may influence delivery time.

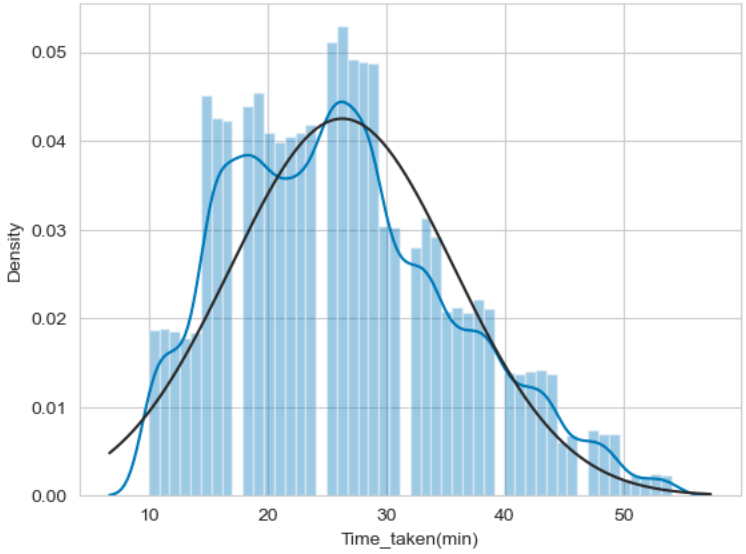
5.Data Visualisation:

Examining the Distribution of Delivery Times:

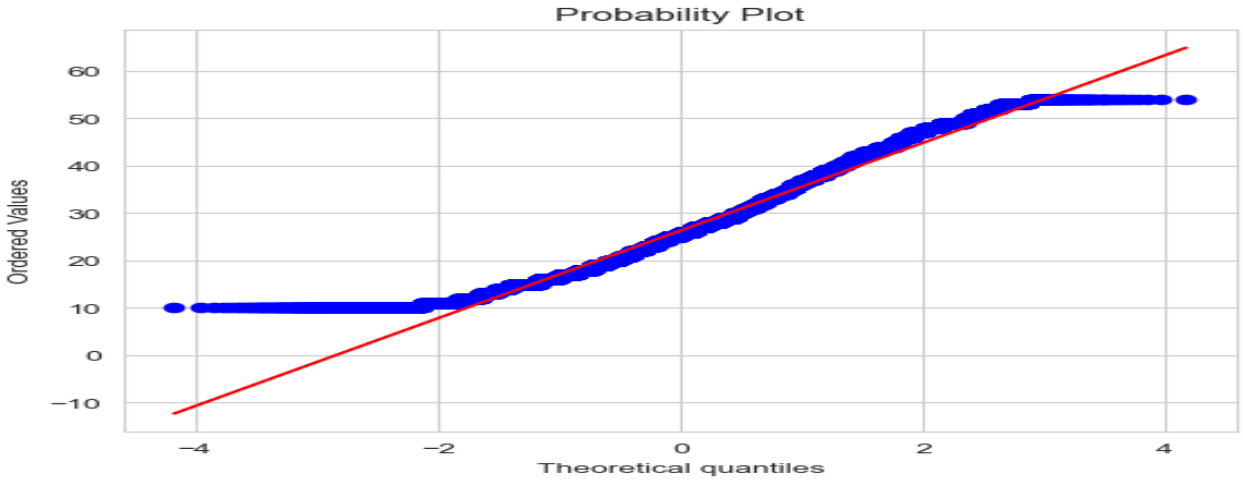
Examining the distribution of delivery times and comparing it to a normal distribution provides valuable insights into the characteristics of the data. This analysis informs subsequent data preprocessing steps and model selection. Here's how we performed this analysis:

- **Distribution Plot:** We visualized the distribution of delivery times using a histogram along with a fitted normal distribution curve. This plot allows us

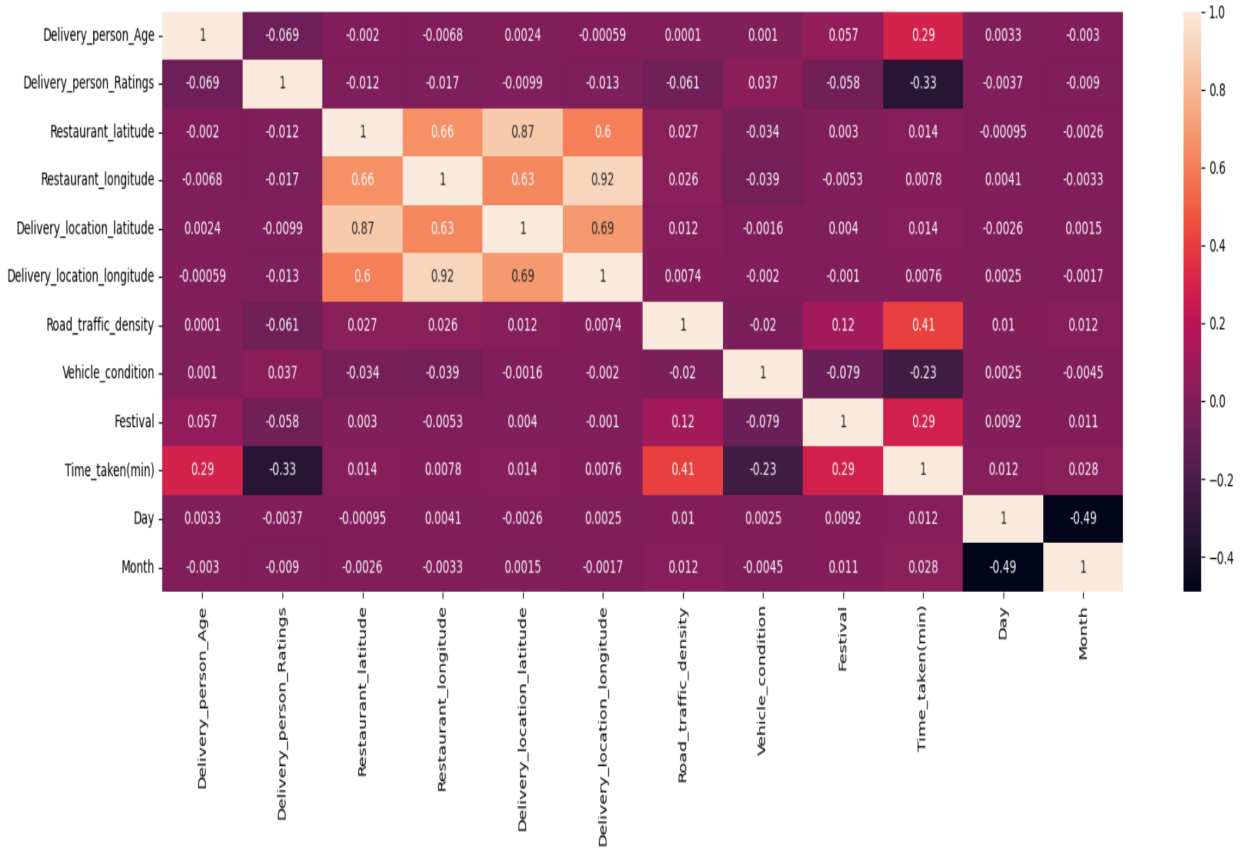
to observe the shape of the delivery time distribution and compare it to a theoretical normal distribution.



- **Fitting a Normal Distribution:**
  We fitted a normal distribution to the delivery time data and extracted the mean (mu) and standard deviation (sigma) to characterize the distribution.
- **Q-Q Plot:**
  We generated a Quantile-Quantile (Q-Q) plot to visually assess whether the delivery time data follows a normal distribution. The Q-Q plot compares the quantiles of the delivery time data to the quantiles of a theoretical normal distribution.



6.Correlation Analysis:



7.Data Transformation:

Categorical feature encoding is a critical preprocessing step in machine learning, enabling the transformation of categorical values into numerical representations. This conversion facilitates the utilization of machine learning models designed to handle numerical data effectively. In our project, we employ Label Encoding to encode categorical variables.

Label Encoding:

Label Encoding is chosen for encoding categorical features due to the presence of multiple features with categorical values. This method assigns a unique numerical label to each category within a feature. The numerical labels are typically integers ranging from 0 to (n-1), where n represents the number of unique categories in the feature.

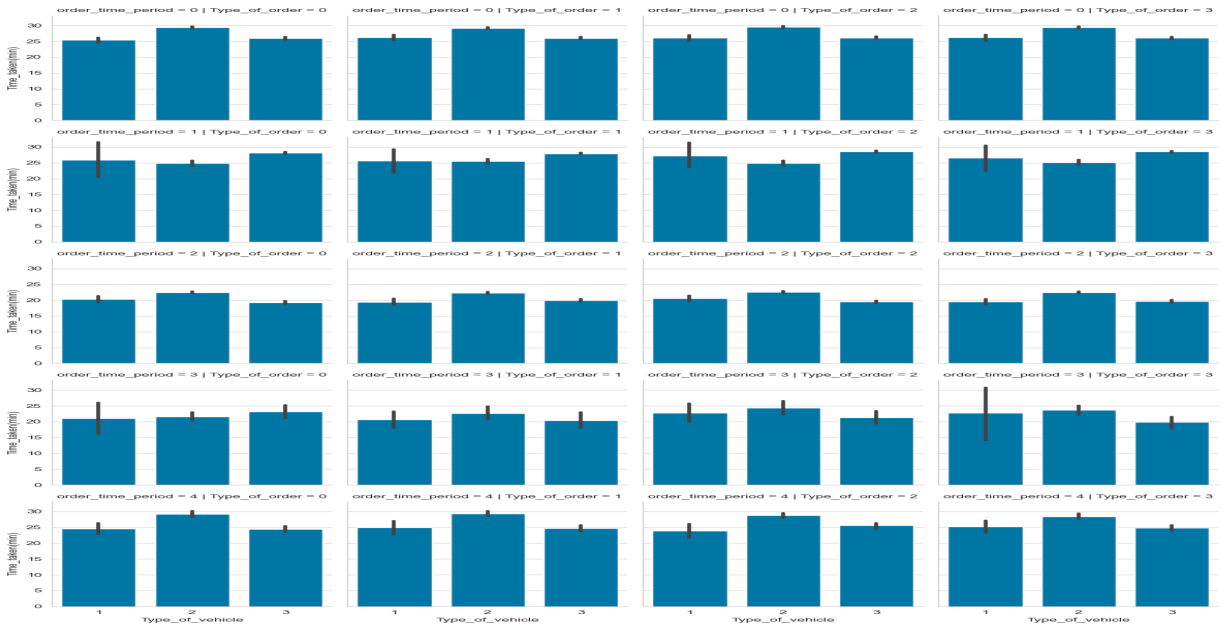
After applying Label Encoding, categorical features are transformed into numerical representations suitable for machine learning algorithms. These encoded features enable the seamless integration of categorical data into our predictive models, enhancing their accuracy and performance.

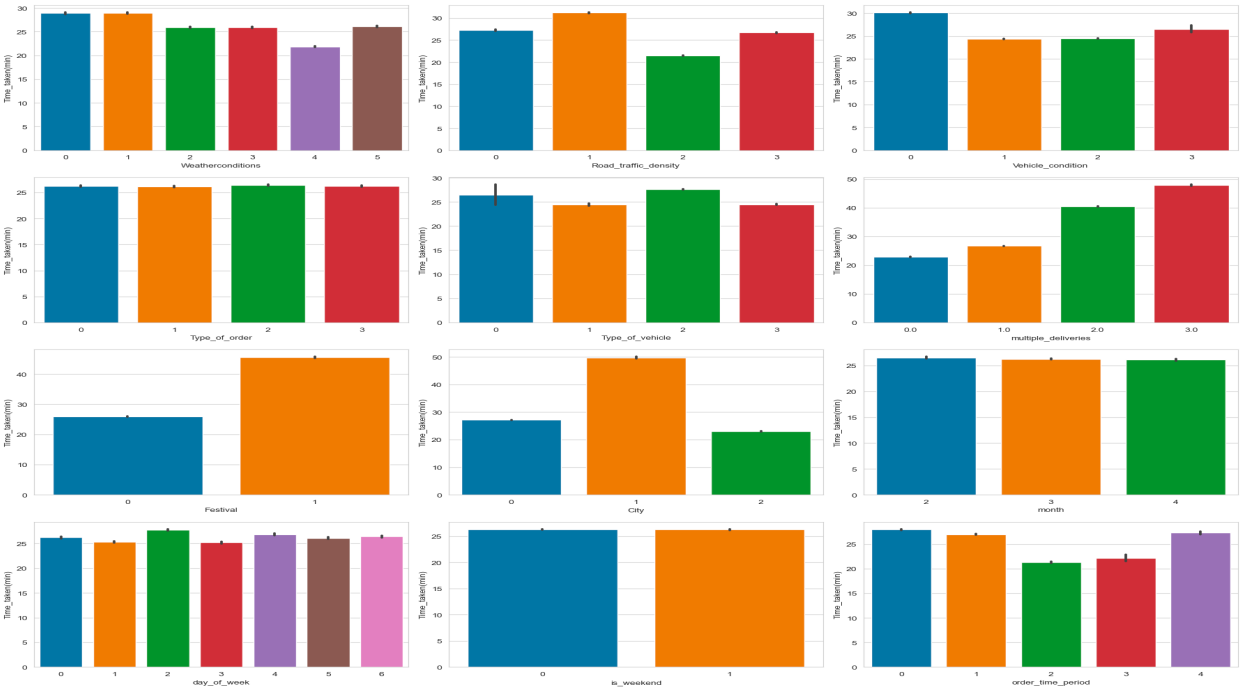
8. Exploratory Data Analysis (EDA):

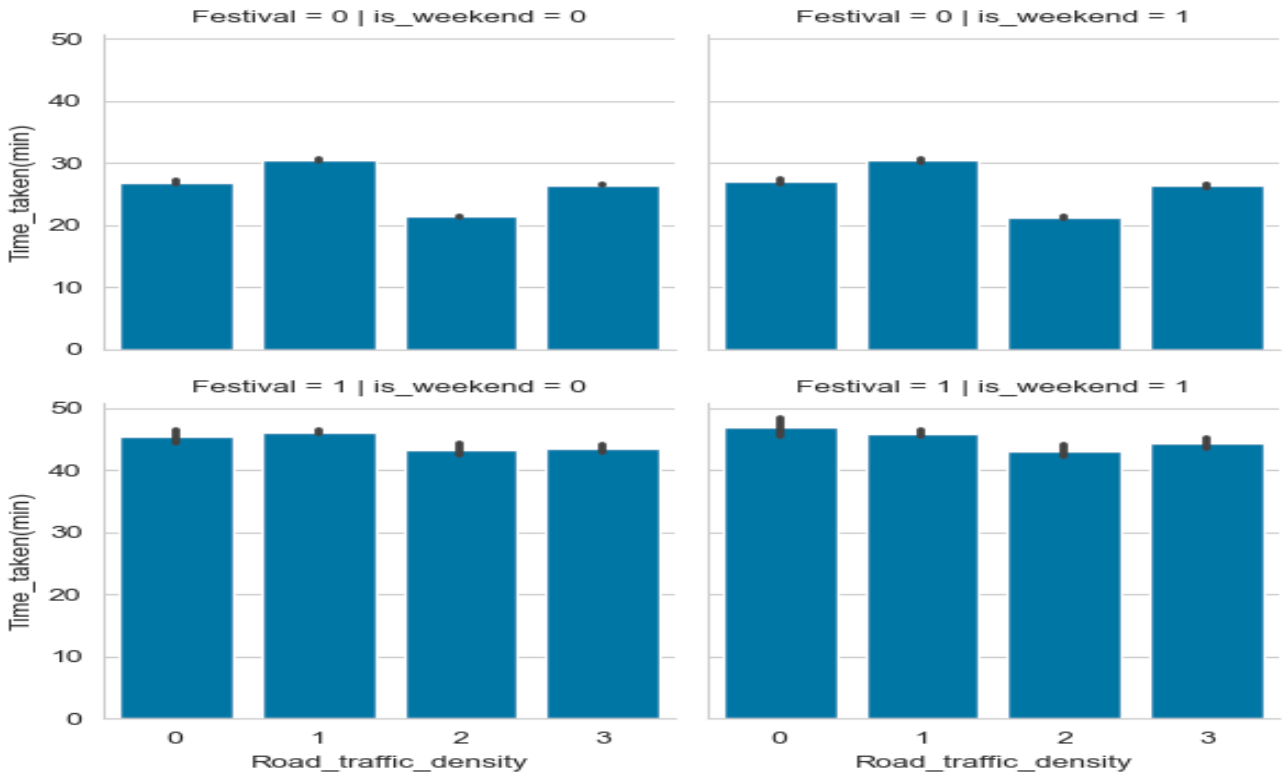
Exploratory Data Analysis (EDA) is a critical step in understanding the underlying patterns, trends, and relationships within the dataset. In our food delivery prediction project, EDA provides valuable insights into various factors influencing delivery times. Here's a summary of our EDA findings:

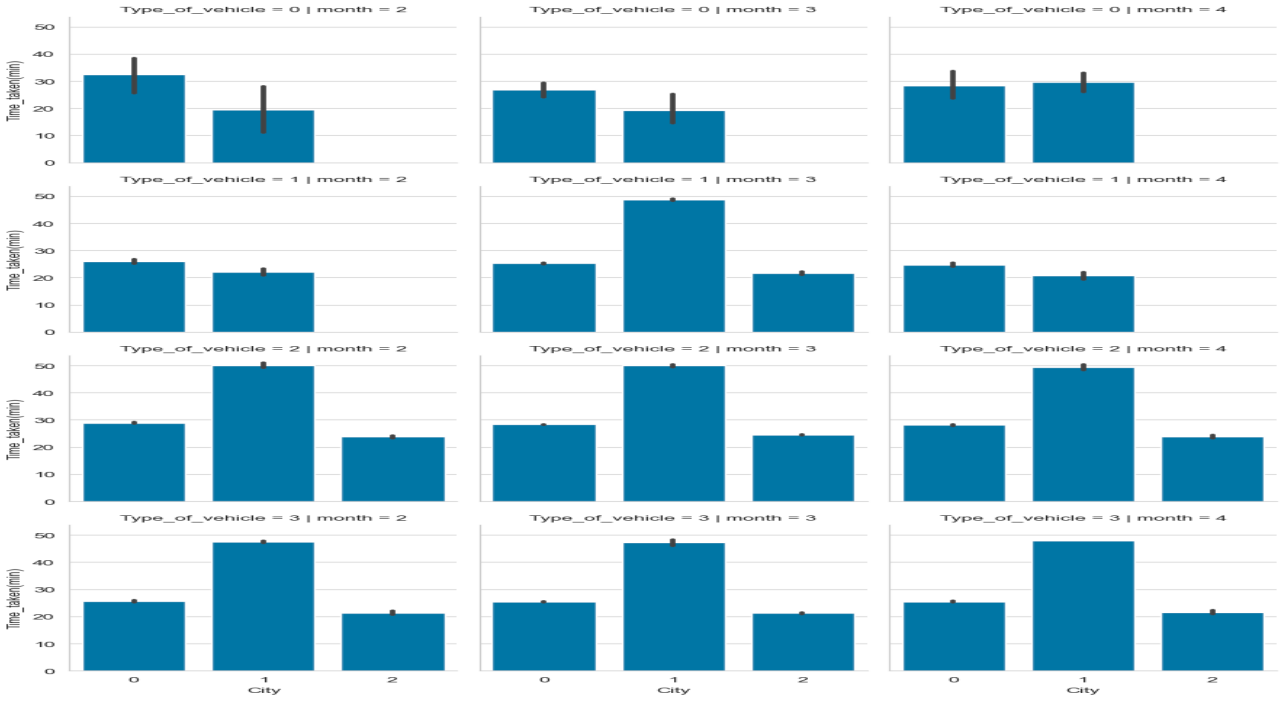
- **Analysis of Categorical Variables:** We analyzed the impact of categorical variables on delivery times by visualizing their distributions and relationships with the target variable (Time_taken(min)).
    1. Bar Plots: Bar plots were used to visualize the average delivery times for different categories within each categorical variable. For instance, we explored the impact of variables such as month, is_weekend, Type_of_order, Type_of_vehicle, Festival, Road_traffic_density, and City on delivery times. These visualizations revealed significant variations in delivery times across different categories, highlighting the influence of categorical variables on the delivery process.
    2. FacetGrid Plots: To delve deeper into categorical interactions, we utilized FacetGrid plots to visualize the relationship between multiple categorical variables and delivery times. For example, we examined how the type of order, type of vehicle, festival occurrence, and road traffic density collectively affect delivery times. These plots provided insights into how combinations of categorical variables impact delivery efficiency.
- **Analysis of Numerical Variables:** We explored the relationship between numerical variables and delivery times to identify potential correlations or patterns.
    1. Scatter Plots: Scatter plots were used to visualize the relationship between numerical variables such as Delivery_person_Age and

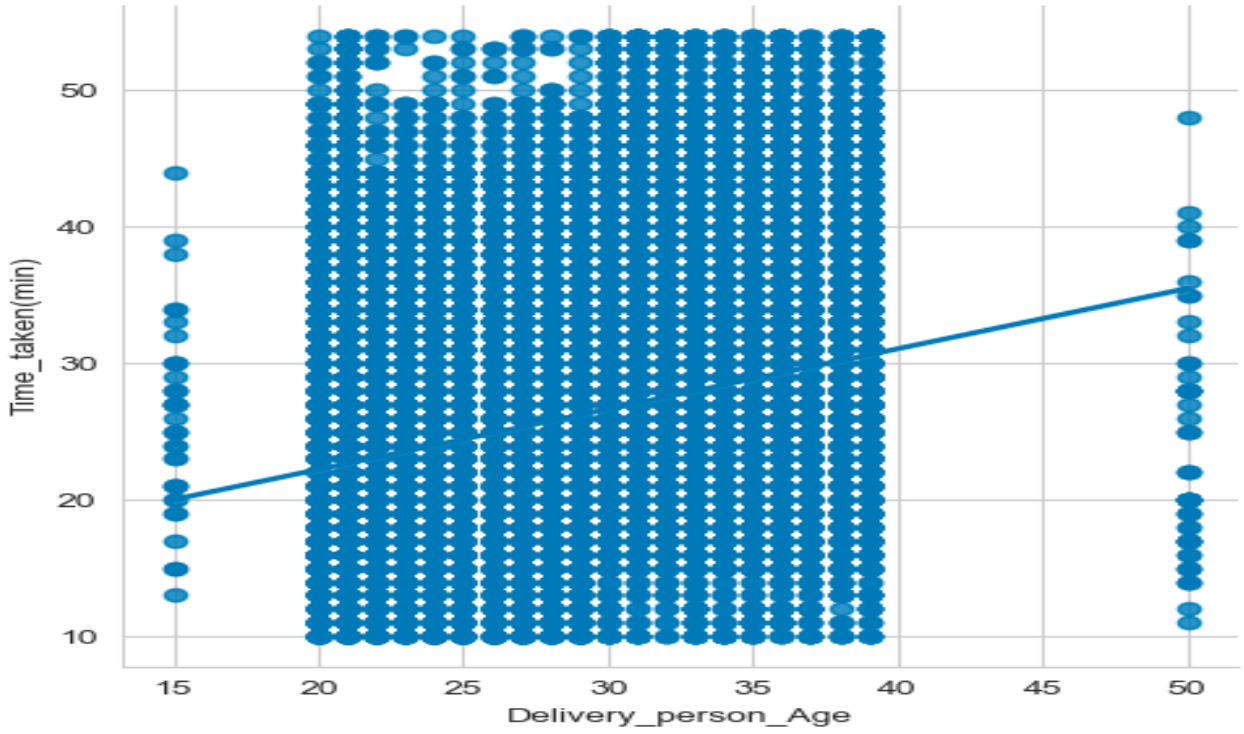
order_prepare_time with delivery times. Notably, we observed a linear relationship between the courier's age and delivery time, suggesting potential implications for workforce management.

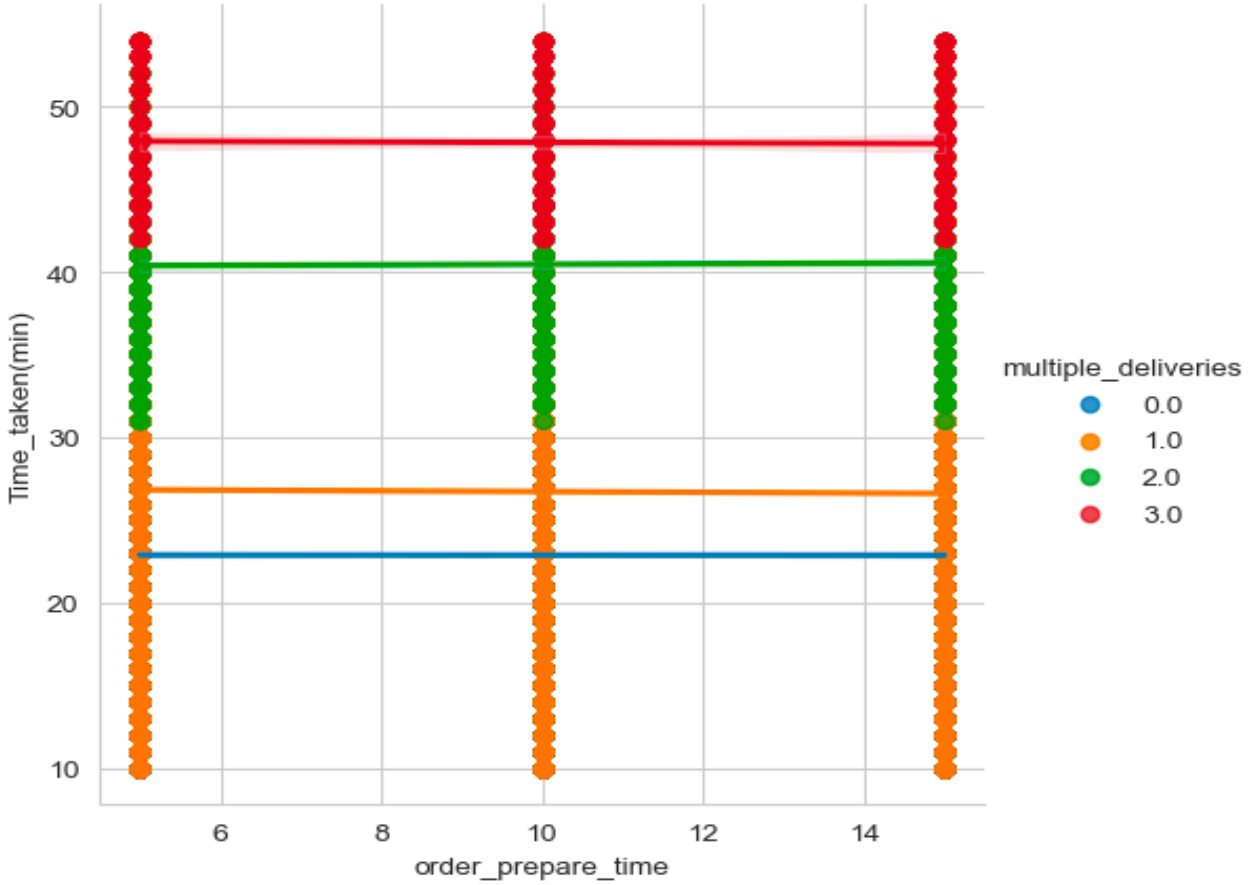


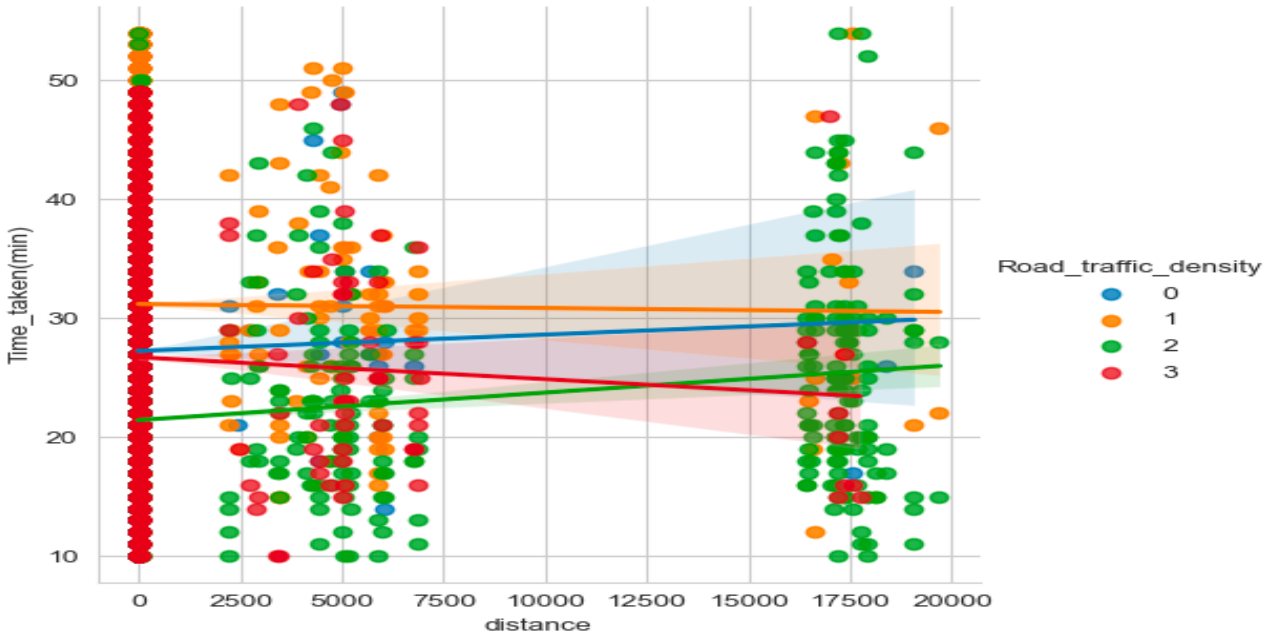



Festival = 0 | is_weekend = 0
Festival = 0 | is_weekend = 1
Festival = 1 | is_weekend = 0
Festival = 1 | is_weekend = 1
Time_taken(min)
Road_traffic_density
0
10
20
30
40
50
0
1
2
3


Type_of_vehicle = 0 | month = 2
Type_of_vehicle = 0 | month = 3
Type_of_vehicle = 0 | month = 4
Type_of_vehicle = 1 | month = 2
Type_of_vehicle = 1 | month = 3
Type_of_vehicle = 1 | month = 4
Type_of_vehicle = 2 | month = 2
Type_of_vehicle = 2 | month = 3
Type_of_vehicle = 2 | month = 4
Type_of_vehicle = 3 | month = 2
Type_of_vehicle = 3 | month = 3
Type_of_vehicle = 3 | month = 4
Time_taken(min)
City
0
10
20
30
40
50
0
1
2


Time_taken(min)
50
40
30
20
10
15
20
25
30
35
40
45
50
Delivery_person_Age


Time_taken(min)
50
40
30
20
10
6
8
10
12
14
order_prepare_time
multiple_deliveries
0.0
1.0
2.0
3.0



9.Data Modelling and Model Evaluation:

After conducting Exploratory Data Analysis (EDA), the next step is to build predictive models to estimate food delivery durations accurately. Here's how we approached the data modelling and evaluation process:

Train-Test Split:

We divided the dataset into training, validation, and test sets to develop and evaluate our models effectively. The train_test_split function from Scikit-Learn was used for this purpose.

- Training Set: Used to train various machine learning models.
- Validation Set: Utilized for fine-tuning hyperparameters and model selection.
- Test Set: Reserved for final model evaluation and assessing performance on unseen data.

Model Building:

We experimented with several regression algorithms to predict delivery durations:

- Linear Regression
- K-Nearest Neighbors (KNN)
- Random Forest
- Gradient Boosting
- AdaBoost
- Decision Tree
- XGBoost

Model Evaluation:

To evaluate the performance of each model, we used cross-validation with multiple metrics, including R-squared (R^2) and Root Mean Squared Error (RMSE).

- **R-squared (R^2):** Indicates the proportion of variance in the target variable that is predictable from the independent variables. A higher R^2 value indicates a better fit of the model to the data.
- **Root Mean Squared Error (RMSE):** Represents the square root of the average squared differences between predicted and actual values. It provides a measure of the model's accuracy, with lower values indicating better performance**.**

Hyperparameter Optimization:

We performed hyperparameter optimization for the top-performing algorithms, namely Random Forest, Gradient Boosting, and XGBoost. This involved tuning the algorithm's parameters to further improve model performance.

Model Selection and Comparison:

After hyperparameter optimization, we compared the performance of the models based on R-squared and RMSE scores. The XGBoost algorithm with optimized hyperparameters yielded the best results among the models considered.

Model Visualization:

To visualize the performance of the XGBoost model, we plotted the actual delivery durations against the predicted values for the first 50 observations in the test set.

This provided a graphical representation of how well the model predicts delivery times compared to the actual values.

Through rigorous data modelling and evaluation, we have developed an accurate predictive model using XGBoost, which can effectively estimate food delivery durations. This model holds the potential to optimize delivery operations, enhance customer satisfaction, and drive business efficiency in the food delivery service.

Conclusion:

Through rigorous data modelling and evaluation, we have developed an accurate predictive model using XGBoost, which can effectively estimate food delivery durations. This model holds the potential to optimize delivery operations, enhance customer satisfaction, and drive business efficiency in the food delivery service.

This visualization allows us to observe how well the XGBoost model predicts the delivery durations compared to the actual values.