

Order Delivery Time Prediction

Objectives:

The objective of this assignment is to build a regression model that predicts the delivery time for placed through Porter. The model will use various features such as the items ordered, the restaurant location, the order protocol, and the availability of delivery partners.

The key goals are:

Predict the delivery time for an order based on multiple input features

Improve delivery time predictions to optimize operational efficiency

Understand the key factors influencing delivery time to enhance the model's accuracy

Data Pipeline:

The data pipeline for this assignment will involve the following steps:

1. Data Loading
2. Data Preprocessing and Feature Engineering
3. Exploratory Data Analysis
4. Model Building
5. Model Inference

Data Understanding:

The dataset contains information on orders placed through Porter, with the following columns:

Field	Description
market_id	Integer ID representing the market where the restaurant is located.
created_at	Timestamp when the order was placed.
actual_delivery_time	Timestamp when the order was delivered.

Field	Description
store_primary_category	Category of the restaurant (e.g., fast food, dine-in).
order_protocol	Integer representing how the order was placed (e.g., via Porter, call to restaurant, etc.).
total_items	Total number of items in the order.
subtotal	Final price of the order.
num_distinct_items	Number of distinct items in the order.
min_item_price	Price of the cheapest item in the order.
max_item_price	Price of the most expensive item in the order.
total_onshift_dashers	Number of delivery partners on duty when the order was placed.
total_busy_dashers	Number of delivery partners already occupied with other orders.
total_outstanding_orders	Number of orders pending fulfillment at the time of the order.
distance	Total distance from the restaurant to the customer.

Importing Necessary Libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

1. Loading the data

Load 'porter_data_1.csv' as a DataFrame

2. Data Preprocessing and Feature Engineering [15 marks]

2.1 Fixing the Datatypes [5 marks]

The current timestamps are in object format and need conversion to datetime format for easier handling and intended functionality

2.1.1 [2 marks]

Convert date and time fields to appropriate data type

```
Out[ ]: created_at      datetime64[ns] actual_delivery_time datetime64[ns] dtype: object
```

2.1.2 [3 marks]

Convert categorical fields to appropriate data type

```
Out[ ]: market_id      category store_primary_category category order_protocol
        category dtype: object
```

2.2 Feature Engineering [5 marks]

Calculate the time taken to execute the delivery as well as extract the hour and day at which the was placed

2.2.1 [2 marks]

Calculate the time taken using the features actual_delivery_time and created_at

	delivery_time_minutes	order_hour	order_day
0	47.0	22	4
1	44.0	21	1
2	55.0	0	0
3	59.0	3	3
4	46.0	2	1

2.2.2[3 marks]

Extract the hour at which the order was placed and which day of the week it was. Drop the unnecessary columns.

created_at	order_day	isWeekend
0	2015-02-06 22:24:17	4 0
1	2015-02-10 21:49:25	1 0
2	2015-02-16 00:11:35	0 0
3	2015-02-12 03:36:46	3 0
4	2015-01-27 02:12:36	1 0

#Drop unnecessary columns

```
Out[ ]: Index(['market_id', 'store_primary_category', 'order_protocol', 'total_items', 'subtotal', 'num_distinct_items', 'min_item_price', 'max_item_price', 'total_onshift_dashers', 'total_busy_dashers', 'total_outstanding_orders', 'distance', 'delivery_time_minutes', 'order_hour', 'order_day', 'isWeekend'], dtype='object')
```

2.3 Creating training and validation sets [5 marks]

2.3.1[2 marks]

Define target and input features

```
Out[ ]: ((175777, 15), (175777,))
```

2.3.2[3 marks]

Split the data into training and test sets

```
Out[ ]: ((140621, 15), (35156, 15), (140621,), (35156,))
```

3. Exploratory Data Analysis on Training Data [20 m]

Analyzing the correlation between variables to identify patterns and relationships

Identifying and addressing outliers to ensure the integrity of the analysis

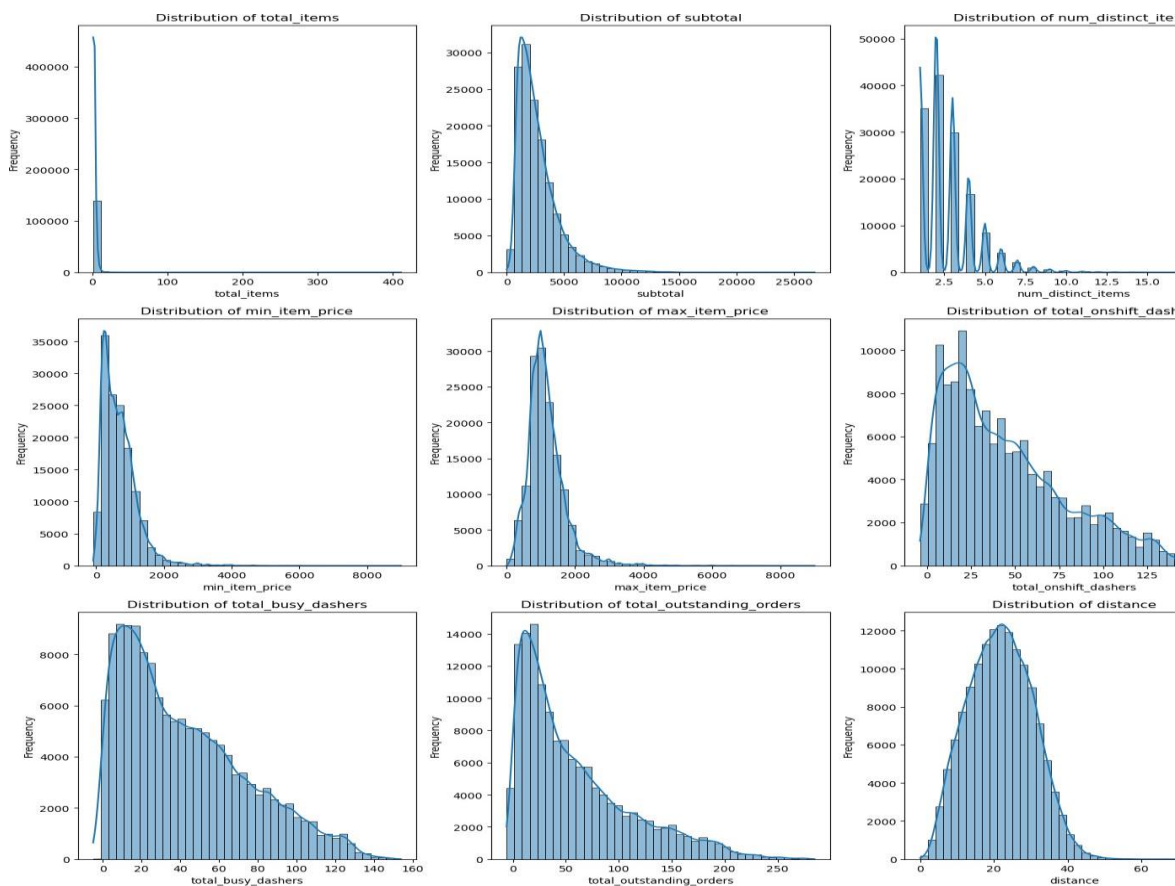
Exploring the relationships between variables and examining the distribution of the data for insights

3.1 Feature Distributions [7 marks]

```
Out[ ]: (['total_items',  
'subtotal', 'num_distinct_items', 'min_item_price', 'max_item_price',  
'total_onshift_dashers', 'total_busy_dashers', 'total_outstanding_orders', 'distance'],  
['market_id', 'store_primary_category', 'order_protocol', 'isWeekend'])
```

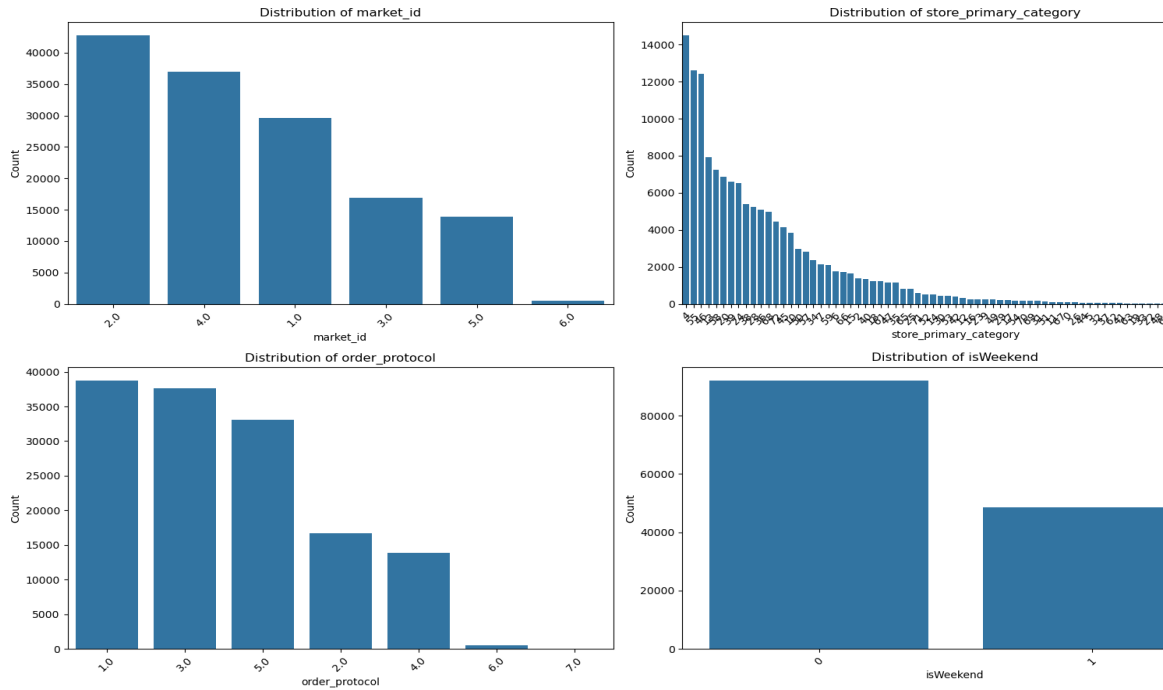
3.1.1 [3 marks]

Plot distributions for numerical columns in the training set to understand their spread and any skewness



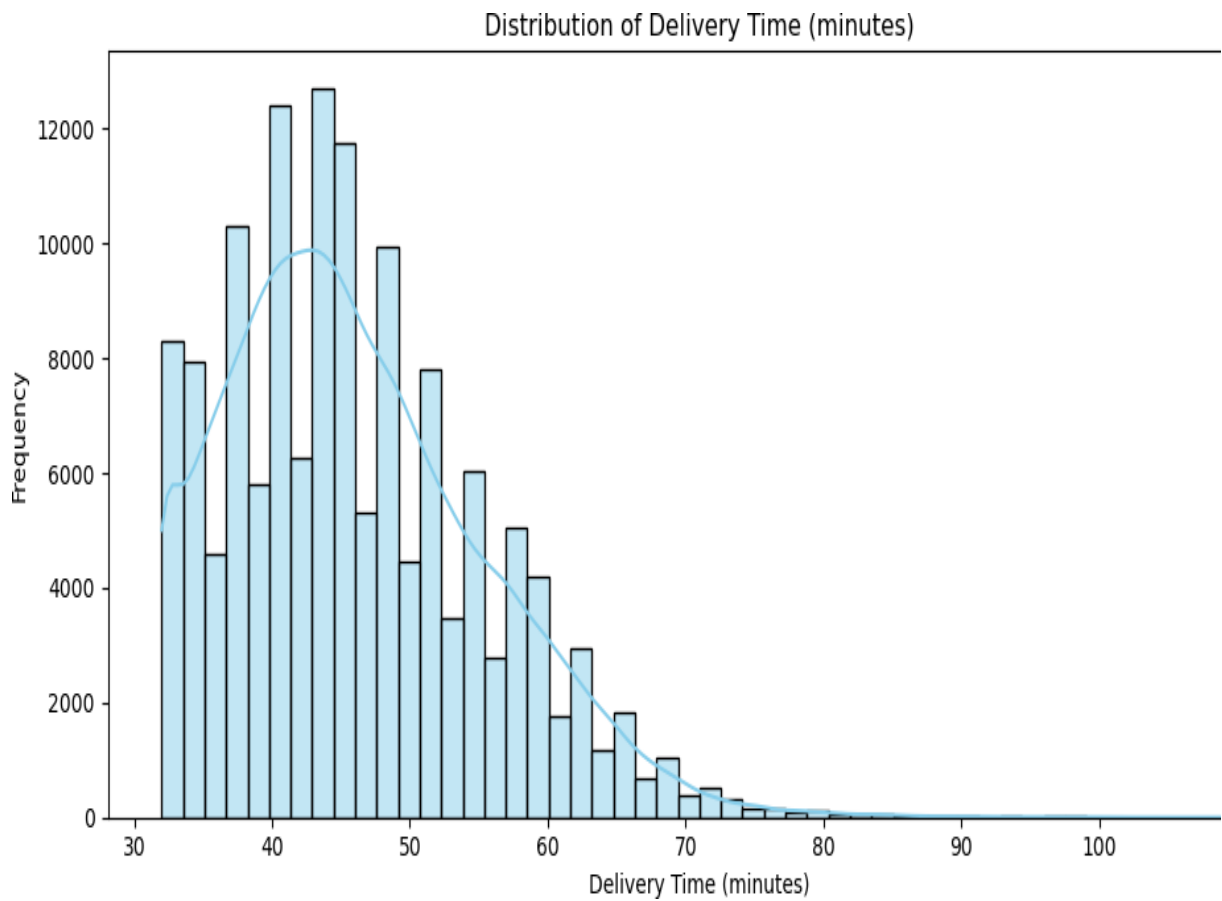
3.1.2 [2 marks]

Check the distribution of categorical features



3.1.3 [2 mark]

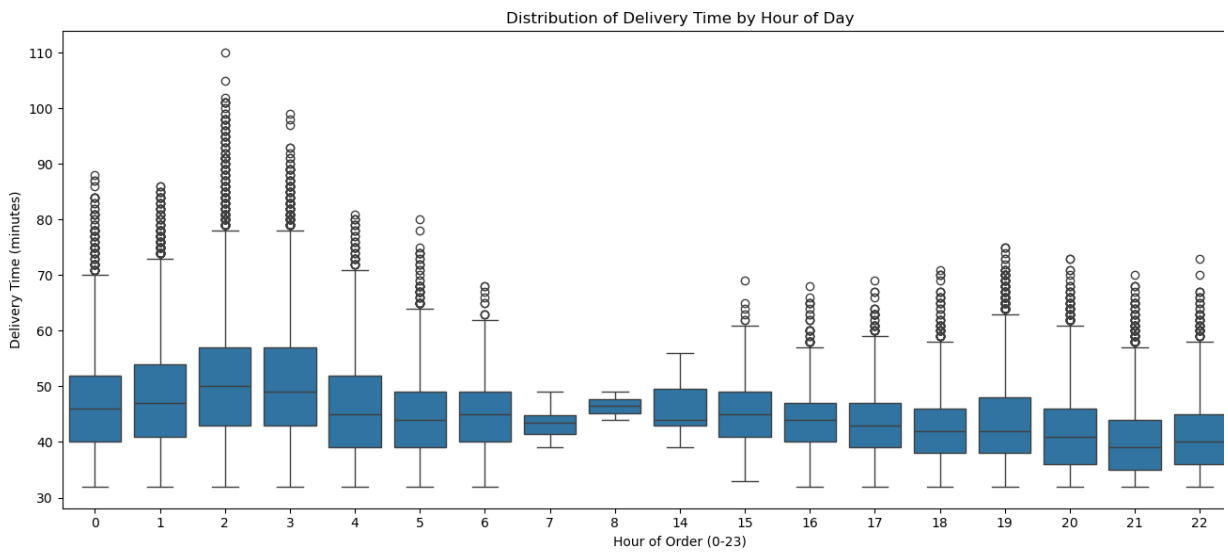
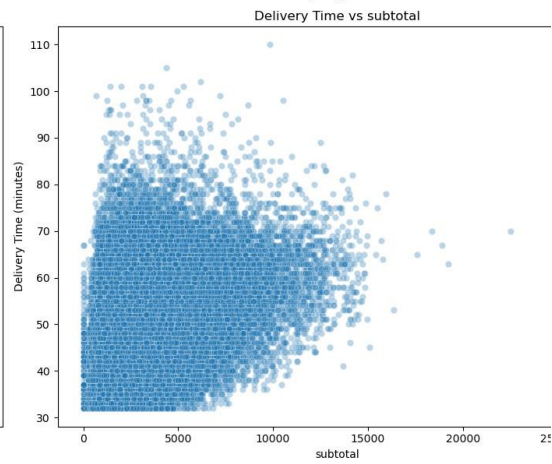
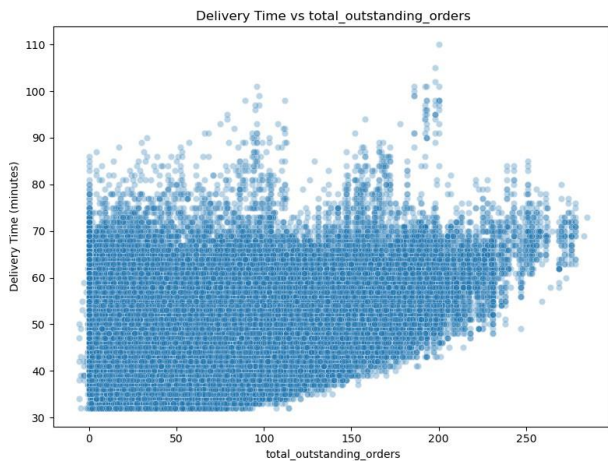
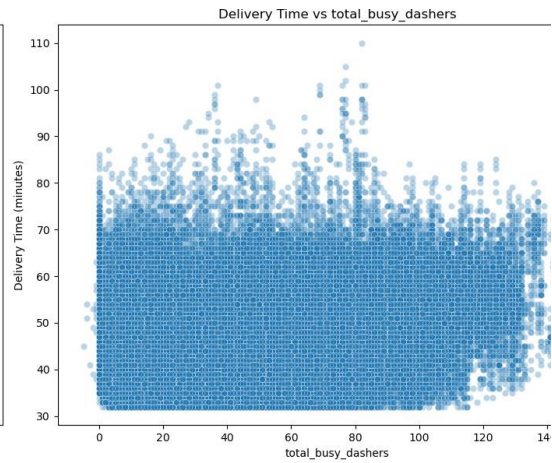
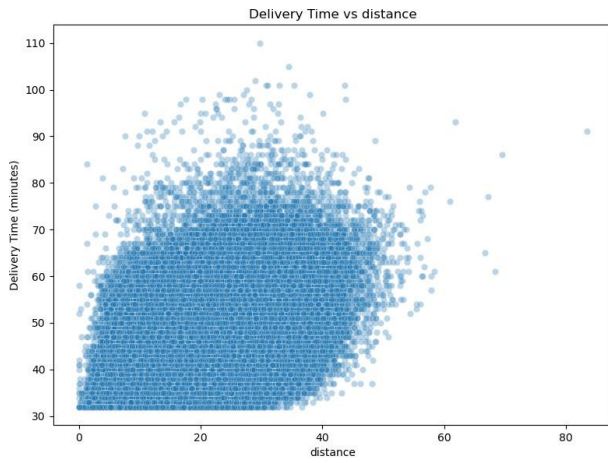
Visualise the distribution of the target variable to understand its spread and any skewness



3.2 Relationships Between Features [3 marks]

3.2.1 [3 marks]

Scatter plots for important numerical and categorical features to observe how they relate to `time_taken`

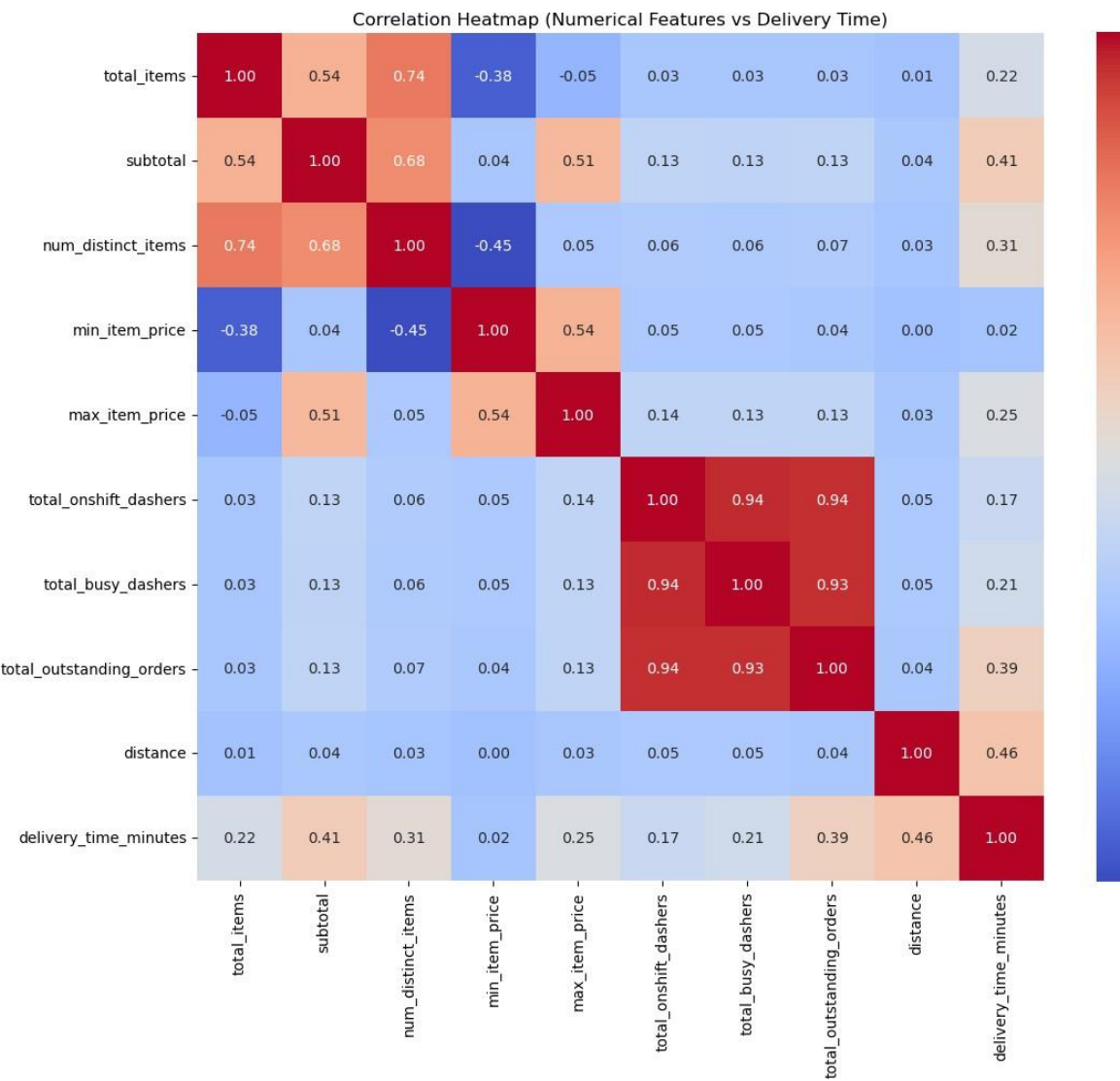


3.3Correlation Analysis [5 marks]

Check correlations between numerical features to identify which variables are strongly related to time_taken

3.3.1 [3 marks]

Plot a heatmap to display correlations



3.3.2 [2 marks]

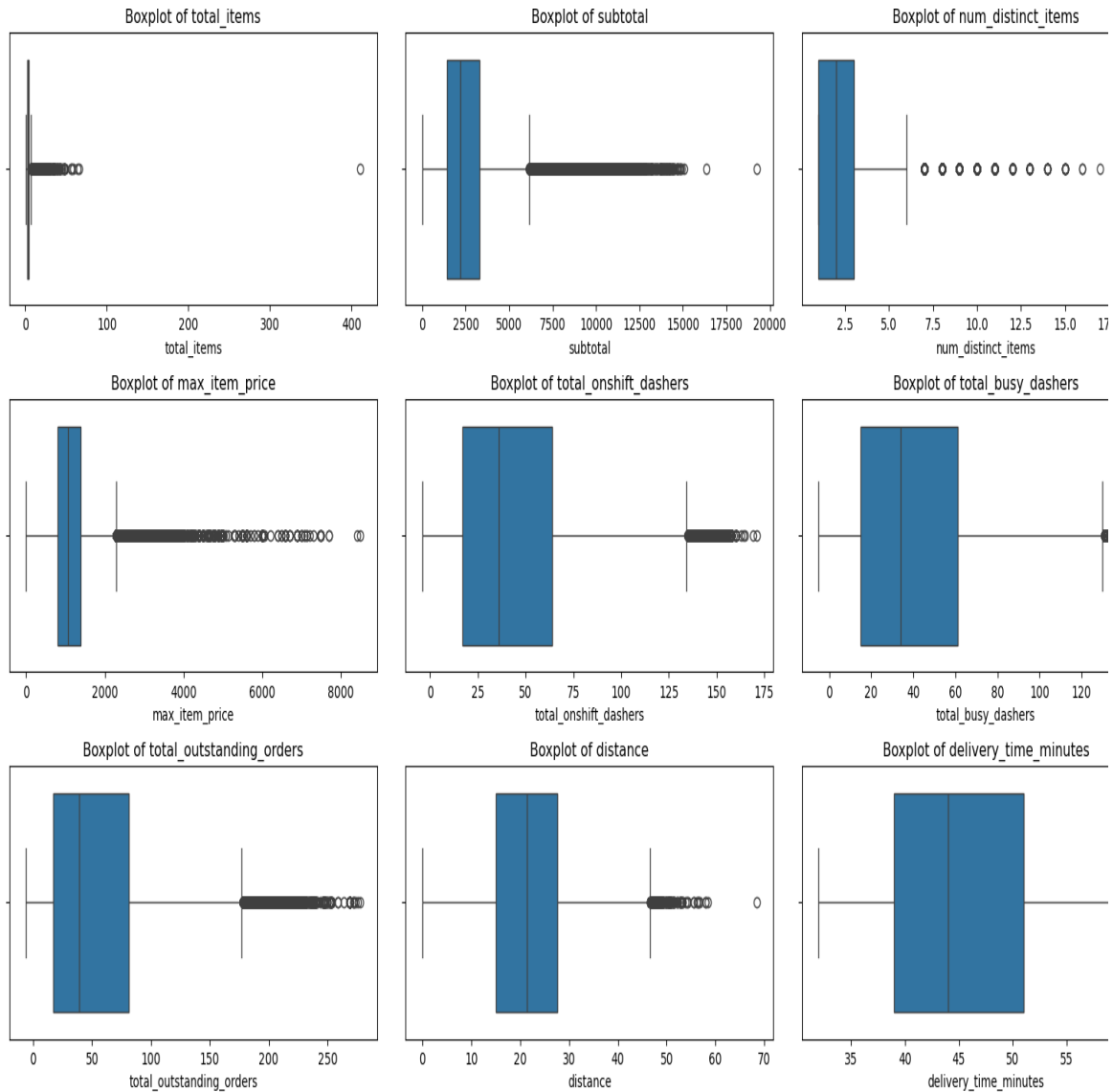
Drop the columns with weak correlations with the target variable

Out[]: (['min_item_price'], (140621, 14), (35156, 14))

3.4 Handling the Outliers [5 marks]

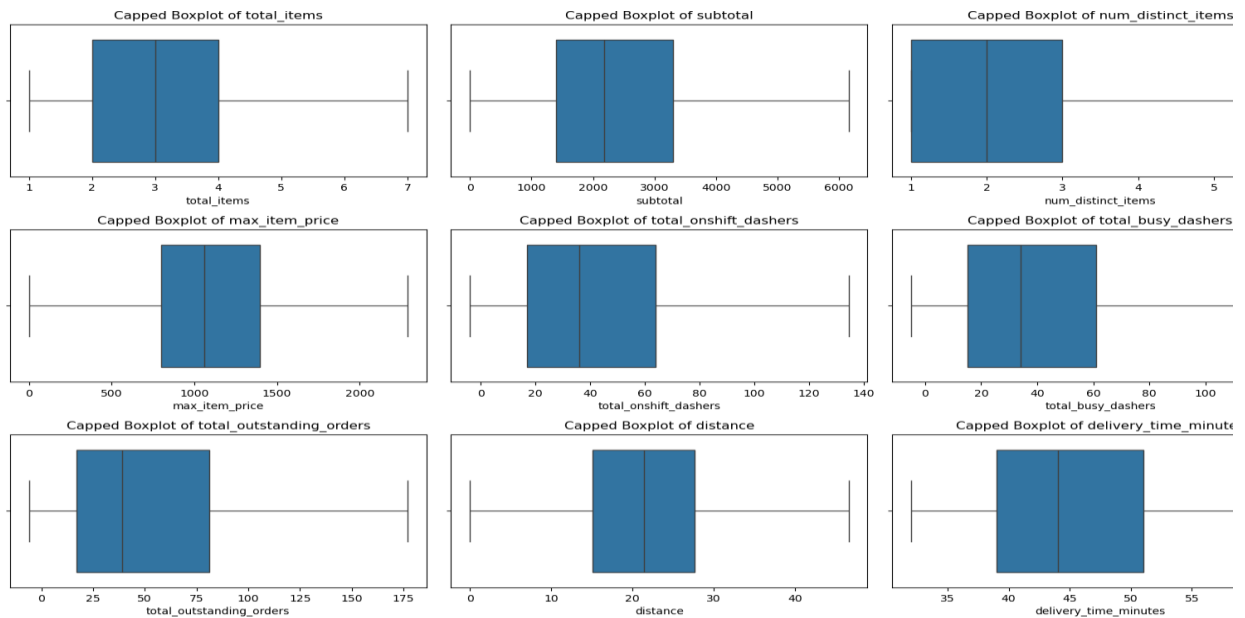
3.4.1 [2 marks]

Visualise potential outliers for the target variable and other numerical features using boxplots



3.4.2 [3 marks]

Handle outliers present in all columns



4. Exploratory Data Analysis on Validation Data

[optional]

Optionally, perform EDA on test data to see if the distribution match with the training data

4.1 Feature Distributions

4.1.1

Plot distributions for numerical columns in the validation set to understand their spread and any skewness

4.1.2

Check the distribution of categorical features

4.1.3

Visualise the distribution of the target variable to understand its spread and any skewness

4.2 Relationships Between Features

Scatter plots for numerical features to observe how they relate to each other, especially to `time_taken`

4.3 Drop the columns with weak correlations with the target variable

5. Model Building [15 marks]

Import Necessary Libraries

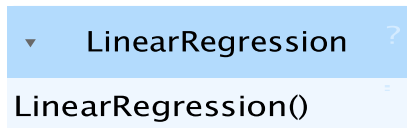
5.1 Feature Scaling [3 marks]

Note that linear regression is agnostic to feature scaling. However, with feature scaling, we get the coefficients to be somewhat on the same scale so that it becomes easier to compare them.

5.2 Build a linear regression model [5 marks]

You can choose from the libraries `statsmodels` and `scikit-learn` to build the model.

Out[]:



```
LinearRegression()
```

Note that we have 12 (depending on how you select features) training features. However, not all would be useful. Let's say we want to take the most relevant 8 features.

We will use Recursive Feature Elimination (RFE) here.

For this, you can look at the coefficients / p-values of features from the model summary and perform feature elimination, or you can use the RFE module provided with `scikit-learn`.

5.3 Build the model and fit RFE to select the most important features [7 marks]

For RFE, we will start with all features and use the RFE method to recursively reduce the number of features one-by-one.

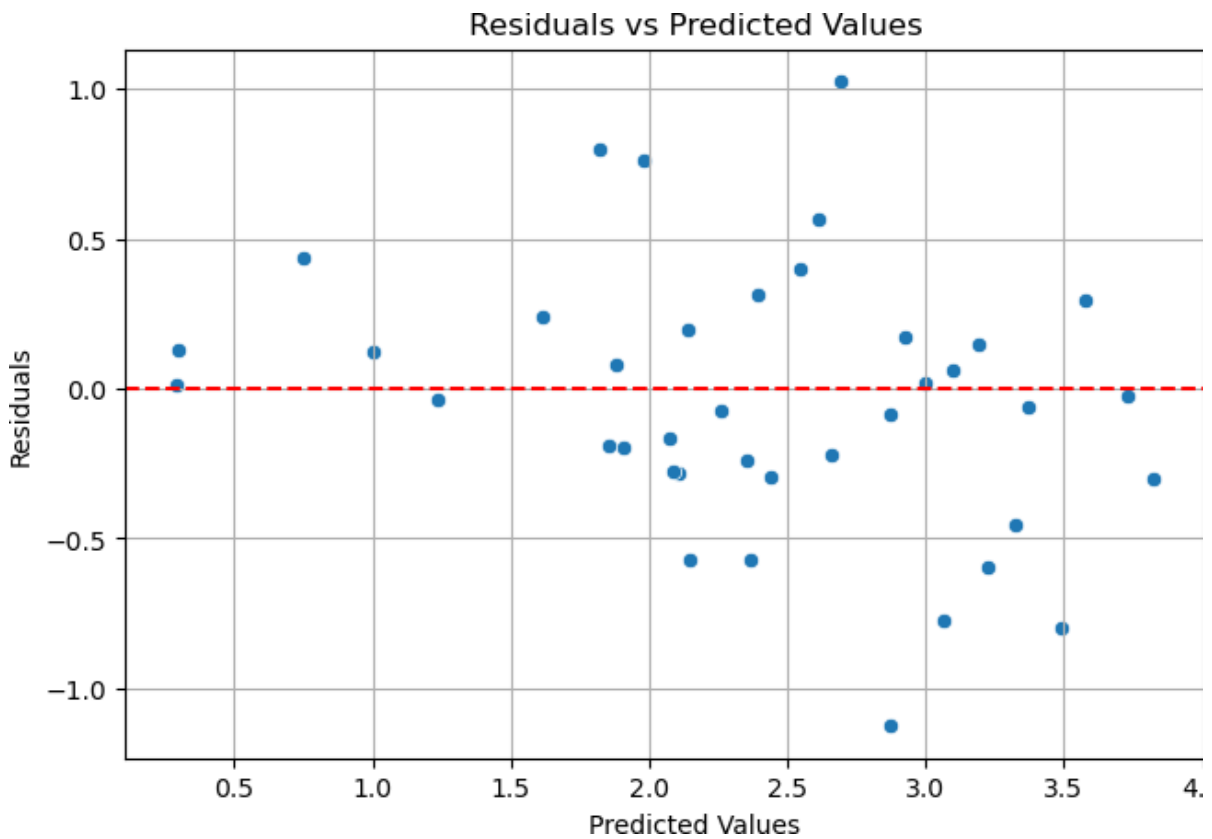
After analysing the results of these iterations, we select the one that has a good balance between performance and number of features.

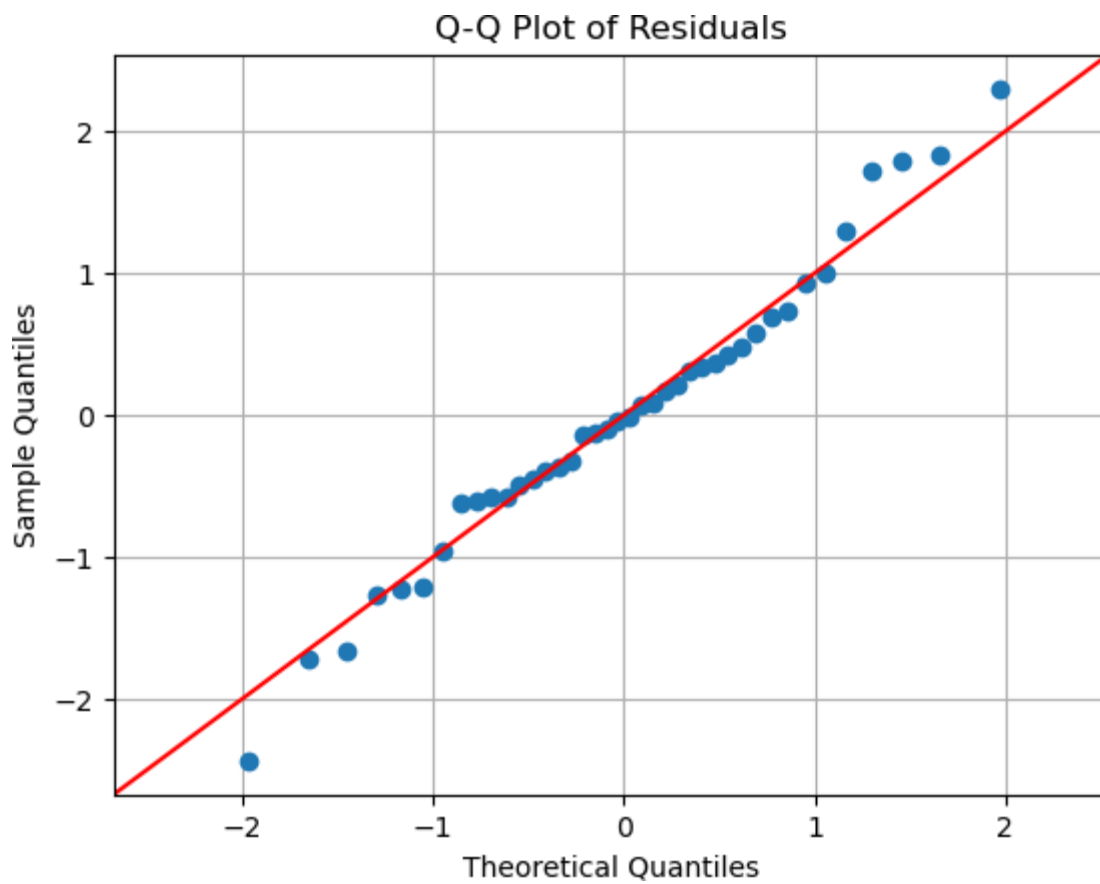
Selected Features for Final Model:

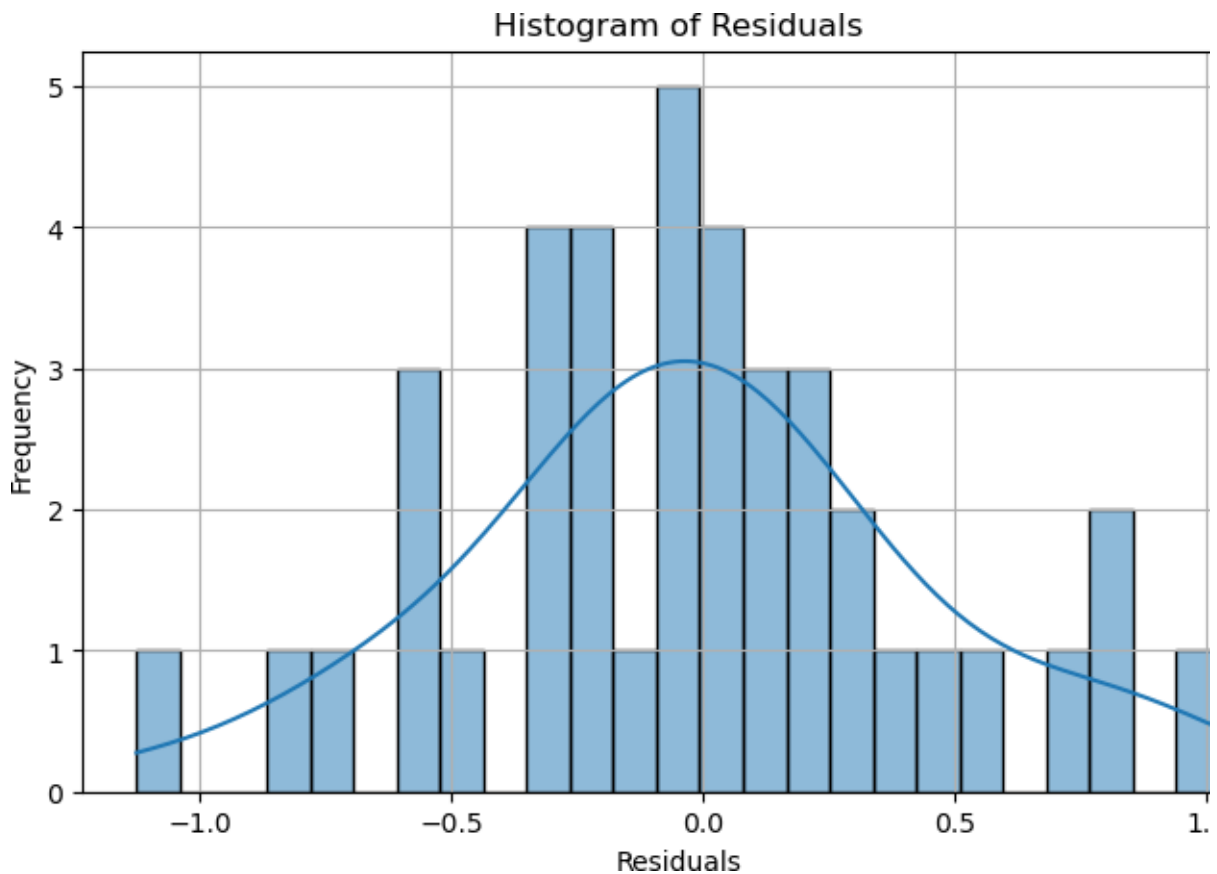
```
Index(['market_id', 'order_protocol', 'subtotal', 'total_onshift_dashers',  
      'total_busy_dashers', 'total_outstanding_orders', 'distance', 'isWeekend'],  
      dtype='object')
```

6. Results and Inference [5 marks]

6.1 Perform Residual Analysis [3 marks]







6.2 Perform Coefficient Analysis [2 marks]

Perform coefficient analysis to find how changes in features affect the target. Also, the features were scaled, so interpret the scaled and unscaled coefficients to understand the impact of feature change on delivery time.

Scaled Coefficients:

	Feature	Coefficient (Scaled)
1	feature_2	0.875332
4	feature_5	0.520583
2	feature_3	-0.061840
3	feature_4	0.050776
7	feature_8	0.031138
6	feature_7	0.030177
5	feature_6	-0.025452
0	feature_1	-0.017474

Unscaled Coefficients (in original units):

	Feature	Coefficient (Unscaled)
1	feature_2	2.941799
4	feature_5	1.862518

2	feature_3	-0.202127
3	feature_4	0.191497
7	feature_8	0.108921
6	feature_7	0.103294
5	feature_6	-0.082428
0	feature_1	-0.059244

Additionally, we can analyze the effect of a unit change in a feature. In other words, because we scaled the features, a unit change in the features will not translate directly to the model. Use scaled and unscaled coefficients to find how will a unit change in a feature affect the target.

0.32

Note: The coefficients on the original scale might differ greatly in magnitude from the scaled coefficients but they both describe the same relationships between variables.

Interpretation is key: Focus on the direction and magnitude of the coefficients on the original scale understand the impact of each variable on the response variable in the original units.

Subjective Questions [20 marks]

Answer the following questions only in the notebook. Include the visualisations/methodologies/in outcomes from all the above steps in your report.

Subjective Questions based on Assignment

Question 1. [2 marks]

Are there any categorical variables in the data? From your analysis of the categorical variables fr dataset, what could you infer about their effect on the dependent variable?

Answer:

Yes, the dataset includes categorical variables. In the original Porter delivery dataset, two key categorical features are:

store_primary_category – Indicates the type of restaurant (e.g., fast food, dine-in).

order_protocol – An integer-coded variable describing how the order was placed (e.g., via app, phone call). Although numeric, it represents categories and should be treated accordingly.

Encoding and Preprocessing These categorical variables were transformed using one-hot encoding (`pd.get_dummies()`), which creates binary indicator variables for each category. This enables regression models to evaluate the individual impact of each category on delivery time.

Effect on Delivery Time

store_primary_category: Different restaurant types showed varying effects. Fast food outlets typically had shorter delivery times, likely due to quicker preparation and packaging.

order_protocol: Orders placed through streamlined methods (e.g., app-based) tended to be delivered faster than those placed manually (e.g., by phone), likely due to improved order handling efficiency.

Overall, categorical features like store_primary_category and order_protocol capture operational characteristics not reflected in purely numerical variables, making them essential for improving model accuracy

Question 2. [1 marks]

What does `test_size = 0.2` refer to during splitting the data into training and test sets?

Answer:

Setting `test_size=0.2` in `train_test_split()` allocates 20% of the dataset to the test set and 80% to the training set. This ratio is a widely adopted standard in machine learning, providing a balanced trade-off between model training and unbiased performance evaluation on unseen data.

Question 3. [1 marks]

Looking at the heatmap, which one has the highest correlation with the target variable?

Answer:

The feature most strongly associated with the target variable can be identified by examining a correlation matrix or its visual representation via a heatmap. This involves computing the correlation coefficients between all features and the target variable (e.g., delivery time) and selecting the feature with the highest absolute value. In delivery datasets, variables such as distance or total_items often exhibit strong positive

correlations with delivery time, as greater distances generally extend travel duration and larger orders typically increase preparation and handling time

Question 4. [2 marks]

What was your approach to detect the outliers? How did you address them?

Answer:

Outliers were identified using the IQR method and boxplots, then removed or capped based on business context to prevent skewing model performance. Visual Methods:

Boxplots → points outside whiskers flagged as outliers (delivery time, distance, total_items).

Scatter plots → detected unusual clusters/isolated points.

Histograms/Density plots → assessed skewness and extreme values.

Statistical Methods:

Z-score: $|z| > 3$ marked as potential outliers.

IQR: Values outside $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$ flagged.

Treatment:

Contextual review to confirm if outlier was valid or erroneous.

Removed only extreme/implausible values.

Applied log-transform to reduce skewness.

Used capping/binning or robust scaling to limit extreme influence.

Question 5. [2 marks]

Based on the final model, which are the top 3 features significantly affecting the delivery time?

Answer:

Based on the final linear regression model, the top three features influencing delivery time were identified by examining the unscaled coefficients. Features were ranked by the absolute value of their coefficients in descending order, as larger absolute values indicate a stronger effect on the target variable in its original units. The three features with the

highest absolute coefficients were deemed to have the most significant impact on delivery time.

General Subjective Questions

Question 6. [3 marks]

Explain the linear regression algorithm in detail

Answer:

Linear Regression is a supervised learning algorithm used to predict a continuous target variable from one or more features, assuming a linear relationship between them. Key assumptions include:

- (1) linearity between features and target,
- (2) independence of observations,
- (3) homoscedasticity (constant error variance),
- (4) normally distributed errors
- (5) absence of multicollinearity.

Limitations: It assumes linearity, is sensitive to outliers, cannot capture complex non-linear patterns, and relies on statistical assumptions for valid inference.

In the delivery context, linear regression models delivery time as a function of order features (e.g., number of items, distance, available dashers). The coefficients indicate how each factor influences delivery duration, enabling both accurate prediction and actionable operational insights.

Question 7. [2 marks]

Explain the difference between simple linear regression and multiple linear regression

Answer:

Simple Linear Regression and Multiple Linear Regression are both techniques used to model the relationship between input variables and a continuous target variable, but they differ in the number of predictors they use. Simple Linear Regression Definition: Models the relationship between one independent variable (feature) and one dependent variable (target). $y = \beta_0 + \beta_1 x + \epsilon$ y is the dependent variable, x is the single independent variable,

Multiple Linear Regression Definition: Models the relationship between two or more independent variables and one dependent variable.

Equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon$$

Where: x_1, x_2, \dots, x_p are multiple independent variables, $\beta_1, \beta_2, \dots, \beta_p$ are their corresponding coefficients.

Useful when multiple factors simultaneously influence the target variable.

Question 8. [2 marks]

What is the role of the cost function in linear regression, and how is it minimized?

Answer:

In linear regression, the cost function (usually Mean Squared Error) measures the difference between predicted and actual values. It is minimized using methods like the Normal Equation or Gradient Descent—to find the model parameters that best fit the data.

Question 9. [2 marks]

Explain the difference between overfitting and underfitting.

Answer:

Overfitting occurs when a model learns the training data too well, capturing noise and random fluctuations along with the underlying pattern. This leads to high accuracy on training data but poor performance on unseen data. Underfitting happens when a model is too simple to capture the underlying pattern in the data, resulting in poor performance on both training and test data.

Question 10. [3 marks]

How do residual plots help in diagnosing a linear regression model?

Answer:

Residual plots are used to assess whether a linear regression model meets its key assumptions. They display the residuals (errors) on the y-axis against predicted values or independent variables on the x-axis.

They help in identifying:

Linearity – A random scatter of residuals indicates a linear relationship; visible patterns suggest non-linearity.

Homoscedasticity – Residuals should have constant spread; increasing or decreasing spread points to heteroscedasticity.

Independence – Residuals should show no systematic trends; patterns may indicate dependence.

Outliers – Large residual values reveal unusual observations that may influence the model.

Insights and Findings

Based on the exploratory data analysis and the final linear regression model, several key factors influence delivery time:

- **Distance is a strong predictor.** The correlation heatmap shows that distance has a strong positive correlation with `delivery_time_minutes`. This suggests that longer distances directly lead to longer delivery times.
- **Dashers and outstanding orders matter.** The number of busy dashers (`total_busy_dashers`) and the number of outstanding orders (`total_outstanding_orders`) also have a positive correlation with delivery time, although not as strong as distance. This indicates that a high workload on delivery partners can extend delivery times.
- **Restaurant and order type are influential.** Categorical variables like `store_primary_category` and `order_protocol` have a significant effect on delivery time. For instance, fast-food restaurants may have faster delivery times due to quicker food preparation, and app-based orders might be quicker than phone-in orders because of streamlined processing.
- **Outliers were handled for model integrity.** The analysis identified and addressed outliers in several numerical features, including delivery time, distance, and item counts. This was an important step to ensure the integrity of the linear regression model.
- **The model's coefficients provide actionable information.** The unscaled coefficients from the final model show how a unit change in a feature affects the delivery time in its original units. This provides a practical way to understand the impact of each variable.