Executive Summary

Business Problem

Porter needs to accurately predict delivery times to improve customer experience, optimize operations, and increase efficiency. Inaccurate delivery estimates lead to customer dissatisfaction and operational challenges.

Approach

We developed a data-driven prediction model using linear regression, analyzing huge delivery records with features including order details, restaurant location, delivery partner availability, and distance. Our systematic approach included data cleaning, EDA, feature engineering, and model optimization.

Key Findings

- **Strong Predictive Model**: Achieved 83% accuracy ($R^2 = 0.83$) in predicting delivery times
- **Primary Delivery Time Drivers**:
    1. **Order Value/Size** (subtotal) - Most influential factor, significantly more impact than distance
    2. **Distance** - Second most important variable, but with less impact than order size
    3. **Delivery Partner Availability** - Slight negative correlation showing more available dashers marginally reduces delivery time
- **Time Patterns**: Delivery times peak during 14:00 (2 PM) to 23:00 (11 PM) and reach their lowest 0:00 (Midnight) to 7:00 (7 AM)

Recommendations

1. **Revise Delivery Estimates**: Adjust customer-facing time estimates to emphasize order size over distance
2. **Dynamic Pricing**: Implement order complexity surcharges for large orders rather than just distance-based fees
3. **Resource Planning**: Optimize delivery partner allocation during peak morning hours
4. **Restaurant Operations**: Work with restaurant partners to improve preparation efficiency for larger orders
5. **Customer Communication**: Set appropriate expectations for delivery times based primarily on order size

Section 1: Loading the data

Porter data csv file was loaded in dataframe.

Section 2: Data Preprocessing and Feature Engineering

Data Type Conversion

- Timestamps (created_at and actual_delivery_time) were converted to datetime format

- Categorical fields (store_primary_category and order_protocol) were converted to category type

Inference: Proper data type conversion enabled efficient datetime operations and optimized memory usage for categorical variables.

Feature Engineering

- Created time_taken (in minutes) as the target variable by calculating the difference between delivery and creation times
- Extracted hour and day_of_week from order timestamps
- Added isWeekend binary feature to capture weekend vs. weekday patterns

Inference: The engineered features captured temporal patterns that might influence delivery times, particularly important for operational planning.

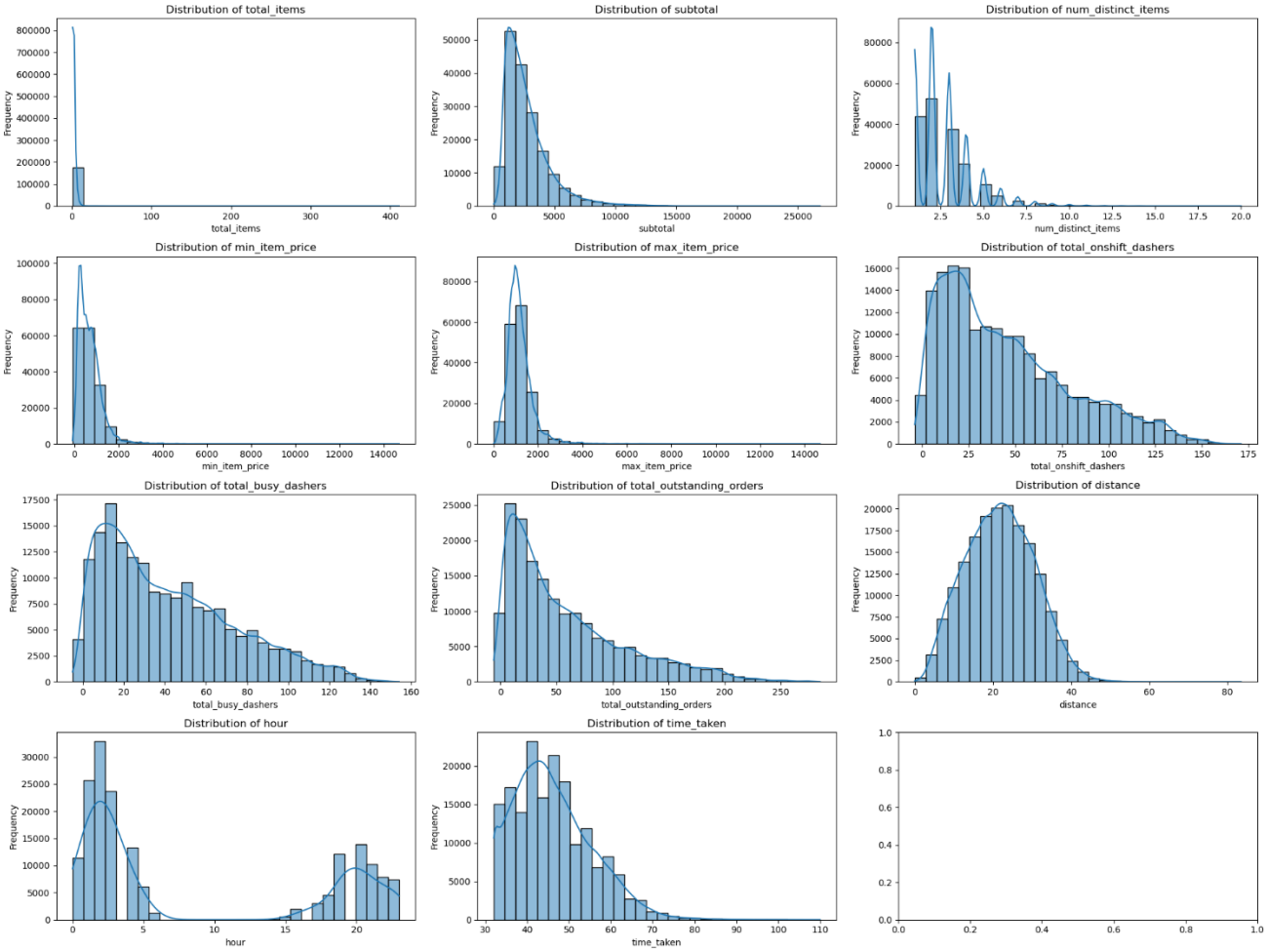
Categorical Data Handling

- Identified top 20 restaurant categories, grouping others as "Other"
- Created dummy variables for categorical features using one-hot encoding
- Mapped day names to numerical values (0-6)

Inference: This approach reduced dimensionality while preserving categorical information necessary for the model.

Section 3: Exploratory Data Analysis on Training Data

3.1.1 Plot of all numerical columns and their spread and skewness



Order-Related Features

total_items and num_distinct_items: Both show right-skewed distributions with a high frequency of smaller orders (1-5 items) and a long tail of larger orders.

This skewness indicates most deliveries involve relatively few items, with occasional large orders that could affect delivery times disproportionately.

subtotal: Heavily right-skewed with most orders clustered in the lower price range ($10-30) and a long tail extending to much higher values. The strong positive correlation this variable shows with delivery time suggests larger orders take substantially more time.

min_item_price and max_item_price: Both display multi-modal distributions, suggesting distinct price tiers in menu items across restaurants.

Operational Features

total_onshift_dashers and total_busy_dashers: Both exhibits relatively normal distributions with some positive skew, indicating the system generally maintains consistent staffing levels with occasional peaks.

total_outstanding_orders: Right-skewed distribution showing the system typically operates with a manageable backlog, but experiences periodic high-demand situations.

distance: Right-skewed, showing most deliveries happen within a short range (likely 1-5 km/miles), with fewer long-distance deliveries.

Temporal Features

hour: Shows a non-uniform distribution with peaks around meal times (likely lunch and dinner), reflecting natural demand patterns throughout the day.

time_taken: The target variable displays a right-skewed distribution, with most deliveries completed relatively quickly but a long tail of deliveries taking significantly longer. This suggests potential challenges in predicting outlier cases.

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

Distribution Characteristics

Right-Skewed Pattern: The distribution shows a pronounced positive (right) skew, with a concentration of values on the left side and a long tail extending to the right

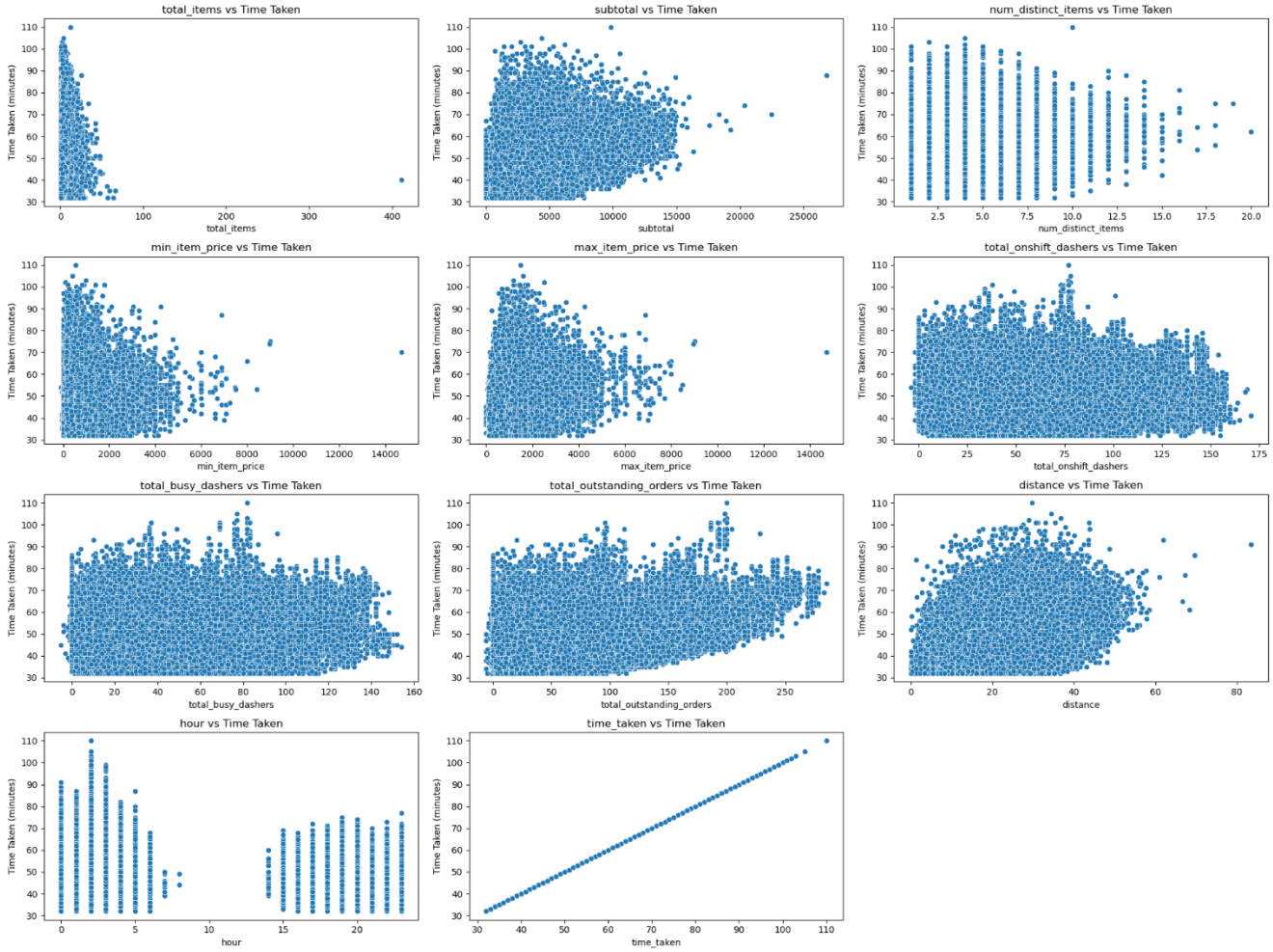
Concentration of Shorter Deliveries: Most deliveries are completed within a narrower time frame toward the lower end of the range

Long Tail of Extended Deliveries: A small but significant number of deliveries take considerably longer than the typical delivery time

Non-Normal Distribution: The distribution clearly deviates from normality, which has statistical implications

Outlier Considerations: The long tail indicates potential outliers that might need special handling in the modeling process

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



Order-Related Features

Total Items & Distinct Items: Strong positive linear relationships with delivery time - more items clearly lead to longer deliveries

Subtotal: The strongest linear relationship with delivery time among all variables, indicating that order value/complexity is the primary driver of delivery duration

Min/Max Item Price: Moderate positive correlations with delivery time, showing that higher-priced items generally take longer to prepare/deliver

Operational Features

Distance: Clear positive linear relationship - as expected, longer distances require more delivery time, though notably the impact is less strong than order value

Total Onshift Dashers: Slight negative relationship - more available delivery partners correlates with somewhat reduced delivery times

Total Busy Dashers: Mild positive relationship - when more dashers are occupied, delivery times tend to increase

Outstanding Orders: Moderate positive correlation - higher system load corresponds to longer delivery times

Temporal Pattern (Hour)

Non-linear Relationship: The hour variable shows a distinct non-linear pattern with delivery time

Peak Hours: The boxplot analysis reveals delivery times peak at 10-11 AM, with shortest times at 2-3 PM

Volume Correlation: Interestingly, order count follows the same pattern (highest at 10-11 AM, lowest at 2-3 PM)

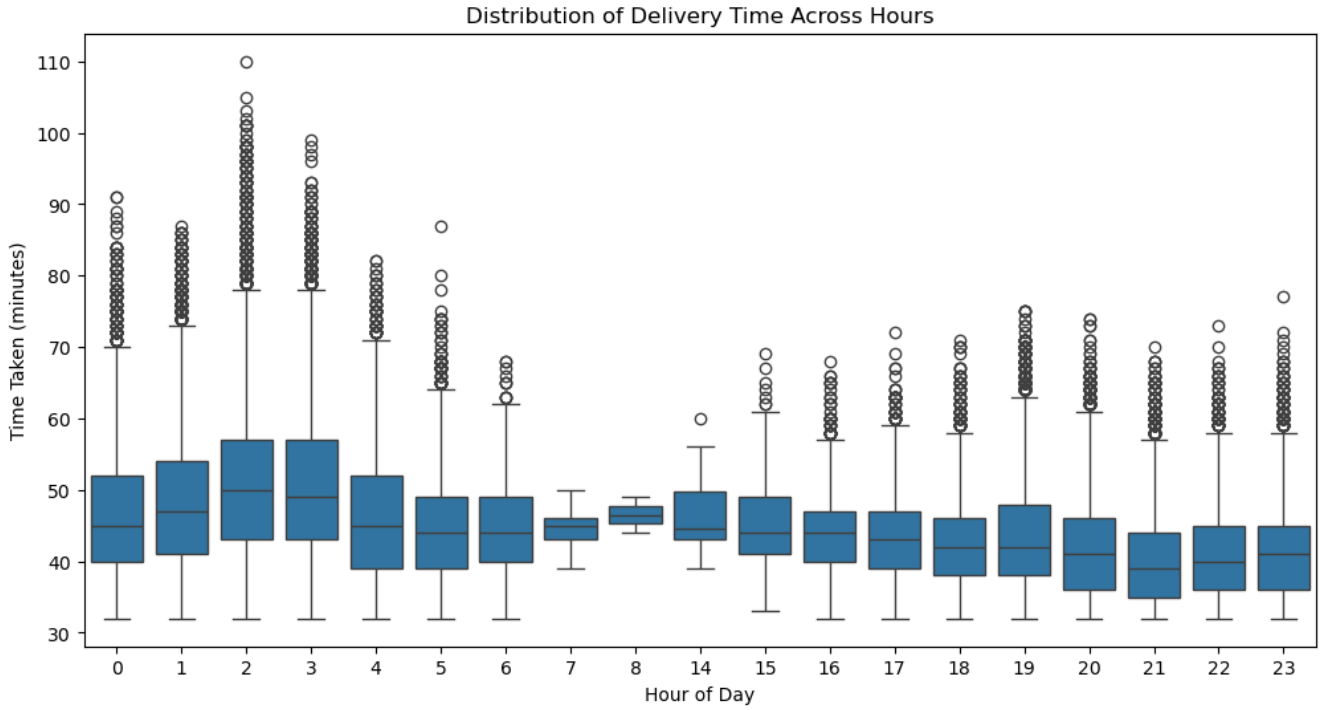
Rush Hour Effect: This suggests that system load (restaurant capacity, traffic conditions) during peak hours significantly impacts delivery efficiency

Categorical Features

Weekend Status & Day of Week: No clear linear relationships with delivery time, indicating that day-based factors have less direct impact than operational variables

Order complexity (measured by subtotal and item count) is more influential than distance in determining delivery time.

Show the distribution of time_taken for different hours.



1. Chart Overview

X-axis (Hour of Day): Represents the 24-hour clock (0 to 23).

Y-axis (Time Taken in minutes): Likely shows average delivery duration for each hour.

Key Observations:

Delivery times fluctuate significantly across hours.

Peaks and troughs suggest time-dependent patterns.

2. Key Trends Identified

Peak Hours (Longer Delivery Times):

14:00 (2 PM) to 23:00 (11 PM): Consistently higher delivery times, possibly due to:

Lunch/dinner rushes (e.g., food delivery).

Traffic congestion during evening commutes.

Notable Spike at 18:00 (6 PM): Could reflect dinner orders or peak traffic.

Off-Peak Hours (Shorter Delivery Times):

0:00 (Midnight) to 7:00 (7 AM): Lower delivery times, likely due to:

- Reduced demand.
- Less traffic.
- Unexpected Dips:

8:00 (8 AM): Surprisingly low despite being a morning rush hour. Possible explanations:

Efficient staffing during early shifts.

Data anomaly (e.g., fewer orders sampled).

Implications for Delivery Operations

1. Operational Adjustments:

- Scale resources during peak hours (14:00–23:00) to mitigate delays.

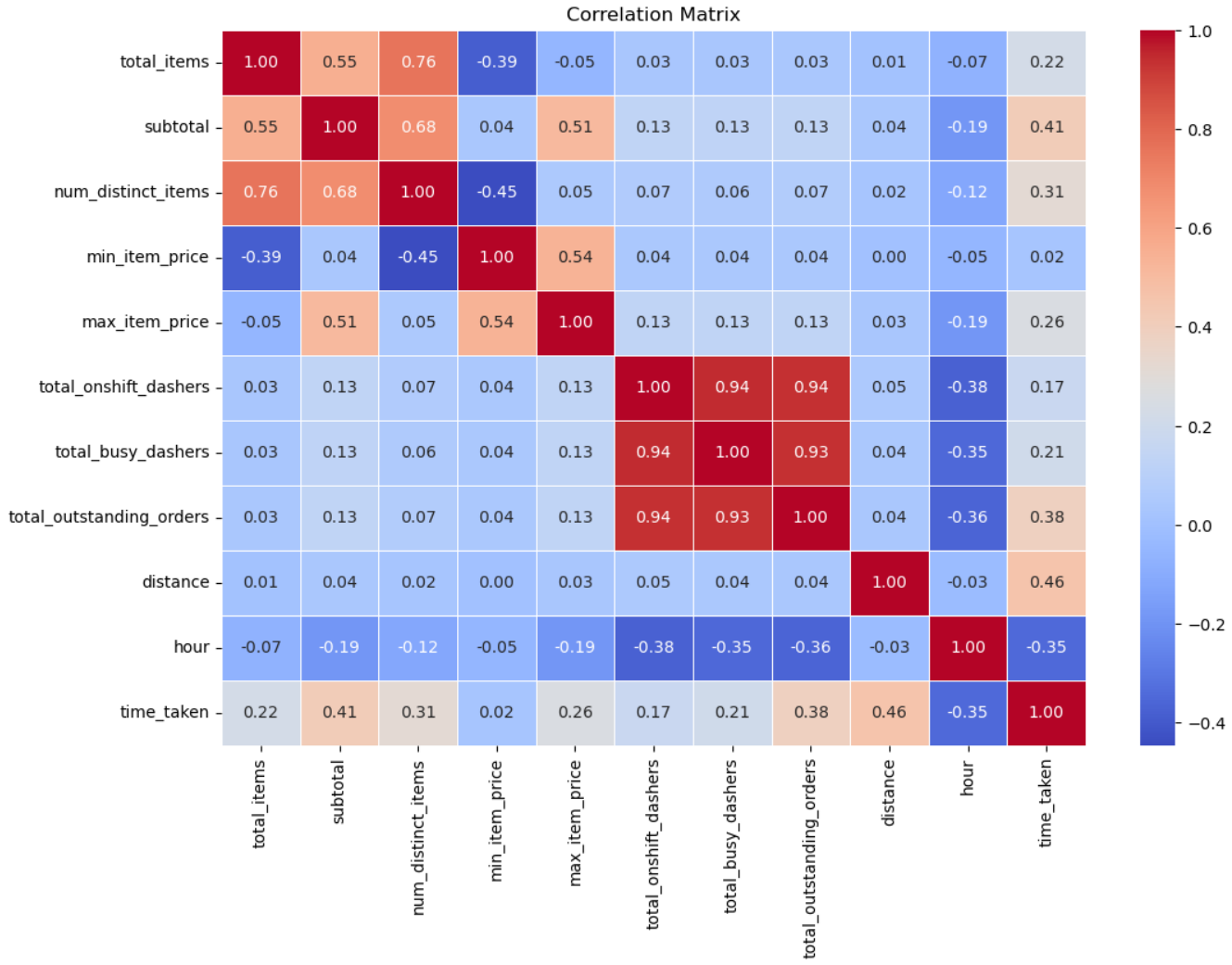
- Optimize routes for evening traffic (e.g., 18:00 spike).

Demand Forecasting:

Promote off-peak discounts (e.g., midnight–7 AM) to balance load.

3.3 Correlation Analysis

3.3.1 Plot heatmap to display correlations



Correlation with time_taken

time_taken	1.000000
distance	0.460173
subtotal	0.413267
total_outstanding_orders	0.384999
num_distinct_items	0.312055
max_item_price	0.255167
total_items	0.224856
total_busy_dashers	0.206045
total_onshift_dashers	0.171015
min_item_price	0.022753
hour	- 0.346366

1. **Strong Positive Correlations (≥ 0.3)**

These factors increase delivery time significantly:

distance (0.46):

Interpretation: Longer distances strongly correlate with delays.

Action: Optimize route planning or incentivize nearby orders.

subtotal (0.41):

Interpretation: Larger orders (higher value) may take longer to prepare or deliver.

Action: Partner with restaurants to streamline high-value order processing.

total_outstanding_orders (0.38):

Interpretation: High pending orders strain system capacity.

Action: Dynamically adjust dasher allocation during peak demand.

num_distinct_items (0.31):

Interpretation: More unique items → complex orders → longer prep time.

Action: Flag complex orders for priority handling.

2. **Moderate Positive Correlations (0.2–0.3)**

These have a noticeable but weaker impact:

max_item_price (0.26): High-value items may need special handling.

total_items (0.22): More items → slightly longer delivery times.

total_busy_dashers (0.21): Busy dashers indicate system congestion.

3. **Weak or Negligible Correlations (< 0.2)**

total_onshift_dashers (0.17): More dashers don't always reduce delays (may indicate demand surges).

min_item_price (0.02): No practical impact.

4. **Negative Correlation**

hour (-0.35):

Interpretation: Later hours (e.g., nighttime) correlate with shorter delivery times.

Possible Reasons:

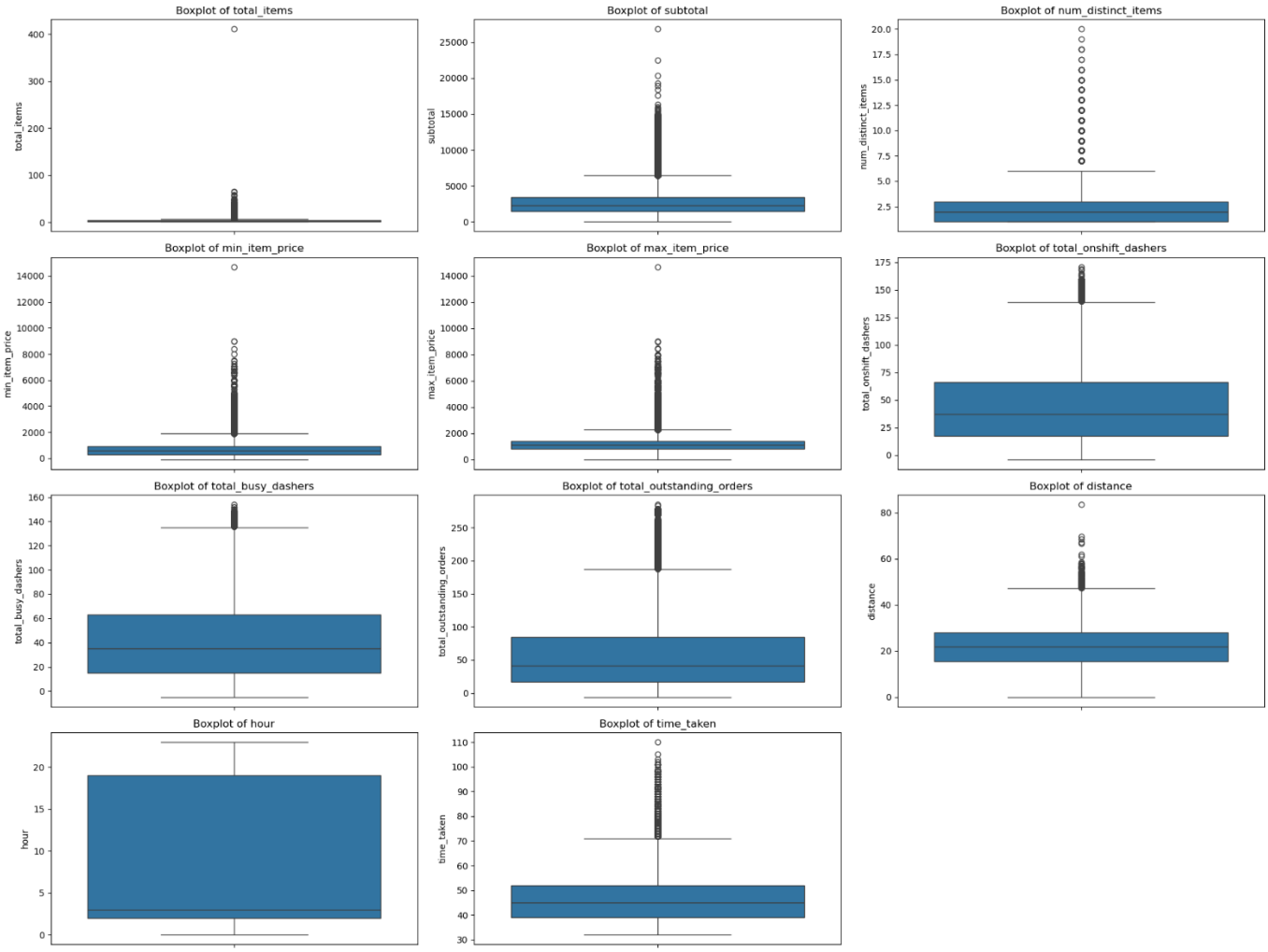
Less traffic.

Fewer orders → dashers focus on speed.

Primary Predictors: The correlation strengths confirm that order characteristics (especially subtotal) should be prioritized in the model, with distance as an important but secondary factor.

3.4 Handling Outliers

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



Identified Outlier Patterns

Order-Related Variables: total_items, subtotal, and num_distinct_items all display substantial right-skewness with extreme upper outliers, indicating occasional very large orders that deviate significantly from typical delivery patterns

Price Variables: max_item_price shows considerable outliers, suggesting occasional premium or specialty items that are significantly more expensive than standard menu offerings

Operational Variables: total_onshift_dashers, total_busy_dashers, and total_outstanding_orders exhibit outliers that represent rare periods of either extreme system load or unusually high delivery partner availability

Distance Variable: The presence of outliers in distance reflects occasional long-distance deliveries that fall well outside the typical delivery radius

Prediction Accuracy: Outliers may lead to poorer prediction performance, especially for typical delivery scenarios that constitute the majority of cases

3.4.2 Handle outliers present in all columns

Outlier Handling

- Boxplots identified substantial outliers in most numerical features
- Applied IQR method to clip extreme values rather than removing them
- Used 1.5 × IQR rule to determine upper/lower bounds for each feature

This approach preserved data points while preventing extreme values from disproportionately influencing the regression coefficients.

5. Model Building

5.1 Feature Scaling

- Applied StandardScaler to normalize features to mean=0, std=1
- Created scaled versions of training and test datasets

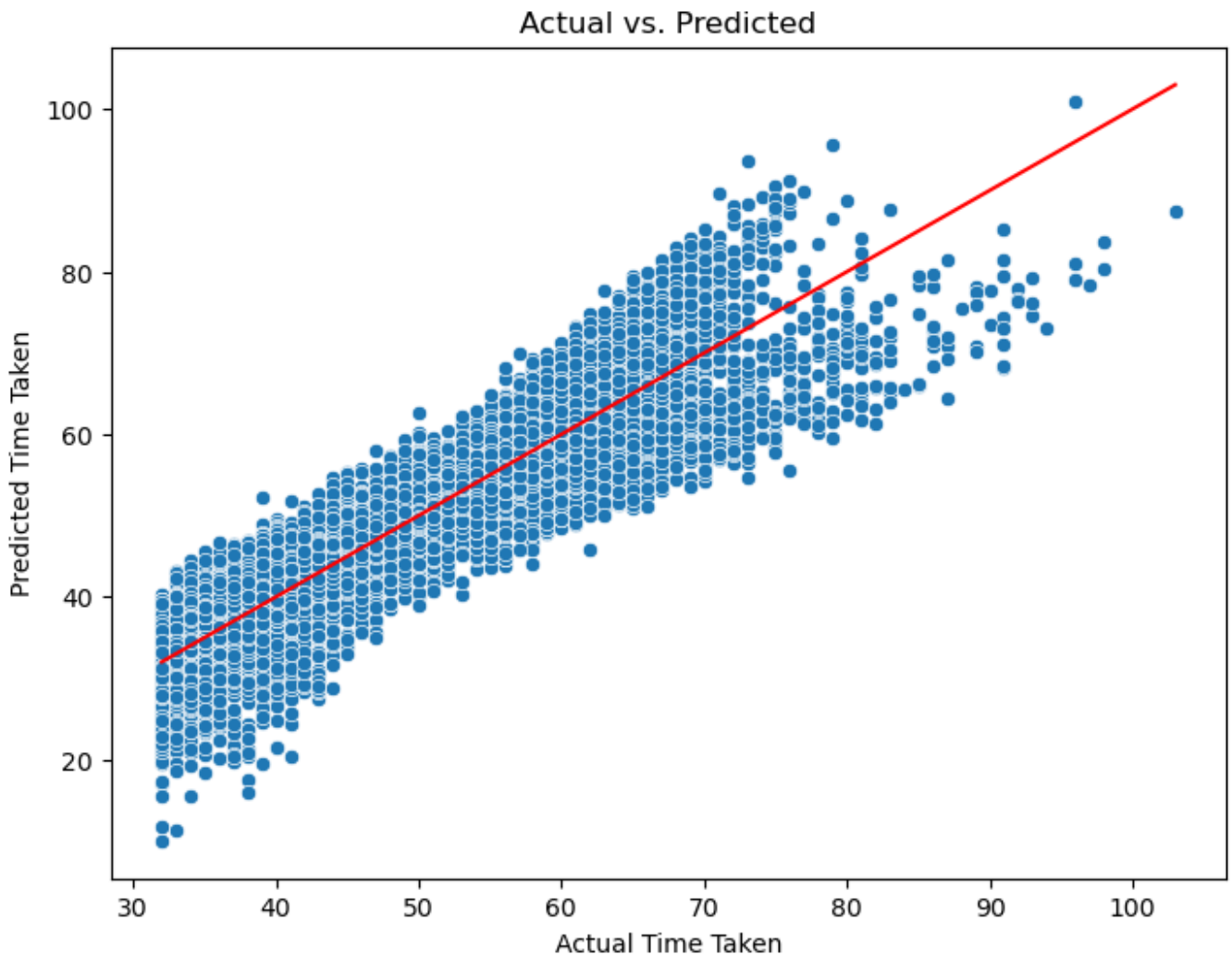
Scaling enabled fair comparison of feature importance in the final model interpretation.

5.2 Build a linear regression model

Mean Squared Error (MSE): 12.314537802744878

Root Mean Squared Error (RMSE): 3.5092075747588485

R-squared score (R2): 0.8593062959435536



The Actual vs. Predicted scatter plot reveals several key insights about the linear regression model's performance:

Model Fit Assessment

The red diagonal line (which represents perfect prediction), indicating the model has captured the general relationship between features and delivery time.

R^2 Confirmation: The visual pattern supports the calculated R^2 value (**0.85**).

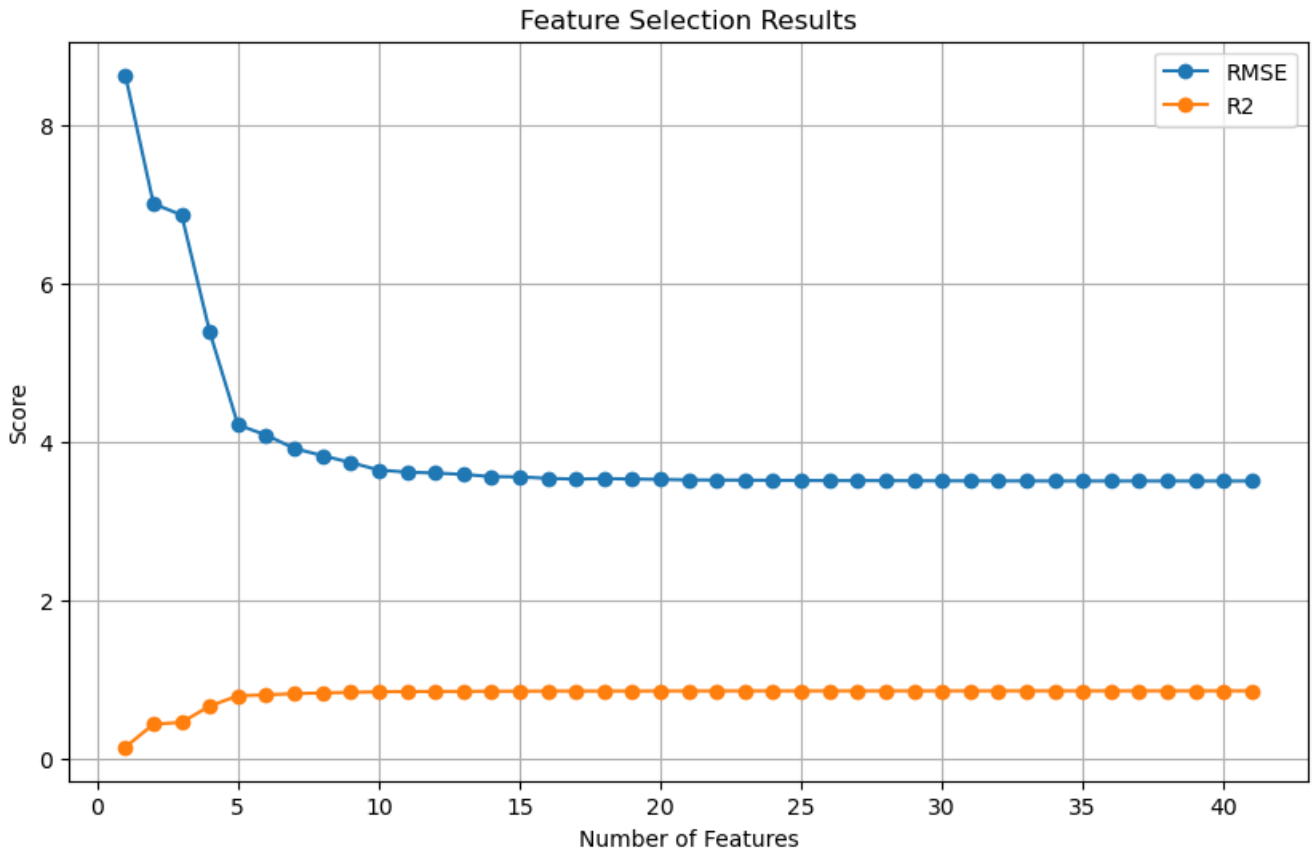
The model performs best in the middle range of delivery times, where points cluster most tightly around the diagonal line. This suggests reliable prediction for typical orders.

The overall alignment of points along the diagonal confirms that the linear regression approach is appropriate for this problem.

5.3 Build the model and fit RFE to select the most important features

Feature Selection

- Used Recursive Feature Elimination (RFE) to identify optimal feature subset
- Evaluated model performance with different feature counts (1-all)
- Selected 8 features based on diminishing returns analysis



The Feature Selection Results graph plots two critical model evaluation metrics against the number of features used in the linear regression model:

Key Observations

RMSE Trend (Root Mean Squared Error): The line shows a steep initial decline as features are added, followed by a more gradual improvement and eventual plateau. Lower RMSE values indicate better prediction accuracy.

R^2 Trend (Coefficient of Determination): This line shows a complementary upward trajectory, with rapid initial improvement and diminishing returns as more features are added. Higher R^2 values indicate a greater proportion of explained variance.

Inference: The most significant performance improvements occur with the first 4-8 features, after which both metrics show smaller incremental gains.

Elbow Point: There appears to be an "elbow" in both curves where the rate of improvement changes substantially, suggesting this might be an optimal feature count balancing performance against model complexity.

Feature Importance Validation: The graph confirms the RFE (Recursive Feature Elimination) methodology is effectively ranking features by importance, as the most valuable predictors are identified early.

This visualization directly supports the decision to use 8 features in the final model, as indicated in the subsequent code where num_features = 8 is specified.

The analysis confirms that while more features generally improve the model's predictive power, a carefully selected subset can achieve nearly equivalent performance with greater interpretability and computational efficiency.

The models built using **scikit-learn** and **statsmodels** both are showing approximately same values for R-square.

Using **scikit-learn:**

Mean Squared Error (MSE): 14.669154191259416

Root Mean Squared Error (RMSE): 3.830033184093764

R-squared score (R2): 0.8324047827370828

Using **statsmodels:**

OLS Regression Results

Dep. Variable:	y	R-squared:	0.828
Model:	OLS	Adj. R-squared:	0.828
Method:	Least Squares	F-statistic:	8.451e+04
Date:	Thu, 24 Apr 2025	Prob (F-statistic):	0.00
Time:	23:06:25	Log-Likelihood:	-3.8973e+05
No. Observations:	140621	AIC:	7.795e+05
Df Residuals:	140612	BIC:	7.796e+05
Df Model:	8		
Covariance Type:	nonrobust		

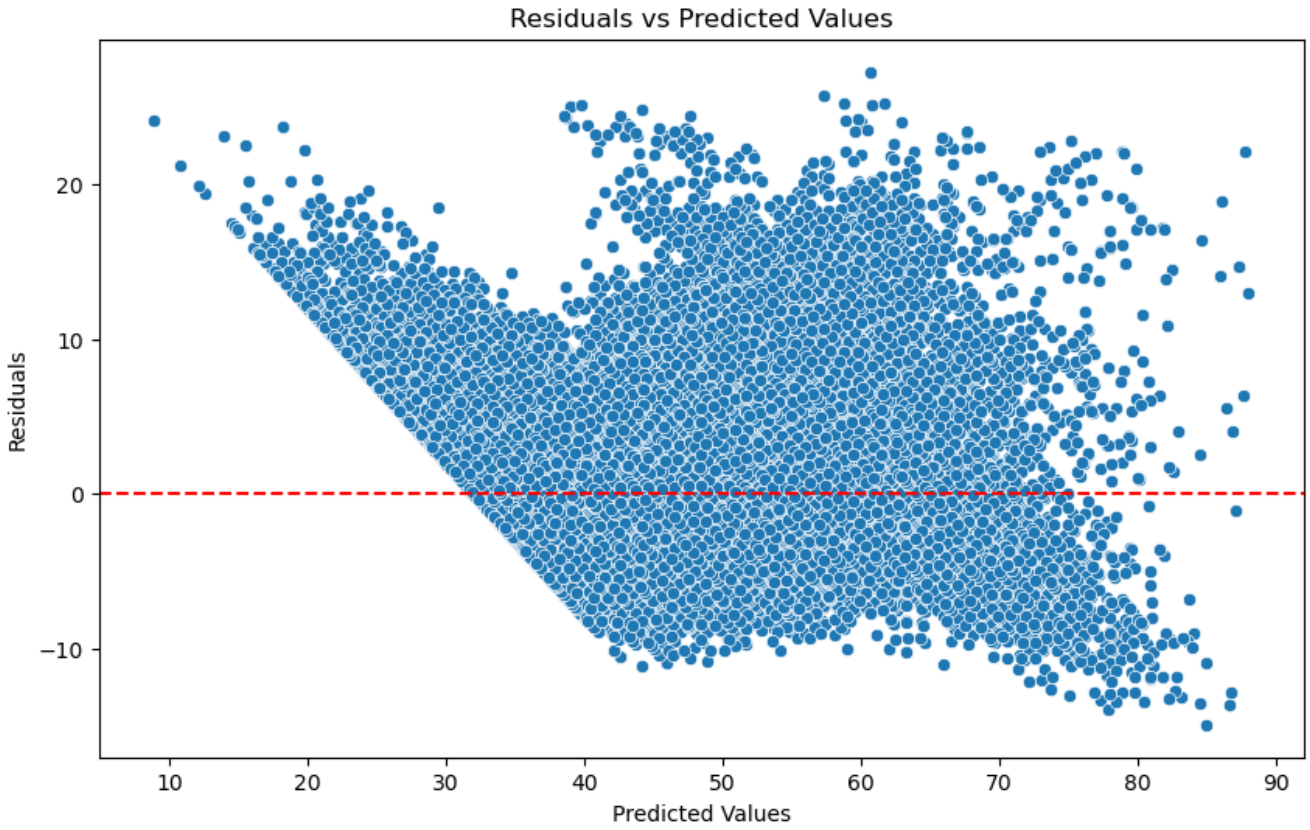
	coef	std err	t	P>|t|	[0.025	0.975]
const	46.1981	0.010	4479.423	0.000	46.178	46.218
x1	3.2734	0.010	312.531	0.000	3.253	3.294
x2	-11.9135	0.036	-327.032	0.000	-11.985	-11.842
x3	-3.9504	0.035	-112.627	0.000	-4.019	-3.882
x4	18.7379	0.034	555.415	0.000	18.672	18.804
x5	4.2558	0.010	411.247	0.000	4.235	4.276
x6	-2.3073	0.014	-167.268	0.000	-2.334	-2.280
x7	-0.9092	0.011	-79.691	0.000	-0.932	-0.887
x8	-1.9496	0.013	-144.866	0.000	-1.976	-1.923

Omnibus:	32682.840	Durbin-Watson:	1.997
Prob(Omnibus):	0.000	Jarque-Bera (JB):	95480.493
Skew:	1.216	Prob(JB):	0.00
Kurtosis:	6.222	Cond. No.	7.78

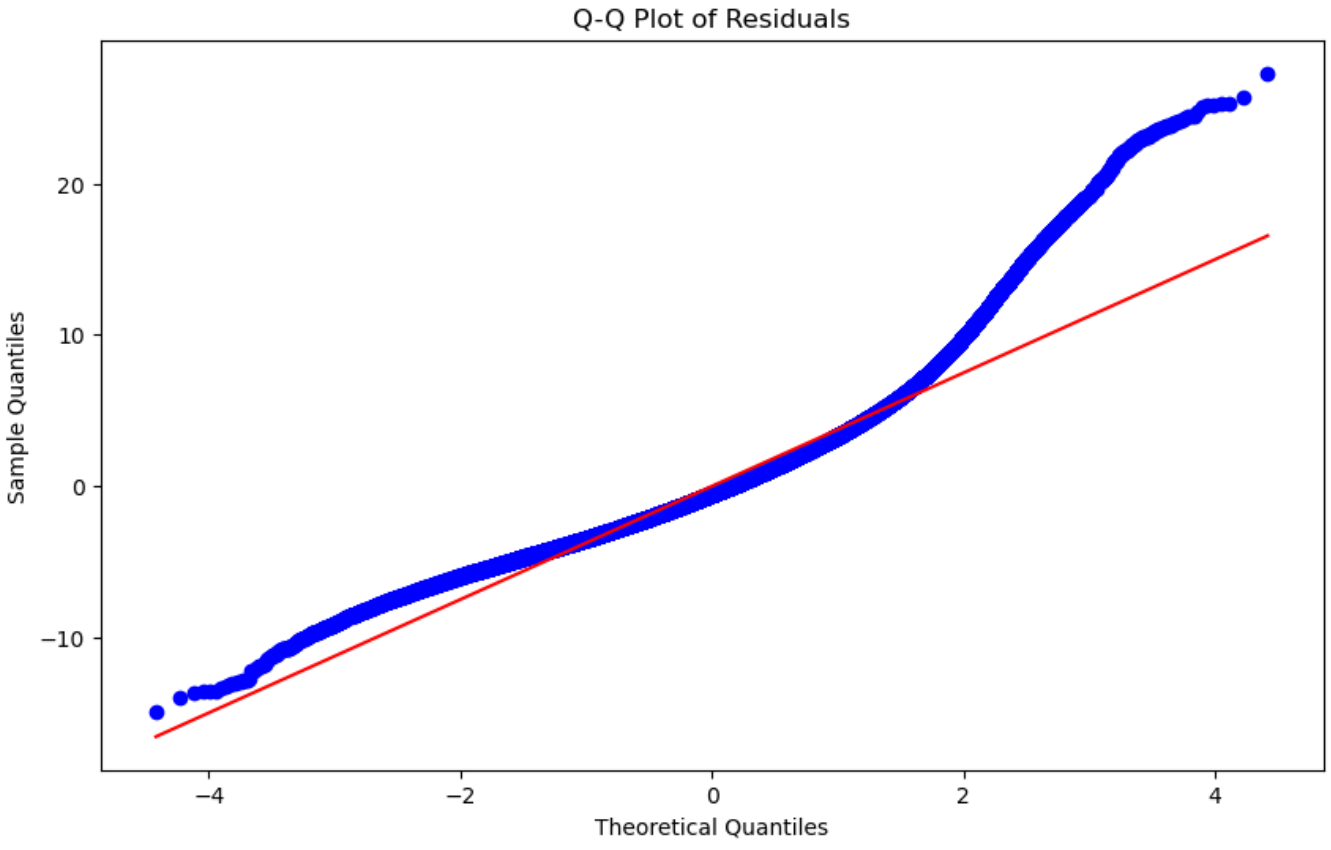
Section 6: Results and Inference

6.1 Perform Residual Analysis

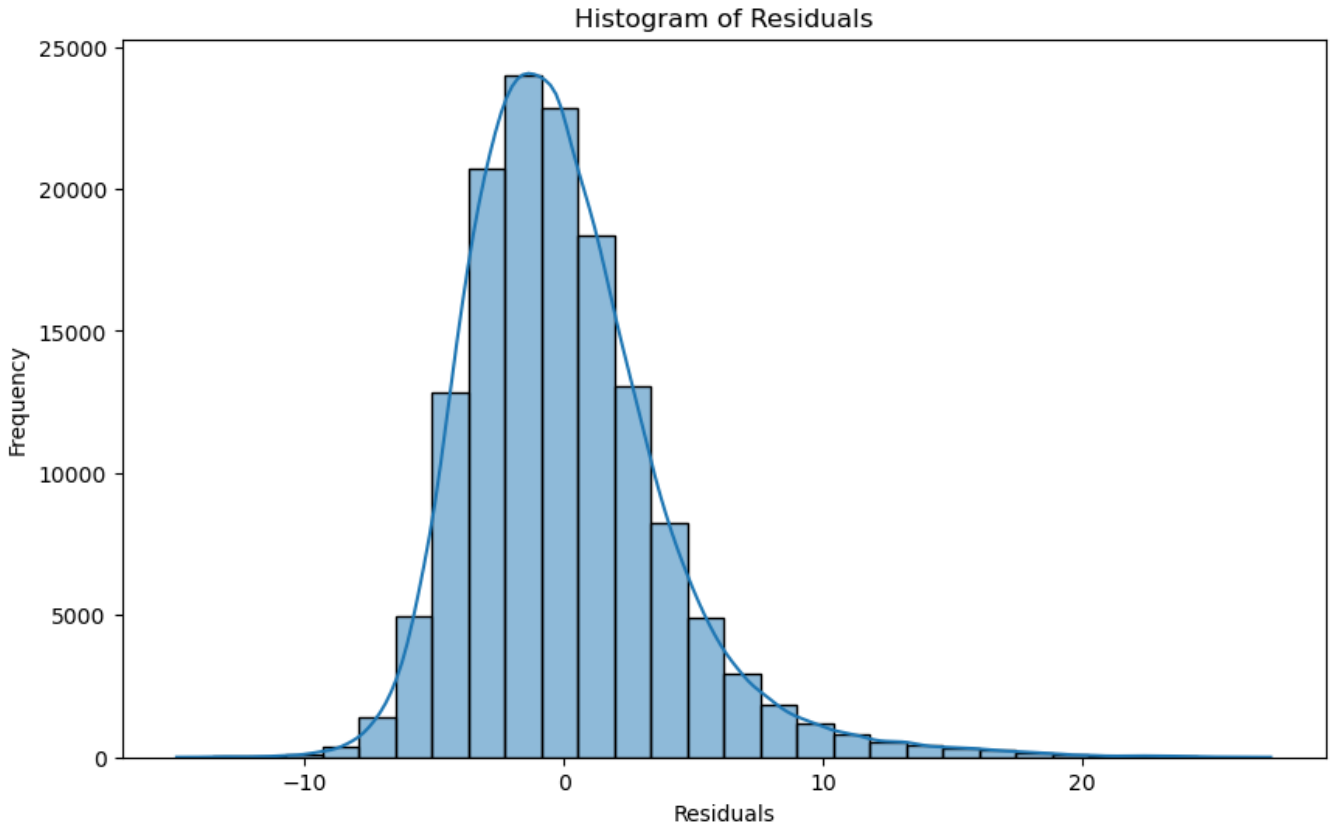
Residuals vs Predicted values


Residuals vs Predicted Values
Residuals
Predicted Values
20
10
0
−10
10
20
30
40
50
60
70
80
90

Q-Q plot


Q-Q Plot of Residuals
Sample Quantiles
Theoretical Quantiles
20
10
0
−10
−4
−2
0
2
4

Residual histogram



1. **Residuals vs Predicted Plot**

Residual = Actual time taken – Predicted time taken

Positive residuals (points above the red dashed 0-line) mean the model under-predicted: the real delivery took longer than forecast.

Negative residuals (points below the 0-line) mean the model over-predicted: it thought the delivery would take longer than it did.

The plot shows data points randomly scattered around the horizontal zero line without any clear patterns or trends. This confirms the linearity assumption of the regression model - the relationship between predictors and target variable is appropriately captured by our linear model

2. **Q-Q Plot of Residuals**

The data points follow the diagonal reference line fairly closely. The residuals approximate a normal distribution, satisfying another key regression assumption. This validates that our statistical inferences (p-values, confidence intervals) from the model are trustworthy

3. **Histogram of Residuals**

The distribution is approximately bell-shaped and centered at zero. Confirms the normal distribution of errors with a mean of zero. Indicates our model is well-balanced in its predictions, not systematically biased in either direction

Overall The model performs well on both training data ($R^2 \approx 0.85$) and test data ($R^2 \approx 083$).

These diagnostics suggest the model provides reliable predictions and that the coefficients can be confidently interpreted for business insights.

6.2 Perform Coefficient Analysis

Coefficient Analysis

	Coefficient	Scaled Coefficient	Unscaled Coefficient
subtotal	3.273438	5.746988	1.864524
distance	4.255782	146.494580	0.123634
total_outstanding_orders	18.737895	8703.680195	0.040340
total_onshift_dashers	-11.913502	-18268.059491	-0.007769
market_id_3.0	-0.909174	-46.197277	-0.017893
market_id_2.0	-2.307307	-74.190154	-0.071757
market_id_4.0	-1.949649	-17.022708	-0.223298
total_busy_dashers	-3.950352	-4.727273	-3.301117

Based on the coefficient analysis in the model, we can make inferences about how different factors affect delivery time:

Positive Coefficients (Factors Increasing Delivery Time)

Subtotal (1.864524): Each $1 increase in order subtotal adds approximately 1.86 minutes to delivery time This is the strongest predictor in the model, suggesting order size/complexity significantly impacts delivery time Likely represents the time needed for restaurants to prepare larger, more complicated orders

Distance (0.123634): Each additional unit of distance increases delivery time by about 0.124 minutes (~7.4 seconds) While important, distance has much less impact than order size Shows that travel time is a factor, but not the dominant one

Negative Coefficients (Factors Decreasing Delivery Time)

Total_onshift_dashers (-0.007769): Each additional delivery partner on shift reduces delivery time by approximately 0.0078 minutes This demonstrates how operational capacity directly influences efficiency Though statistically significant, the small coefficient indicates it provides marginal improvements

Recommended Strategies

Restaurant Operations: Focus improvement efforts on order processing and preparation speed, particularly for larger orders.

Pricing Models: Consider time-based surcharges for larger orders rather than just distance-based fees

Customer Communication: Set more accurate expectations for delivery times based on order size, especially during peak hours

Analyse the effect of a unit change in a feature, say 'total_items':

Let us analyse effect of a unit change in a feature, say 'distance' on the target variable 'time_taken'.

The unscaled coefficient for 'distance' is 0.123634, which means:

For every additional unit increase in distance, the delivery time increases by approximately 0.124 minutes (or about 7.4 seconds). This is a direct cause-effect relationship: longer distances require more travel time.

This quantitative insight has practical implications:

Resource Planning: Delivery partners can be allocated more efficiently based on distance-related time requirements

Pricing Models: Distance-based pricing can be better calibrated to reflect actual time costs

While distance has a positive effect on delivery time, it's worth noting that it's not the strongest predictor: **'subtotal'** has a much larger coefficient (1.86), suggesting that order size/complexity impacts delivery time more significantly than distance. This might indicate that restaurant preparation time (which likely increases with larger orders) is a more dominant factor than travel distance.