Report: LR Delivery time prediction

Include your visualizations, analysis, results, insights, and outcomes. Explain your methodology and approach to the tasks. Add your conclusions to the sections.

1. Data Preprocessing and Feature Engineering

1.1. Fixing the Datatypes

Converted created_at and actual_delivery_time to datetime type

```
dataframe['created_date_time'] = pd.to_datetime(dataframe.created_at)
dataframe['actual_delivery_date_time'] = pd.to_datetime(dataframe.actual_delivery_time)
```

1.2. Calculated duration for order completion

```
result = []
for (actual, created) in zip(dataframe.actual_delivery_date_time,
dataframe.created_date_time):
    duration = actual - created
    result.append(duration)
dataframe['time_for_delivery'] = result
```

1.3. Created two more columns day_of_the_week and month

```
dataframe['day_of_the_week'] = dataframe['created_date_time'].apply(lambda x:
x.weekday() + 1)

dataframe['month'] = dataframe['created_date_time'].dt.month
dataframe['month'] = dataframe['month'].apply(lambda x: x)
```

1.4. Introducing new column isWeekend

```
# Create a categorical feature 'isWeekend'
def isweekend(day):
    if day == 6 or day == 7:
        return 1
    else:
        return 0

dataframe['isWeekend'] = dataframe['created_date_time'].apply(lambda x:
isweekend(x.weekday() + 1))
```

 # Drop unnecessary columns

```
dataframe = dataframe.drop('actual_delivery_time', axis =1 )
dataframe = dataframe.drop('store_primary_category', axis =1 )
```

```
dataframe = dataframe.drop('time_for_delivery', axis =1 )
dataframe = dataframe.drop('created_at', axis =1 )
```

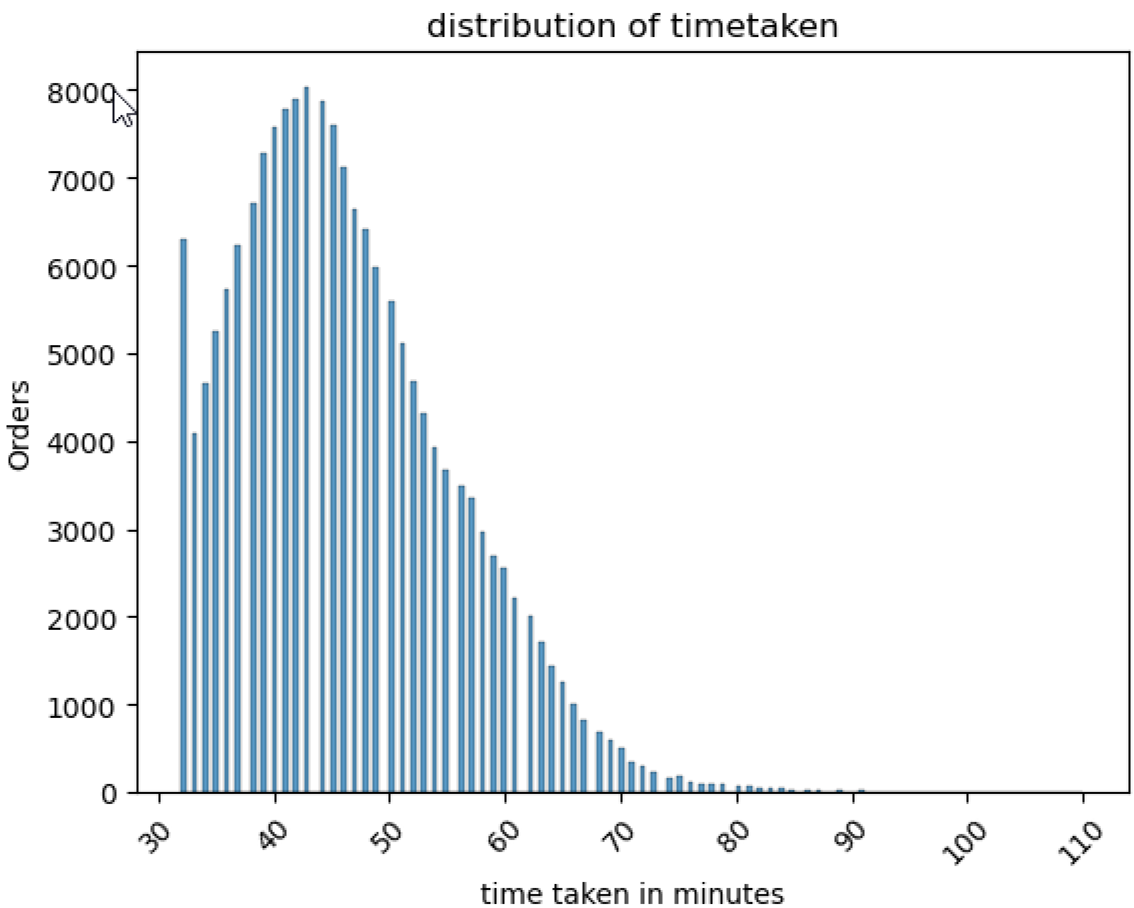
1.5. Added new column for date of order

dataframe['date'] = dataframe['created_date_time'].dt.day

2. Exploratory Data Analysis

Note: I am splitting data into train and test data after EDA.

2.1. Distribution of time_taken



2.2. Finding correlation of columns

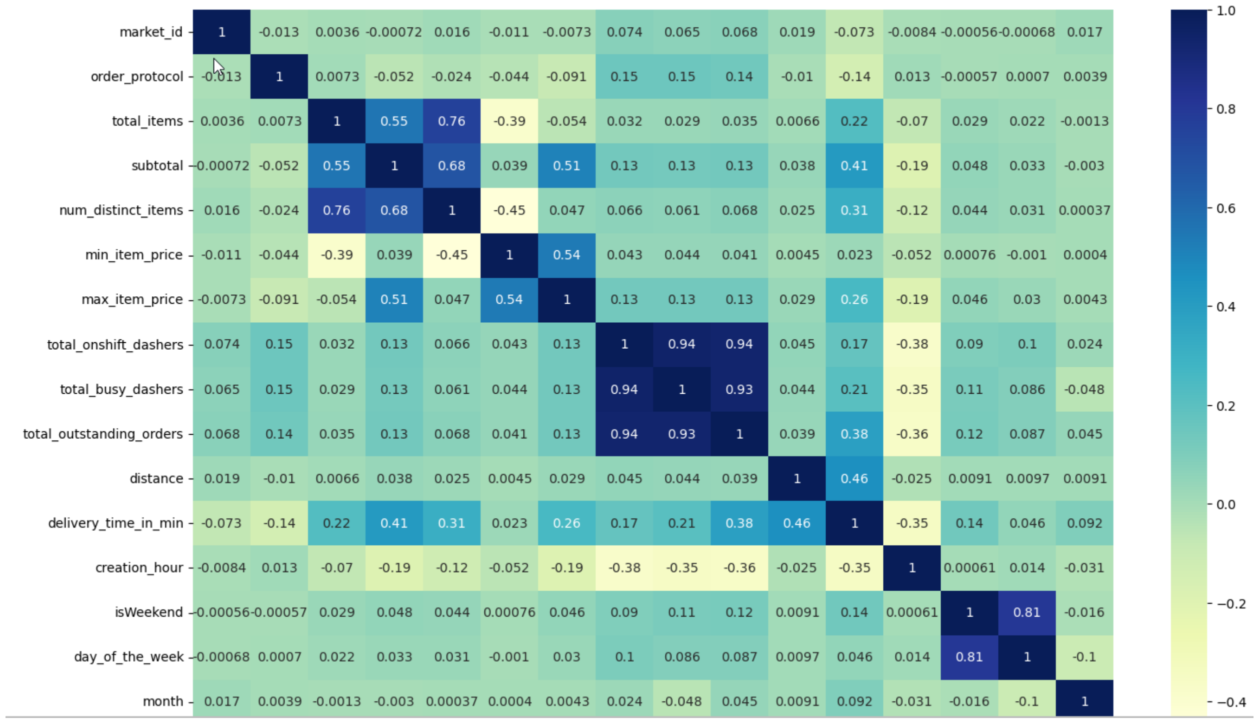
dataframe.corr(numeric_only=True)

gave the correlation details of all columns, which helped drop least correlated columns

plt.figure(figsize = (16, 10))

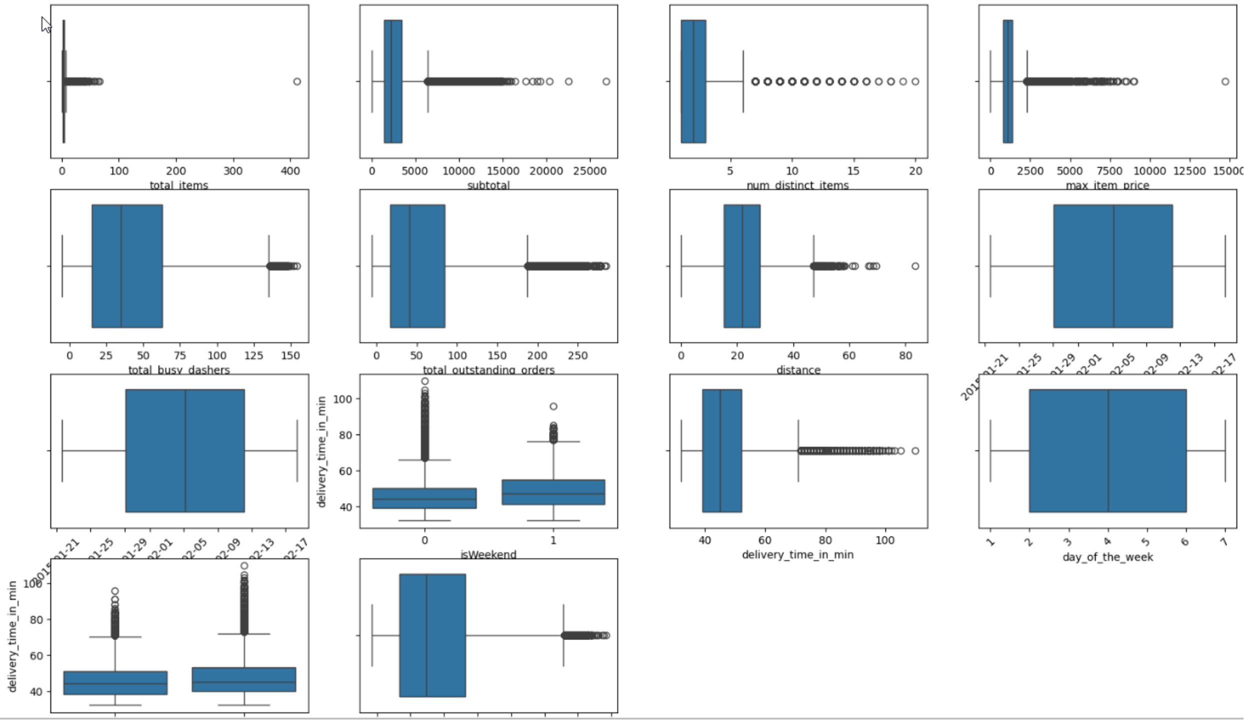
sns.heatmap(dataframe.corr(numeric_only=True), annot = True, cmap="YlGnBu")

plt.show()




```
# Drop 3-5 weakly correlated columns from training dataset
# identified columns to drop market_id, order_protocol, min_item_price, creation_hour,total_onshift_dashers
      tobedel = ['market_id', 'order_protocol', 'min_item_price', 'creation_hour']
      dataframe = dataframe.drop(tobedel, axis=1)
```

2.3 Handle outliers present in all columns



3. Creating training and validation sets

```
df_train, df_test = train_test_split(dataframe, train_size = 0.7, test_size = 0.3, random_state = 100)

y = df_train.pop('delivery_time_in_min')
X = df_train
```

4. Model Building

4.1 Feature Scaling

```python
# Apply scaling to the numerical columns
scaler = StandardScaler()
cols = ['total_items', 'subtotal', 'num_distinct_items','max_item_price','total_onshift_dashers','total_busy_dashers','total_outstanding_orders','distance','isWeekend','day_of_the_week','m
numeric_features = X[cols]
model = scaler.fit(X[cols])
scaled_data = model.transform(X[cols])
                              #[['total_items', 'subtotal', 'num_distinct_items','max_item_price','total_outstanding_orders','distance','isWeekend']])

# print scaled features
print(scaled_data)
```

```
[[-0.07891376  0.65960876  0.20223327 ...  0.73731598 -0.63203715
   0.1760108 ]
 [-0.48995033 -0.32087777 -0.4142855  ... -1.35627061 -0.86233334
   1.04287624]
 [ 0.74315939  1.0151036   1.43527081 ... -1.35627061  1.32548044
   1.69302532]
 ...
 [-0.9009869  -1.11896094 -1.03080427 ...  0.73731598  1.32548044
  -0.25742192]
 [-0.48995033 -0.51642732 -0.4142855  ... -1.35627061 -0.74718524
   0.7178017 ]
 [ 0.33212282 -0.44357552  0.20223327 ...  0.73731598  1.44062853
  -0.79921283]]
```

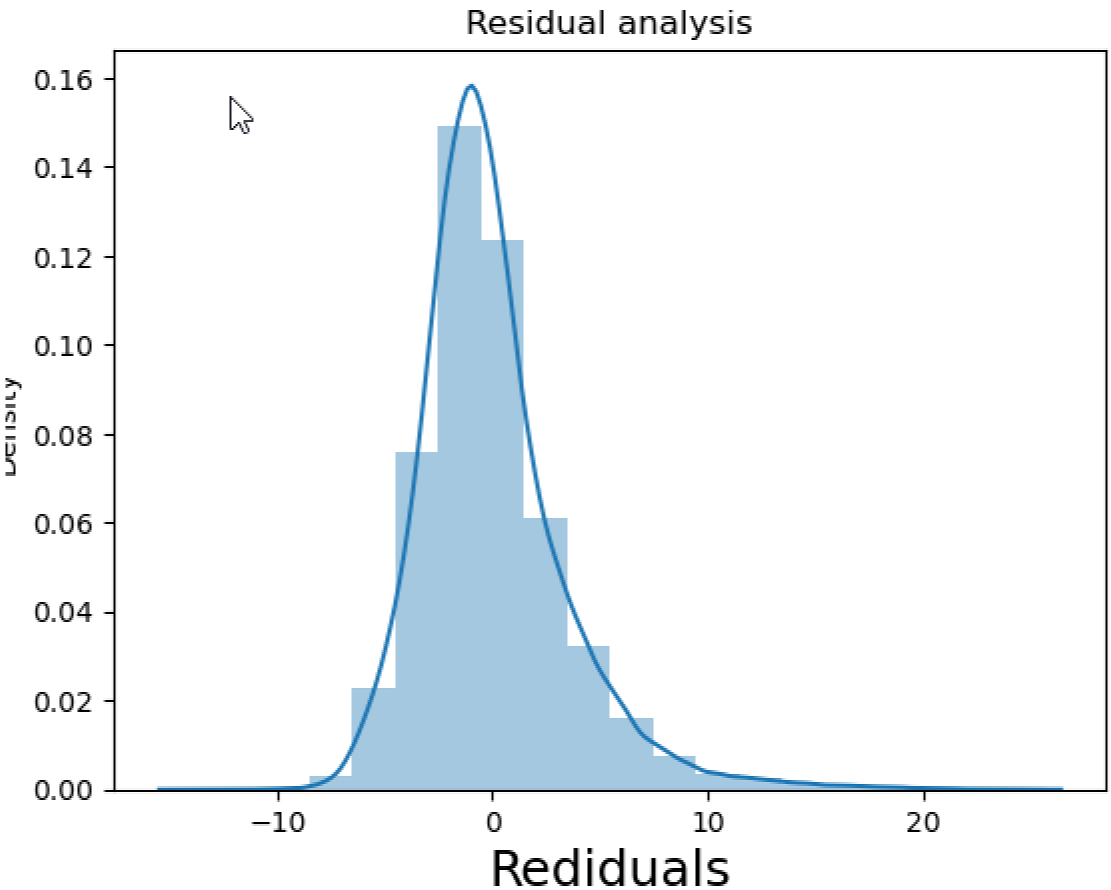
4.2 Build a linear regression model

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.863
Model:                            OLS   Adj. R-squared:                  0.863
Method:                 Least Squares   F-statistic:                 5.971e+04
Date:                Wed, 26 Mar 2025   Prob (F-statistic):               0.00
Time:                        18:41:20   Log-Likelihood:            -3.2669e+05
No. Observations:              123003   AIC:                         6.534e+05
Df Residuals:                  122989   BIC:                         6.535e+05
Df Model:                          13
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                       36.9979      0.119    311.564      0.000      36.765      37.231
total_items                 -0.0721      0.008     -9.503      0.000      -0.087      -0.057
subtotal                     0.0013   1.02e-05    126.184      0.000       0.001       0.001
num_distinct_items           0.5554      0.012     46.889      0.000       0.532       0.579
max_item_price               0.0008   2.46e-05     33.415      0.000       0.001       0.001
total_onshift_dashers       -0.3634      0.001   -358.624      0.000      -0.365      -0.361
total_busy_dashers          -0.1490      0.001   -138.390      0.000      -0.151      -0.147
total_outstanding_orders     0.3482      0.001    579.489      0.000       0.347       0.349
distance                     0.4772      0.001    423.792      0.000       0.475       0.479
isWeekend                    1.6937      0.036     46.753      0.000       1.623       1.765
day_of_the_week             -0.1287      0.008    -15.191      0.000      -0.145      -0.112
month                       -0.9651      0.044    -21.880      0.000      -1.052      -0.879
creation_hour               -0.2589      0.001   -208.051      0.000      -0.261      -0.257
date                        -0.0567      0.002    -25.461      0.000      -0.061      -0.052
==============================================================================
Omnibus:                    34318.214   Durbin-Watson:                   2.006
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           125105.581
Skew:                           1.375   Prob(JB):                         0.00
Kurtosis:                       7.105   Cond. No.                     4.41e+04
==============================================================================
```

	Features	VIF
0	const	146.11
5	total_onshift_dashers	12.64
6	total_busy_dashers	12.33
7	total_outstanding_orders	10.30
11	month	4.60
13	date	4.38
3	num_distinct_items	3.82
2	subtotal	3.60
1	total_items	3.53
10	day_of_the_week	3.10
9	isWeekend	3.07
4	max_item_price	1.98
12	creation_hour	1.21
8	distance	1.00

Hence dropped columns total_onshift_dashers and total_busy_dashers

4.3 Train the model using the training data and Make predictions

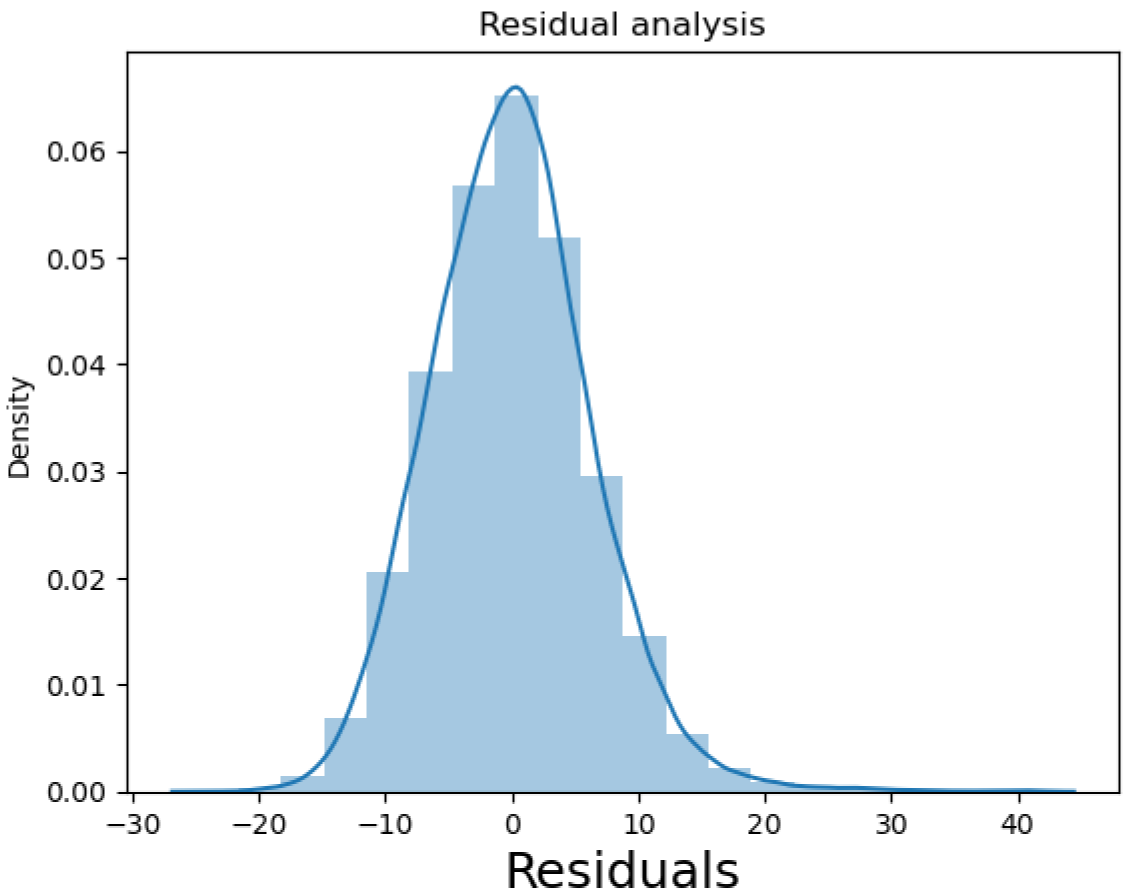


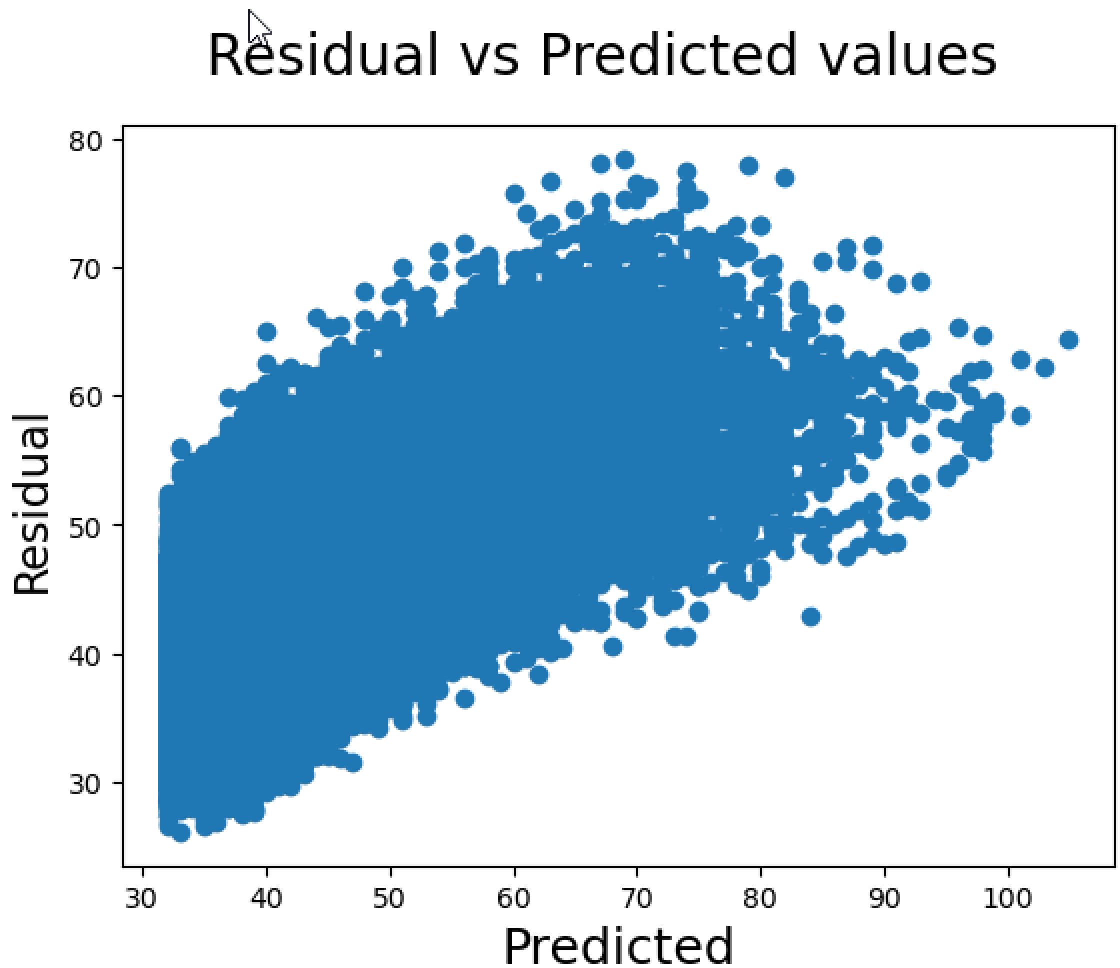
4.4 Build the model and fit RFE to select the most important features

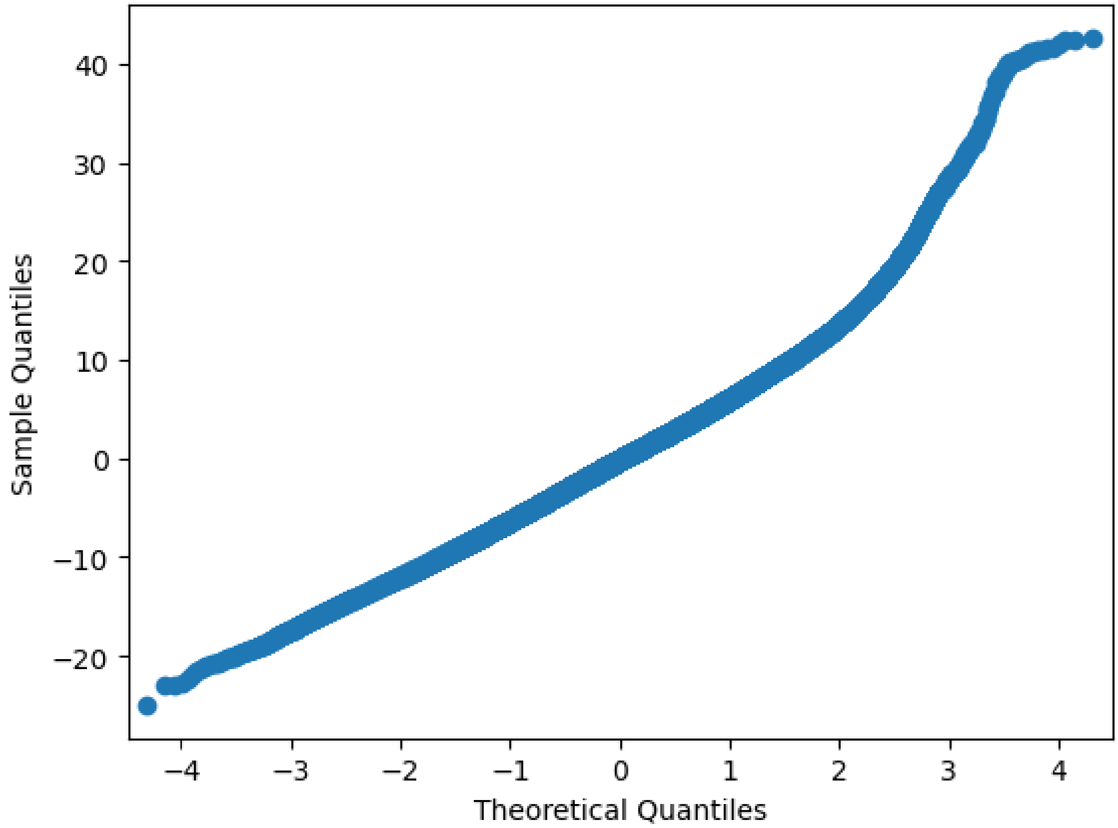
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.523
Model:                            OLS   Adj. R-squared:                  0.523
Method:                 Least Squares   F-statistic:                 1.350e+04
Date:                Wed, 26 Mar 2025   Prob (F-statistic):               0.00
Time:                        18:42:34   Log-Likelihood:            -4.0348e+05
No. Observations:              123003   AIC:                         8.070e+05
Df Residuals:                  122992   BIC:                         8.071e+05
Df Model:                          10
Covariance Type:            nonrobust
============================================================================================
                               coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------
const                       33.3927      0.217    153.954      0.000      32.968      33.818
subtotal                     0.0012   1.82e-05     67.245      0.000       0.001       0.001
num_distinct_items           0.5352      0.018     30.378      0.000       0.501       0.570
max_item_price               0.0007   4.35e-05     16.802      0.000       0.001       0.001
distance                     0.4628      0.002    220.272      0.000       0.459       0.467
isWeekend                    4.2774      0.066     64.786      0.000       4.148       4.407
day_of_the_week             -0.7463      0.015    -48.448      0.000      -0.777      -0.716
month                       -0.4193      0.081     -5.201      0.000      -0.577      -0.261
total_outstanding_orders     0.0434      0.000    114.340      0.000       0.043       0.044
creation_hour               -0.1937      0.002    -83.855      0.000      -0.198      -0.189
date                        -0.0915      0.004    -22.085      0.000      -0.100      -0.083
==============================================================================
Omnibus:                     7194.356   Durbin-Watson:                   1.997
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            12947.299
Skew:                           0.449   Prob(JB):                         0.00
Kurtosis:                       4.311   Cond. No.                     4.31e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 4.31e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

5. Results and Inference




Residual vs Predicted values
Residual
Predicted
80
70
60
50
40
30
30
40
50
60
70
80
90
100



	Features	VIF
0	const	139.87
7	month	4.41
10	date	4.35
1	subtotal	3.27
6	day_of_the_week	2.94
5	isWeekend	2.93
2	num_distinct_items	2.43
3	max_item_price	1.78
9	creation_hour	1.20
8	total_outstanding_orders	1.18
4	distance	1.00

Inference:
The model perform moderately good with following columns :

['subtotal','num_distinct_items','max_item_price','distance','isWeekend','day_of_the_week','month','total_outstanding_orders','creation_hour','date']
With r-squared 52% and VIF of all columns

6. Subjective Questions

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

Ans: These are categorical columns in the data market_id, created_at, actual_delivery_time, order_protocol

I extracted hours, months, day of the order, to make better predictions from datetime columns. Market_id had order_protocol had week correlation with other columns hence dropped them. Month, day of the week had little better correlation and helped get better regression model.

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

Ans: which means split data into 2-part train data and test data, where test data size 20% of total data.

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

Ans:
distance 0.460173

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

Ans:
I drew box plots for each column to identify potentials outliers, and cleaned columns one by one, starting from column with highest outliers to lowest.

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

Ans:
Distance, isWeekend and num_distinct_items

6.6 Explain the linear regression algorithm in detail

Ans:

1. I started modeling with all 12 columns(['total_items', 'subtotal', 'num_distinct_items','max_item_price','total_onshift_dashers','total_busy_dashers','total_outstanding_orders','distance','isWeekend','day_of_the_week','month','creation_hour','date'])
2. Computed VIF for the model, which indicated potential multicollinearity for the columns (total_onshift_dashers : 12.64, total_busy_dashers: 12.33, total_outstanding_orders: 10.30)
3. In the second model dropped total_onshift_dashers.
4. Computed VIF(total_outstanding_orders : 8.34, total_busy_dashers: 8.26)
5. In the third and final model dropped (total_outstanding_orders)
6. R-squared = 0.523
7. VIF for all columns is less than 5, which strongly eliminates multicollinearity.
8. Residual histogram also is normally distributed around 0.

6.7 Explain the difference between simple linear regression and multiple linear regression

Ans:
Simple linear regression involves only one feature that helps in prediction.
Multiple linear regression invloves 2 or more features for prediction, this comes with risk of having multicollinearity, which might wrongly impact model by overfitting.

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

Ans:
Measures the performance of a machine learning model for a given dataset.
Minimize this error by adjusting its parameters either by adding new parameters or removing tightly correlated columns.

6.9 Explain the difference between overfitting and underfitting.

Overfitting: The model is too complex and fits the training data too closely, which is like memorizing all datapoints.
Underfitting: The model is too simple leading to low model accuracy.

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

Residual plot represents the "leftover" variation in the data that the model hasn't explained. Which visually reveals patterns or trends in the residuals. Helps in identifying Homoscedasticity **,** Outliers