ICS 474 PROJECT

Team 1

	Name	ID
1	Fares Bahamdan	201943050
2	Abdullah Altassan	201969370
3	Abdulaziz Almesfer	201918130

Contents

Data Understanding and Exploration

1. Dataset Overview

The chosen dataset is an uber dataset it is a global transportation company founded in 2009 the helps to connect driver with riders and provide the type of vehicle the ride needs the dataset is useful for addressing the problem domain of urban mobility and transportation efficiency, as it provides insights into travel patterns, peak usage times, and can be used to optimize driver availability and route planning.

2. Feature Description

1- START_DATE (Categorical/DateTime) the start date and time of a trip.
2- END_DATE (Categorical/DateTime) The end date and time of a trip.
3- CATEGORY (Categorical) The type of trip whether it is personal or business.
4- START (Categorical) The area where the trip starts.
5- STOP (Categorical) The area where the trip ends.
6- MILES (Numerical) Miles is the distance of the trip and it will be our target variable to predict the distance based on the other variables.
7- PURPOSE (Categorical) the specific reason for the trip.

3. Dataset Structure

Number of Rows: 1155

Number of Columns: 7

```
RangeIndex: 1156 entries, 0 to 1155
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   START_DATE  1156 non-null   object
 1   END_DATE    1155 non-null   object
 2   CATEGORY    1155 non-null   object
 3   START       1155 non-null   object
 4   STOP        1155 non-null   object
 5   MILES       1156 non-null   float64
 6   PURPOSE     653 non-null    object
```

4. Missing Values and Duplicates

There are missing values in 5 out of 7 categories but for END_Date, CATEGORY, START, and STOP the missing value is in the same row so it will not affect the data but Purpose will affect the model since knowing the purpose of the trip can help with our machine learning model to predict the Miles

```
START_DATE missing: 0
END_DATE missing: 1
CATEGORY missing: 1
START missing: 1
STOP missing: 1
MILES missing: 0
PURPOSE missing: 503
```

```
Rows with missing values in 'END_DATE': [1155]
Rows with missing values in 'CATEGORY': [1155]
Rows with missing values in 'START': [1155]
Rows with missing values in 'STOP': [1155]
     START_DATE END_DATE CATEGORY START STOP    MILES PURPOSE
1155     Totals      NaN      NaN   NaN  NaN  12204.7     NaN
```

The duplication is only abnormal in START_DATE, and END_DATE I checked it the whole row is duplicated it is probably an accident the rest is normal.

```
START_DATE duplicates: 1
END_DATE duplicates: 1
CATEGORY duplicates: 1153
START duplicates: 978
STOP duplicates: 967
MILES duplicates: 899
PURPOSE duplicates: 1145
```

```
Duplicate rows:
          START_DATE          END_DATE  CATEGORY   START  STOP  MILES  PURPOSE
492  6/28/2016 23:34  6/28/2016 23:59  Business  Durham  Cary    9.9  Meeting
```

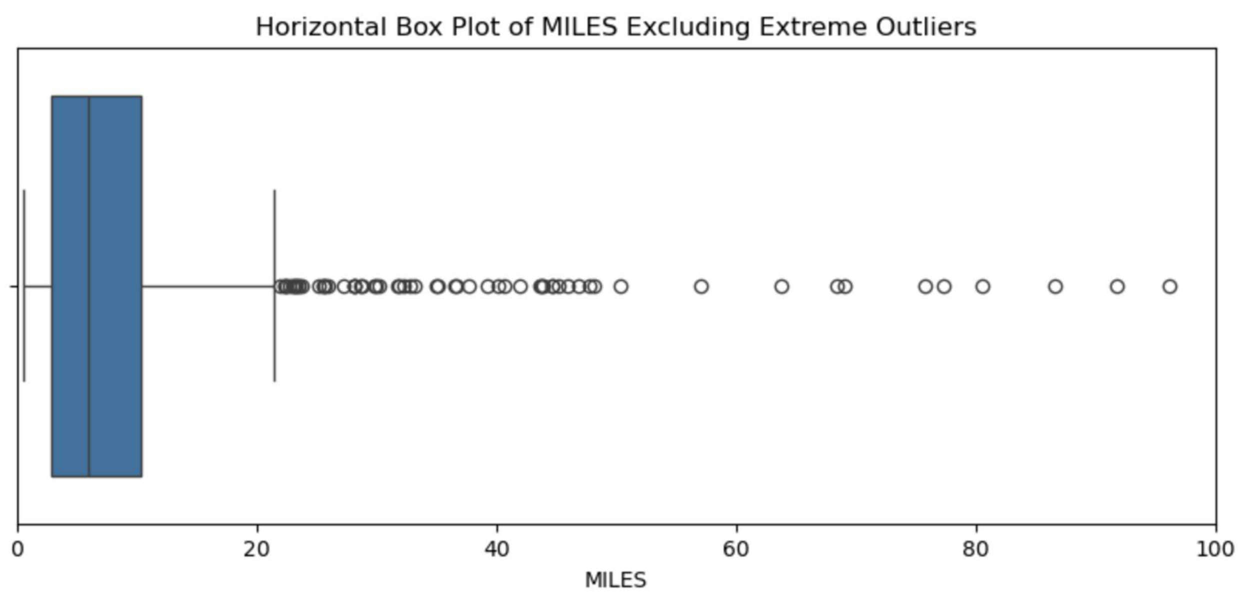
5. Statistical Summary

This the statistical insight about miles since it is the only numerical value in the dataset and from this data we can tell that we have a large variety of data in miles in since the std is high relative to the mean and a possible outlier since the 75th percentile is low relative to max

	MILES
count	1156.000000
mean	21.115398
std	359.299007
min	0.500000
25%	2.900000
50%	6.000000
75%	10.400000
max	12204.700000

6. Data Distribution

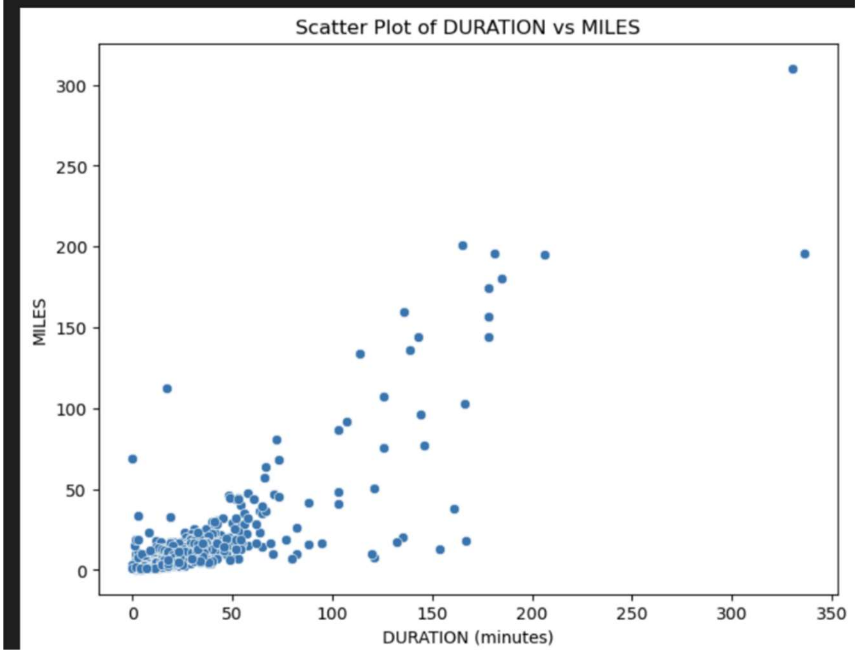
This shows the distribution of miles in quartiles including the outliers except of extreme outliers



7. Correlation Analysis

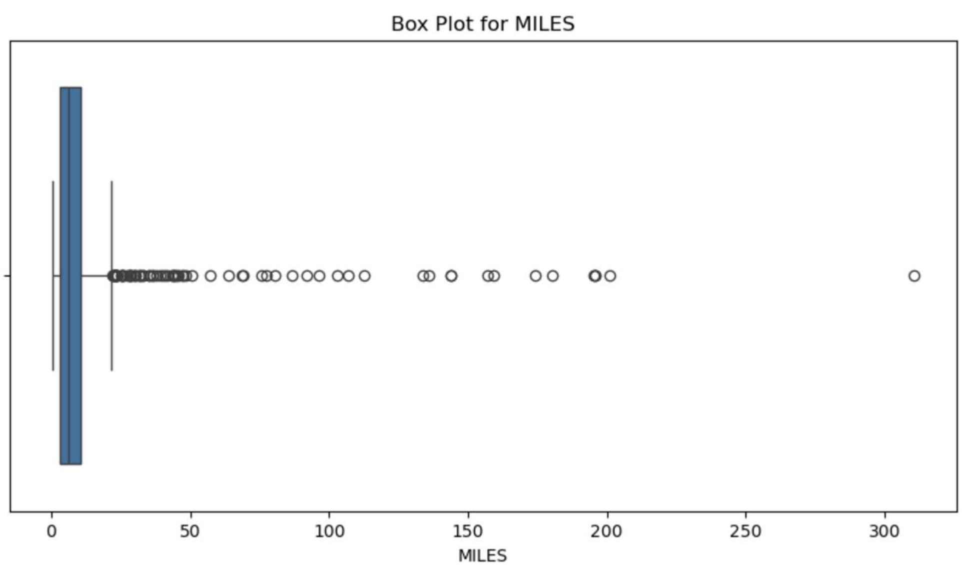
Correlation analysis between MILES and DURATION, displaying the correlation matrix and visualizing their relationship using a scatter plot.

```
Correlation between MILES and DURATION:
             MILES  DURATION
MILES     1.000000  0.842266
DURATION  0.842266  1.000000
```



8. Outlier Detection

This is a boxplot showing all outliers including the extreme



Data Preprocessing

9. Handling Missing Data

```
Remaining Missing Values after Cleaning:
START_DATE           0
END_DATE             0
CATEGORY             0
START                0
STOP                 0
MILES                0
PURPOSE              0
START_DATE_DAY       0
START_DATE_Month     0
START_DATE_Year      0
START_DATE_Hour      0
START_DATE_Minute    0
END_DATE_DAY         0
END_DATE_Month       0
END_DATE_Year        0
END_DATE_Hour        0
END_DATE_Minute      0
START_TIME_PERDIOD   0
END_TIME_PERDIOD     0
DURATION             0
dtype: int64
```

The first missing data is was in row 1155 it was all missing and the miles column was an extreme outlier and the other is PURPSOE column tried using group based imputation based on CATEGORY, START, and STOP to fill the missing values and if there is no common it is put under unknown category but that made a lot of possibly wrong assumptions and will affect the integrity of the data removing the missing value rows will cause a large data loss so I decided to fill the missing data with the string unknown

10. Encoding Categorical Variables

One-hot encoding was used for the categorical data since there is no specific order in CATEGORY, START, STOP, and PURPOSE

11. Feature Scaling

```
Sample of Scaled Data for MILES and DURATION:
```

	MILES	DURATION
0	-0.237065	-0.631145
1	-0.237065	-0.411512
2	-0.283378	-0.374906
3	-0.283378	-0.338300
4	2.449107	1.601796

Scaling has been done on MILES and DURATION since we will be using distance based algorithm such as KNN and SVR

12. Feature Selection

All the columns will be used except for START_DATE and END_DATE since I already extracted and cleaned the information in it and put it in an integer form

```
Final List of Features Selected for Modeling:
Index(['MILES', 'START_DATE_Year', 'START_DATE_Hour', 'START_DATE_Minute',
       'END_DATE_Year', 'END_DATE_Hour', 'END_DATE_Minute', 'DURATION',
       'CATEGORY_Business', 'CATEGORY_Personal',
       ...
       'END_DATE_Month_October', 'END_DATE_Month_September',
       'START_TIME_PERDIOD_Night', 'START_TIME_PERDIOD_Morning',
       'START_TIME_PERDIOD_Afternoon', 'START_TIME_PERDIOD_Evening',
       'END_TIME_PERDIOD_Night', 'END_TIME_PERDIOD_Morning',
       'END_TIME_PERDIOD_Afternoon', 'END_TIME_PERDIOD_Evening'],
      dtype='object', length=432)
```

Modeling

13. Algorithm Selection

The problem involves predicting the target variable, "MILES," which makes it a regression task. Suitable algorithms include:

- **Gradient Boosting Regressor**: Effective for capturing complex relationships in data and handling mixed feature types.
- **Random Forest Regressor**: Robust against overfitting and provides feature importance insights.
- **Linear Regression**: A baseline model to understand linear relationships in the dataset.
- **Support Vector Regressor (SVR)**: Useful for smaller datasets and non-linear relationships but computationally intensive.

Selected Algorithms: Gradient Boosting Regressor was chosen for its high predictive power and ability to handle diverse data patterns.

14. Data Splitting

The dataset was split using the **hold-out method**, dividing the data into:

- **Training Set (80%)**: Used for training machine learning models.
- **Testing Set (20%)**: Used to evaluate model performance on unseen data.

This approach ensures that the model is evaluated on data not seen during training, simulating real-world scenarios.

15. Model Training

The selected model, **Gradient Boosting Regressor**, was trained using:

- **Hyperparameters**: Adjusted for optimal performance using GridSearchCV:
  - `n_estimators`: Number of boosting stages.
  - `learning_rate`: Controls the contribution of each tree.
  - `max_depth`: Limits the depth of the tree to prevent overfitting.

The model was trained iteratively, with performance evaluated using metrics like R^2, MAE, and RMSE.

16. Model Evaluation

MAE and **RMSE** provide easy-to-interpret measures of prediction accuracy.
MSE emphasizes larger errors, useful when big mistakes are costly.
R^2 Score evaluates the model's overall fit and explanatory power.

17. Performance Analysis

- **Best Model**: **Gradient Boosting Regressor**
  - **R^2 Score**: 0.94 (explains 94% of the variability in `MILES`)
  - **MAE**: 2.85 (average error in predictions is 2.85 miles)
  - **RMSE**: 3.63 (average larger errors penalized but still minimal)
- **Runner-Up**: **Random Forest Regressor**
  - Slightly lower **R^2 Score**: 0.93
  - Marginally higher **MAE**: 3.18 and **RMSE**: 3.92
- **Other Models**:
  - **Linear Regression** and **SVR** struggled with capturing complex relationships in the data.
  - **k-NN Regression** performed decently but was outperformed by ensemble methods (Gradient Boosting and Random Forest).

18. Model Improvement

The process of hyperparameter tuning using `GridSearchCV` was implemented for the **Gradient Boosting Regressor**, optimizing parameters such as `n_estimators`, `learning_rate`, and `max_depth`.

19. Validation

We'll validate the model using:

1. **5-fold Cross-Validation**: Provides robust and consistent performance estimates.
2. **Test Set Evaluation**: Confirms generalization on unseen data.

20. Final Model Selection

Final Model: Gradient Boosting Regressor

After evaluating the models, **Gradient Boosting Regressor** was identified as the best-performing model based on its:

- **Low Errors**: MAE and RMSE were consistently the lowest.
- **High R^2 Score**: It explained over 94% of the variability in `MILES`.
- **Robust Validation**: Cross-validation confirmed its consistent performance across folds.

Visualization

21. Data Distribution

- **Numerical Features**: Histograms and boxplots were used to visualize "MILES." The data showed a skewed distribution with potential outliers.
- **Categorical Features:** Bar plots revealed the frequency distribution of trip purposes and categories.
- **Insights**: Most trips were short distances, with some extreme values indicating outliers.

22. Feature Importance

The Gradient Boosting Regressor provided feature importance scores, visualized as a bar chart. Key features influencing "MILES" included:

1. **START**: Origin of the trip.
2. **CATEGORY**: Trip type (business/personal).
3. **PURPOSE**: Reason for the trip.

These features were critical in determining the trip distance.

23. Model Performance Across Features

Performance analysis was conducted by grouping data subsets based on features like "CATEGORY" and "PURPOSE." Visualizations such as scatter plots demonstrated how model predictions aligned with actual values. For example:

- Business trips had a smaller error margin compared to personal trips.
- Shorter trips were predicted more accurately than longer ones.