GitHub Repository

- Project code and resources are available at:

Git Hub URL: AndresLonghorn/presentation-module

Executive Summary Presentation

Predicting Co2 emissions by other vehicle features (engine size, weight, mpg).

Finding which feature is most dependable in forecasting future datasets.

Presenting findings and visualizations.

Introduction Slide

Using data-driven decision making is essential in avoiding contributing to Co2 emissions.

This analysis will help anyone who wishes to better estimate Co2 emissions when purchasing a car.

Interactive Map with Folium

- The interactive map was generated using Folium and saved as an HTML file.
- To view the map, open 'interactive_map.html' in a web browse.
- It displays markers for major Canadian cities relevant to the dataset.

Data Collection & Wrangling

- Data Source: Contains model-specific fuel consumption ratings and estimated carbon dioxide emissions for new light-duty vehicles for retail sale in Canada.

Data link: Fuel consumption ratings - Open Government Portal

Data Collection & Wrangling

Cleaning: selecting features for our prediction model. 75% of cars have between 31 MPG – 11 MPG.

```
cdf = df[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_COMB','CO2EMISSIONS']]
cdf.sample(9)
```

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
695	2.0	4	8.0	184
723	1.8	4	9.4	216
101	3.0	6	10.1	232
593	2.4	4	10.2	235
878	3.8	6	10.8	248
25	2.0	4	10.2	235
834	3.7	6	11.4	262
634	5.2	10	15.7	361
978	1.8	4	8.5	196

```
df.describe()
```

	MODELYEAR	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	FUELCONSUMPTION_COMB	FUELCONSUMPTION_COMB_MPG	CO2EMISSIONS
count	1067.0	1067.000000	1067.000000	1067.000000	1067.000000	1067.000000	1067.000000	1067.000000
mean	2014.0	3.346298	5.794752	13.296532	9.474602	11.580881	26.441425	256.228679
std	0.0	1.415895	1.797447	4.101253	2.794510	3.485595	7.468702	63.372304
min	2014.0	1.000000	3.000000	4.600000	4.900000	4.700000	11.000000	108.000000
25%	2014.0	2.000000	4.000000	10.250000	7.500000	9.000000	21.000000	207.000000
50%	2014.0	3.400000	6.000000	12.600000	8.800000	10.900000	26.000000	251.000000
75%	2014.0	4.300000	8.000000	15.550000	10.850000	13.350000	31.000000	294.000000
max	2014.0	8.400000	12.000000	30.200000	20.500000	25.800000	60.000000	488.000000

Data Collection & Wrangling

- Feature Engineering: Created derived features for analysis.
- The data provide is mostly clean.

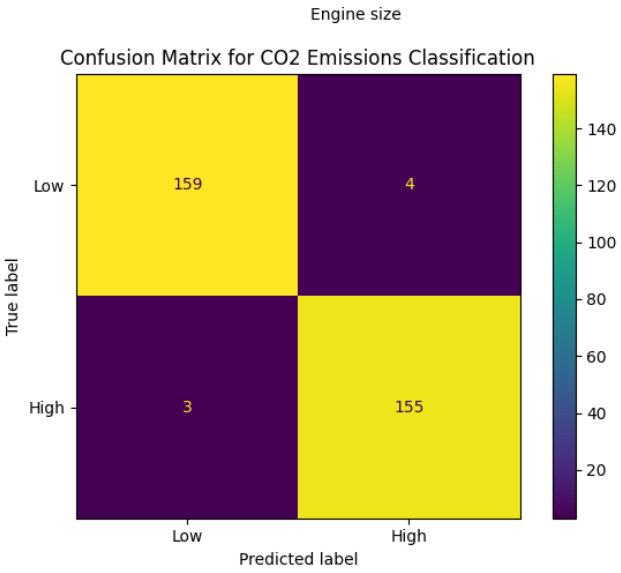
Select a few features that might be indicative of CO2 emission to explore more.

```
cdf = df[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_COMB','CO2EMISSIONS']]
cdf.sample(9)
```

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
240	5.3	8	16.9	270
425	3.5	6	14.3	229
404	2.0	4	10.3	165
426	3.5	6	11.2	258
485	3.5	6	9.6	221
123	4.4	8	13.8	317
869	3.4	6	10.5	242
392	1.6	4	7.6	175
305	3.6	6	10.9	251

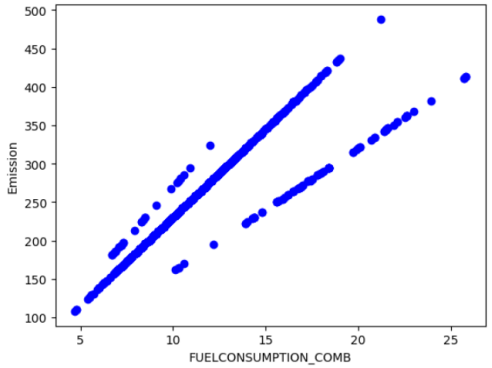
Predictive Analysis Results

- Confusion matrix.



- Slope of the features.

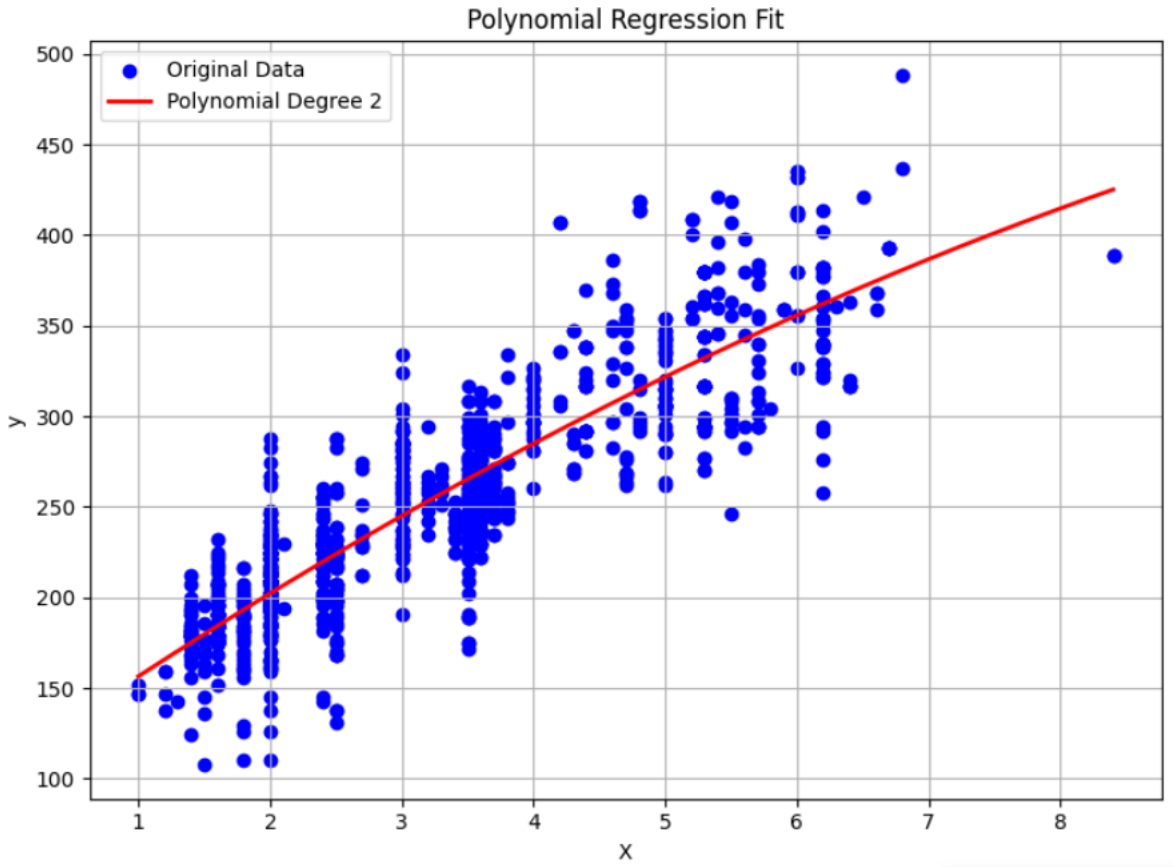
```
plt.scatter(cdf.FUELCONSUMPTION_COMB, cdf.CO2EMISSIONS,  color='blue')
plt.xlabel("FUELCONSUMPTION_COMB")
plt.ylabel("Emission")
plt.show()
```



This is an informative result. Three car groups each have a strong linear relationship between their combined fuel consumption and their CO2 emissions. Their intercepts are similar, while they noticeably differ in their slopes.

Conclusion and Insights

The model can predict 76% of Co2 emissions.

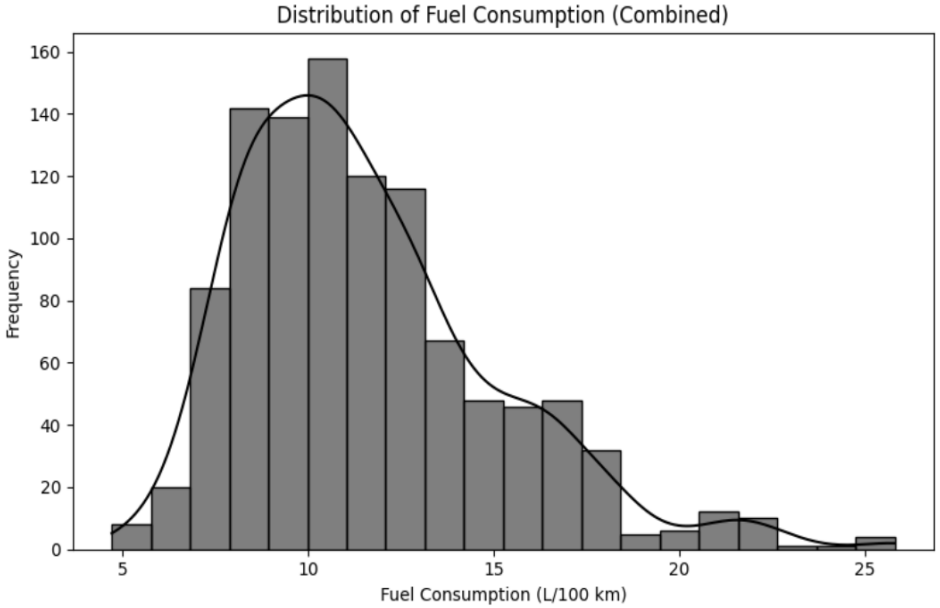



```
R-squared: 0.7615595731934374
Mean Squared Error: 985.9381692274995
Coefficients: [38.99297872]
Intercept: 126.2897021740873
```

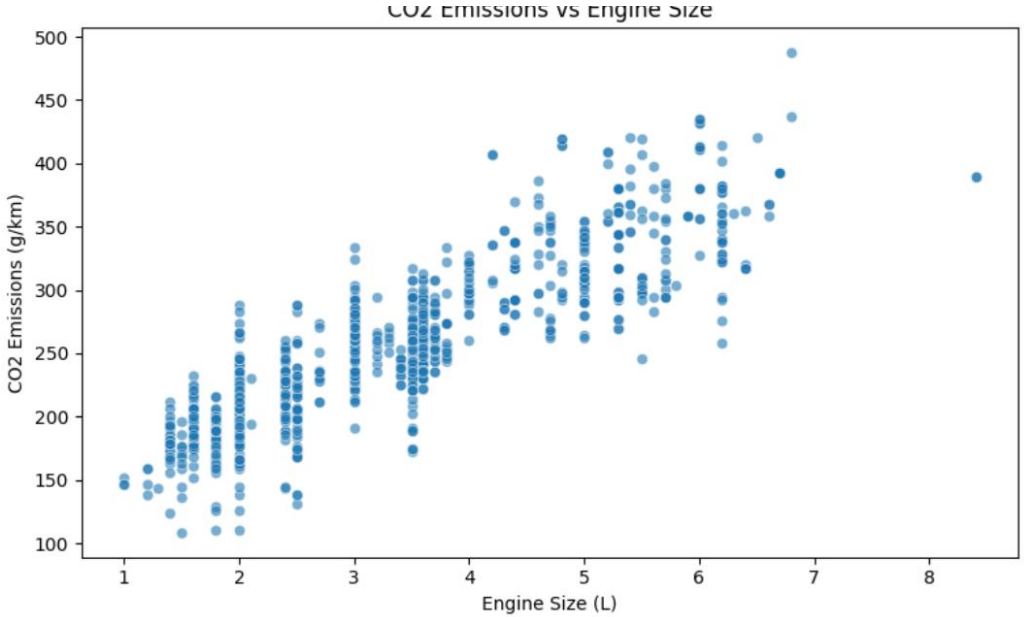
Introduction

- Background: Data-driven decision making is essential in modern analytics.
- Problem Statement: Need to explore, visualize, and predict using diverse data.
- Objectives: Perform EDA, build predictive models, and create interactive visuals.

EDA & Visual Analytics Methodology



The distribution of fuel consumption is mostly between 10 Km per liter. This and the engine size can be used to find a correlation and ultimately predict Co2 emissions.



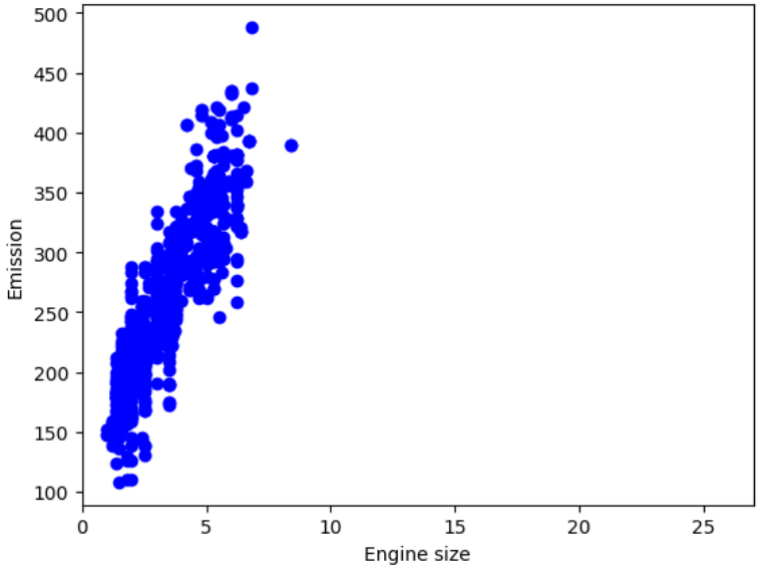
Predictive Analysis Methodology

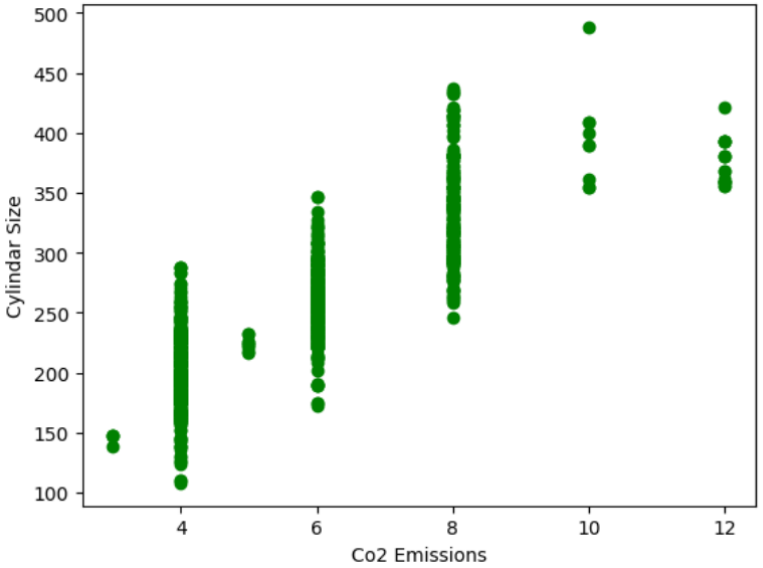
- Model: Random Forest Classifier.
- Data Split: 80% training, 20% testing.
- Evaluation: Confusion matrix and accuracy score.

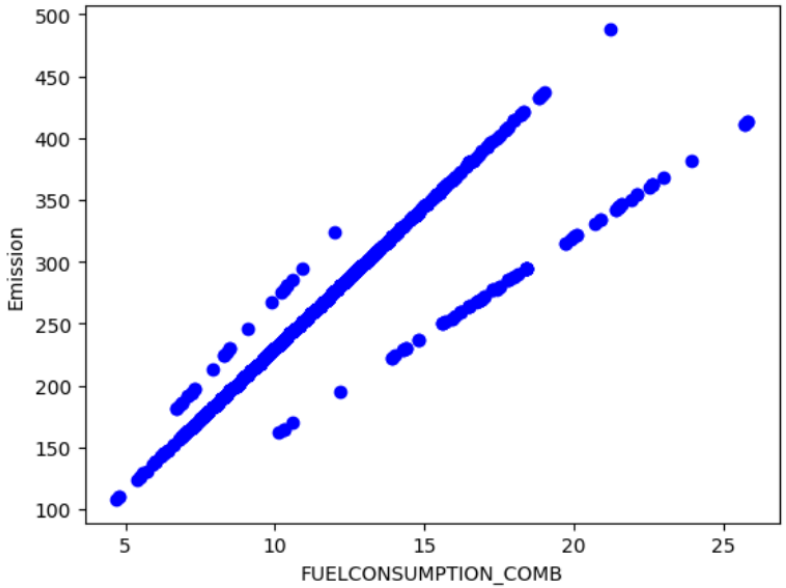
```
from sklearn.model_selection import train_test_split
from sklearn.datasets import make_classification

X, y = make_classification(n_samples=100, n_features=4, random_state=42)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

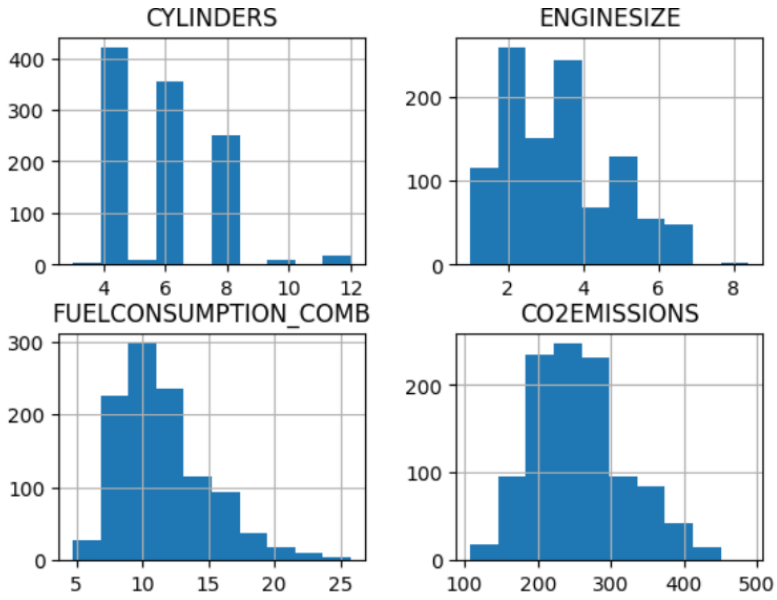
```
R-squared: 0.7615595731934374
Mean Squared Error: 985.9381692274995
Coefficients: [38.99297872]
Intercept: 126.2897021740873
```

EDA with Visualization Results









EDA with SQL Results

Calgary has the highest average age, 35.
Montreal has the youngest average age, 20.

```
Average Age by City:
        City  Average_Age
0    Calgary         40.0
1    Toronto         30.0
2  Vancouver         29.5

Count by City:
        City  Count
0    Calgary      1
1    Toronto      2
2  Vancouver      2

Overall Average Age:
   Overall_Average_Age
0                 31.8

People Older Than 30:
      Name  Age     City
0  Charlie   35  Toronto
1    David   40  Calgary
```

Plotly Dash Dashboard Results