ICBP 2.0

Carbon Footprint Optimization in Supply Chain Logistics

Your Name : UMAMAHESWARI R

Project GitHub Link: https://github.com/radhauma/Carbon-Footprint-Optimization-in-Supply-Chain-Logistics

Content

edunet
foundation

- Problem Statement
- Objective
- Dataset
- Exploratory Data Analysis (EDA)
- Model Selection
- Model Architecture (For Deep Learning Projects)
- Training & Evaluation
- Results
- Conclusion & Future Work
- References

Problem Statement

•**The Challenge:** Vehicular carbon dioxide (CO_2) emissions are a significant contributor to greenhouse gases and climate change.

•**Need for Prediction:** Accurately predicting CO_2 emissions based on vehicle characteristics is crucial for:

- Environmental policy-making and regulation.
- Informing consumer choices towards more eco-friendly vehicles.
- Automotive manufacturers in designing more fuel-efficient and lower-emission vehicles.

•**Complexity:** CO_2 emissions depend on a variety of inter-related vehicle specifications and fuel consumption patterns.

Objective

•**Primary Goal:** To develop a deep learning model capable of accurately predicting CO_2 emissions (g/km) for vehicles based on their specifications.

•**Specific Aims:**

- Perform Exploratory Data Analysis (EDA) to understand the dataset and feature relationships.
- Preprocess the data, handling categorical features and scaling numerical features.
- Build, train, and evaluate a Neural Network model.
- Assess the model's performance using appropriate regression metrics (MAE, RMSE).

Dataset

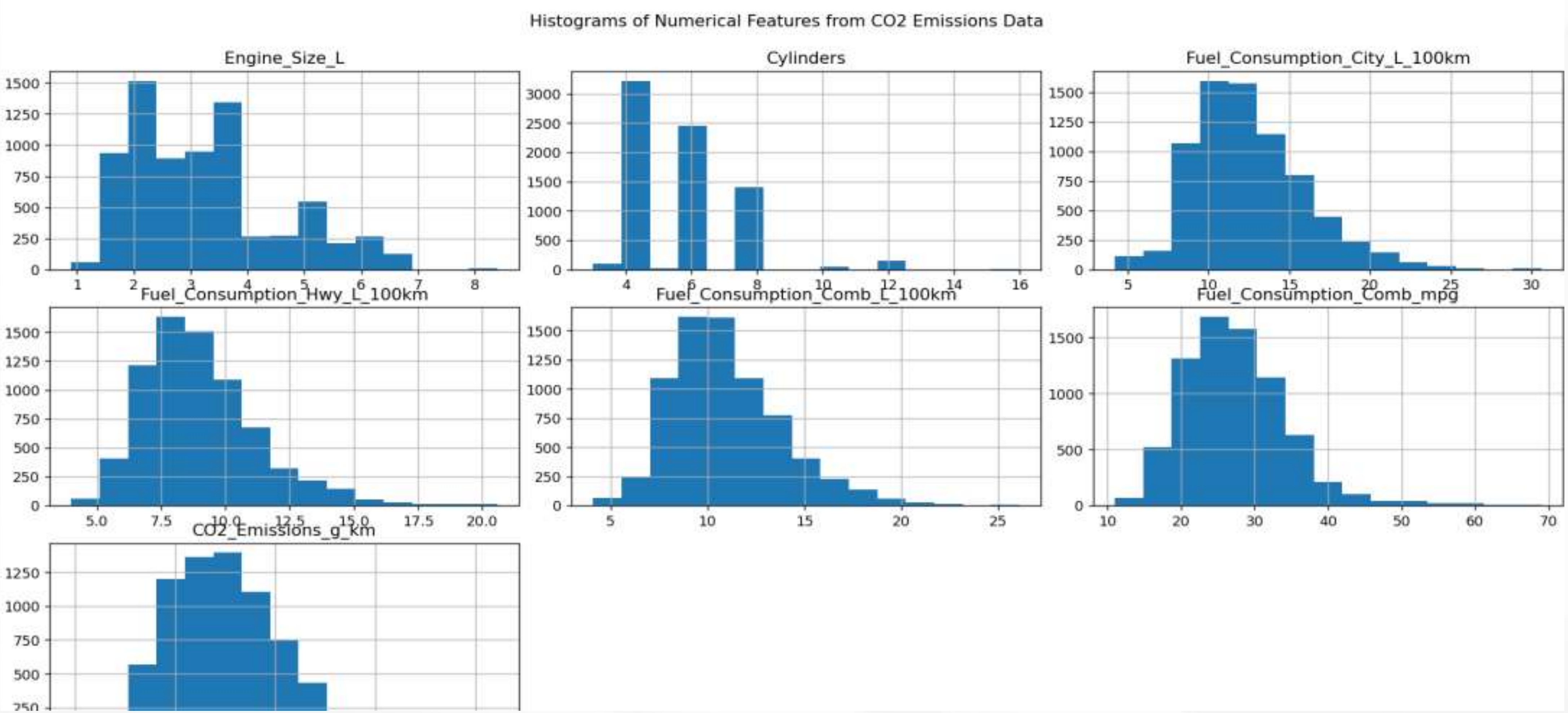
•**Source:** CO2_Emissions_Canada.csv
  •Contains specifications and CO2 emission data for various vehicle models.

•**Size:**
  •7385 entries (vehicles)
  •12 original columns

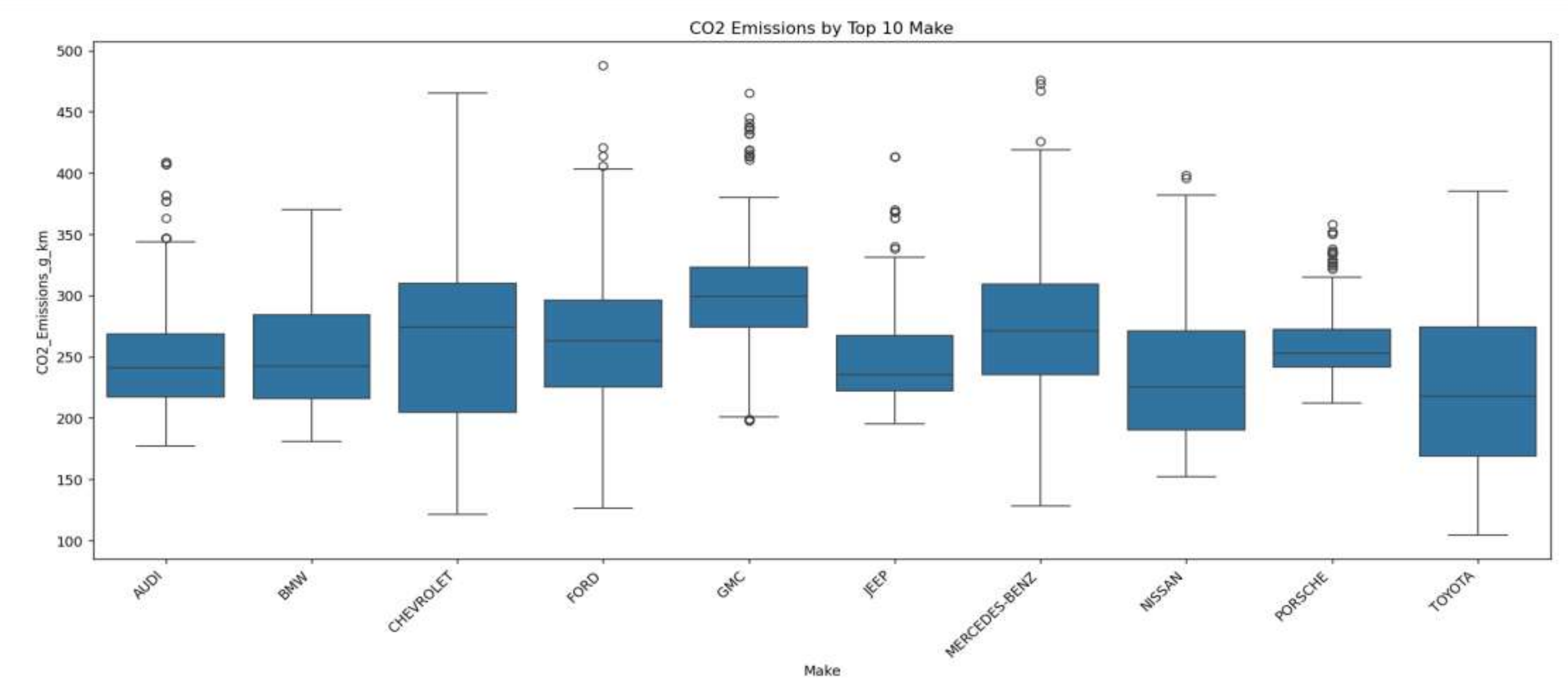
•**Key Features (Original):**
  •Make, Model, Vehicle Class (Categorical)
  •Engine Size(L), Cylinders (Numerical)
  •Transmission, Fuel Type (Categorical)
  •Fuel Consumption City (L/100 km), Fuel Consumption Hwy (L/100 km), Fuel Consumption Comb (L/100 km), Fuel Consumption Comb (mpg) (Numerical)

•**Target Variable:** CO2 Emissions(g/km) (Numerical)
•**Data Quality:** The dataset was complete with no missing values across all 7385 entries.
•**Initial Cleaning:** Column names were standardized (e.g., "Engine Size(L)" to "Engine_Size_L") for easier processing.

Numerical Features

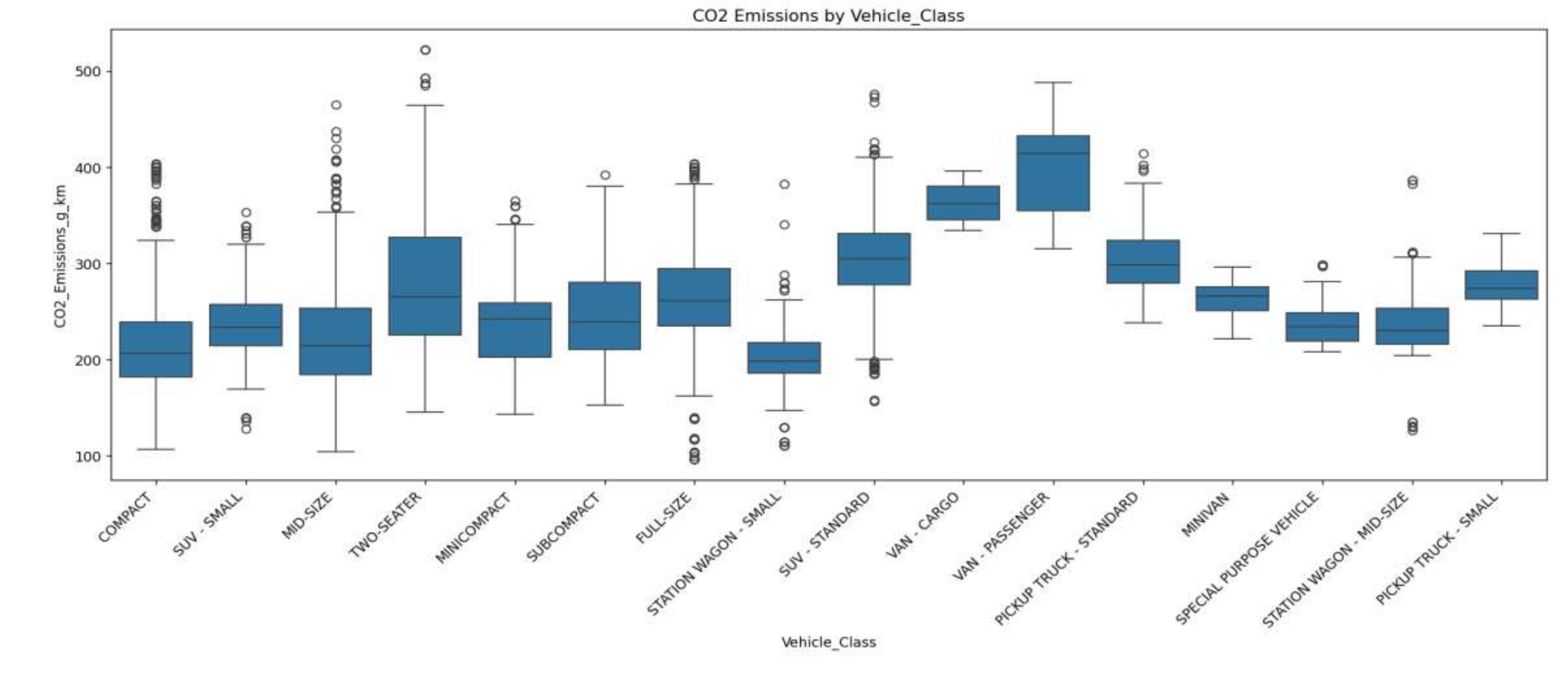


Categorical Features & Correlations(CO2 Emissions by Top 10 Make- boxplot)



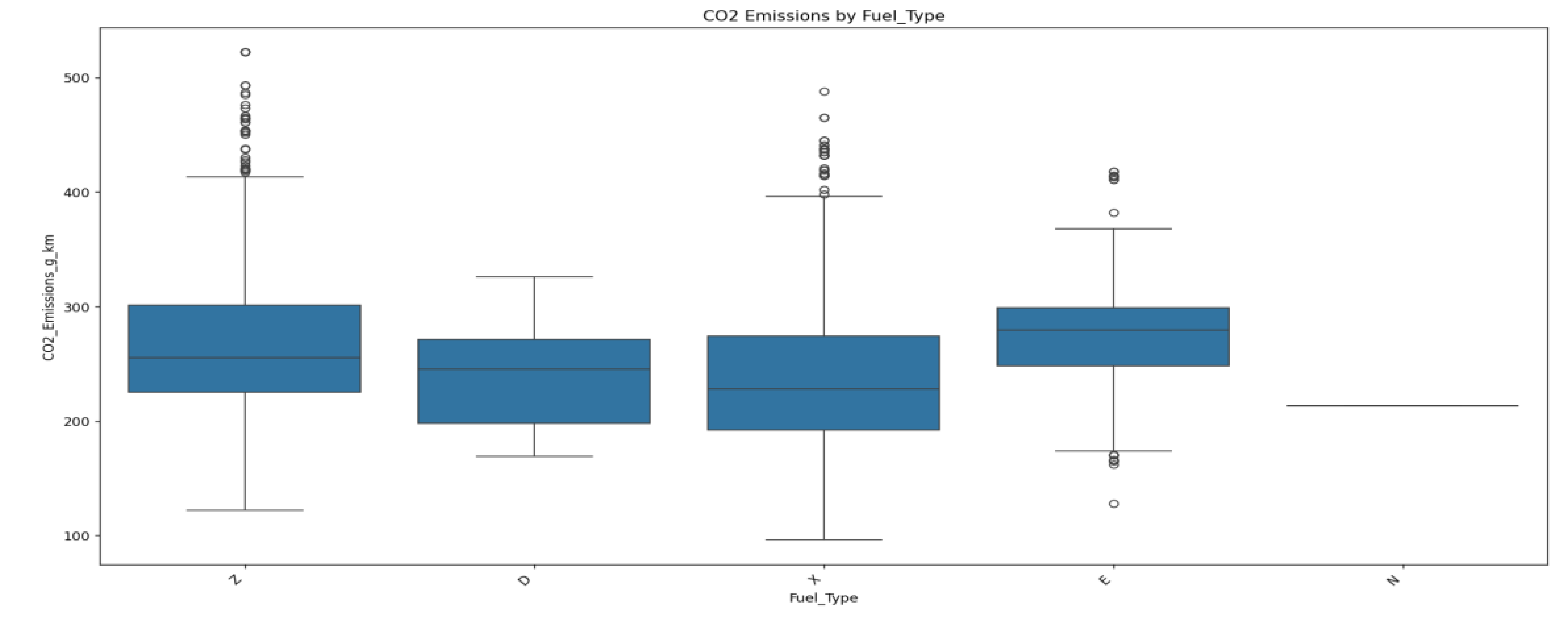
Exploratory Data Analysis (EDA)

Categorical Features & Correlations (CO2 Emissions by Vehicle_Class – boxplot)



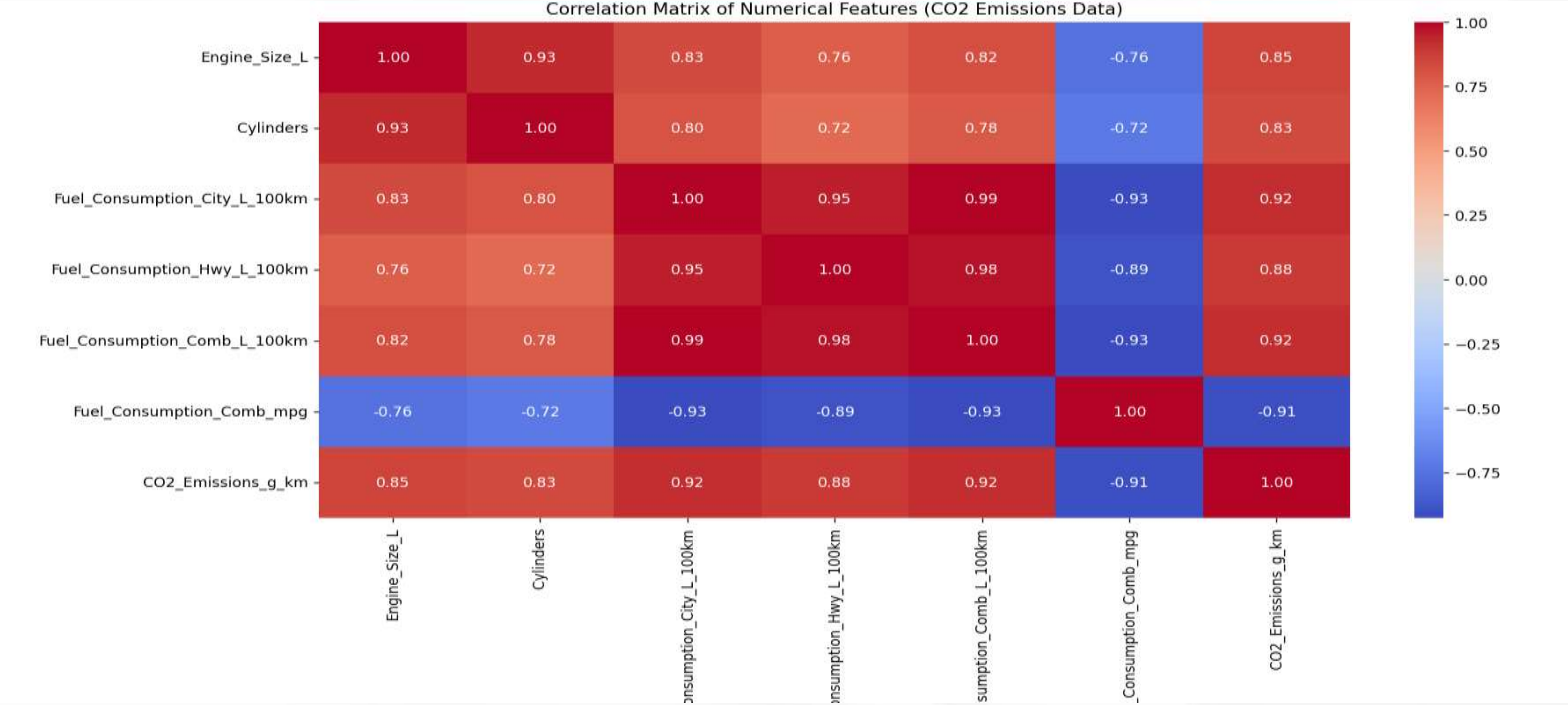
Exploratory Data Analysis (EDA)

Categorical Features & Correlations(CO2 Emissions by Fuel_Type - boxplot)



Exploratory Data Analysis (EDA)

Categorical Features & Correlations(Correlation Matrix of Numerical Features - heatmap)



Model Architecture (For Deep Learning Projects)

- **Model Choice:** A Deep Learning model (Feedforward Neural Network) was selected.
  - **Rationale:** Capable of learning complex, non-linear relationships between vehicle features and CO2 emissions.
- **Target Variable:** CO2_Emissions_g_km
- **Feature Selection for Model:**
  - Numerical: Engine_Size_L, Cylinders, Fuel_Consumption_City_L_100km, Fuel_Consumption_Hwy_L_100km, Fuel_Consumption_Comb_L_100km, Fuel_Consumption_Comb_mpg.
  - Categorical: Make, Vehicle_Class, Transmission, Fuel_Type.
  - Excluded Model due to very high cardinality.
- **Preprocessing Steps:**
  - **Numerical Features:**
    - SimpleImputer (strategy='mean'): Applied, though no NaNs were in the original dataset used for this run.
    - StandardScaler: To scale features to have zero mean and unit variance.
  - **Categorical Features:**
    - SimpleImputer (strategy='most_frequent'): Applied for robustness.
    - OneHotEncoder: To convert categorical variables into a numerical format suitable for the neural network. This resulted in an increase in feature dimensions.
- **Processed Features:** After preprocessing, the input features for the model expanded to 95 columns.

Model Architecture (For Deep Learning Projects)

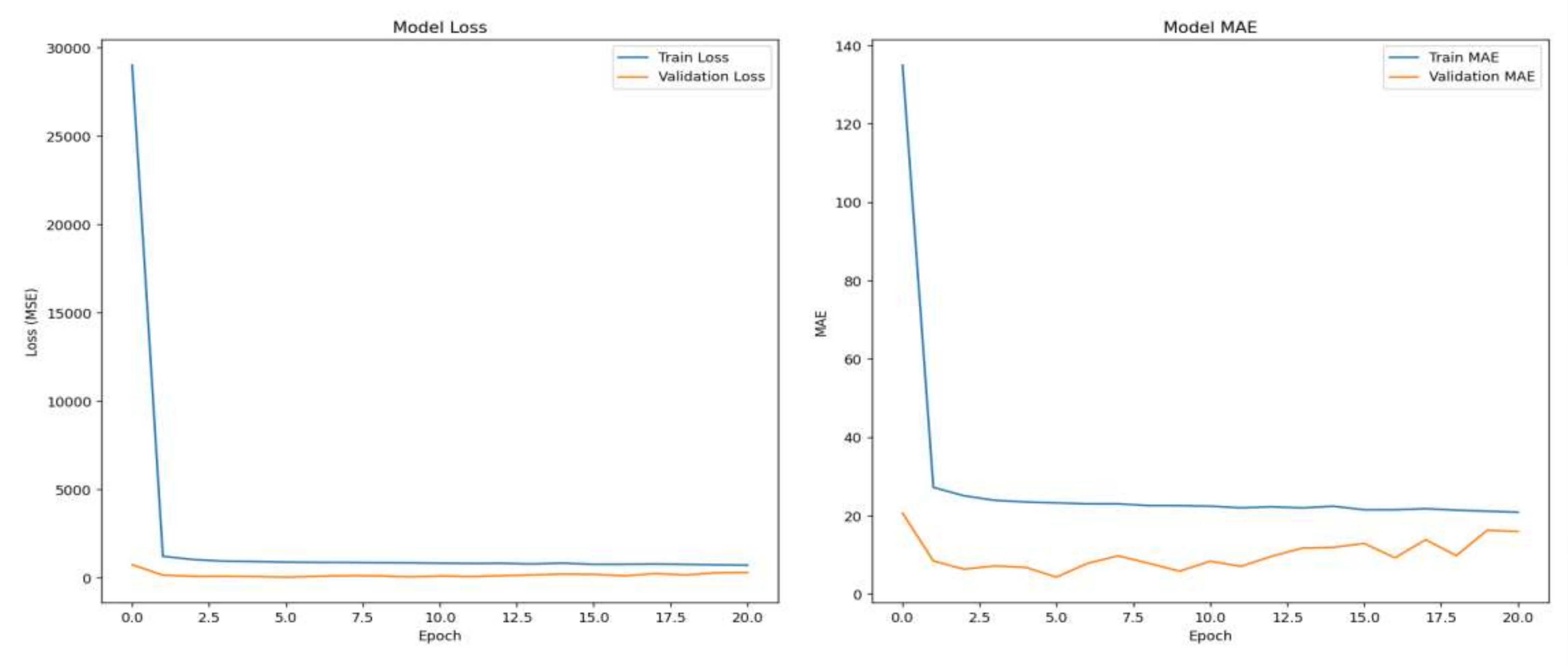
- **Type:** Sequential Feedforward Neural Network
- **Input Layer:**
    - Shape: 95 features (derived from preprocessed data)
- **Hidden Layers:**
    - Layer 1: Dense layer with 128 neurons, ReLU activation.
    - Dropout layer with a rate of 0.3 (to prevent overfitting).
    - Layer 2: Dense layer with 64 neurons, ReLU activation.
    - Dropout layer with a rate of 0.3.
    - Layer 3: Dense layer with 32 neurons, ReLU activation.
- **Output Layer:**
    - Dense layer with 1 neuron, linear activation (for regression task).
- **Total Parameters:** 22,657 (all trainable)

Training & Evaluation

- **Data Splitting:**
  - Training set: 5169 samples
  - Validation set: 1108 samples
  - Test set: 1108 samples
- **Training Configuration:**
  - Optimizer: Adam (learning rate = 0.001)
  - Loss Function: Mean Squared Error (MSE)
  - Metrics: Mean Absolute Error (MAE), MSE
  - Epochs: Maximum 150, Batch Size: 32
  - Callbacks: EarlyStopping (monitor='val_loss', patience=15, restore_best_weights=True)

Training & Evaluation

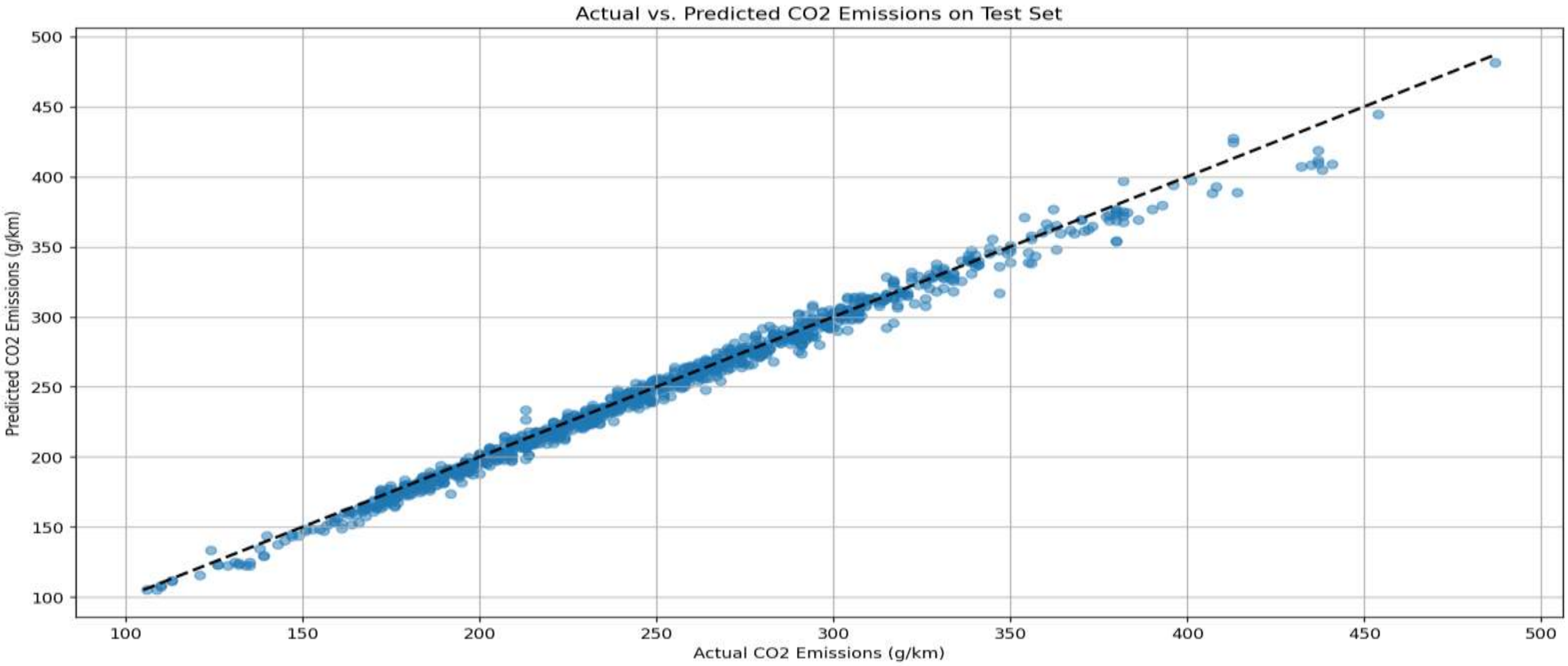
- **Training Process:**



Results

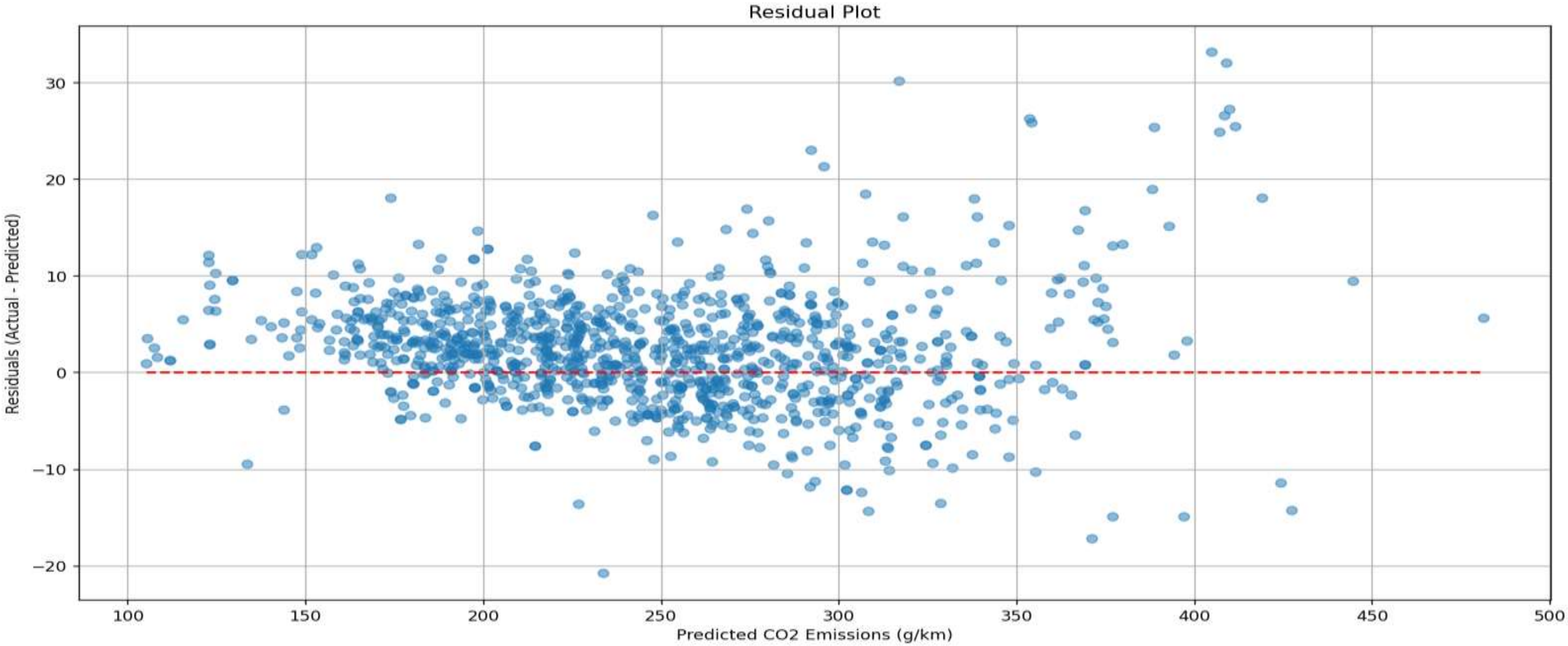
- **Test Set Performance:**
  - Test Loss (MSE): **38.32**
  - Test Mean Absolute Error (MAE): **4.59 g CO2/km**
    - Interpretation: On average, the model's CO2 emission predictions are off by approximately 4.59 g/km from the actual values on unseen test data.
  - Test Root Mean Squared Error (RMSE): **6.19 g CO2/km**
    - Interpretation: Provides another measure of prediction error in the same units as the target, penalizing larger errors more.

Results



Actual vs. Predicted CO2 Emissions on Test Set
Predicted CO2 Emissions (g/km)
Actual CO2 Emissions (g/km)
100
150
200
250
300
350
400
450
500

Results



Residual Plot
Residuals (Actual - Predicted)
Predicted CO2 Emissions (g/km)
30
20
10
0
−10
−20
100
150
200
250
300
350
400
450
500

Conclusion & Future Work

•Conclusion:

- The developed deep learning model demonstrates strong performance in predicting CO_2 emissions for vehicles based on the provided Canadian dataset.
- Achieved a Test MAE of 4.59 g/km, indicating high accuracy.
- EDA revealed key features like fuel consumption metrics, engine size, and cylinders are highly correlated with CO_2 emissions.
- Preprocessing techniques (scaling, one-hot encoding) and model regularization (dropout, early stopping) were effective.

Conclusion & Future Work

•**Future Work:**

- **Feature Engineering:** Explore interaction terms (e.g., Engine Size * Cylinders) or polynomial features.
- **Advanced Categorical Encoding:** Investigate alternatives to one-hot encoding for high-cardinality features like Model (e.g., target encoding, embedding layers).
- **Hyperparameter Tuning:** Systematic optimization of learning rate, number of layers/neurons, dropout rates, and batch size using techniques like KerasTuner or Optuna.
- **Alternative Models:** Compare performance with other machine learning algorithms (e.g., Gradient Boosting Machines like XGBoost or LightGBM, Random Forest).
- **Error Analysis:** Deeper dive into instances where the model performs poorly to identify patterns or data issues.
- **Deployment:** Consider pathways for deploying the model for real-world use (e.g., as a web API).

References

Dataset: **https://www.kaggle.com/datasets/debajyotipodder/co2-emission-by-vehicles?resource=download**

Libraries:

- **Pandas:** https://pandas.pydata.org/
- **NumPy:** https://numpy.org/
- **Scikit-learn:** https://scikit-learn.org/
- **TensorFlow:** https://www.tensorflow.org/
- **Keras:** https://keras.io/
- **Matplotlib:** https://matplotlib.org/
- **Seaborn:** https://seaborn.pydata.org/

http://jmlr.org/papers/v15/srivastava14a.html

https://arxiv.org/abs/1604.06737

https://www.sciencedirect.com/science/article/abs/pii/S1361920921000651?via%3Dihub

https://papers.nips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

Thank You