Assessment Report

on

"4. Classify Vehicles Based on Engine Emissions"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY

DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Name : Anant Kumar

Roll Number : 202401100400034

Section: A

Under the supervision of

"BIKKI KUMAR"

Classify Vehicles Based on Engine Emissions

Title Page

Project Title: Classify Vehicles Based on Engine Emissions
Name: Anant kumar
Roll No: 202401100400034

Introduction

Here's a short introduction for the topic:

Introduction

Classifying vehicles based on engine emissions is an important step toward promoting environmental sustainability and enforcing emission regulations. By analyzing factors such as fuel type, engine size, and CO_2 emissions, machine learning models can categorize vehicles into emission classes like low, moderate, or high emitters. This classification helps governments, manufacturers, and consumers make informed decisions to reduce environmental impact and improve air quality.

Methodology

The approach used in this project involves:

1. **Obtain vehicle data including engine size, fuel type, CO_2 emissions, fuel consumption, etc.**

2. Clean the dataset by handling missing or inconsistent values.
3. **Visualize distributions and relationships between emissions and vehicle features.**

4. **Select key attributes influencing emission levels (e.g., engine size, fuel consumption).**

5. **Create emission categories (e.g., Low, Medium, High) based on emission thresholds or regulations.**

CODE

```python
# Import necessary modules
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

# Train the Random Forest model
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)

# Predict on the test set
y_pred = model.predict(X_test)

# Create confusion matrix
cm = confusion_matrix(y_test, y_pred)

# Define label names if categories are numeric (0 = Low, 1 = Medium, 2 = High)
label_names = ['Low', 'Medium', 'High']

# Plot the confusion matrix heatmap
plt.figure(figsize=(6, 4))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=label_names, yticklabels=label_names)
```

OUTPUT/RESULT

```python
# Load the uploaded CSV file
file_path = "/content/vehicle_emissions.csv"
df = pd.read_csv(file_path)

# Display the first few rows
df.head()
```

	engine_size	fuel_type	co2_emissions	emission_category
0	1.887889	petrol	215.413794	B
1	1.924299	electric	234.463942	C
2	3.687571	diesel	180.042027	A
3	1.078842	petrol	156.704264	A
4	1.416434	diesel	269.166344	A

Visualization of code

Confusion Matrix
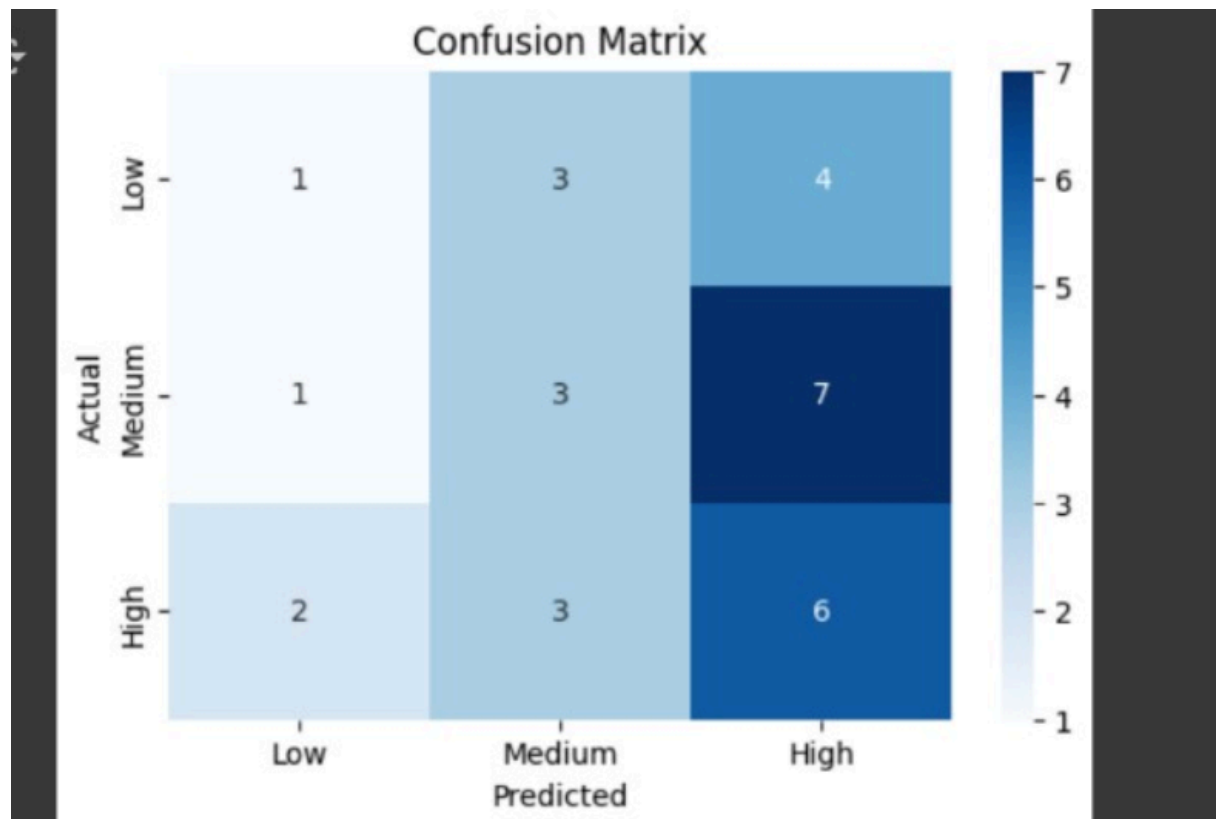
References/credits

Libraries Used

- **Pandas**: For loading, cleaning, and handling the dataset.
  https://pandas.pydata.org

- **NumPy**: For numerical operations and generating synthetic data.
  https://numpy.org

- **Matplotlib**: For plotting graphs and visualizing model results.
  https://matplotlib.org

- **Seaborn**: For advanced data visualizations and statistical plotting.
  https://seaborn.pydata.org

- **Scikit-learn (sklearn)**: For model training, feature scaling, evaluation, and classification.
  https://scikit-learn.org

Dataset Used

- **Student Dropout Dataset** (custom/synthetic): Contains features such as `Grades`, `Marks`, `Participation`, and `DropoutRisk`.

- Dataset created/generated for educational purposes to simulate student dropout prediction scenarios.

Development Environment

- Python programming language (v3.8+)

- Jupyter Notebook or any standard Python IDE (e.g., VSCode, PyCharm, Google Colab)

- **AI Assistance**

    - **OpenAI ChatGPT**: Provided assistance in designing the code structure, writing functions, and documentation.