import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import LabelEncoder, StandardScaler

from sklearn.metrics import classification\_report, confusion\_matrix

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read\_csv('city\_day.csv')

# Drop rows with missing AQI or AQI\_Bucket

df = df.dropna(subset=['AQI', 'AQI\_Bucket'])

# Select relevant features (pollutants)

pollutants = ['PM2.5', 'PM10', 'NO', 'NO2', 'NOx', 'NH3', 'CO', 'SO2', 'O3', 'Benzene', 'Toluene']

# Fill missing pollutant values with median

for col in pollutants:

    df[col].fillna(df[col].median(), inplace=True)

# Encode target variable

label\_encoder = LabelEncoder()

df['AQI\_Bucket\_Encoded'] = label\_encoder.fit\_transform(df['AQI\_Bucket'])

# Feature scaling

scaler = StandardScaler()

X = scaler.fit\_transform(df[pollutants])

y = df['AQI\_Bucket\_Encoded']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

model = RandomForestClassifier(n\_estimators=100, random\_state=42)

model.fit(X\_train, y\_train)

RandomForestClassifier

RandomForestClassifier(random\_state=42)

precision recall f1-score support

Good 0.85 0.65 0.74 282

Moderate 0.81 0.86 0.84 1745

Poor 0.69 0.64 0.66 555

Satisfactory 0.85 0.86 0.85 1655

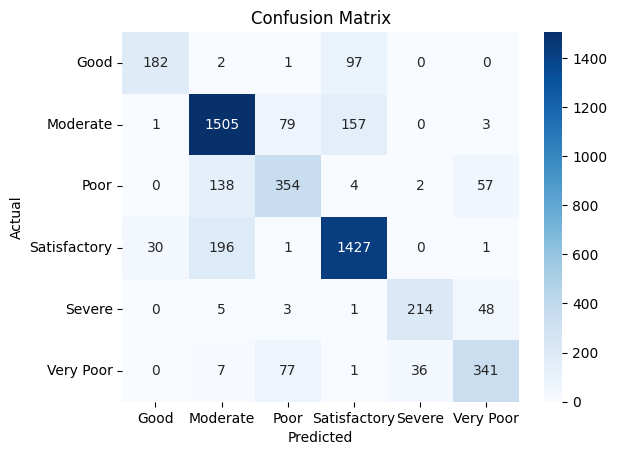
Severe 0.85 0.79 0.82 271

Very Poor 0.76 0.74 0.75 462

accuracy 0.81 4970

macro avg 0.80 0.76 0.78 4970

weighted avg 0.81 0.81 0.81 4970



importances = model.feature\_importances\_

feature\_names = df[pollutants].columns

sns.barplot(x=importances, y=feature\_names)

plt.title('Feature Importance for AQI Bucket Prediction')

plt.xlabel('Importance')

plt.ylabel('Pollutant')

plt.show()

