**PRACTICAL 8**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name:** | Harsh Shah | **Semester:** | VII | **Division:** | 6 |
| **Roll No.:** | 21BCP359 | **Date:** | 10-09-24 | **Batch:** | G11 |
| **Aim:** | Feature Selection in Dataset. | | | | |

**Code**

*import* pandas *as* pd

*import* matplotlib.pyplot *as* plt

*from* statsmodels.tsa.seasonal *import* seasonal\_decompose

*from* statsmodels.tsa.api *import* SimpleExpSmoothing

*from* sklearn.metrics *import* mean\_squared\_error

*# Step 1: Load the data*

data *=* pd.read\_csv("IPG2211A2N.csv", index\_col*=*"DATE", parse\_dates*=True*)

*# Step 2: Plot the raw data*

plt.figure(figsize*=*(12, 6))

plt.plot(data, label*=*"Industrial Production: Utilities (Electric & Gas)")

plt.title("Industrial Production: Electric & Gas Utilities")

plt.xlabel("Date")

plt.ylabel("Production")

plt.legend()

plt.grid(*True*)

plt.show()

*# Step 3: Trend and Seasonal Variation (Seasonal Decomposition)*

decompose\_result *=* seasonal\_decompose(

    data, model*=*"multiplicative", period*=*12

)  *# Assuming monthly data*

decompose\_result.plot()

plt.show()

*# Step 4: Moving Averages*

*def* plot\_moving\_average(data, window\_sizes):

    plt.figure(figsize*=*(12, 6))

    plt.plot(data, label*=*"Original", color*=*"blue")

*for* window *in* window\_sizes:

        data[*f*"MA\_{window}"] *=* data["IPG2211A2N"].rolling(window*=*window).mean()

        plt.plot(data[*f*"MA\_{window}"], label*=f*"Moving Average {window}-months")

    plt.title("Moving Averages for Industrial Production")

    plt.xlabel("Date")

    plt.ylabel("Production")

    plt.legend()

    plt.grid(*True*)

    plt.show()

*# Moving averages for 3, 6, and 12 months*

plot\_moving\_average(data.copy(), window\_sizes*=*[3, 6, 12])

*# Step 5: Time Series Forecasting*

*# Using Simple Exponential Smoothing to predict for 2020-2024*

*# Split data into training and testing*

train *=* data[:"2019"]

test *=* data["2020":]

*# Fit the model on training data*

model *=* SimpleExpSmoothing(train).fit(smoothing\_level*=*0.2, optimized*=True*)

*# Forecast for 2020-2024*

forecast *=* model.forecast(steps*=len*(test))

*# Plot the forecasted data*

plt.figure(figsize*=*(12, 6))

plt.plot(train, label*=*"Training Data")

plt.plot(test, label*=*"Actual Data (2020-2024)", color*=*"orange")

plt.plot(forecast, label*=*"Forecast (2020-2024)", color*=*"green")

plt.title("Forecasting Industrial Production for Electric & Gas Utilities (2020-2024)")

plt.xlabel("Date")

plt.ylabel("Production")

plt.legend()

plt.grid(*True*)

plt.show()

*# Step 6: Analysis*

*print*(*f*"Mean Squared Error: {mean\_squared\_error(test, forecast)}")









