

Ex.No: 1B CONVERSION OF NON STATIONARY TO STATIONARY DATA

Date:

AIM:

To perform regular differencing, seasonal adjustment and log transformation on international airline passenger data

ALGORITHM:

1. Import the required packages like pandas and numpy
2. Read the data using the pandas
3. Perform the data preprocessing if needed and apply regular differencing, seasonal adjustment, log transformation.
4. Plot the data according to need, before and after regular differencing, seasonal adjustment, log transformation.
5. Display the overall results.

PROGRAM:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

data = pd.read_csv("/kaggle/input/time-series/daily-minimum-temperatures-in-me.csv",
on_bad_lines='skip')
data.columns = ["Date", "Temp"]
data["Date"] = pd.to_datetime(data["Date"])
data["Temp"] = pd.to_numeric(data["Temp"], errors="coerce")
data = data.dropna()

seasonal_period = 365

data["regular_diff"] = data["Temp"].diff()

data["seasonal_diff"] = data["Temp"].diff(seasonal_period)

data["log_temp"] = np.log(data["Temp"])
```

```
data["log_regular_diff"] = data["log_temp"].diff()

data["log_regular_seasonal_diff"] = data["log_temp"].diff().diff(seasonal_period)

data_clean = data.dropna()

plt.figure(figsize=(14, 25))

plt.subplot(6,1,1)
plt.plot(data["Date"], data["Temp"])
plt.title("Original Data")
plt.grid(True)

plt.subplot(6,1,2)
plt.plot(data_clean["Date"], data_clean["regular_diff"])
plt.title("Regular Differencing")
plt.grid(True)

plt.subplot(6,1,3)
plt.plot(data_clean["Date"], data_clean["seasonal_diff"])
plt.title("Seasonal Adjustment (Seasonal Differencing)")
plt.grid(True)

plt.subplot(6,1,4)
plt.plot(data_clean["Date"], data_clean["log_temp"])
plt.title("Log Transformation")
plt.grid(True)

plt.subplot(6,1,5)
plt.plot(data_clean["Date"], data_clean["log_regular_diff"])
plt.title("Log Transformation + Regular Differencing")
plt.grid(True)

plt.subplot(6,1,6)
plt.plot(data_clean["Date"], data_clean["log_regular_seasonal_diff"])
plt.title("Log + Regular + Seasonal Differencing")
plt.grid(True)

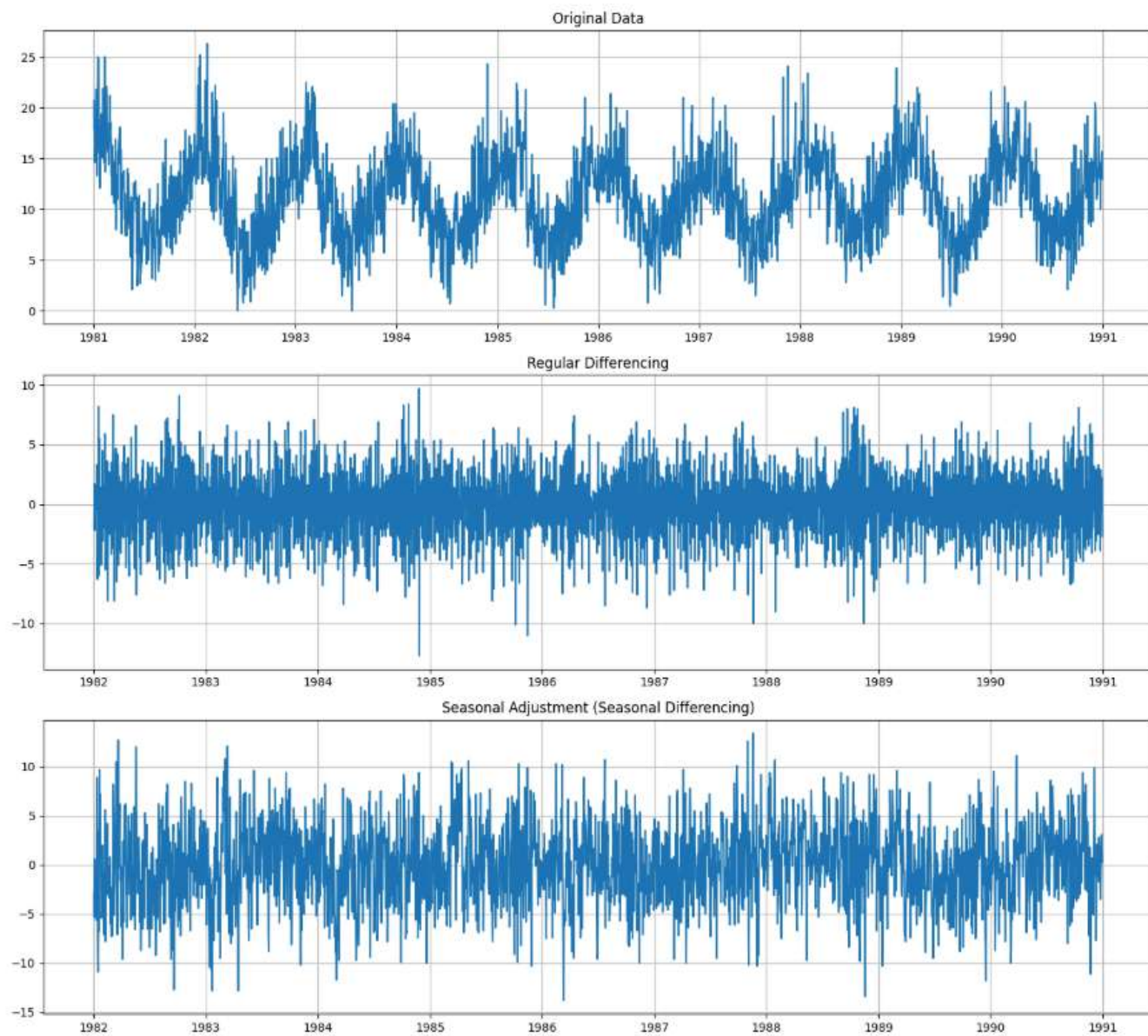
plt.tight_layout()
plt.show()
```

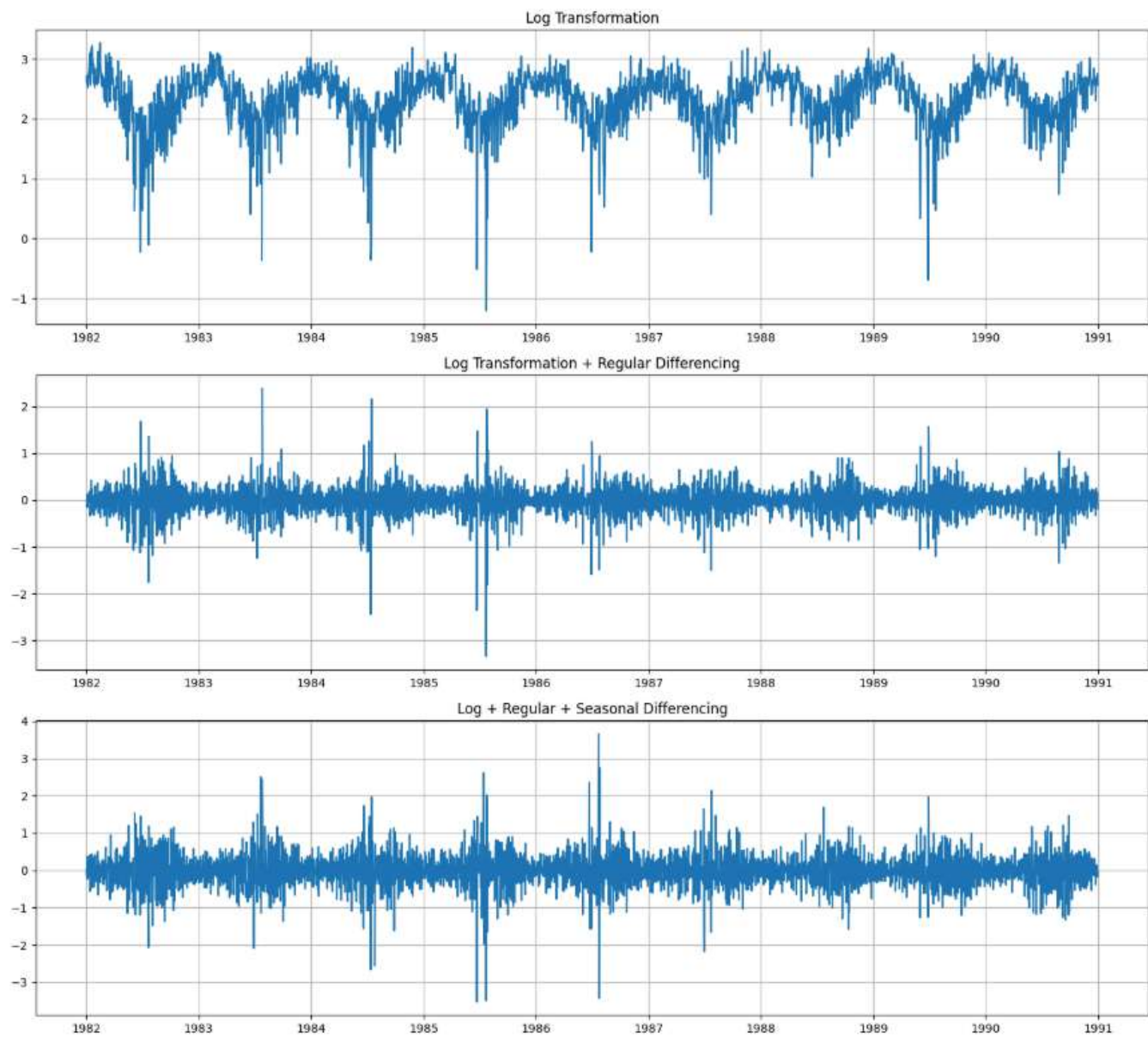
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose

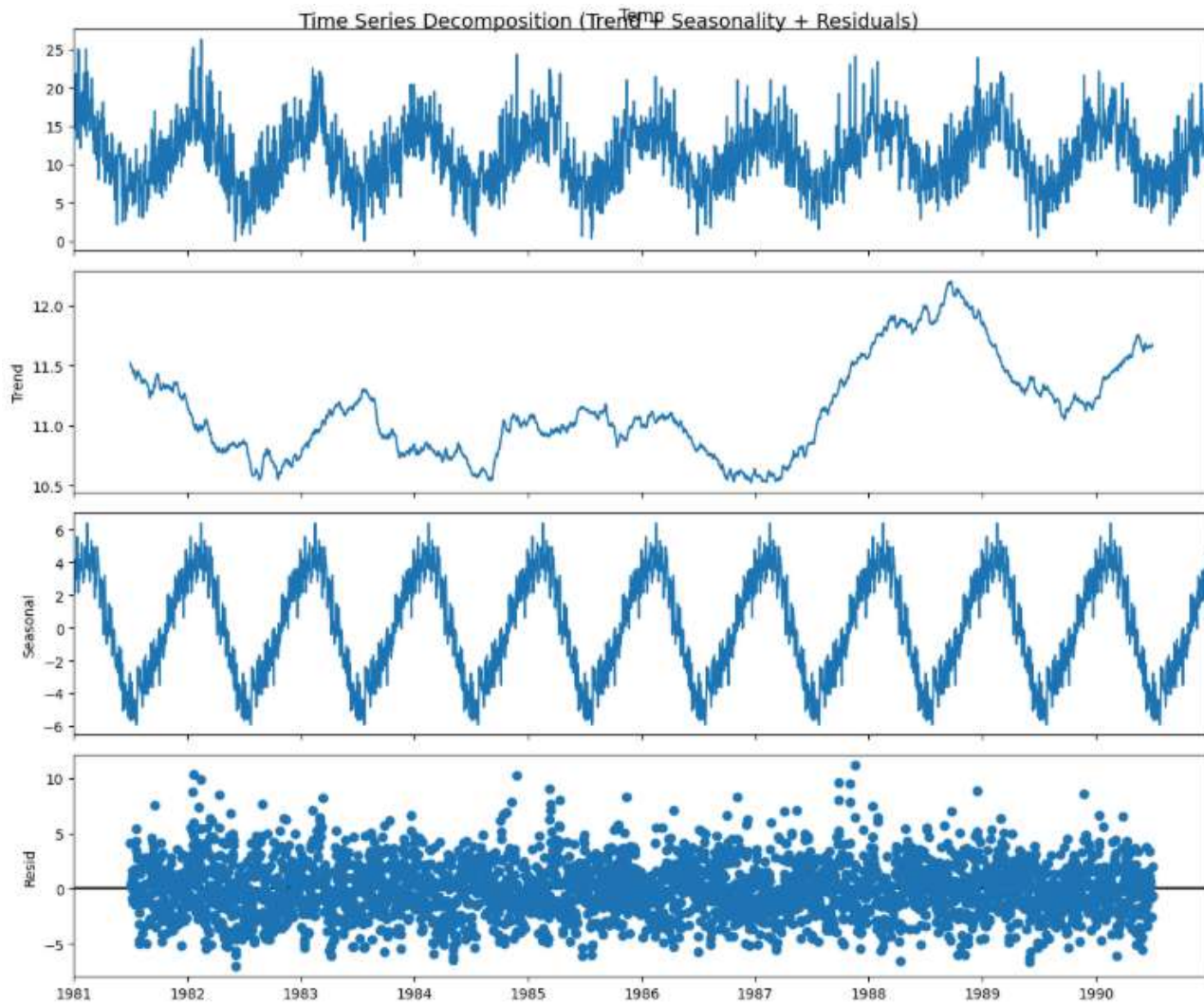
data = pd.read_csv("/kaggle/input/time-series/daily-minimum-temperatures-in-me.csv",
on_bad_lines='skip')
data.columns = ["Date", "Temp"]
data["Date"] = pd.to_datetime(data["Date"])
data["Temp"] = pd.to_numeric(data["Temp"], errors="coerce")
data = data.dropna()
data = data.set_index("Date")
decomposition = seasonal_decompose(data["Temp"], model="additive", period=365)

plt.rcParams["figure.figsize"] = (12, 10)
decomposition.plot()
plt.suptitle("Time Series Decomposition (Trend + Seasonality + Residuals)",
fontsize=1)
plt.show()
```

OUTPUT:







RESULT:

Thus we have created the python code for the conversion of non stationary to stationary data on international airline passenger data.