Exp no: 3 Develop a linear regression model for forecasting time series data

Date: 18/2/25

Objectives:

The objective of developing a linear regression model for forecasting time series data is to predict future values based on historical patterns. In this case, the AirPassengers time series dataset is used to forecast metrics such as customer growth, revenue, or production consumption. Linear regression is a simple yet powerful statistical method that models the relationship between a dependent variable (target) and one or more independent variables (features). By analyzing trends and seasonality in the dataset, the model aims to provide actionable insights for decision- making, such as optimizing content strategies or improving user engagement.

Background Scope:

Time series forecasting is a critical tool in industries like entertainment, finance, and retail, where understanding future trends is essential for strategic planning. AirPassenger, as a large industry, generates vast amounts of time series data, including daily no. of customers travelling and profit rates. Linear regression can be applied to this dataset to identify relationships between variables, such as the impact of passenger growth or the effect of marketing campaigns on passengers . While linear regression is a straightforward approach, it serves as a foundational model for more complex techniques like ARIMA or machine learning models. This project focuses on leveraging historical AirPassenger data to build a predictive model that can inform business strategies and enhance operational efficiency.

Implementation:

Step 1: Import libraries

```
import pandas as pd import numpy
as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split from
sklearn.linear_model import LinearRegression from sklearn.metrics import
mean_squared_error
```

Step 2: load the dataset and follow preprocessing steps

```
url = 'AirPassengers.csv'

data = pd.read_csv(url, parse_dates=['Month'], index_col='Month')

data.head()

plt.figure(figsize=(10, 6))

plt.plot(data, label='Monthly airline passengers')

plt.title('Airline Passengers Over Time')

plt.xlabel('Date')

plt.ylabel('Number of Passengers')

plt.legend()

# saving the plot as an image

plt.savefig('timeseries_plot.png')

plt.show()
```

Step 3 : split the data into train and test sets

```
data['Month'] = data.index
data['Month'] = data['Month'].map(pd.Timestamp.toordinal)

X = data[['Month']]
y = data['#Passengers']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)
```

Step 4: Train the model using linear regression

```
model = LinearRegression()
model.fit(X_train, y_train)
```

Step 5: Predictions and Calculate error rate

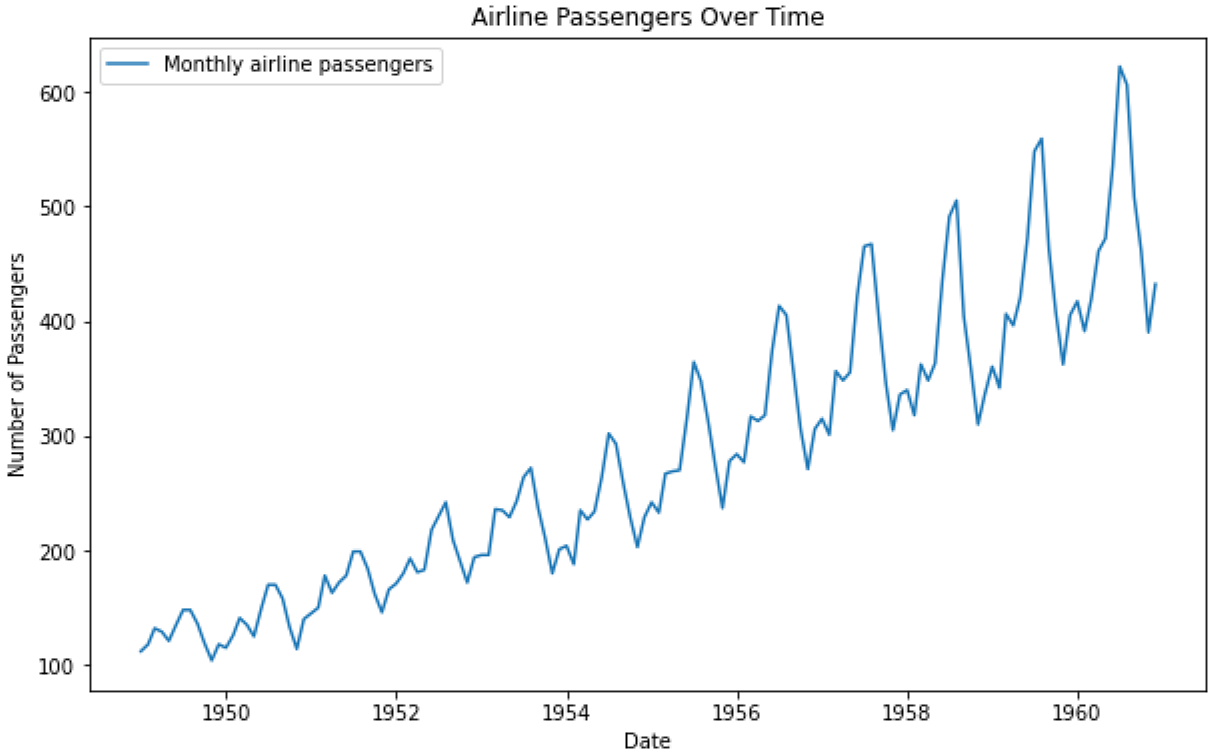
```
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')
```

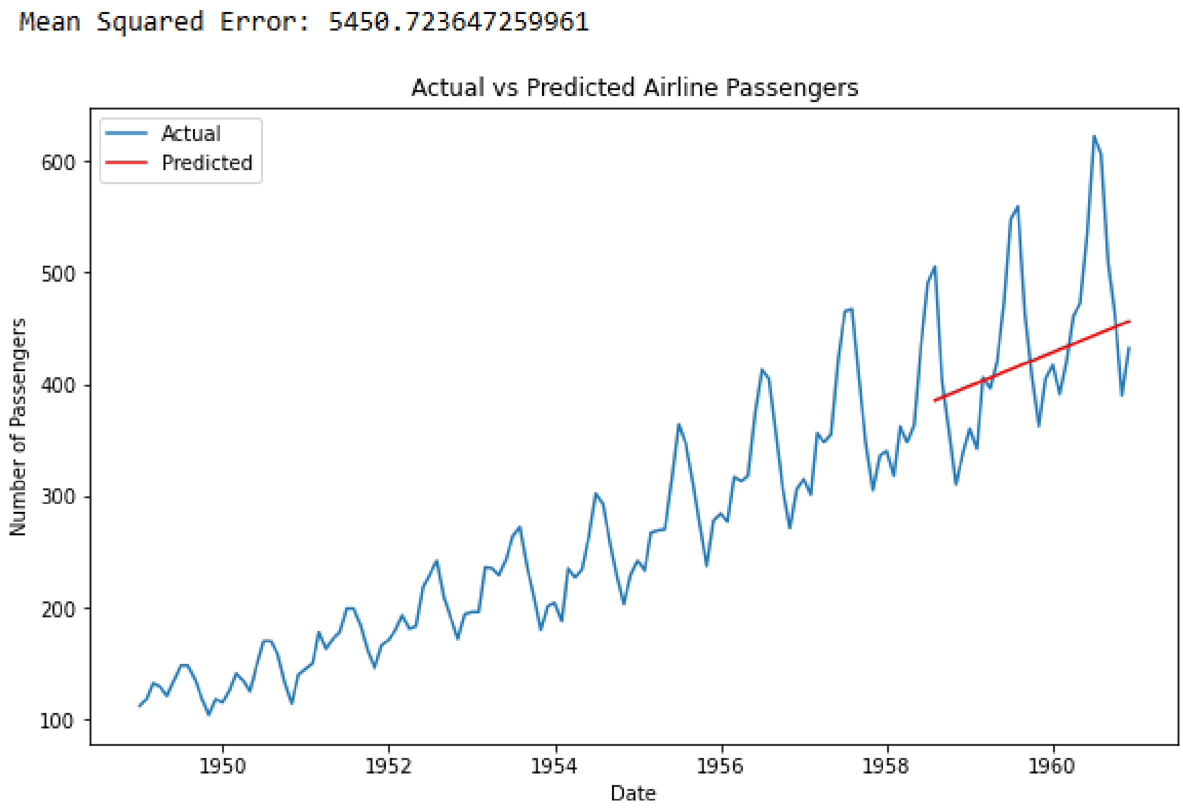
Step 6: Plot the graph

```
plt.figure(figsize=(10, 6))
plt.plot(data.index, data['#Passengers'], label='Actual')
plt.plot(X_test.index, y_pred, label='Predicted', color='red')
plt.title('Actual vs Predicted Airline Passengers')
plt.xlabel('Date')
plt.ylabel('Number of Passengers')
plt.legend()

plt.savefig('actual_vs_predicted.png')
plt.show()
```

Output :





```
Mean Squared Error: 5450.723647259961
```

Result: The program has been executed successfully using linear regression .