

Linear Regression for Time Series Forecasting

Linear Regression for Time Series Forecasting

1. Importing Necessary Libraries

In this section, we import essential Python libraries that facilitate data handling, visualization, and model training for time series forecasting.

```
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_absolute_error, mean_squared_error, r2_score
```

2. Loading the Dataset

The dataset containing time series values is loaded from a CSV file. The 'Date' column is converted to datetime format, and missing values are handled.

```
file_path = "/mnt/data/108,110.csv"
df = pd.read_csv(file_path)
df.columns = ['Date', 'Value']
df['Date'] = pd.to_datetime(df['Date'], format='%Y-%m')
df.dropna(inplace=True)
```

3. Feature Engineering

We create a lag feature 'Prev_Value' by shifting previous values to help the model learn from past trends.

```
df['Prev_Value'] = df['Value'].shift(1)
df.dropna(inplace=True)
```

4. Splitting Data into Training and Testing Sets

The dataset is split into training (80%) and testing (20%), ensuring the order of the time series is preserved.

```
X = df[['Prev_Value']]
y = df['Value']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)
```

5. Training the Linear Regression Model

Linear Regression for Time Series Forecasting

A Linear Regression model is trained using the training dataset to learn patterns in the data.

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

6. Making Predictions

The trained model is used to predict the values for the test set.

```
y_pred = model.predict(X_test)
```

7. Model Evaluation

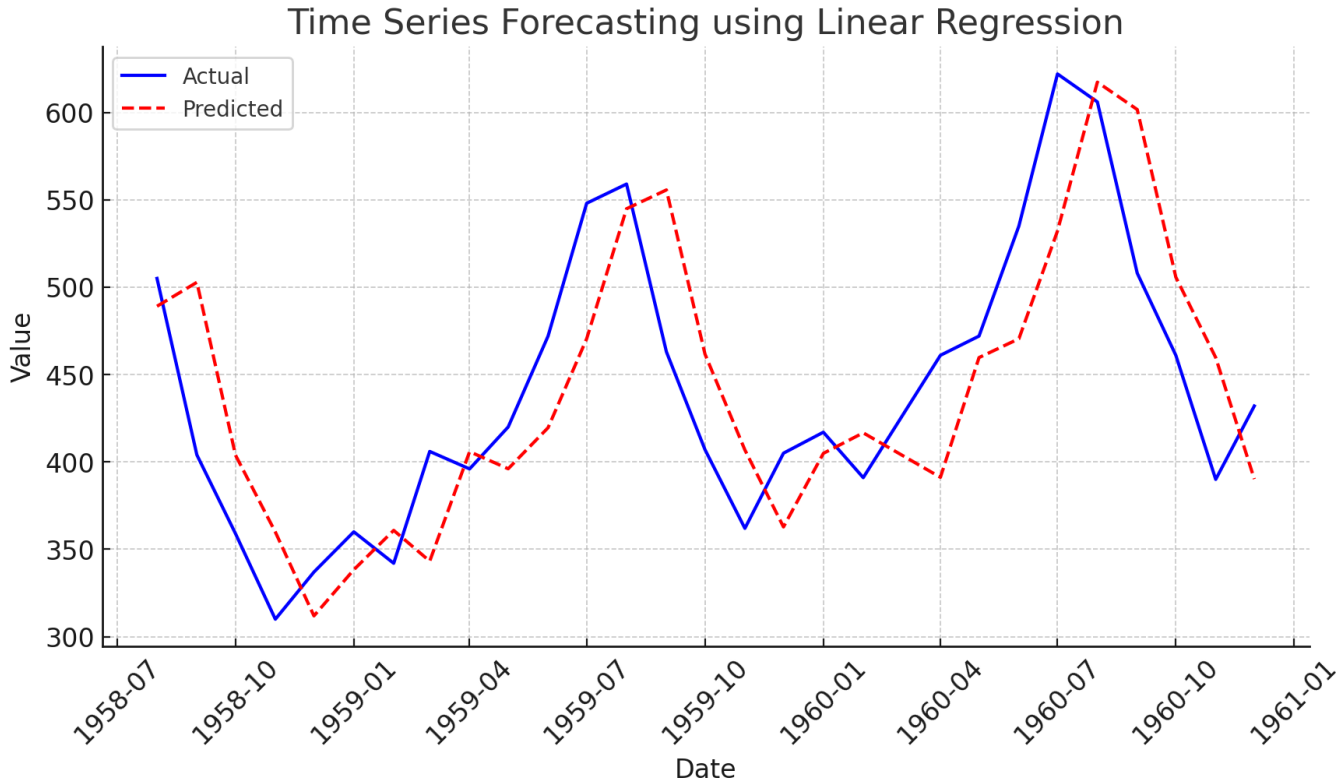
The model is evaluated using standard metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R² Score.

```
mae = mean_absolute_error(y_test, y_pred)  
mse = mean_squared_error(y_test, y_pred)  
rmse = np.sqrt(mse)  
r2 = r2_score(y_test, y_pred)  
  
print(f"MAE: {mae}")  
print(f"MSE: {mse}")  
print(f"RMSE: {rmse}")  
print(f"R2 Score: {r2}")
```

Linear Regression for Time Series Forecasting

8. Visualization of Results

The following plot shows the actual vs. predicted values from the model. The blue line represents actual values, while



Linear Regression for Time Series Forecasting

9. Result

Experiment Results:

Mean Absolute Error (MAE): 45.9099

Mean Squared Error (MSE): 2873.7871

Root Mean Squared Error (RMSE): 53.6077

R² Score: 0.5443

The experiment successfully applied Linear Regression for time series forecasting.