10.Develop vector auto regression model for multivariate time series data forecasting

AIM:

To implement vector auto regression model for multivariate time series data forecasting.

PROCEDURE:

1.Import the necessary libraries:

```
import pandas as pd

import matplotlib.pyplot as plt

from statsmodels.tsa.arima.model import ARIMA

from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
```

2.Load dataset:

```
df = pd.read_csv('PRICE_AND_DEMAND_201801_NSW1.csv')

df['SETTLEMENTDATE'] = pd.to_datetime(df['SETTLEMENTDATE'], format='%Y/%m/%d
%H:%M:%S')

df.set_index('SETTLEMENTDATE', inplace=True)
```

3. Normalize

```
scaler = StandardScaler()

data_scaled = scaler.fit_transform(data)

data_scaled = pd.DataFrame(data_scaled, columns=data.columns, index=data.index)
```

4. Train-test split

```
n_obs = 24 * 3  # forecast next 3 days (assuming hourly data)

train = data_scaled[:-n_obs]

test = data_scaled[-n_obs:]
```

5. Fit VAR model

```
model = VAR(train)

results = model.fit(maxlags=15, ic='aic’)
```

6. Forecast

```
forecast_df = pd.DataFrame(forecast, index=test.index, columns=test.columns)

# Inverse transform to get original scale
forecast_original = pd.DataFrame(scaler.inverse_transform(forecast_df),
                                 index=forecast_df.index,
                                 columns=forecast_df.columns)
test_original = pd.DataFrame(scaler.inverse_transform(test),
                             index=test.index,
                             columns=test.columns)
```

7. Plotting results

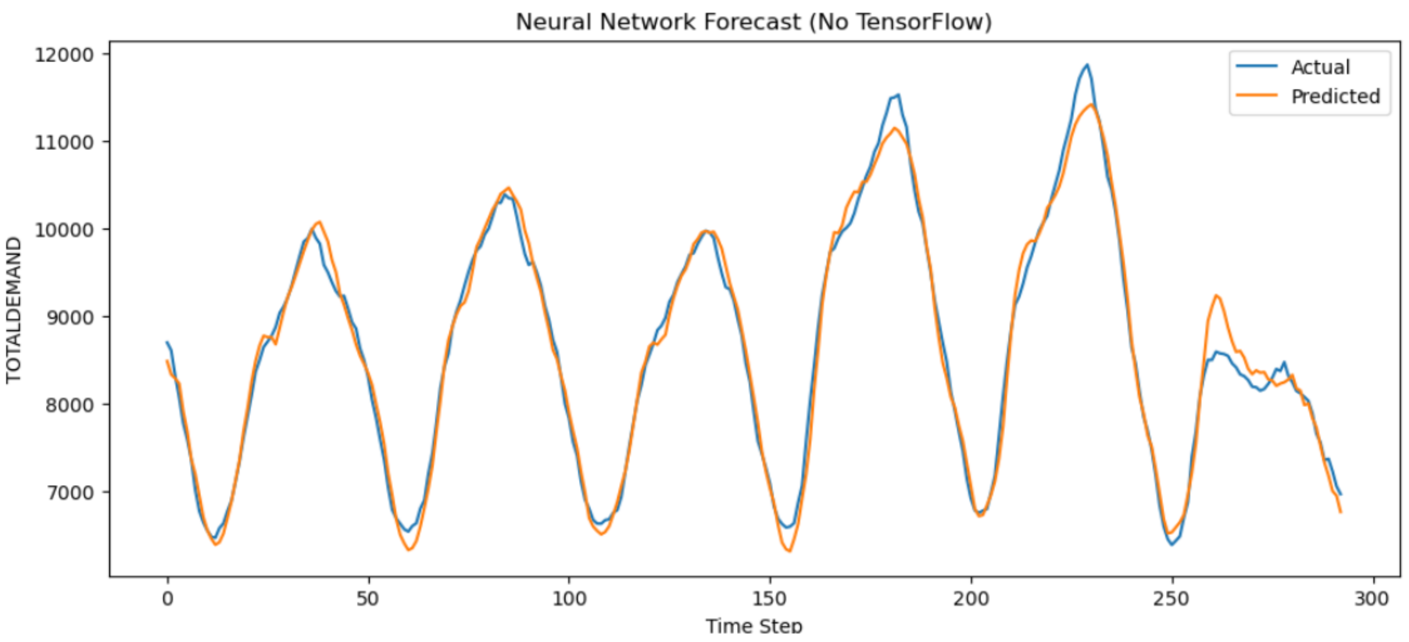
```
plt.figure(figsize=(14, 6))
plt.plot(test_original['RRP'], label='Actual RRP')
plt.plot(forecast_original['RRP'], label='Forecasted RRP')
plt.title("Forecast vs Actual - RRP")
plt.legend()
plt.show()

plt.figure(figsize=(14, 6))
plt.plot(test_original['TOTALDEMAND'], label='Actual TOTALDEMAND')
plt.plot(forecast_original['TOTALDEMAND'], label='Forecasted TOTALDEMAND')
plt.title("Forecast vs Actual - TOTALDEMAND")
plt.legend()
plt.show()
```

RESULT:

Thus the program has been executed and implemented successfully.

OUTPUT:



RESULT:

Thus the program has been executed and implemented successfully.