**Develop neural network-based time series forecasting model run in python code**

**EX.No:9**

**DATE:**

**AIM:**

To build an LSTM-based model for forecasting future values in a time series using historical data.

**ALGORITHM:**

1. Load and preprocess the time series data.
2. Normalize the data and create input sequences.
3. Split the data into training and testing sets.
4. Build and train the LSTM model.
5. Make predictions and evaluate performance.

**CODE:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.preprocessing import MinMaxScaler

from keras.models import Sequential

from keras.layers import LSTM, Dense

df = pd.read\_csv('Plant\_1\_Generation\_Data.csv')

df['DATE\_TIME'] = pd.to\_datetime(df['DATE\_TIME'], dayfirst=True)

df = df.set\_index('DATE\_TIME').resample('h').mean(numeric\_only=True)

data = df['DC\_POWER'].fillna(0).values.reshape(-1, 1)

scaler = MinMaxScaler()

data\_scaled = scaler.fit\_transform(data)

def create\_sequences(data, seq\_len):

X, y = [], []

for i in range(len(data) - seq\_len):

X.append(data[i:i+seq\_len])

y.append(data[i+seq\_len])

return np.array(X), np.array(y)

seq\_length = 24

X, y = create\_sequences(data\_scaled, seq\_length)

split = int(len(X) \* 0.8)

X\_train, y\_train = X[:split], y[:split]

X\_test, y\_test = X[split:], y[split:]

model = Sequential([

LSTM(50, return\_sequences=False, input\_shape=(seq\_length, 1)),

Dense(1)

])

model.compile(optimizer='adam', loss='mse')

model.fit(X\_train, y\_train, epochs=10, batch\_size=32, verbose=1)

predicted = model.predict(X\_test)

predicted\_actual = scaler.inverse\_transform(predicted)

y\_actual = scaler.inverse\_transform(y\_test)

plt.figure(figsize=(10,4))

plt.plot(y\_actual, label='Actual')

plt.plot(predicted\_actual, label='Predicted', alpha=0.7)

plt.legend()

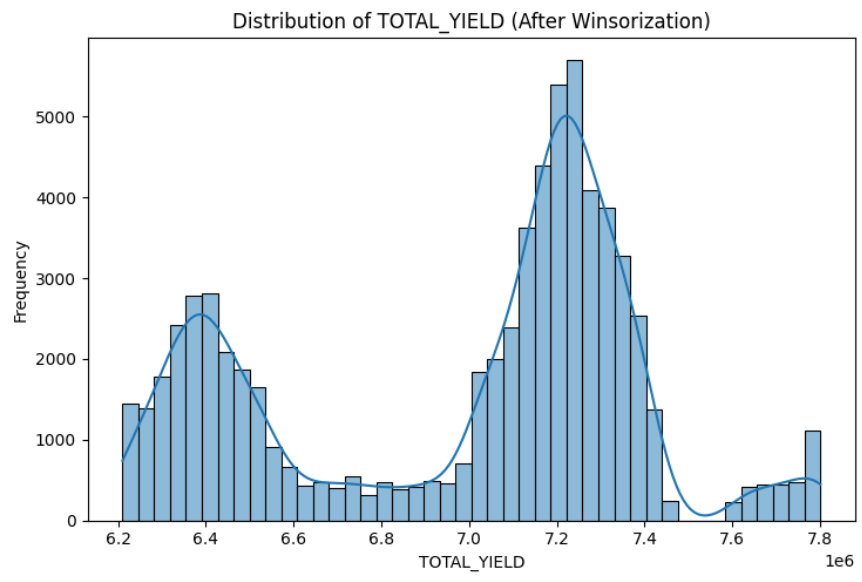
plt.title('LSTM Forecasting of DC Power')

plt.xlabel('Time Steps')

plt.ylabel('DC Power')

plt.show()

**OUTPUT:**

****

**\**

**RESULT:**

Thus the program has been completed and verified successfully.