

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.arima.model import ARIMA

# Load the dataset
df = pd.read_csv('covid_19_india.csv')

# Convert 'Date' column to datetime format
df['Date'] = pd.to_datetime(df['Date'], format='%d/%m/%y')

# Aggregate data for India
india_data = df.groupby('Date')['Confirmed'].sum().reset_index()
india_data.set_index('Date', inplace=True)

# Explore and Visualize the Data
plt.figure(figsize=(10, 6))
plt.plot(india_data['Confirmed'])
plt.title('COVID-19 Confirmed Cases in India')
plt.xlabel('Date')
plt.ylabel('Confirmed Cases')
plt.show()

# Stationarity Check and Differencing
result = adfuller(india_data['Confirmed'])
print('ADF Statistic:', result[0])
print('p-value:', result[1])
print('Critical Values:')
for key, value in result[4].items():
    print(f' {key}: {value}')

# First-order differencing
india_data['Confirmed_diff'] = india_data['Confirmed'].diff()
india_data['Confirmed_diff'].dropna(inplace=True)

# ACF and PACF plots
plot_acf(india_data['Confirmed_diff'], lags=20)
plt.xlabel('Lags')
plt.ylabel('ACF')
plt.show()

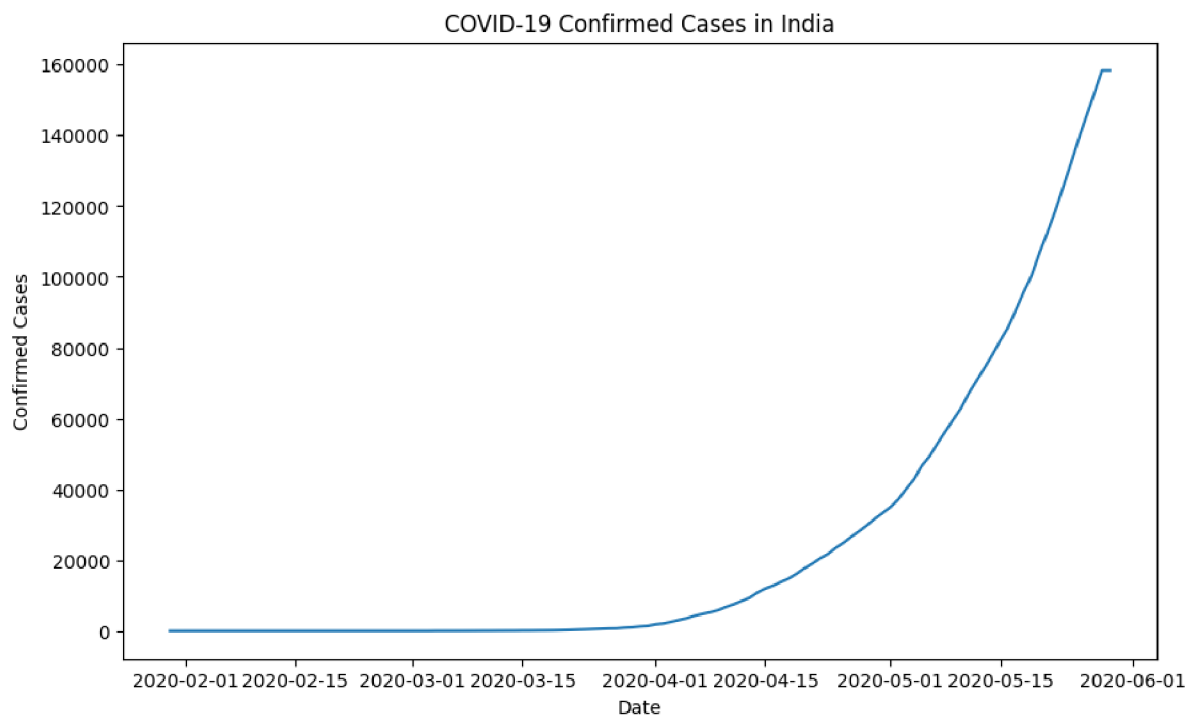
plot_pacf(india_data['Confirmed_diff'], lags=20)
plt.xlabel('Lags')
plt.ylabel('PACF')
plt.show()

# Fit ARIMA model with appropriate values for p, d, q
p = 1 # Placeholder, replace with actual value
d = 1 # Placeholder, replace with actual value
q = 1 # Placeholder, replace with actual value
model = ARIMA(india_data['Confirmed'], order=(p, d, q))
model_fit = model.fit()

# Print model summary
print(model_fit.summary())

# Make predictions
forecast = model_fit.forecast(steps=10)

# Plot actual vs predicted values
plt.figure(figsize=(10, 6))
plt.plot(india_data.index, india_data['Confirmed'], label='Actual')
plt.plot(pd.date_range(start=india_data.index[-1], periods=11, freq='D')[1:], forecast, label='Forecast', linestyle='--')
plt.title('ARIMA Forecast for COVID-19 Confirmed Cases in India')
plt.xlabel('Date')
plt.ylabel('Confirmed Cases')
plt.legend()
plt.show()
```



ADF Statistic: -3.843036195866195

p-value: 0.002496448399268513

Critical Values:

1%: -3.492995948509562

5%: -2.888954648057252

10%: -2.58139291903223

/usr/local/lib/python3.10/dist-packages/matplotlib/axes/_base.py:2503: UserWarning: Warning: converting a masked element to nan.
xys = np.asarray(xys)

