```python
# coding: utf-8
"""
Created on Sat Mar 17 06:38:50 2018

@author: Tawanda Vera

"""

# Import the libraries need
import pandas_datareader.data as web
import datetime
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
from statsmodels.tsa.stattools import adfuller
import statsmodels.graphics.tsaplots
from statsmodels.tsa.arima_model import _arma_predict_out_of_sample
plt.style.use('fivethirtyeight')


###############################################################################
# Q1. Implement the Augmented Dickey-Fuller Test for checking the existence
# of a unit root in the Case-Shiller Index series.
###############################################################################
#
# Set the data period from January 1975 to now.
# However, the FRED data only starts from 1987
#

start1 = datetime.datetime(1975, 1, 1)

end1 = datetime.date.today()


#
# Download the S&P/Case-Shiller U.S. National Home Price Index data series
# from FRED using the series' FRED ID.
#
df = web.DataReader("CSUSHPINSA", "fred", start1, end1)
df.plot(figsize=(15, 6))  # Plot the Time Series
plt.show()

#
# Convert the pandas dataframe to numpy ndarray and ploy the series
#
hpdata = df.CSUSHPINSA.values

#
# Perform the Augumented Dickey-Fuller test:
"""We use the Augmented Dickey Fuller (ADF) Test to see if the data presents\
 a unit root or not. We use adfuller function from Statsmodels module\
Null hypothesis: There is a unit root (nonstationarity)\
Alternative hypothesis: There is no unit root(stationarity or\
trend-stationarity)
"""
```

```python
print('Results of the Augumented Dickey-Fuller Test:')

dftest = adfuller(hpdata, autolag='AIC')
dfoutput = pd.Series(dftest, index=['Test statistic', 'p-value', '#Lags Used',
                                    'Number of Observations Used',
                                    'Critical Value',
                                    'Maximized information criterion'])
for key, value in dftest[4].items():
    dfoutput['Critical Value (%s)' % key] = value
print(dfoutput)

#
# Comments
#
print('\n The pvalue = 0.97 is the higher as compared to the critical values\
      of(1%)= -3.45, (5%) = -2.86  and (10%)= -2.57\
      Therefore we cannot reject the null hypothesis. As such,\
      The U.S. National Home Price Index data presents a unit root\
      therefore, it is nonstationary.\n')

##############################################################################
# Q2. Implement an ARIMA(p,d,q) model. Determine p, d, and q using
# the Box-Jenkins methodology. Discuss the results provided by the
# Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF).
##############################################################################
#
# Stationarize the series by logging & differencing
#

print("Assuming a Deterministic Trend:")

diff = np.diff(hpdata, axis=-1)  # First Log Difference of HPI data

fig = plt.figure(figsize=(15, 6))
ax1 = fig.add_subplot(311)
fig = statsmodels.graphics.tsaplots.plot_acf(diff, lags=12, ax=ax1)
ax2 = fig.add_subplot(312)
fig = statsmodels.graphics.tsaplots.plot_pacf(diff, lags=12, ax=ax2)
plt.show()

print("Assuming a Stochastic Trend:")
hpi_log = np.log(hpdata)  # create the log evolution of the HPI data
diff = np.diff(hpi_log, axis=-1)  # First Log Difference of HPI data


fig = plt.figure(figsize=(15, 6))

ax1 = fig.add_subplot(311)
fig = statsmodels.graphics.tsaplots.plot_acf(diff, lags=12, ax=ax1)
# If the ACF(k) differs significantly from zero, the serial dependence among
# the observations must be included in the ARIMA model.
#
ax2 = fig.add_subplot(312)
fig = statsmodels.graphics.tsaplots.plot_pacf(diff, lags=12, ax=ax2)
plt.show()
#
```

```
# If the PACF displays a sharp cutoff while the ACF decays more slowly
# (i.e., has significant spikes at higher lags),
# we say that the series displays an "AR signature"
# The lag at which the PACF cuts off is the indicated number of AR terms
#
# Discuss the results provided by the Autocorrelation Function (ACF) and
# Partial Autocorrelation Function (PACF).
#
print('The graphs plotted, show that ACF is decreasing very very slowly.\
      WHile, PACF cuts off at 2. Thus, we can conclude this to be a\
      ARIMA(1,0,0) = AR(1) [The lag at which the PACF cuts off is\
      the indicated number of AR terms]\n')

###############################################################################
# 3. Forecast the future evolution of the Case-Shiller Index (HPI)
# using the ARMA model. Provide one month, two-month and three-month forecasts.
# Test the model using in-sample forecasts.
###############################################################################
#
# Implement ARIMA(1,0,0) model:
#
arima100 = sm.tsa.ARIMA(diff, (1, 0, 0)).fit()
#
# Obtain ARIMA(1,0,0) parameters:
#
params = arima100.params
residuals = arima100.resid
p = arima100.k_ar
q = arima100.k_ma
k_exog = arima100.k_exog
k_trend = arima100.k_trend
steps = 4
#
# Obtain the Information Criterion (IC) values
#
arima100.aic  # Akaike Information Criterion (AIC)
arima100.bic  # Bayesian Information Criterion (BIC)

# Forecast the evolution of HPI using predict function
pred = _arma_predict_out_of_sample(params, steps, residuals, p, q, k_trend,
                                   k_exog, endog=hpi_log, exog=None,
                                   start=len(hpi_log))
#
# Provide one month, two-month and three-month forecasts
#
output = pd.Series(np.exp(pred), index=['One-month Forecast',
                   'Two-month Forecast', 'Three-month Forecast',
                   'Four-month Forecast'])

print('Results for one month, two-month and three-month forecasts:',
      output[0:4])

###############################################################################
# 4. Suggest exogenous variables that can be introduced,
# which can improve the forecasts.
###############################################################################
print('\n We can add other indices related to home price Index\
```

```
      to improve the prediction. These include the economic growth rate,\
      the consumer price index, nominal wages as a percentage of GDP,\
      the short-term interest rate, mortgage loans as a share of GDP\
      and population in the 15-64 cohort as a percentage of GDP.\n')

##############################################################################
```