stationarity_tests

June 2, 2024

Setting up to test each column in the DataFrame for a unit root using the Augmented Dickey Fuller test

[ ]:
```
import pandas as pd
from statsmodels.tsa.stattools import adfuller
```

[ ]:
```
file_path = '/Users/asger/Documents/GitHub/Deep_Learning_Techniques/Master/Data/
 ↪select_technical_all_nonstationary.csv'

df = pd.read_csv(file_path)

df.head()
```

[ ]:
```
   usd_eur_exchange   SMA_10    SMA_20   MACD_12        RSI  brent  eu_cpi  \
0           -0.0003  1.35163  1.341430  0.009364  40.107019  40.75     2.3
1           -0.0105  1.35065  1.341630  0.007291  39.923894  41.00     2.3
2           -0.0125  1.34761  1.341195  0.004746  33.903826  43.25     2.3
3            0.0047  1.34287  1.340385  0.001700  28.266007  43.28     2.3
4            0.0052  1.33771  1.339385 -0.000330  32.925766  44.71     2.3

   eu_mro_rate  eu_unemployment_rate  eu_yield_3m  …  eur_stoxx_vix  \
0          2.0                   8.9     2.055906  …        14.5103
1          2.0                   8.9     2.064858  …        14.4159
2          2.0                   8.9     2.060515  …        13.8016
3          2.0                   8.9     2.048016  …        13.5709
4          2.0                   8.9     2.034898  …        12.9192

   eur_stoxx    us_cpi  us_federal_fund_rate  us_sp500  us_sp500_vix  \
0    2971.12  0.052383                  2.25    118.83         13.98
1    2947.19  0.052383                  2.25    118.01         14.09
2    2966.24  0.052383                  2.25    118.61         13.58
3    2979.82  0.052383                  2.24    118.44         13.49
4    2977.21  0.052383                  2.26    119.00         13.23

   us_treasury_yield_3m  us_treasury_yield_10y  us_treasury_yield_30y  \
0                  2.33                   4.29                   4.91
1                  2.33                   4.29                   4.88
2                  2.31                   4.29                   4.89
```

```
3                 2.32                 4.29                 4.88
4                 2.36                 4.29                 4.86

   us_unemployment
0              5.4
1              5.4
2              5.4
3              5.4
4              5.4

[5 rows x 22 columns]
```

[ ]:
```
# Lets set up a for loop that rolls through each column except the date column, testing for a unit root using the statsmodel adfuller test.

def check_stationarity(df):
    for column in df.columns:
        result = adfuller(df[column], autolag = 'AIC')
        print(f'ADF Statistic: {result[0]}')
        print(f'p-value: {result[1]}')
        print(f'Critical Values:')
        for key, value in result[4].items():
            print(f'\t{key}: {value}')
        print(f'Number of lags used: {result[2]}')
        print(f'Is {column} stationary? {"Yes" if result[1] < 0.05 else "No"}')
        print('\n')

check_stationarity(df)
```

```
ADF Statistic: -67.93618243102965
p-value: 0.0
Critical Values:
        1%: -3.431725411436783
        5%: -2.8621477762930767
        10%: -2.5670935212314934
Number of lags used: 0
Is usd_eur_exchange stationary? Yes


ADF Statistic: -1.9947782871253386
p-value: 0.28886732134328374
Critical Values:
        1%: -3.431734731460428
        5%: -2.8621518937308172
        10%: -2.5670957131211334
Number of lags used: 32
Is SMA_10 stationary? No
```

```
ADF Statistic: -1.7820117926330934
p-value: 0.38942819732725265
Critical Values:
        1%: -3.4317344382982022
        5%: -2.8621517642166197
        10%: -2.567095644175108
Number of lags used: 31
Is SMA_20 stationary? No


ADF Statistic: -11.152376374910144
p-value: 2.9219852888432915e-20
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
        10%: -2.567093589281407
Number of lags used: 1
Is MACD_12 stationary? Yes


ADF Statistic: -14.979056368225612
p-value: 1.1694609181940813e-27
Critical Values:
        1%: -3.431725411436783
        5%: -2.8621477762930767
        10%: -2.5670935212314934
Number of lags used: 0
Is RSI stationary? Yes


ADF Statistic: -2.9451648482663852
p-value: 0.04034240200349569
Critical Values:
        1%: -3.4317338523459866
        5%: -2.8621515053526343
        10%: -2.567095506370587
Number of lags used: 29
Is brent stationary? Yes


ADF Statistic: -1.8158089777415216
p-value: 0.3726417005357108
Critical Values:
        1%: -3.431732681928294
        5%: -2.862150988281327
        10%: -2.5670952311111415
Number of lags used: 25
```

```
Is eu_cpi stationary? No


ADF Statistic: -0.025693643860809023
p-value: 0.9563549544205178
Critical Values:
        1%: -3.431734731460428
        5%: -2.8621518937308172
        10%: -2.5670957131211334
Number of lags used: 32
Is eu_mro_rate stationary? No


ADF Statistic: 0.0058708463248437195
p-value: 0.9590149163716398
Critical Values:
        1%: -3.4317320974615604
        5%: -2.8621507300734508
        10%: -2.5670950936559214
Number of lags used: 23
Is eu_unemployment_rate stationary? No


ADF Statistic: -0.7598017485805751
p-value: 0.8306458505982424
Critical Values:
        1%: -3.4317341452600814
        5%: -2.8621516347572373
        10%: -2.5670955752582656
Number of lags used: 30
Is eu_yield_3m stationary? No


ADF Statistic: -1.2575993313219294
p-value: 0.6483406913562688
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
        10%: -2.567093589281407
Number of lags used: 1
Is eu_yield_10y stationary? No


ADF Statistic: -1.5362827136390707
p-value: 0.5154925916075703
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
```

```
        10%: -2.567093589281407
Number of lags used: 1
Is eu_yield_30y stationary? No


ADF Statistic: -4.8347892335939076
p-value: 4.663704757017224e-05
Critical Values:
        1%: -3.431734731460428
        5%: -2.8621518937308172
        10%: -2.5670957131211334
Number of lags used: 32
Is eur_stoxx_vix stationary? Yes


ADF Statistic: -1.9735379138315896
p-value: 0.29829494831004266
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
        10%: -2.567093589281407
Number of lags used: 1
Is eur_stoxx stationary? No


ADF Statistic: -7.811096273211807
p-value: 7.066452802864785e-12
Critical Values:
        1%: -3.4317320974615604
        5%: -2.8621507300734508
        10%: -2.5670950936559214
Number of lags used: 23
Is us_cpi stationary? Yes


ADF Statistic: -0.5036862730942259
p-value: 0.8912951353671354
Critical Values:
        1%: -3.431734731460428
        5%: -2.8621518937308172
        10%: -2.5670957131211334
Number of lags used: 32
Is us_federal_fund_rate stationary? No


ADF Statistic: 0.7231178334247574
p-value: 0.9902881119293114
Critical Values:
```

```
        1%: -3.431729182523975
        5%: -2.8621494423007805
        10%: -2.567094408118962
Number of lags used: 13
Is us_sp500 stationary? No


ADF Statistic: -5.451447447515676
p-value: 2.6388211731922126e-06
Critical Values:
        1%: -3.431728601011308
        5%: -2.862149185397661
        10%: -2.567094271358372
Number of lags used: 11
Is us_sp500_vix stationary? Yes


ADF Statistic: -0.3394280846921287
p-value: 0.919731530330405
Critical Values:
        1%: -3.4317344382982022
        5%: -2.8621517642166197
        10%: -2.567095644175108
Number of lags used: 31
Is us_treasury_yield_3m stationary? No


ADF Statistic: -1.8388387884124693
p-value: 0.36135682137550673
Critical Values:
        1%: -3.431728310438826
        5%: -2.862149057027305
        10%: -2.5670942030213086
Number of lags used: 10
Is us_treasury_yield_10y stationary? No


ADF Statistic: -2.011658245403414
p-value: 0.28148487564421165
Critical Values:
        1%: -3.4317320974615604
        5%: -2.8621507300734508
        10%: -2.5670950936559214
Number of lags used: 23
Is us_treasury_yield_30y stationary? No


ADF Statistic: -2.6507660948407996
```

```
p-value: 0.0829521634367566
Critical Values:
        1%: -3.431732681928294
        5%: -2.862150988281327
        10%: -2.5670952311111415
Number of lags used: 25
Is us_unemployment stationary? No
```

[ ]:
```
non_stationary = df[['SMA_10', 'SMA_20', 'eu_cpi', 'eu_mro_rate',␣
 ↪'eu_unemployment_rate', 'eu_yield_3m','eu_yield_10y', 'eu_yield_30y',␣
 ↪'eur_stoxx', 'us_federal_fund_rate','us_sp500', 'us_treasury_yield_3m',␣
 ↪'us_treasury_yield_10y', 'us_treasury_yield_30y', 'us_unemployment']]

# Differencing the non stationary DataFrame

non_stationary_diff = non_stationary.diff().dropna()

check_stationarity(non_stationary_diff)
```

```
ADF Statistic: -10.014888199729953
p-value: 1.7393897806437205e-17
Critical Values:
        1%: -3.431734731460428
        5%: -2.8621518937308172
        10%: -2.5670957131211334
Number of lags used: 31
Is SMA_10 stationary? Yes


ADF Statistic: -10.19183710435024
p-value: 6.29248851996659e-18
Critical Values:
        1%: -3.4317344382982022
        5%: -2.8621517642166197
        10%: -2.567095644175108
Number of lags used: 30
Is SMA_20 stationary? Yes


ADF Statistic: -10.442247394999049
p-value: 1.5114209402190425e-18
Critical Values:
        1%: -3.431732681928294
        5%: -2.862150988281327
        10%: -2.5670952311111415
Number of lags used: 24
```

```
Is eu_cpi stationary? Yes


ADF Statistic: -8.232377310573472
p-value: 6.01464055323788e-13
Critical Values:
        1%: -3.4317350247468372
        5%: -2.8621520232998643
        10%: -2.5670957820963594
Number of lags used: 32
Is eu_mro_rate stationary? Yes


ADF Statistic: -10.600738460773202
p-value: 6.180711890184115e-19
Critical Values:
        1%: -3.4317320974615604
        5%: -2.8621507300734508
        10%: -2.5670950936559214
Number of lags used: 22
Is eu_unemployment_rate stationary? Yes


ADF Statistic: -8.062634177400591
p-value: 1.6274899981352503e-12
Critical Values:
        1%: -3.4317341452600814
        5%: -2.8621516347572373
        10%: -2.5670955752582656
Number of lags used: 29
Is eu_yield_3m stationary? Yes


ADF Statistic: -66.47102881805678
p-value: 0.0
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
        10%: -2.567093589281407
Number of lags used: 0
Is eu_yield_10y stationary? Yes


ADF Statistic: -65.49989980263621
p-value: 0.0
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
```

```
        10%: -2.567093589281407
Number of lags used: 0
Is eu_yield_30y stationary? Yes


ADF Statistic: -71.07915031571855
p-value: 0.0
Critical Values:
        1%: -3.4317257007880992
        5%: -2.862147904124065
        10%: -2.567093589281407
Number of lags used: 0
Is eur_stoxx stationary? Yes


ADF Statistic: -8.098063884700268
p-value: 1.3225088970103313e-12
Critical Values:
        1%: -3.4317350247468372
        5%: -2.8621520232998643
        10%: -2.5670957820963594
Number of lags used: 32
Is us_federal_fund_rate stationary? Yes


ADF Statistic: -19.408416335790832
p-value: 0.0
Critical Values:
        1%: -3.431729182523975
        5%: -2.8621494423007805
        10%: -2.567094408118962
Number of lags used: 12
Is us_sp500 stationary? Yes


ADF Statistic: -9.13553138122683
p-value: 2.942787169751023e-15
Critical Values:
        1%: -3.4317350247468372
        5%: -2.8621520232998643
        10%: -2.5670957820963594
Number of lags used: 32
Is us_treasury_yield_3m stationary? Yes


ADF Statistic: -20.920708137384313
p-value: 0.0
Critical Values:
```

```
        1%: -3.431728310438826
        5%: -2.862149057027305
        10%: -2.5670942030213086
Number of lags used: 9
Is us_treasury_yield_10y stationary? Yes


ADF Statistic: -13.793622857261509
p-value: 8.83227330305889e-26
Critical Values:
        1%: -3.4317320974615604
        5%: -2.8621507300734508
        10%: -2.5670950936559214
Number of lags used: 22
Is us_treasury_yield_30y stationary? Yes


ADF Statistic: -13.16488983307847
p-value: 1.2849879263029858e-24
Critical Values:
        1%: -3.431732681928294
        5%: -2.862150988281327
        10%: -2.5670952311111415
Number of lags used: 24
Is us_unemployment stationary? Yes
```

[ ]:
```
# Dropping the non stationary columns from the original dataframe df
df = df.drop(columns = ['SMA_10', 'SMA_20', 'eu_cpi', 'eu_mro_rate',␣
 ↪'eu_unemployment_rate', 'eu_yield_3m','eu_yield_10y', 'eu_yield_30y',␣
 ↪'eur_stoxx', 'us_federal_fund_rate','us_sp500', 'us_treasury_yield_3m',␣
 ↪'us_treasury_yield_10y', 'us_treasury_yield_30y', 'us_unemployment'])

# combining the stationary df with the differenced non stationary df
df = pd.concat([df, non_stationary_diff], axis = 1)

# Removing the first observation
df = df.iloc[2:]

df.to_csv('/Users/asger/Documents/GitHub/Deep_Learning_Techniques/Master/Data/
 ↪final_dataset.csv', index = False)
```

[ ]:
```
# Creating data for second iteration

df_2 = df
```

[ ]:
```
# Removing the following columns; brent, eu_yield_3m, eu_yield_10y,␣
 ↪eu_yield_30y, us_treasury_yield_3m, us_treasury_yield_10y,␣
 ↪us_treasury_yield_30y, us_sp500, eur_stoxx

df_2 = df_2.drop(columns = ['brent', 'eu_yield_3m', 'eu_yield_10y',␣
 ↪'eu_yield_30y', 'us_treasury_yield_3m', 'us_treasury_yield_10y',␣
 ↪'us_treasury_yield_30y', 'us_sp500', 'eur_stoxx'])
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4756 entries, 2 to 4757
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   usd_eur_exchange      4756 non-null   float64
 1   MACD_12               4756 non-null   float64
 2   RSI                   4756 non-null   float64
 3   eur_stoxx_vix         4756 non-null   float64
 4   us_cpi                4756 non-null   float64
 5   us_sp500_vix          4756 non-null   float64
 6   SMA_10                4756 non-null   float64
 7   SMA_20                4756 non-null   float64
 8   eu_cpi                4756 non-null   float64
 9   eu_mro_rate           4756 non-null   float64
 10  eu_unemployment_rate  4756 non-null   float64
 11  us_federal_fund_rate  4756 non-null   float64
 12  us_unemployment       4756 non-null   float64
dtypes: float64(13)
memory usage: 483.2 KB
```

[ ]:
```
df_2.to_csv('/Users/asger/Documents/GitHub/Deep_Learning_Techniques/Master/Data/
 ↪final_dataset_iteration2.csv', index = False)
```