Aroon_Oscillator

September 29, 2021

1 Aroon Oscillator

https://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:aroon_oscillator

```python
[1]: import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt

     import warnings
     warnings.filterwarnings("ignore")

     # fix_yahoo_finance is used to fetch data
     import fix_yahoo_finance as yf
     yf.pdr_override()
```

```python
[2]: # input
     symbol = 'AAPL'
     start = '2017-01-01'
     end = '2019-01-01'

     # Read data
     df = yf.download(symbol,start,end)

     # View Columns
     df.head()
```

```
[*********************100%***********************]  1 of 1 downloaded
```

```
[2]:                  Open        High         Low       Close    Adj Close  \
     Date
     2017-01-03  115.800003  116.330002  114.760002  116.150002  112.140007
     2017-01-04  115.849998  116.510002  115.750000  116.019997  112.014503
     2017-01-05  115.919998  116.860001  115.809998  116.610001  112.584129
     2017-01-06  116.779999  118.160004  116.470001  117.910004  113.839249
     2017-01-09  117.949997  119.430000  117.940002  118.989998  114.881950

                    Volume
     Date
```

```
2017-01-03  28781900
2017-01-04  21118100
2017-01-05  22193600
2017-01-06  31751900
2017-01-09  33561900
```

[3]:
```python
n = 25
high_max = lambda xs: np.argmax(xs[::-1])
low_min = lambda xs: np.argmin(xs[::-1])

df['Days since last High'] = df['High'].
 ↪rolling(center=False,min_periods=0,window=n).apply(func=high_max).astype(int)

df['Days since last Low'] = df['Low'].
 ↪rolling(center=False,min_periods=0,window=n).apply(func=low_min).astype(int)

df['Aroon_Up'] = ((25-df['Days since last High'])/25) * 100
df['Aroon_Down'] = ((25-df['Days since last Low'])/25) * 100

df['Aroon_Oscillator'] = df['Aroon_Up'] - df['Aroon_Down']
```

[4]:
```python
df = df.drop(['Days since last High','Days since last Low', 'Aroon_Up',
 ↪'Aroon_Down'],axis=1)
```

[6]:
```python
fig = plt.figure(figsize=(14,10))
ax1 = plt.subplot(2, 1, 1)
ax1.plot(df['Adj Close'])
ax1.set_title('Stock '+ symbol +' Closing Price')
ax1.set_ylabel('Price')
ax1.legend(loc='best')

ax2 = plt.subplot(2, 1, 2)
ax2.plot(df['Aroon_Oscillator'], label='Aroon Oscillator', color='g')
ax2.axhline(y=0, color='darkblue')
ax2.grid()
ax2.legend(loc='best')
ax2.set_ylabel('Aroon Oscillator')
ax2.set_xlabel('Date')
```
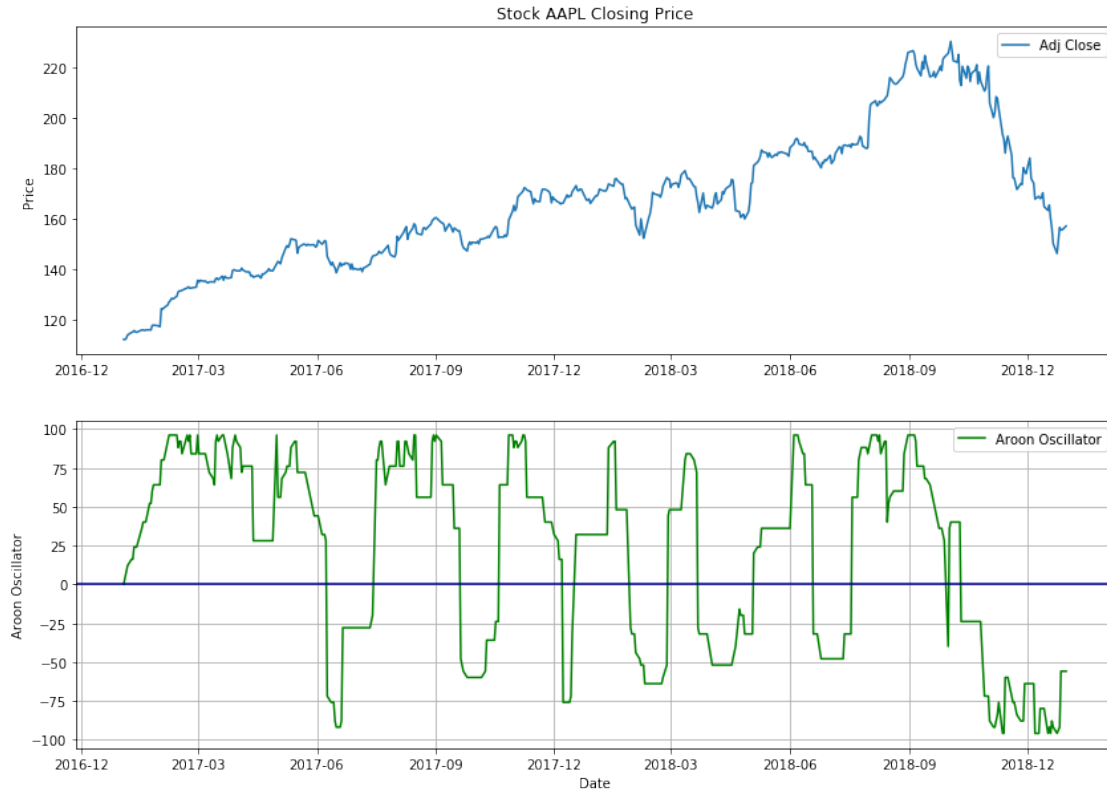
[6]: Text(0.5,0,'Date')

Stock AAPL Closing Price

```
[8]: fig = plt.figure(figsize=(14,10))
     ax1 = plt.subplot(2, 1, 1)
     ax1.plot(df['Adj Close'])
     ax1.set_title('Stock '+ symbol +' Closing Price')
     ax1.set_ylabel('Price')
     ax1.legend(loc='best')

     df['Positive'] = df['Aroon_Oscillator'] > 0
     ax2 = plt.subplot(2, 1, 2)
     ax2.bar(df.index, df['Aroon_Oscillator'], color=df.Positive.map({True: 'g',␣
      ↪False: 'r'}))
     ax2.axhline(y=0, color='red')
     ax2.grid()
     ax2.set_ylabel('Aroon Oscillator')
     ax2.set_xlabel('Date')
```

```
[8]: Text(0.5,0,'Date')
```

Stock AAPL Closing Price
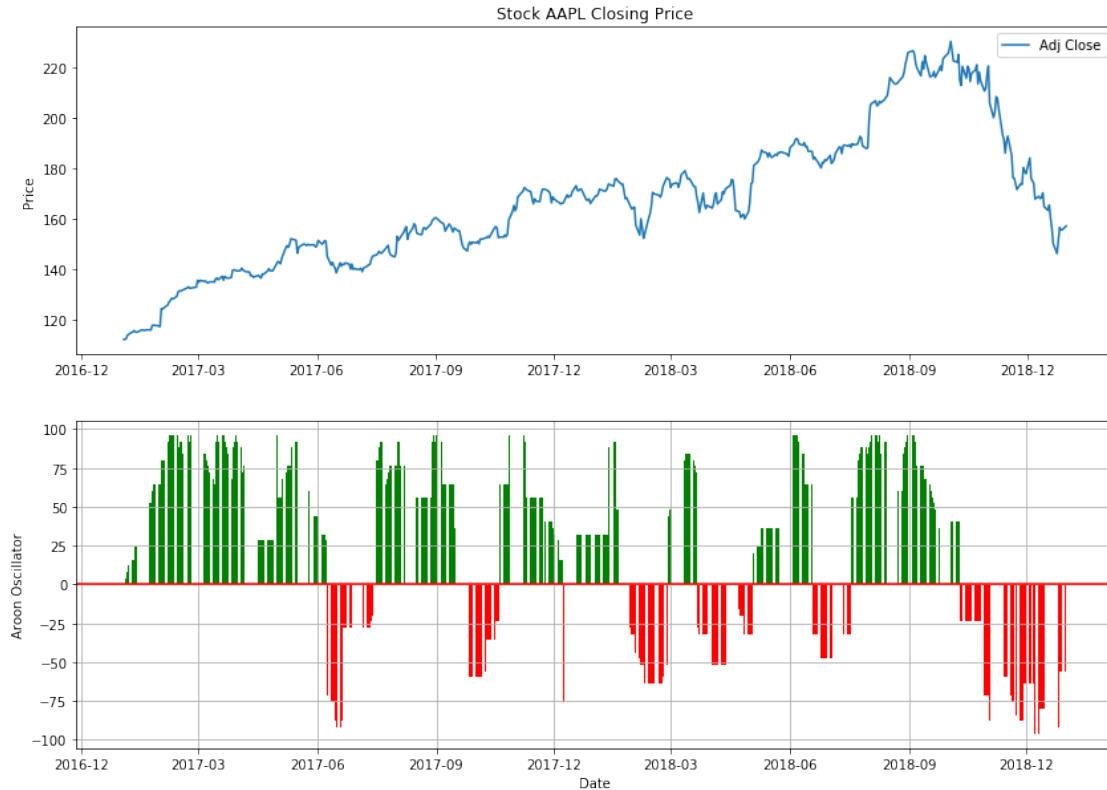
1.1 Candlestick with Aroon Oscillator

```python
from matplotlib import dates as mdates
import datetime as dt

dfc = df.copy()
dfc['VolumePositive'] = dfc['Open'] < dfc['Adj Close']
#dfc = dfc.dropna()
dfc = dfc.reset_index()
dfc['Date'] = mdates.date2num(dfc['Date'].astype(dt.date))
dfc.head()
```

```
[9]:       Date        Open        High         Low       Close    Adj Close  \
     0  736332.0  115.800003  116.330002  114.760002  116.150002  112.140007
     1  736333.0  115.849998  116.510002  115.750000  116.019997  112.014503
     2  736334.0  115.919998  116.860001  115.809998  116.610001  112.584129
     3  736335.0  116.779999  118.160004  116.470001  117.910004  113.839249
     4  736338.0  117.949997  119.430000  117.940002  118.989998  114.881950

          Volume  Aroon_Oscillator  Positive  VolumePositive
     0  28781900               0.0     False           False
     1  21118100               4.0      True           False
```

```
2   22193600               8.0      True            False
3   31751900              12.0      True            False
4   33561900              16.0      True            False
```

```python
[10]: from mpl_finance import candlestick_ohlc

      fig = plt.figure(figsize=(14,10))
      ax1 = plt.subplot(2, 1, 1)
      candlestick_ohlc(ax1,dfc.values, width=0.5, colorup='g', colordown='r', alpha=1.
       ↪0)
      ax1.xaxis_date()
      ax1.xaxis.set_major_formatter(mdates.DateFormatter('%d-%m-%Y'))
      ax1.grid(True, which='both')
      ax1.minorticks_on()
      ax1v = ax1.twinx()
      colors = dfc.VolumePositive.map({True: 'g', False: 'r'})
      ax1v.bar(dfc.Date, dfc['Volume'], color=colors, alpha=0.4)
      ax1v.axes.yaxis.set_ticklabels([])
      ax1v.set_ylim(0, 3*df.Volume.max())
      ax1.set_title('Stock '+ symbol +' Closing Price')
      ax1.set_ylabel('Price')

      df['Positive'] = df['Aroon_Oscillator'] > 0
      ax2 = plt.subplot(2, 1, 2)
      ax2.bar(df.index, df['Aroon_Oscillator'], color=df.Positive.map({True: 'g',␣
       ↪False: 'r'}))
      ax2.axhline(y=0, color='red')
      ax2.grid()
      ax2.set_ylabel('Aroon Oscillator')
      ax2.set_xlabel('Date')
```

```
[10]: Text(0.5,0,'Date')
```

Stock AAPL Closing Price