

Absolute_Return_Indicator

September 29, 2021

1 Absolute Return Indicator

<https://www.investopedia.com/terms/a/absolutereturn.asp>

```
[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()
```

```
[2]: # input
symbol = 'AAPL'
start = '2018-01-01'
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 | | | | | | |
| 2018-01-02 | 170.160004 | 172.300003 | 169.259995 | 172.259995 | 168.339050 | |
| 2018-01-03 | 172.529999 | 174.550003 | 171.960007 | 172.229996 | 168.309738 | |
| 2018-01-04 | 172.539993 | 173.470001 | 172.080002 | 173.029999 | 169.091522 | |
| 2018-01-05 | 173.440002 | 175.369995 | 173.050003 | 175.000000 | 171.016678 | |
| 2018-01-08 | 174.350006 | 175.610001 | 173.929993 | 174.350006 | 170.381485 | |

| | Volume |
|------|--------|
| Date | |

```

2018-01-02 25555900
2018-01-03 29517900
2018-01-04 22434600
2018-01-05 23660000
2018-01-08 20567800

```

```
[3]: df['Absolute_Return'] = 100 * (df['Adj Close'] - df['Adj Close'].shift(1))/
      ↪df['Adj Close'].shift(1)
```

```
[4]: df.head(20)
```

```
[4]:
```

| | Open | High | Low | Close | Adj Close \ |
|------------|------------|------------|------------|------------|-------------|
| Date | | | | | |
| 2018-01-02 | 170.160004 | 172.300003 | 169.259995 | 172.259995 | 168.339050 |
| 2018-01-03 | 172.529999 | 174.550003 | 171.960007 | 172.229996 | 168.309738 |
| 2018-01-04 | 172.539993 | 173.470001 | 172.080002 | 173.029999 | 169.091522 |
| 2018-01-05 | 173.440002 | 175.369995 | 173.050003 | 175.000000 | 171.016678 |
| 2018-01-08 | 174.350006 | 175.610001 | 173.929993 | 174.350006 | 170.381485 |
| 2018-01-09 | 174.550003 | 175.059998 | 173.410004 | 174.330002 | 170.361954 |
| 2018-01-10 | 173.160004 | 174.300003 | 173.000000 | 174.289993 | 170.322845 |
| 2018-01-11 | 174.589996 | 175.490005 | 174.490005 | 175.279999 | 171.290329 |
| 2018-01-12 | 176.179993 | 177.360001 | 175.649994 | 177.089996 | 173.059113 |
| 2018-01-16 | 177.899994 | 179.389999 | 176.139999 | 176.190002 | 172.179611 |
| 2018-01-17 | 176.149994 | 179.250000 | 175.070007 | 179.100006 | 175.023361 |
| 2018-01-18 | 179.369995 | 180.100006 | 178.250000 | 179.259995 | 175.179718 |
| 2018-01-19 | 178.610001 | 179.580002 | 177.410004 | 178.460007 | 174.397949 |
| 2018-01-22 | 177.300003 | 177.779999 | 176.600006 | 177.000000 | 172.971176 |
| 2018-01-23 | 177.300003 | 179.440002 | 176.820007 | 177.039993 | 173.010254 |
| 2018-01-24 | 177.250000 | 177.300003 | 173.199997 | 174.220001 | 170.254440 |
| 2018-01-25 | 174.509995 | 174.949997 | 170.529999 | 171.110001 | 167.215210 |
| 2018-01-26 | 172.000000 | 172.000000 | 170.059998 | 171.509995 | 167.606140 |
| 2018-01-29 | 170.160004 | 170.160004 | 167.070007 | 167.960007 | 164.136932 |
| 2018-01-30 | 165.529999 | 167.369995 | 164.699997 | 166.970001 | 163.169464 |

| | Volume | Absolute_Return |
|------------|----------|-----------------|
| Date | | |
| 2018-01-02 | 25555900 | NaN |
| 2018-01-03 | 29517900 | -0.017412 |
| 2018-01-04 | 22434600 | 0.464491 |
| 2018-01-05 | 23660000 | 1.138529 |
| 2018-01-08 | 20567800 | -0.371422 |
| 2018-01-09 | 21584000 | -0.011463 |
| 2018-01-10 | 23959900 | -0.022956 |
| 2018-01-11 | 18667700 | 0.568029 |
| 2018-01-12 | 25418100 | 1.032623 |
| 2018-01-16 | 29565900 | -0.508209 |
| 2018-01-17 | 34386800 | 1.651618 |

| | | |
|------------|----------|-----------|
| 2018-01-18 | 31193400 | 0.089335 |
| 2018-01-19 | 32425100 | -0.446267 |
| 2018-01-22 | 27108600 | -0.818113 |
| 2018-01-23 | 32689100 | 0.022592 |
| 2018-01-24 | 51105100 | -1.592862 |
| 2018-01-25 | 41529000 | -1.785111 |
| 2018-01-26 | 39143000 | 0.233789 |
| 2018-01-29 | 50640400 | -2.069857 |
| 2018-01-30 | 46048200 | -0.589427 |

```
[5]: 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')

ax2 = plt.subplot(2, 1, 2)
ax2.plot(df['Absolute_Return'] , label='Absolute Return', color='red')
#ax2.axhline(y=0, color='blue', linestyle='--')
#ax2.axhline(y=0.5, color='darkblue')
#ax2.axhline(y=-0.5, color='darkblue')
ax2.grid()
ax2.set_ylabel('Absolute Return')
ax2.set_xlabel('Date')
ax2.legend(loc='best')
```

```
[5]: <matplotlib.legend.Legend at 0x1de01b253c8>
```



1.1 Candlestick with Triple Exponential Weighted Moving Average

```
[6]: 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'] = pd.to_datetime(dfc['Date'])
dfc['Date'] = dfc['Date'].apply(mdates.date2num)
dfc.head()
```

```
[6]:      Date      Open      High      Low      Close  Adj Close  \
0  736696.0  170.160004  172.300003  169.259995  172.259995  168.339050
1  736697.0  172.529999  174.550003  171.960007  172.229996  168.309738
2  736698.0  172.539993  173.470001  172.080002  173.029999  169.091522
3  736699.0  173.440002  175.369995  173.050003  175.000000  171.016678
4  736702.0  174.350006  175.610001  173.929993  174.350006  170.381485

      Volume  Absolute_Return  VolumePositive
```

| | | | |
|---|----------|-----------|-------|
| 0 | 25555900 | NaN | False |
| 1 | 29517900 | -0.017412 | False |
| 2 | 22434600 | 0.464491 | False |
| 3 | 23660000 | 1.138529 | False |
| 4 | 20567800 | -0.371422 | False |

```
[7]: 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')

ax2 = plt.subplot(2, 1, 2)
ax2.plot(df['Absolute_Return'] , label='Absolute Return', color='red')
#ax2.axhline(y=0, color='blue', linestyle='--')
#ax2.axhline(y=0.5, color='darkblue')
#ax2.axhline(y=-0.5, color='darkblue')
ax2.grid()
ax2.set_ylabel('Absolute Return')
ax2.set_xlabel('Date')
ax2.legend(loc='best')
```

```
[7]: <matplotlib.legend.Legend at 0x1de0357eba8>
```

