

TWAP

September 29, 2021

1 Time Weighted Average Price (TWAP)

https://en.wikipedia.org/wiki/Time-weighted_average_price

```
[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]: TP = (df[['Open', 'High', 'Low', 'Adj Close']].sum(axis=1))/4
```

```
[4]: n=10
df['TWAP'] = TP.rolling(n).mean()
```

```
[5]: df.head(15)
```

```
[5]:
```

	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	

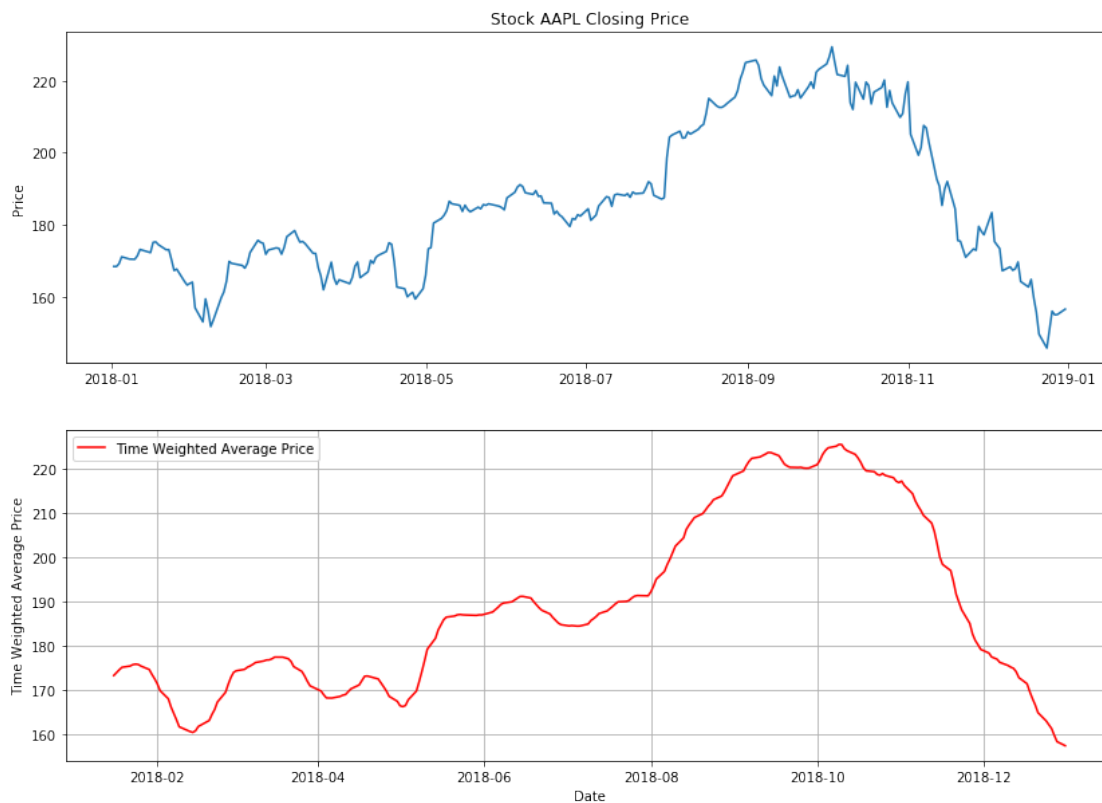
	Volume	TWAP
Date		
2018-01-02	25555900	NaN
2018-01-03	29517900	NaN
2018-01-04	22434600	NaN
2018-01-05	23660000	NaN
2018-01-08	20567800	NaN
2018-01-09	21584000	NaN
2018-01-10	23959900	NaN
2018-01-11	18667700	NaN
2018-01-12	25418100	NaN
2018-01-16	29565900	173.240558
2018-01-17	34386800	173.876416
2018-01-18	31193400	174.515165
2018-01-19	32425100	175.085576
2018-01-22	27108600	175.379939

2018-01-23 32689100 175.687408

```
[6]: 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['TWAP'], label='Time Weighted Average Price', 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('Time Weighted Average Price')
ax2.set_xlabel('Date')
ax2.legend(loc='best')
```

[6]: <matplotlib.legend.Legend at 0x1e8bd550710>



1.1 Candlestick with Time Weighted Average Price (TWAP)

```
[7]: 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()
```

```
[7]:
```

	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	TWAP	VolumePositive
0	25555900	NaN	False
1	29517900	NaN	False
2	22434600	NaN	False
3	23660000	NaN	False
4	20567800	NaN	False

```
[8]: 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['TWAP'], label='Time Weighted Average Price', 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('Time Weighted Average Price')
ax2.set_xlabel('Date')
ax2.legend(loc='best')

```

[8]: <matplotlib.legend.Legend at 0x1e8bef93a20>

