

Keltners_Channels

September 29, 2021

1 Keltner Channels

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

```
[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 = '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	

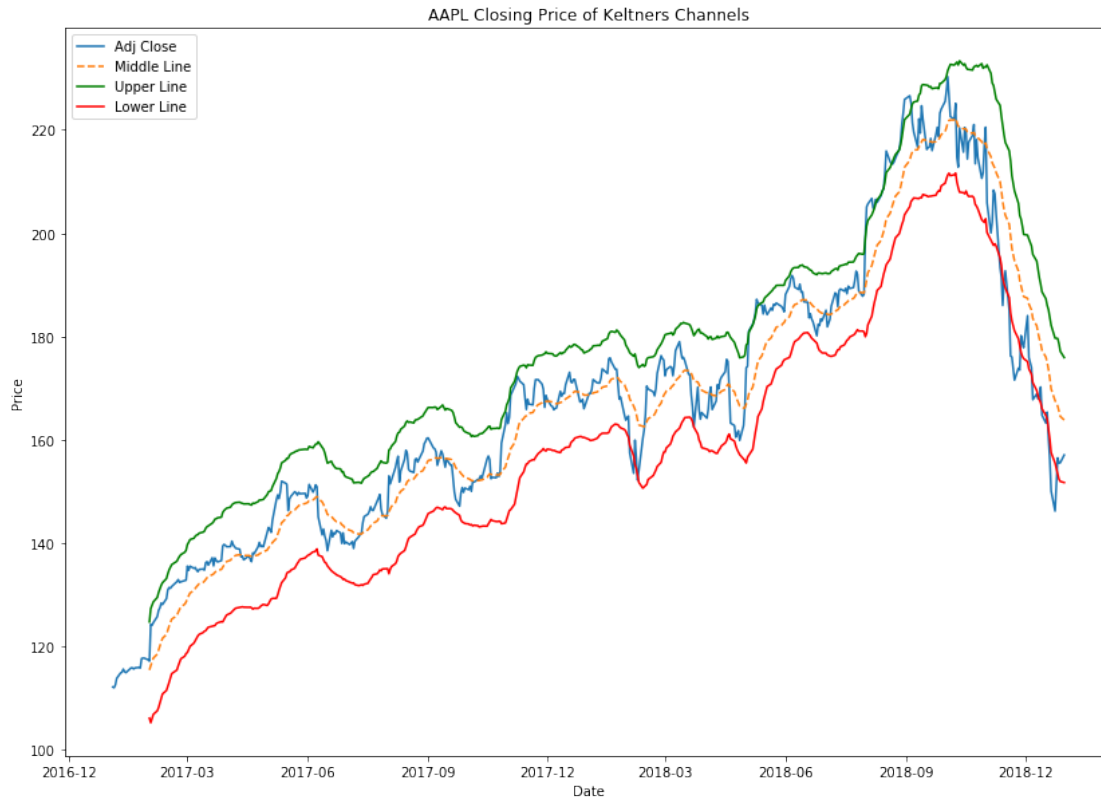
```
2017-01-03  28781900
2017-01-04  21118100
2017-01-05  22193600
2017-01-06  31751900
2017-01-09  33561900
```

```
[3]: import talib as ta
```

```
[4]: n = 20
df['EMA'] = ta.EMA(df['Adj Close'], timeperiod=20)
df['ATR'] = ta.ATR(df['High'], df['Low'], df['Adj Close'], timeperiod=10)
df['Upper Line'] = df['EMA'] + 2*df['ATR']
df['Lower Line'] = df['EMA'] - 2*df['ATR']
del df['ATR']
```

```
[5]: plt.figure(figsize=(14,10))
plt.plot(df['Adj Close'])
plt.plot(df['EMA'], label='Middle Line', linestyle='--')
plt.plot(df['Upper Line'], color='g')
plt.plot(df['Lower Line'], color='r')
plt.ylabel('Price')
plt.xlabel('Date')
plt.title(symbol + ' Closing Price of Keltners Channels')
plt.legend(loc='best')
```

```
[5]: <matplotlib.legend.Legend at 0x22d4e106240>
```



1.1 Candlestick with Keltners Channels

```
[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'] = mdates.date2num(dfc['Date'].astype(dt.date))
dfc.head()
```

```
[6]:
```

	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	EMA	Upper Line	Lower Line	VolumePositive
0	28781900	NaN	NaN	NaN	False

1	21118100	NaN	NaN	NaN	False
2	22193600	NaN	NaN	NaN	False
3	31751900	NaN	NaN	NaN	False
4	33561900	NaN	NaN	NaN	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.plot(df['EMA'], label='Middle Line', linestyle='--')
ax1.plot(df['Upper Line'], color='g')
ax1.plot(df['Lower Line'], color='r')
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*dfc.Volume.max())
ax1.set_title('Stock ' + symbol + ' Closing Price')
ax1.legend(loc='best')
ax1.set_ylabel('Price')
ax1.set_xlabel('Date')
```

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

