

Kijun_Sen

September 29, 2021

1 Kijun-sen (Base Line)

<https://www.investopedia.com/terms/k/kijunsen.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

```

```

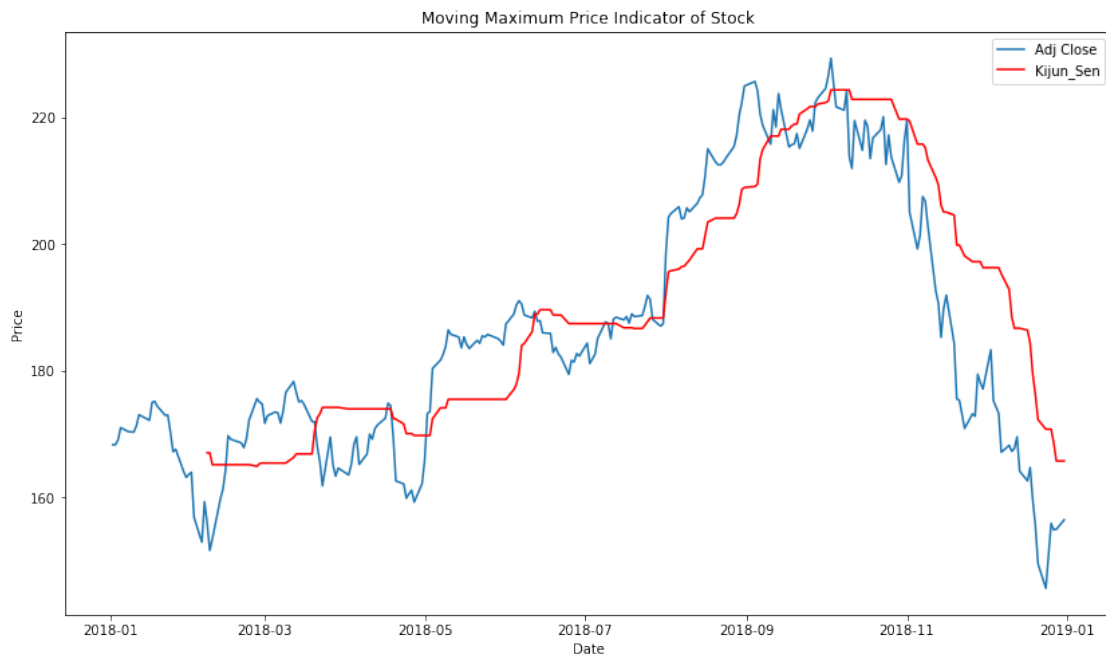
[3]: # Kijun-sen (Base Line): (26-Period High + 26-Period Low)/2))
Period_High = df['High'].rolling(window=26).max()
Period_Low = df['Low'].rolling(window=26).min()
df['Kijun_Sen'] = (Period_High + Period_Low) / 2

```

```

[4]: plt.figure(figsize=(14,8))
plt.plot(df['Adj Close'])
plt.plot(df['Kijun_Sen'], color='red')
plt.title('Moving Maximum Price Indicator of Stock')
plt.legend()
plt.xlabel('Date')
plt.ylabel('Price')
plt.show()

```



1.1 Candlestick with Kijun-Sen

```
[5]: 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()
```

```
[5]:      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  Kijun_Sen  VolumePositive
0  25555900      NaN      False
1  29517900      NaN      False
2  22434600      NaN      False
3  23660000      NaN      False
4  20567800      NaN      False
```

```
[6]: 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['Kijun_Sen'], color='orange')
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')
ax1.legend()
```

```

ax2 = plt.subplot(2, 1, 2)
df['VolumePositive'] = df['Open'] < df['Adj Close']
ax2.bar(df.index, df['Volume'], color=df.VolumePositive.map({True: 'g', False: 'r'}), label='macdhist')
ax2.grid()
ax2.set_ylabel('Volume')
ax2.set_xlabel('Date')

```

[6]: Text(0.5,0,'Date')

