

Elder_Ray_Index

September 29, 2021

1 Elder Ray Index

<https://www.investopedia.com/terms/e/elderray.asp>

<https://library.tradingtechnologies.com/trade/chrt-ti-elder-ray-index.html>

```
[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 = '2018-08-01'
end = '2018-12-31'

# 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						
2018-08-01	199.130005	201.759995	197.309998	201.500000	198.478760	
2018-08-02	200.580002	208.380005	200.350006	207.389999	204.280457	
2018-08-03	207.029999	208.740005	205.479996	207.990005	204.871445	
2018-08-06	208.000000	209.250000	207.070007	209.070007	205.935257	
2018-08-07	209.320007	209.500000	206.759995	207.110001	204.004639	

	Volume
Date	
2018-08-01	67935700
2018-08-02	62404000
2018-08-03	33447400
2018-08-06	25425400
2018-08-07	25587400

```
[3]: n = 13
df['EMA'] = df['Adj Close'].
    ewm(ignore_na=False,span=n,min_periods=n,adjust=True).mean()
df['BullPower'] = [i-j for (i,j) in zip(df['High'],df['EMA'])]
df['BearPower'] = [i-j for (i,j) in zip(df['EMA'],df['Low'])]
```

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

```
[4]:
```

	Open	High	Low	Close	Adj Close \
Date					
2018-08-01	199.130005	201.759995	197.309998	201.500000	198.478760
2018-08-02	200.580002	208.380005	200.350006	207.389999	204.280457
2018-08-03	207.029999	208.740005	205.479996	207.990005	204.871445
2018-08-06	208.000000	209.250000	207.070007	209.070007	205.935257
2018-08-07	209.320007	209.500000	206.759995	207.110001	204.004639
2018-08-08	206.050003	207.809998	204.520004	207.250000	204.142532
2018-08-09	207.279999	209.779999	207.199997	208.880005	205.748108
2018-08-10	207.360001	209.100006	206.669998	207.529999	205.135254
2018-08-13	207.699997	210.949997	207.699997	208.869995	206.459793
2018-08-14	210.160004	210.559998	208.259995	209.750000	207.329651
2018-08-15	209.220001	210.740005	208.330002	210.240005	207.813995
2018-08-16	211.750000	213.809998	211.470001	213.320007	210.858459
2018-08-17	213.440002	217.949997	213.160004	217.580002	215.069290
2018-08-20	218.100006	219.179993	215.110001	215.460007	212.973755
2018-08-21	216.800003	217.190002	214.029999	215.039993	212.558609
2018-08-22	214.100006	216.360001	213.839996	215.050003	212.568481
2018-08-23	214.649994	217.050003	214.600006	215.490005	213.003418
2018-08-24	216.600006	216.899994	215.110001	216.160004	213.665680
2018-08-27	217.149994	218.740005	216.330002	217.940002	215.425140
2018-08-28	219.009995	220.539993	218.919998	219.699997	217.164825

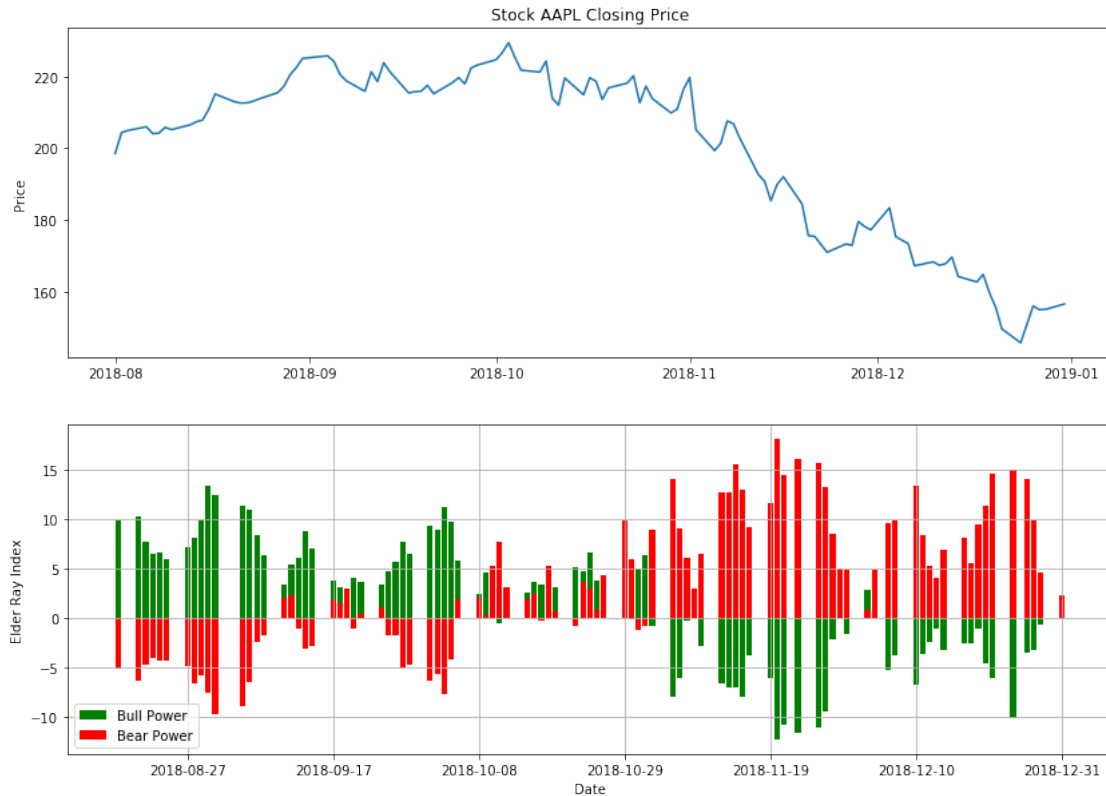
	Volume	EMA	BullPower	BearPower
Date				
2018-08-01	67935700	NaN	NaN	NaN
2018-08-02	62404000	NaN	NaN	NaN
2018-08-03	33447400	NaN	NaN	NaN
2018-08-06	25425400	NaN	NaN	NaN
2018-08-07	25587400	NaN	NaN	NaN
2018-08-08	22525500	NaN	NaN	NaN

2018-08-09	23469200	NaN	NaN	NaN
2018-08-10	24611200	NaN	NaN	NaN
2018-08-13	25869100	NaN	NaN	NaN
2018-08-14	20748000	NaN	NaN	NaN
2018-08-15	28807600	NaN	NaN	NaN
2018-08-16	28500400	NaN	NaN	NaN
2018-08-17	35427000	208.011611	9.938386	-5.148393
2018-08-20	30287700	208.813095	10.366898	-6.296906
2018-08-21	26159800	209.406986	7.783016	-4.623013
2018-08-22	19018100	209.900524	6.459477	-3.939472
2018-08-23	18883200	210.378579	6.671424	-4.221427
2018-08-24	18476400	210.879399	6.020595	-4.230602
2018-08-27	20525100	211.565467	7.174538	-4.764535
2018-08-28	22776800	212.403788	8.136205	-6.516210

```
[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.bar(df.index, df['BullPower'], label='Bull Power', color='green')
ax2.bar(df.index, df['BearPower'], label='Bear Power', color='red')
ax2.grid()
ax2.set_ylabel('Elder Ray Index')
ax2.set_xlabel('Date')
ax2.legend(loc='best')
```

```
[5]: <matplotlib.legend.Legend at 0x1bf3ee97828>
```



1.1 Candlestick with Elder Ray Index

```
[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	736907.0	199.130005	201.759995	197.309998	201.500000	198.478760	
1	736908.0	200.580002	208.380005	200.350006	207.389999	204.280457	
2	736909.0	207.029999	208.740005	205.479996	207.990005	204.871445	
3	736912.0	208.000000	209.250000	207.070007	209.070007	205.935257	
4	736913.0	209.320007	209.500000	206.759995	207.110001	204.004639	

	Volume	EMA	BullPower	BearPower	VolumePositive
0	67935700	NaN	NaN	NaN	False

1	62404000	NaN	NaN	NaN	True
2	33447400	NaN	NaN	NaN	False
3	25425400	NaN	NaN	NaN	False
4	25587400	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.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.bar(df.index, df['BullPower'], label='Bull Power', color='green')
ax2.bar(df.index, df['BearPower'], label='Bear Power', color='red')
ax2.grid()
ax2.set_ylabel('Elder Ray Index')
ax2.set_xlabel('Date')
ax2.legend(loc='best')
```

```
[7]: <matplotlib.legend.Legend at 0x1bf3f1b9080>
```

