McClellan_Oscillator

September 29, 2021

1 McClellan Oscillator

https://stockcharts.com/school/doku.php?id=chart_school:market_indicators:mcclellan_oscillator

Market Indicator

[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
dfs = yf.download(symbol,start,end)

# View Columns
dfs.head()
```

```
[**********************100%***********************]  1 of 1 downloaded
```

[2]:

Date	Open	High	Low	Close	Adj Close	Volume
2018-01-02	170.160004	172.300003	169.259995	172.259995	168.987320	
2018-01-03	172.529999	174.550003	171.960007	172.229996	168.957886	
2018-01-04	172.539993	173.470001	172.080002	173.029999	169.742706	
2018-01-05	173.440002	175.369995	173.050003	175.000000	171.675278	
2018-01-08	174.350006	175.610001	173.929993	174.350006	171.037628	

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

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

https://en.wikipedia.org/wiki/Advance%E2%80%93decline_line

https://www.investopedia.com/terms/m/mcclellanoscillator.asp

[4]:
```
change = dfs['Adj Close'].diff()
Advances = change[change > 0]
Declines = change[change <= 0]
```

[5]:
```
# df[['Advances', 'Declines']] = df[['Advances', 'Declines']].fillna(0)
# df['ADL'] = df['Advances'].fillna(df['Declines'])
# ADL for stocks
dfs['ADL_Stock'] = Advances.combine_first(Declines)
```

[6]:
```
dfs.head()
```

[6]:

Date	Open	High	Low	Close	Adj Close	Volume	ADL_Stock
2018-01-02	170.160004	172.300003	169.259995	172.259995	168.987320	25555900	NaN
2018-01-03	172.529999	174.550003	171.960007	172.229996	168.957886	29517900	-0.029434
2018-01-04	172.539993	173.470001	172.080002	173.029999	169.742706	22434600	0.784820
2018-01-05	173.440002	175.369995	173.050003	175.000000	171.675278	23660000	1.932572
2018-01-08	174.350006	175.610001	173.929993	174.350006	171.037628	20567800	-0.637650

https://stockcharts.com/school/doku.php?id=chart_school:market_indicators:mcclellan_oscillator

[7]:
```
import quandl as q

Advances = q.get('URC/NYSE_ADV', start_date = "2018-01-01")['Numbers of Stocks']
Declines = q.get('URC/NYSE_DEC', start_date = "2018-01-01")['Numbers of Stocks']
```

[8]:
```
df = pd.DataFrame()
df['Advances'] = Advances
df['Declines'] = Declines
df.head()
```

[8]:

Date	Advances	Declines
2018-01-02	1852.0	1160.0
2018-01-03	1757.0	1211.0
2018-01-04	1755.0	1208.0
2018-01-05	1794.0	1172.0
2018-01-08	1719.0	1240.0

[9]:
```
#Ratio Adjusted Net Advances (RANA): (Advances - Declines)/(Advances +␣
 ↪Declines)
#RANA = (advances - declines) / (advances + declines)
# df['Net_Advances'] = df['Advances'] - df['Declines']
# df['Ratio_Adjusted'] = (df['Net_Advances']/(df['Advances'] +␣
 ↪df['Declines']))*1000
df['Net_Advances'] = df['Advances'] - df['Declines']
df['Ratio_Adjusted'] = (df['Net_Advances']/(df['Advances'] + df['Declines'])) *␣
 ↪1000
df['19_EMA'] = ta.EMA(df['Ratio_Adjusted'], timeperiod=19)
df['39_EMA'] = ta.EMA(df['Ratio_Adjusted'], timeperiod=39)
df['RANA'] = (df['Advances'] - df['Declines']) / (df['Advances'] +␣
 ↪df['Declines']) * 1000
```

[10]:
```
df.tail(20)
```

[10]:

Date	Advances	Declines	Net_Advances	Ratio_Adjusted	19_EMA	\
2019-02-06	1183.0	1774.0	-591.0	-199.864728	187.895896	
2019-02-07	922.0	2024.0	-1102.0	-374.066531	131.699653	
2019-02-08	1414.0	1523.0	-109.0	-37.112700	114.818418	
2019-02-11	1934.0	1017.0	917.0	310.742121	134.410788	
2019-02-12	2218.0	734.0	1484.0	502.710027	171.240712	
2019-02-13	1907.0	1035.0	872.0	296.397009	183.756342	
2019-02-14	1577.0	1356.0	221.0	75.349472	172.915655	
2019-02-15	2293.0	656.0	1637.0	555.103425	211.134432	
2019-02-19	1879.0	1068.0	811.0	275.195114	217.540500	
2019-02-20	1784.0	1162.0	622.0	211.133741	216.899824	
2019-02-21	1136.0	1785.0	-649.0	-222.184183	172.991423	
2019-02-22	2145.0	807.0	1338.0	453.252033	201.017484	
2019-02-25	1453.0	1487.0	-34.0	-11.564626	179.759273	
2019-02-26	1213.0	1724.0	-511.0	-173.987062	144.384640	
2019-02-27	1530.0	1385.0	145.0	49.742710	134.920447	
2019-02-28	1340.0	1591.0	-251.0	-85.636302	112.864772	

```
2019-03-01    1895.0    1044.0         851.0      289.554270  130.533722
2019-03-04    1291.0    1625.0        -334.0     -114.540466  106.026303
2019-03-05    1382.0    1524.0        -142.0      -48.864418   90.537231
2019-03-06     763.0    2185.0       -1422.0     -482.360923   33.247415
```

Date	39_EMA	RANA
2019-02-06	138.145820	-199.864728
2019-02-07	112.535203	-374.066531
2019-02-08	105.052808	-37.112700
2019-02-11	115.337273	310.742121
2019-02-12	134.705911	502.710027
2019-02-13	142.790466	296.397009
2019-02-14	139.418416	75.349472
2019-02-15	160.202667	555.103425
2019-02-19	165.952289	275.195114
2019-02-20	168.211362	211.133741
2019-02-21	148.691584	-222.184183
2019-02-22	163.919607	453.252033
2019-02-25	155.145395	-11.564626
2019-02-26	138.688772	-173.987062
2019-02-27	134.241469	49.742710
2019-02-28	123.247581	-85.636302
2019-03-01	131.562915	289.554270
2019-03-04	119.257746	-114.540466
2019-03-05	110.851638	-48.864418
2019-03-06	81.191010	-482.360923

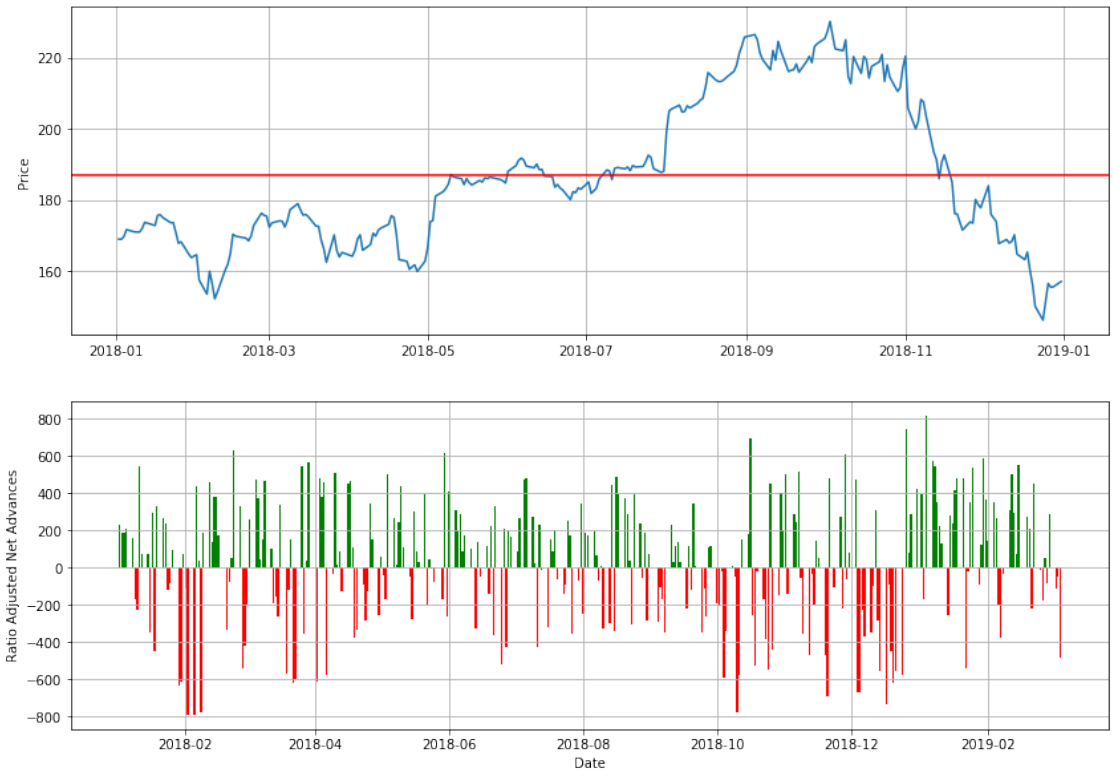
[11]:
```
plt.figure(figsize=(12,6))
plt.plot(dfs.index, dfs['Adj Close'])
plt.axhline(y=dfs['Adj Close'].mean(),color='r')
plt.title('Stock Close Price')
plt.grid()
plt.ylabel('Price')
plt.show()
```

1.1 Comparing Stock and McClellan Oscillator

[12]:
```
# Line Chart
# See if the stock correlate with Market Indicator
fig = plt.figure(figsize=(14,10))
ax1 = plt.subplot(2, 1, 1)
ax1.plot(dfs.index, dfs['Adj Close'])
ax1.axhline(y=dfs['Adj Close'].mean(),color='r')
ax1.grid()
ax1.set_ylabel('Price')

df['Positive'] = df['RANA'] > 0
ax2 = plt.subplot(2, 1, 2)
ax2.bar(df.index, df['RANA'], color=df.Positive.map({True: 'g', False: 'r'}))
ax2.grid()
ax2.set_ylabel('Ratio Adjusted Net Advances')
ax2.set_xlabel('Date')
```

[12]:
```
Text(0.5,0,'Date')
```

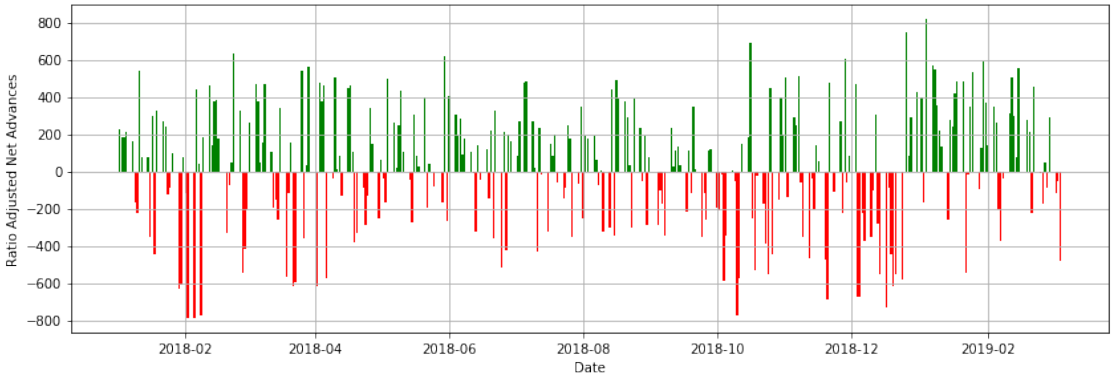


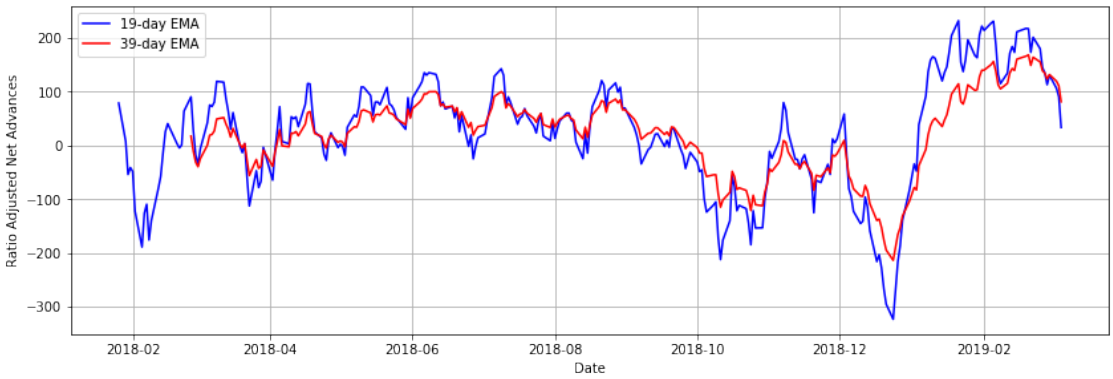
1.2 NYSE Advance and Declines

[13]:
```
fig = plt.figure(figsize=(14,10))
df['Positive'] = df['RANA'] > 0
ax = plt.subplot(2, 1, 1)
ax.bar(df.index, df['RANA'], color=df.Positive.map({True: 'g', False: 'r'}))
ax.grid()
ax.set_ylabel('Ratio Adjusted Net Advances')
ax.set_xlabel('Date')

ax2 = plt.subplot(2, 1, 2)
ax2.plot(df.index, df['19_EMA'], color='b', label='19-day EMA')
ax2.plot(df.index, df['39_EMA'], color='r', label='39-day EMA')
ax2.grid()
ax2.set_ylabel('Ratio Adjusted Net Advances')
ax2.legend(loc='best')
ax2.set_xlabel('Date')
```

[13]:
```
Text(0.5,0,'Date')
```


Ratio Adjusted Net Advances
800
600
400
200
0
−200
−400
−600
−800
2018-02
2018-04
2018-06
2018-08
2018-10
2018-12
2019-02
Date


19-day EMA
39-day EMA
Ratio Adjusted Net Advances
200
100
0
−100
−200
−300
2018-02
2018-04
2018-06
2018-08
2018-10
2018-12
2019-02
Date