Breadth_Indicators

September 29, 2021

1 Breadth Indicators

Breadth Indicators include:

On Balance Volume

McClellan Summation Index Arms Index (TRIN)

Force Index

Chaikin Oscillator

Up/Down Volume Ratio

Up/Down Volume Spread

Cumulative Volume Index

[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 = 'SPY'
start = '2012-01-01'
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
2012-01-03  127.760002  128.380005  127.430000  127.500000  110.244629
2012-01-04  127.199997  127.809998  126.709999  127.699997  110.417557
2012-01-05  127.010002  128.229996  126.430000  128.039993  110.711548
2012-01-06  128.199997  128.220001  127.290001  127.709999  110.426208
2012-01-09  128.000000  128.179993  127.410004  128.020004  110.694260

                Volume
Date
2012-01-03  193697900
2012-01-04  127186500
2012-01-05  173895000
2012-01-06  148050000
2012-01-09   99530200
```

[3]:
```
df['Adj Close'][1:]
```

[3]:
```
Date
2012-01-04    110.417557
2012-01-05    110.711548
2012-01-06    110.426208
2012-01-09    110.694260
2012-01-10    111.654045
2012-01-11    111.714554
2012-01-12    111.982590
2012-01-13    111.403275
2012-01-17    111.835632
2012-01-18    113.072098
2012-01-19    113.668709
2012-01-20    114.092407
2012-01-23    113.798370
2012-01-24    113.668709
2012-01-25    114.619827
2012-01-26    114.031883
2012-01-27    113.980011
2012-01-30    113.590881
2012-01-31    113.547661
2012-02-01    114.542015
2012-02-02    114.723579
2012-02-03    116.331863
2012-02-06    116.254051
2012-02-07    116.548042
2012-02-08    116.893898
2012-02-09    117.040901
2012-02-10    116.176231
2012-02-13    117.040901
```

```
2012-02-14    116.893898
2012-02-15    116.349159
                 …
2018-11-15    270.244537
2018-11-16    270.947357
2018-11-19    266.364441
2018-11-20    261.435028
2018-11-21    262.325867
2018-11-23    260.573883
2018-11-26    264.780670
2018-11-27    265.671539
2018-11-28    271.788696
2018-11-29    271.194794
2018-11-30    272.847839
2018-12-03    276.460693
2018-12-04    267.502716
2018-12-06    267.096893
2018-12-07    260.890656
2018-12-10    261.385559
2018-12-11    261.444946
2018-12-12    262.761414
2018-12-13    262.672302
2018-12-14    257.822144
2018-12-17    252.764099
2018-12-18    252.486938
2018-12-19    248.705765
2018-12-20    244.657349
2018-12-21    239.644424
2018-12-24    233.312317
2018-12-26    245.100388
2018-12-27    246.982117
2018-12-28    246.663513
2018-12-31    248.823990
Name: Adj Close, Length: 1759, dtype: float64
```

[4]:
```
import talib as ta
```

1.1 On Balance Volume

[5]:
```
OBV = ta.OBV(df['Adj Close'], df['Volume'])
```

1.2 McClellan Summation Index Arms Index (TRIN)

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

[6]:
```
import quandl as q
```

```
# For NASDAQ
#Advances = q.get('URC/NASDAQ_ADV')['Numbers of Stocks']
#Declines = q.get('URC/nASDAQ_DEC')['Numbers of Stocks']
#n = Advances - Declines
```

[7]:
```
Advances = q.get('URC/NYSE_ADV', start_date = "2017-07-27")['Numbers of Stocks']
Declines = q.get('URC/NYSE_DEC', start_date = "2017-07-27")['Numbers of Stocks']
```

[8]:
```
adv_vol = q.get("URC/NYSE_ADV_VOL", start_date = "2017-07-27")['Numbers of␣
↪Stocks']
dec_vol = q.get("URC/NYSE_DEC_VOL", start_date = "2017-07-27")['Numbers of␣
↪Stocks']
```

[9]:
```
data = pd.DataFrame()
data['Advances'] = Advances
data['Declines'] = Declines
data['adv_vol'] = adv_vol
data['dec_vol'] = dec_vol
```

[10]:
```
data['Net_Advances'] = data['Advances'] - data['Declines']
data['Ratio_Adjusted'] = (data['Net_Advances']/(data['Advances'] +␣
↪data['Declines'])) * 1000
data['19_EMA'] = ta.EMA(data['Ratio_Adjusted'], timeperiod=19)
data['39_EMA'] = ta.EMA(data['Ratio_Adjusted'], timeperiod=39)
data['RANA'] = (data['Advances'] - data['Declines']) / (data['Advances'] +␣
↪data['Declines']) * 1000
```

[11]:
```
# Finding the TRIN Value
data['ad_ratio'] = data['Advances'].divide(data['Declines'] ) # AD Ratio
data['ad_vol'] = data['adv_vol'].divide(data['dec_vol']) # AD Volume Ratio
data['TRIN'] = data['ad_ratio'].divide(data['adv_vol']) # TRIN Value
```

[12]:
```
data.head()
```

[12]:

Date	Advances	Declines	adv_vol	dec_vol	Net_Advances
2017-07-27	1320.0	1610.0	1.795852e+09	2.081253e+09	-290.0
2017-07-28	1585.0	1347.0	1.392776e+09	1.757435e+09	238.0
2017-07-31	1583.0	1400.0	1.727800e+09	1.574906e+09	183.0
2017-08-01	1774.0	1169.0	1.777532e+09	1.535265e+09	605.0
2017-08-02	1172.0	1764.0	1.335177e+09	2.024331e+09	-592.0

Date	Ratio_Adjusted	19_EMA	39_EMA	RANA	ad_ratio	ad_vol
2017-07-27	-98.976109	NaN	NaN	-98.976109	0.819876	0.862871
2017-07-28	81.173261	NaN	NaN	81.173261	1.176689	0.792505
2017-07-31	61.347637	NaN	NaN	61.347637	1.130714	1.097082

```
2017-08-01      205.572545     NaN     NaN  205.572545  1.517536  1.157801
2017-08-02     -201.634877     NaN     NaN -201.634877  0.664399  0.659564
```

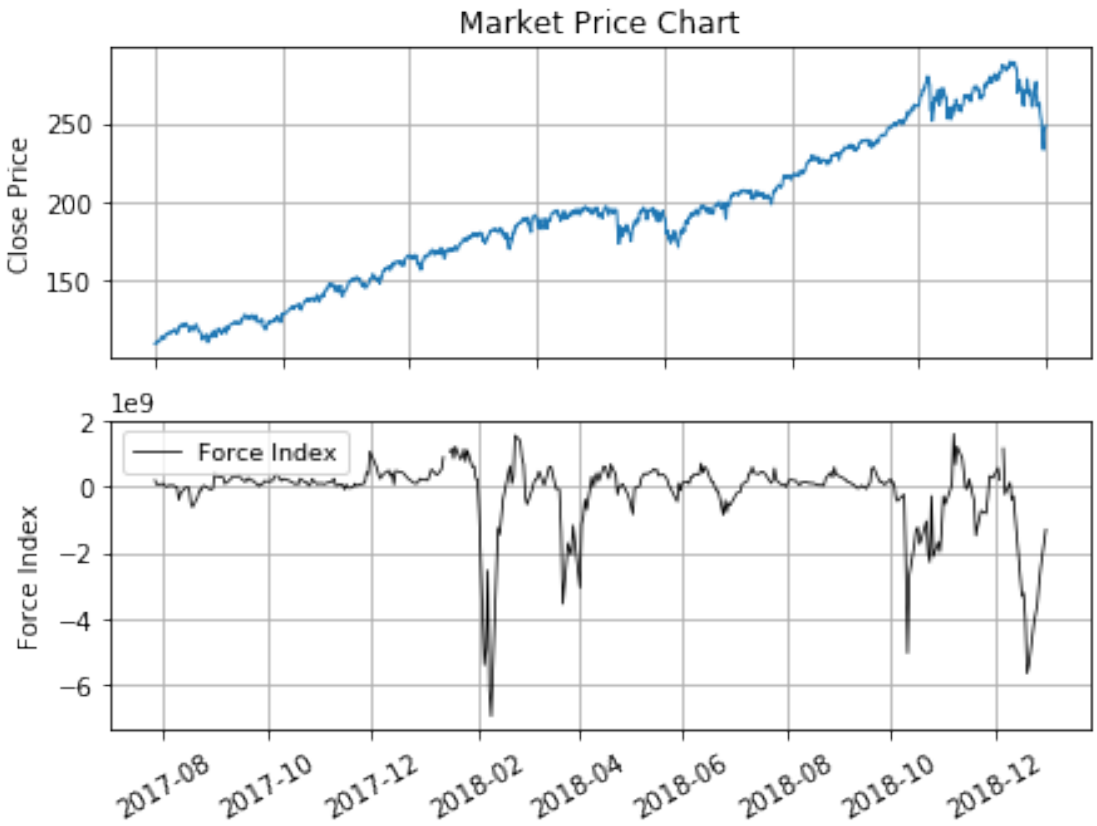
```
                    TRIN
Date
2017-07-27  4.565387e-10
2017-07-28  8.448513e-10
2017-07-31  6.544241e-10
2017-08-01  8.537323e-10
2017-08-02  4.976114e-10
```

1.3 Force Index

[13]:
```
def ForceIndex(data,n):
    ForceIndex=pd.Series(df['Adj Close'].diff(n)*␣
 ↪df['Volume'],name='ForceIndex')
    data = data.join(ForceIndex)
    return data
```

[14]:
```
n=10
ForceIndex = ForceIndex(data,n)
ForceIndex = ForceIndex['ForceIndex']
```

[15]:
```
fig=plt.figure(figsize=(7,5))
ax=fig.add_subplot(2,1,1)
ax.set_xticklabels([])
plt.plot(df['Adj Close'],lw=1)
plt.title('Market Price Chart')
plt.ylabel('Close Price')
plt.grid(True)
bx=fig.add_subplot(2,1,2)
plt.plot(ForceIndex,'k',lw=0.75,linestyle='-',label='Force Index')
plt.legend(loc=2,prop={'size':9.5})
plt.ylabel('Force Index')
plt.grid(True)
plt.setp(plt.gca().get_xticklabels(),rotation=30)
plt.show()
```



1.4 Chaikin Oscillator

[16]:
```
def Chaikin(data):
    money_flow_volume = (2 * df['Adj Close'] - df['High'] - df['Low']) /␣
↪(df['High'] - df['Low']) * df['Volume']
    ad = money_flow_volume.cumsum()
    Chaikin = pd.Series(ad.ewm(com=(3-1)/2).mean() - ad.ewm(com=(10-1)/2).
↪mean(), name='Chaikin')
    data = data.join(Chaikin)
    return data
```

[17]:
```
Chaikin(df)
```

[17]:

Date	Open	High	Low	Close	Adj Close	\
2012-01-03	127.760002	128.380005	127.430000	127.500000	110.244629	
2012-01-04	127.199997	127.809998	126.709999	127.699997	110.417557	
2012-01-05	127.010002	128.229996	126.430000	128.039993	110.711548	
2012-01-06	128.199997	128.220001	127.290001	127.709999	110.426208	
2012-01-09	128.000000	128.179993	127.410004	128.020004	110.694260	

2012-01-10	129.389999	129.649994	128.949997	129.130005	111.654045
2012-01-11	128.729996	129.369995	128.520004	129.199997	111.714554
2012-01-12	129.570007	129.699997	128.539993	129.509995	111.982590
2012-01-13	128.639999	129.050003	127.720001	128.839996	111.403275
2012-01-17	130.080002	130.320007	128.899994	129.339996	111.835632
2012-01-18	129.309998	130.839996	129.080002	130.770004	113.072098
2012-01-19	131.220001	131.570007	130.800003	131.460007	113.668709
2012-01-20	131.240005	131.949997	130.919998	131.949997	114.092407
2012-01-23	131.509995	132.250000	130.979996	131.610001	113.798370
2012-01-24	130.800003	131.500000	130.600006	131.460007	113.668709
2012-01-25	131.259995	132.869995	130.750000	132.559998	114.619827
2012-01-26	133.149994	133.399994	131.360001	131.880005	114.031883
2012-01-27	131.240005	132.050003	131.149994	131.820007	113.980011
2012-01-30	130.509995	131.440002	130.059998	131.369995	113.590881
2012-01-31	132.020004	132.179993	130.679993	131.320007	113.547661
2012-02-01	132.289993	133.139999	132.130005	132.470001	114.542015
2012-02-02	132.729996	133.020004	132.210007	132.679993	114.723579
2012-02-03	134.000000	134.619995	133.770004	134.539993	116.331863
2012-02-06	133.979996	134.509995	133.830002	134.449997	116.254051
2012-02-07	134.169998	135.020004	133.639999	134.789993	116.548042
2012-02-08	134.860001	135.220001	134.309998	135.190002	116.893898
2012-02-09	135.410004	135.589996	134.559998	135.360001	117.040901
2012-02-10	134.160004	134.470001	133.839996	134.360001	116.176231
2012-02-13	135.320007	135.520004	134.740005	135.360001	117.040901
2012-02-14	135.000000	135.270004	134.250000	135.190002	116.893898
…	…	…	…	…	…
2018-11-15	268.779999	273.540009	267.010010	273.019989	270.244537
2018-11-16	271.790009	274.750000	271.209991	273.730011	270.947357
2018-11-19	273.049988	273.380005	268.070007	269.100006	266.364441
2018-11-20	265.359985	267.000000	263.149994	264.119995	261.435028
2018-11-21	265.859985	267.149994	265.010010	265.019989	262.325867
2018-11-23	263.179993	264.820007	263.070007	263.250000	260.573883
2018-11-26	265.779999	267.750000	265.339996	267.500000	264.780670
2018-11-27	266.339996	268.399994	265.660004	268.399994	265.671539
2018-11-28	269.600006	274.579987	268.329987	274.579987	271.788696
2018-11-29	273.709991	275.549988	272.429993	273.980011	271.194794
2018-11-30	273.809998	276.279999	273.450012	275.649994	272.847839
2018-12-03	280.279999	280.399994	277.510010	279.299988	276.460693
2018-12-04	278.369995	278.850006	269.899994	270.250000	267.502716
2018-12-06	265.920013	269.970001	262.440002	269.839996	267.096893
2018-12-07	269.459991	271.220001	262.630005	263.570007	260.890656
2018-12-10	263.369995	265.160004	258.619995	264.070007	261.385559
2018-12-11	267.660004	267.869995	262.480011	264.130005	261.444946
2018-12-12	267.470001	269.000000	265.369995	265.459991	262.761414
2018-12-13	266.519989	267.489990	264.119995	265.369995	262.672302
2018-12-14	262.959991	264.029999	259.850006	260.470001	257.822144
2018-12-17	259.399994	260.649994	253.529999	255.360001	252.764099

```
2018-12-18  257.200012  257.950012  253.279999  255.080002  252.486938
2018-12-19  255.169998  259.399994  249.350006  251.259995  248.705765
2018-12-20  249.860001  251.619995  244.649994  247.169998  244.657349
2018-12-21  246.740005  249.710007  239.979996  240.699997  239.644424
2018-12-24  239.039993  240.839996  234.270004  234.339996  233.312317
2018-12-26  235.970001  246.179993  233.759995  246.179993  245.100388
2018-12-27  242.570007  248.289993  238.960007  248.070007  246.982117
2018-12-28  249.580002  251.399994  246.449997  247.750000  246.663513
2018-12-31  249.559998  250.190002  247.470001  249.919998  248.823990

               Volume       Chaikin
Date
2012-01-03  193697900  0.000000e+00
2012-01-04  127186500 -4.543919e+08
2012-01-05  173895000 -1.035752e+09
2012-01-06  148050000 -2.213366e+09
2012-01-09   99530200 -3.386944e+09
2012-01-10  115282000 -4.898310e+09
2012-01-11  111540700 -6.153837e+09
2012-01-12  118983700 -6.979759e+09
2012-01-13  179836200 -7.907538e+09
2012-01-17  132209200 -8.458773e+09
2012-01-18  163395200 -8.784849e+09
2012-01-19  126328900 -9.797472e+09
2012-01-20  138230200 -1.068820e+10
2012-01-23  129295800 -1.115189e+10
2012-01-24  103083300 -1.152341e+10
2012-01-25  198613200 -1.159977e+10
2012-01-26  184880500 -1.159130e+10
2012-01-27  135259100 -1.217402e+10
2012-01-30  147311800 -1.243961e+10
2012-01-31  157212000 -1.258395e+10
2012-02-01  166234500 -1.336666e+10
2012-02-02  113090400 -1.404009e+10
2012-02-03  160598500 -1.516611e+10
2012-02-06  107694500 -1.603744e+10
2012-02-07  135528100 -1.603639e+10
2012-02-08  139361400 -1.630917e+10
2012-02-09  148602900 -1.658528e+10
2012-02-10  167907500 -1.822975e+10
2012-02-13  115841900 -1.894844e+10
2012-02-14  165329500 -1.935730e+10
…                 …            …
2018-11-15  135101400 -4.923658e+08
2018-11-16  126668000 -4.767755e+08
2018-11-19  103061700 -4.809125e+08
2018-11-20  136021300 -5.207224e+08
```

```
2018-11-21   75563700 -5.740263e+08
2018-11-23   42807900 -5.961248e+08
2018-11-26   79981400 -5.882331e+08
2018-11-27   75502400 -5.553501e+08
2018-11-28  127629600 -4.870751e+08
2018-11-29   82346400 -4.618119e+08
2018-11-30   98204200 -4.540384e+08
2018-12-03  103176300 -4.662504e+08
2018-12-04  177986000 -5.158297e+08
2018-12-06  204185400 -4.738282e+08
2018-12-07  161018900 -4.855518e+08
2018-12-10  151445900 -4.536402e+08
2018-12-11  121504400 -4.528543e+08
2018-12-12   97976700 -4.873435e+08
2018-12-13   96662700 -5.143298e+08
2018-12-14  116961100 -5.519359e+08
2018-12-17  165492300 -5.811294e+08
2018-12-18  134515100 -5.975792e+08
2018-12-19  214992800 -6.271603e+08
2018-12-20  252053400 -6.622767e+08
2018-12-21  255345600 -7.032859e+08
2018-12-24  147311600 -7.166639e+08
2018-12-26  218485400 -5.995531e+08
2018-12-27  186267300 -4.544885e+08
2018-12-28  153100200 -3.983379e+08
2018-12-31  144299400 -3.393576e+08

[1760 rows x 7 columns]
```

1.5 Up/Down Volume Ratio

Volume Spread = Up Volume - Down Volume

[18]:
```
Up = q.get('URC/NYSE_ADV', start_date = "2017-07-27")['Numbers of Stocks']
Down = q.get('URC/NYSE_DEC', start_date = "2017-07-27")['Numbers of Stocks']
Volume_Spread = Up - Down
```

[19]:
```
Volume_Spread
```

[19]:
```
Date
2017-07-27    -290.0
2017-07-28     238.0
2017-07-31     183.0
2017-08-01     605.0
2017-08-02    -592.0
2017-08-03    -429.0
2017-08-04     385.0
```

```
2017-08-07    -121.0
2017-08-08    -787.0
2017-08-09   -1156.0
2017-08-10   -2218.0
2017-08-11     204.0
2017-08-14    1576.0
2017-08-15    -808.0
2017-08-16     616.0
2017-08-17   -1884.0
2017-08-18     199.0
2017-08-21      46.0
2017-08-22    1277.0
2017-08-23     170.0
2017-08-24      46.0
2017-08-25    1023.0
2017-08-28    -200.0
2017-08-29     -53.0
2017-08-30     707.0
2017-08-31    1394.0
2017-09-01    1164.0
2017-09-05   -1178.0
2017-09-06     710.0
2017-09-07      71.0
               …
2019-02-07   -1102.0
2019-02-08    -109.0
2019-02-11     917.0
2019-02-12    1484.0
2019-02-13     872.0
2019-02-14     221.0
2019-02-15    1637.0
2019-02-19     811.0
2019-02-20     622.0
2019-02-21    -649.0
2019-02-22    1338.0
2019-02-25     -34.0
2019-02-26    -511.0
2019-02-27     145.0
2019-02-28    -251.0
2019-03-01     851.0
2019-03-04    -334.0
2019-03-05    -142.0
2019-03-06   -1422.0
2019-03-07   -1152.0
2019-03-08    -284.0
2019-03-11    1733.0
2019-03-12     520.0
```

```
2019-03-13    1079.0
2019-03-14    -182.0
2019-03-15     499.0
2019-03-18    1040.0
2019-03-19    -590.0
2019-03-20    -316.0
2019-03-21    1271.0
Name: Numbers of Stocks, Length: 418, dtype: float64
```

1.6 Up/Down Volume Spread

Volume Ratio = Up Volume / Down Volume

[20]:
```
Up = q.get('URC/NYSE_ADV', start_date = "2017-07-27")['Numbers of Stocks']
Down = q.get('URC/NYSE_DEC', start_date = "2017-07-27")['Numbers of Stocks']
Volume_Ratio = Up/Down
```

[21]:
```
Volume_Ratio
```

[21]:
```
Date
2017-07-27    0.819876
2017-07-28    1.176689
2017-07-31    1.130714
2017-08-01    1.517536
2017-08-02    0.664399
2017-08-03    0.746304
2017-08-04    1.297527
2017-08-07    0.921224
2017-08-08    0.578694
2017-08-09    0.439651
2017-08-10    0.155048
2017-08-11    1.147292
2017-08-14    3.194986
2017-08-15    0.572713
2017-08-16    1.521151
2017-08-17    0.227869
2017-08-18    1.147298
2017-08-21    1.031944
2017-08-22    2.513033
2017-08-23    1.122744
2017-08-24    1.032123
2017-08-25    2.068966
2017-08-28    0.873498
2017-08-29    0.964453
2017-08-30    1.628444
2017-08-31    2.798710
2017-09-01    2.309336
```

```
2017-09-05    0.432015
2017-09-06    1.637343
2017-09-07    1.049512
                 …
2019-02-07    0.455534
2019-02-08    0.928431
2019-02-11    1.901672
2019-02-12    3.021798
2019-02-13    1.842512
2019-02-14    1.162979
2019-02-15    3.495427
2019-02-19    1.759363
2019-02-20    1.535284
2019-02-21    0.636415
2019-02-22    2.657993
2019-02-25    0.977135
2019-02-26    0.703596
2019-02-27    1.104693
2019-02-28    0.842238
2019-03-01    1.815134
2019-03-04    0.794462
2019-03-05    0.906824
2019-03-06    0.349199
2019-03-07    0.436675
2019-03-08    0.821944
2019-03-11    3.808752
2019-03-12    1.433695
2019-03-13    2.160215
2019-03-14    0.883557
2019-03-15    1.413079
2019-03-18    2.089005
2019-03-19    0.665343
2019-03-20    0.804697
2019-03-21    2.523981
Name: Numbers of Stocks, Length: 418, dtype: float64
```

1.7 Cumulative Volume Index

https://www.marketinout.com/technical_analysis.php?t=Cumulative_Volume_Index_(CVI)&id=38

[22]:
```
# CVI = Yesterday's CVI + (Advancing Volume - Declining Volume)
data['CVI'] = data['Net_Advances'][1:] + (data['Advances'] - data['Declines'])
```

[23]:
```
data.head()
```

[23]:
```
            Advances  Declines       adv_vol       dec_vol  Net_Advances  \
Date
```

```
2017-07-27    1320.0    1610.0  1.795852e+09  2.081253e+09        -290.0
2017-07-28    1585.0    1347.0  1.392776e+09  1.757435e+09         238.0
2017-07-31    1583.0    1400.0  1.727800e+09  1.574906e+09         183.0
2017-08-01    1774.0    1169.0  1.777532e+09  1.535265e+09         605.0
2017-08-02    1172.0    1764.0  1.335177e+09  2.024331e+09        -592.0

            Ratio_Adjusted  19_EMA  39_EMA        RANA  ad_ratio    ad_vol  \
Date
2017-07-27      -98.976109     NaN     NaN  -98.976109  0.819876  0.862871
2017-07-28       81.173261     NaN     NaN   81.173261  1.176689  0.792505
2017-07-31       61.347637     NaN     NaN   61.347637  1.130714  1.097082
2017-08-01      205.572545     NaN     NaN  205.572545  1.517536  1.157801
2017-08-02     -201.634877     NaN     NaN -201.634877  0.664399  0.659564

                   TRIN     CVI
Date
2017-07-27  4.565387e-10     NaN
2017-07-28  8.448513e-10   476.0
2017-07-31  6.544241e-10   366.0
2017-08-01  8.537323e-10  1210.0
2017-08-02  4.976114e-10 -1184.0
```