stocks-visualization

July 27, 2020

[1]:
```
import numpy as np
import yfinance as yf
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
```

[2]:
```
tickers = []

with open('stocks-list.txt') as f:
    data = f.readlines()

for datum in data:
    tickers.append(datum.split(' ')[0])

print(tickers)
```

```
['AAPL', 'FB', 'GOOG', 'NFLX', 'TSLA', 'AMZN', 'MSFT', 'KO', 'COST', 'AMD',
'AAL', 'INTC']
```

[3]:
```
colors = sns.color_palette(None, len(tickers))

print(colors)
```

```
[(0.12156862745098039, 0.4666666666666667, 0.7058823529411765), (1.0,
0.4980392156862745, 0.054901960784313725), (0.17254901960784313,
0.6274509803921569, 0.17254901960784313), (0.8392156862745098,
0.15294117647058825, 0.1568627450980392), (0.5803921568627451,
0.403921568627451, 0.7411764705882353), (0.5490196078431373,
0.33725490196078434, 0.29411764705882354), (0.8901960784313725,
0.4666666666666667, 0.7607843137254902), (0.4980392156862745,
0.4980392156862745, 0.4980392156862745), (0.7372549019607844,
0.7411764705882353, 0.13333333333333333), (0.09019607843137255,
0.7450980392156863, 0.8117647058823529), (0.12156862745098039,
0.4666666666666667, 0.7058823529411765), (1.0, 0.4980392156862745,
0.054901960784313725)]
```

[4]:
```
assert len(tickers) == len(colors)
```

[5]:
```
stocks = yf.download(tickers, start="2015-01-02", end='2020-07-24',␣
↪interval="1wk", actions=True, auto_adjust=True)
```

```
[**********100%**********]  12 of 12 completed
```

[6]:
```
stocks
```

[6]:

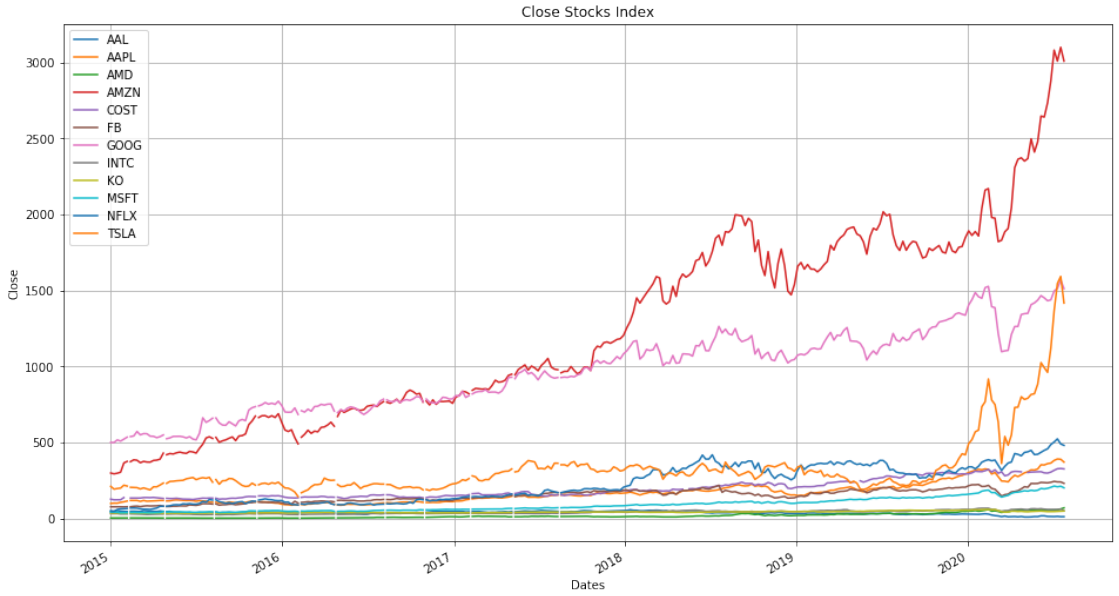
	Adj Close				Close			…	Volume				
	AAL	AAPL	INTC	KO	AAL	AAPL	AMD	…	AMD	AMZN	COST	FB	…
Date								…					…
2015-01-01	NaN	NaN	NaN	NaN	50.227158	98.501518	2.580000	…	35168300.0	11716700.0	8576300.0	94074300.0	…
2015-01-08	NaN	NaN	NaN	NaN	46.816154	100.375572	2.630000	…	57921100.0	18777300.0	10415900.0	115406400.0	…
2015-01-15	NaN	NaN	NaN	NaN	50.852512	100.147011	2.450000	…	151180500.0	21037600.0	7122500.0	103843800.0	…
2015-01-22	NaN	NaN	NaN	NaN	47.962620	105.412628	2.630000	…	160440100.0	19012800.0	8879900.0	128943200.0	…
2015-01-29	NaN	NaN	NaN	NaN	46.863522	109.297836	2.850000	…	113808700.0	53127100.0	33149100.0	193133400.0	…
…	…	…	…	..	…	…	…	…	…	…	…	…	…
2020-06-25	NaN	NaN	NaN	NaN	12.810000	364.109985	52.580002	…	241167700.0	23826000.0	11279800.0	230889300.0	…
2020-07-02	NaN	NaN	NaN	NaN	11.990000	381.369995	53.430000	…	133612200.0	23769100.0			
2020-07-09	NaN	NaN	NaN	NaN	13.440000	390.899994	55.340000	…	302048200.0	32615900.0			
2020-07-16	NaN	NaN	NaN	NaN	11.360000	389.089996	61.790001	…	296591900.0	28992900.0			
2020-07-23	NaN	NaN	NaN	NaN	11.390000	370.459991	69.400002	…	311886700.0	11282800.0			

	GOOG	INTC	KO
Date			
2015-01-01	8472200.0	114963400.0	66523500.0
2015-01-08	12357600.0	149745000.0	71870000.0
2015-01-15	9514700.0	173366000.0	54075900.0
2015-01-22	10090100.0	183529100.0	67771200.0
2015-01-29	16344600.0	168114000.0	77642200.0
…	…	…	…
2020-06-25	11126000.0	128824900.0	91385000.0

```
2020-07-02  10584000.0  114518900.0   6131000.0   65389700.0  57152700.0
2020-07-09  13996400.0  112344200.0   8321700.0  101686100.0  89238300.0
2020-07-16   8220200.0   92742900.0   6683900.0   84407600.0  99356600.0
2020-07-23   2985100.0   40086500.0   3171600.0  225581200.0  32130900.0
```

```
                 MSFT         NFLX         TSLA
Date
2015-01-01  133149800.0   57527400.0   19363200.0
2015-01-08  142231500.0   70730800.0   30090300.0
2015-01-15  143689100.0  138819100.0   17475900.0
2015-01-22  358306200.0  130517800.0   16720600.0
2015-01-29  285639900.0   70224700.0   18835900.0
...                 ...          ...          ...
2020-06-25  175552800.0   29687600.0   57381200.0
2020-07-02  128414100.0   25552100.0   75621000.0
2020-07-09  167206100.0   70995100.0  113826500.0
2020-07-16  186172300.0   77512000.0   71070600.0
2020-07-23  107256500.0   15466300.0   43673200.0

[303 rows x 88 columns]
```

[7]:
```
stocks.Close.plot(figsize=[16, 9], grid=True, legend=True, color=colors)
plt.xlabel('Dates')
plt.ylabel('Close')
plt.title('Close Stocks Index')
plt.show()
```



[8]:
```
df = stocks.Close
```

[9]:
```
df
```

[9]:

Date	AAL	AAPL	AMD	AMZN	COST	FB	GOOG	INTC	KO	MSFT	NFLX	TSLA
2015-01-01	50.227158	98.501518	2.580000	298.420013	126.671837	76.150002	499.727997	31.000975	35.881165	41.275364	46.742859	210.949997
2015-01-08	46.816154	100.375572	2.630000	293.269989	122.274490	76.279999	499.498627	31.284983	35.522259	41.034302	46.320000	192.690002
2015-01-15	50.852512	100.147011	2.450000	297.250000	122.125259	76.739998	516.621643	31.396870	36.189976	40.998585	58.468571	196.570007
2015-01-22	47.962620	105.412628	2.630000	303.910004	122.546577	76.239998	508.603638	29.073093	34.988091	36.775524	63.208572	199.369995
2015-01-29	46.863522	109.297836	2.850000	364.750000	136.853317	75.629997	521.328674	28.918175	34.787788	37.355858	64.101425	218.550003
…	…	…	…	…	…	…	…	…	…	…	…	…
2020-06-25	12.810000	364.109985	52.580002	2878.699951	304.750000	237.550003	1438.040039	58.810001	44.820000	204.699997	485.640015	1119.630005
2020-07-02	11.990000	381.369995	53.430000	3081.110107	316.320007	243.580002	1496.000000	58.610001	45.070000	212.830002	502.779999	1365.880005
2020-07-09	13.440000	390.899994	55.340000	3008.870117	326.700012	240.279999	1513.640015	59.029999	46.400002	208.039993	523.260010	1546.010010
2020-07-16	11.360000	389.089996	61.790001	3099.909912	328.299988	239.869995	1568.489990	61.049999	48.480000	211.750000	489.820007	1592.329956
2020-07-23	11.390000	370.459991	69.400002	3008.909912	325.779999	230.710007	1511.869995	50.590000	48.490002	201.300003	480.450012	1417.000000

[303 rows x 12 columns]

[10]:
```
for ticker in tickers:
    df['diff_{}'.format(ticker)] = ((df['{}'.format(ticker)].shift(-1) -␣
↪df['{}'.format(ticker)]) / df['{}'.format(ticker)] * 100).shift(1)
```

```
<ipython-input-10-ceb8fc1a26f5>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df['diff_{}'.format(ticker)] = ((df['{}'.format(ticker)].shift(-1) -
df['{}'.format(ticker)]) / df['{}'.format(ticker)] * 100).shift(1)
```

[11]:
```
df
```

[11]:

Date	AAL	AAPL	AMD	AMZN	COST	\
2015-01-01	50.227158	98.501518	2.580000	298.420013	126.671837	
2015-01-08	46.816154	100.375572	2.630000	293.269989	122.274490	
2015-01-15	50.852512	100.147011	2.450000	297.250000	122.125259	
2015-01-22	47.962620	105.412628	2.630000	303.910004	122.546577	
2015-01-29	46.863522	109.297836	2.850000	364.750000	136.853317	
…	…	…	…	…	…	
2020-06-25	12.810000	364.109985	52.580002	2878.699951	304.750000	
2020-07-02	11.990000	381.369995	53.430000	3081.110107	316.320007	
2020-07-09	13.440000	390.899994	55.340000	3008.870117	326.700012	
2020-07-16	11.360000	389.089996	61.790001	3099.909912	328.299988	
2020-07-23	11.390000	370.459991	69.400002	3008.909912	325.779999	

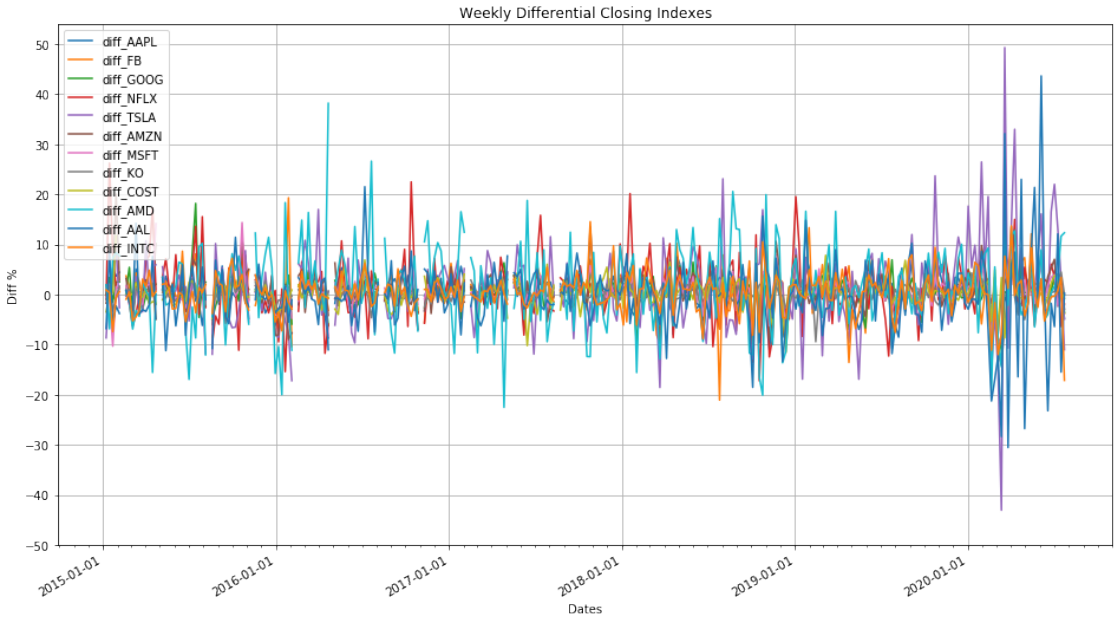
Date	FB	GOOG	INTC	KO	MSFT	…	\
						…	
2015-01-01	76.150002	499.727997	31.000975	35.881165	41.275364	…	
2015-01-08	76.279999	499.498627	31.284983	35.522259	41.034302	…	
2015-01-15	76.739998	516.621643	31.396870	36.189976	40.998585	…	
2015-01-22	76.239998	508.603638	29.073093	34.988091	36.775524	…	
2015-01-29	75.629997	521.328674	28.918175	34.787788	37.355858	…	
…	…	…	…	…	…	…	
2020-06-25	237.550003	1438.040039	58.810001	44.820000	204.699997	…	
2020-07-02	243.580002	1496.000000	58.610001	45.070000	212.830002	…	
2020-07-09	240.279999	1513.640015	59.029999	46.400002	208.039993	…	
2020-07-16	239.869995	1568.489990	61.049999	48.480000	211.750000	…	
2020-07-23	230.710007	1511.869995	50.590000	48.490002	201.300003	…	

Date	diff_GOOG	diff_NFLX	diff_TSLA	diff_AMZN	diff_MSFT	diff_KO	\
2015-01-01	NaN	NaN	NaN	NaN	NaN	NaN	
2015-01-08	-0.045899	-0.904650	-8.656077	-1.725764	-0.584034	-1.000262	

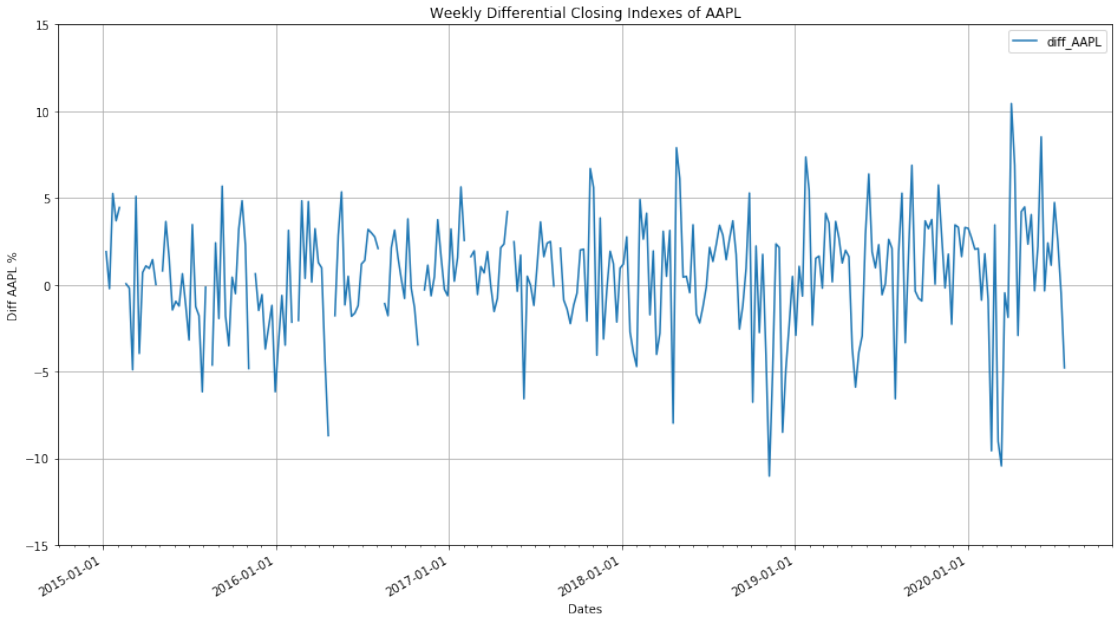
```
2015-01-15   3.428041  26.227485   2.013599   1.357115  -0.087042  1.879714
2015-01-22  -1.552007   8.106922   1.424423   2.240539 -10.300503 -3.321045
2015-01-29   2.501955   1.412550   9.620308  20.019083   1.578043 -0.572487
…                 …          …          …          …          …         …
2020-06-25   0.423896   6.069675  16.524955   5.277211   3.467449  0.380735
2020-07-02   4.030483   3.529360  21.993873   7.031304   3.971668  0.557787
2020-07-09   1.179145   4.073354  13.187835  -2.344609  -2.250627  2.950969
2020-07-16   3.623713  -6.390705   2.996096   3.025714   1.783314  4.482754
2020-07-23  -3.609841  -1.912947 -11.010906  -2.935569  -4.935063  0.020631

            diff_COST   diff_AMD   diff_AAL  diff_INTC
Date
2015-01-01        NaN        NaN        NaN        NaN
2015-01-08  -3.471448   1.937992  -6.791153   0.916126
2015-01-15  -0.122046  -6.844109   8.621720   0.357638
2015-01-22   0.344988   7.346941  -5.682890  -7.401299
2015-01-29  11.674532   8.365011  -2.291573  -0.532859
…                 …          …          …          …
2020-06-25   2.258241   0.362669  -1.763800  -0.473851
2020-07-02   3.796557   1.616581  -6.401254  -0.340080
2020-07-09   3.281489   3.574770  12.093410   0.716598
2020-07-16   0.489738  11.655224 -15.476190   3.421990
2020-07-23  -0.767587  12.315910   0.264091 -17.133496

[303 rows x 24 columns]
```

[12]:
```
df.to_excel('stocks_diff_all.xlsx')
```

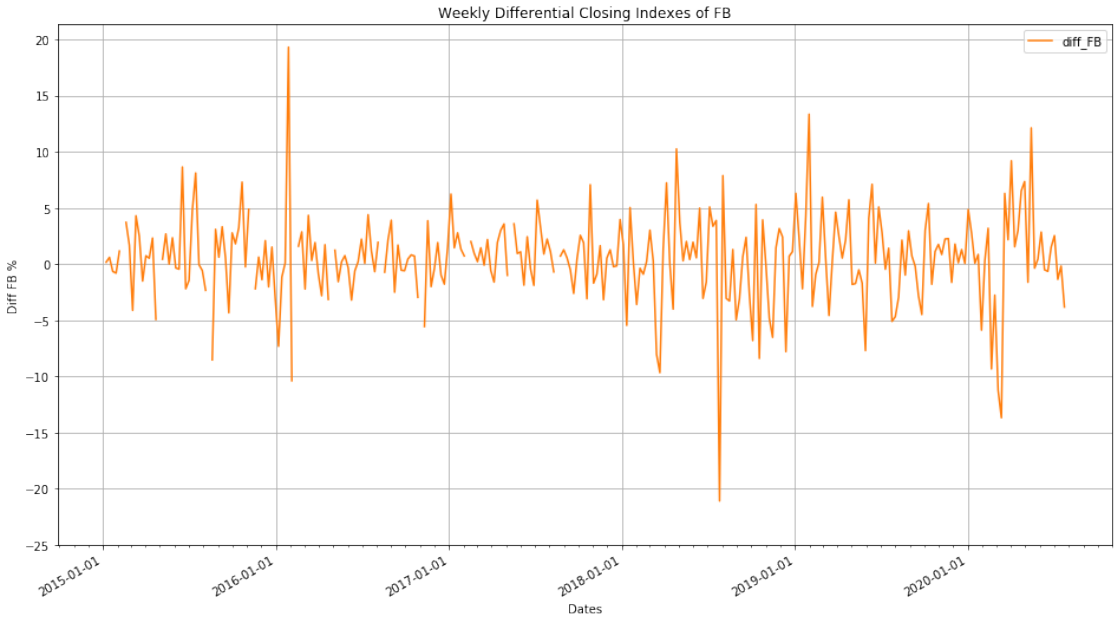
[13]:
```
for i in range(len(tickers)):
    df['diff_{}'.format(tickers[i])].plot(figsize=[16, 9], grid=True,␣
 ↪legend=True, color=colors[i])

plt.xlabel('Dates')
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m-%d'))
plt.gca().xaxis.set_major_locator(mdates.YearLocator())
plt.gca().xaxis.set_minor_locator(mdates.MonthLocator())
plt.gcf().autofmt_xdate()
plt.ylabel('Diff %')
plt.yticks([-50, -40, -30, -20, -10, 0, 10, 20, 30, 40, 50])
plt.title('Weekly Differential Closing Indexes')
plt.show()
```



[14]:
```
for i in range(len(tickers)):
    diff_max = np.nanmax(df['diff_{}'.format(tickers[i])])
    diff_min = np.nanmin(df['diff_{}'.format(tickers[i])])
    diff_yticks = 5 * np.arange(int(np.floor(diff_min / 5)), int(np.
↪ceil(diff_max / 5)) + 1)

    df['diff_{}'.format(tickers[i])].plot(figsize=[16, 9], grid=True,␣
↪legend=True, color=colors[i])
    plt.xlabel('Dates')
    plt.ylabel('Diff {} %'.format(tickers[i]))
    plt.yticks(diff_yticks)
    plt.title('Weekly Differential Closing Indexes of {}'.format(tickers[i]))
    plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m-%d'))
    plt.gca().xaxis.set_major_locator(mdates.YearLocator())
    plt.gca().xaxis.set_minor_locator(mdates.MonthLocator())
    plt.gcf().autofmt_xdate()
    plt.show()

    print(df[['{}'.format(tickers[i]), 'diff_{}'.format(tickers[i])]][np.
↪abs(df['diff_{}'.format(tickers[i])]) >= 10])
```

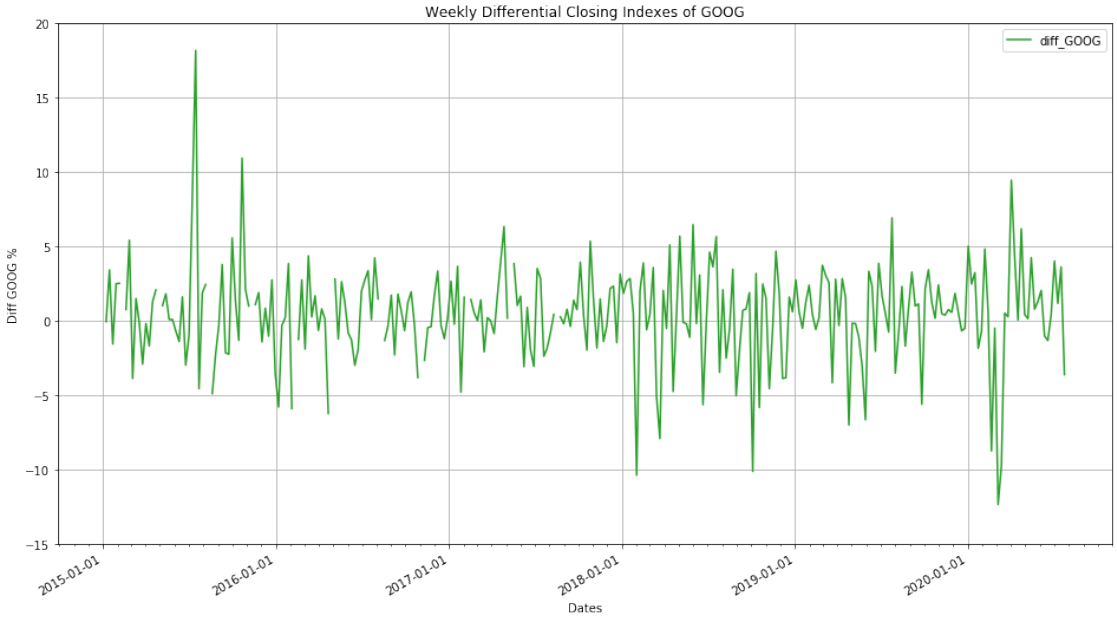



```
                AAPL  diff_AAPL
Date
2018-11-08  182.466843 -11.026420
2020-03-12  246.004074 -10.441855
2020-04-02  265.351715  10.443738
```

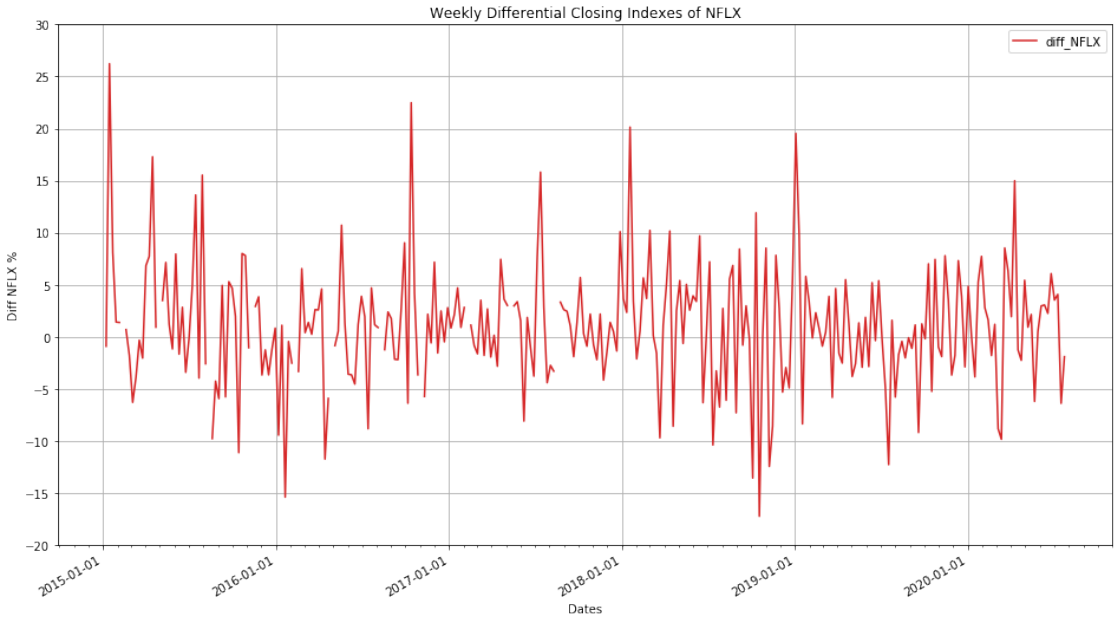



```
                FB    diff_FB
```

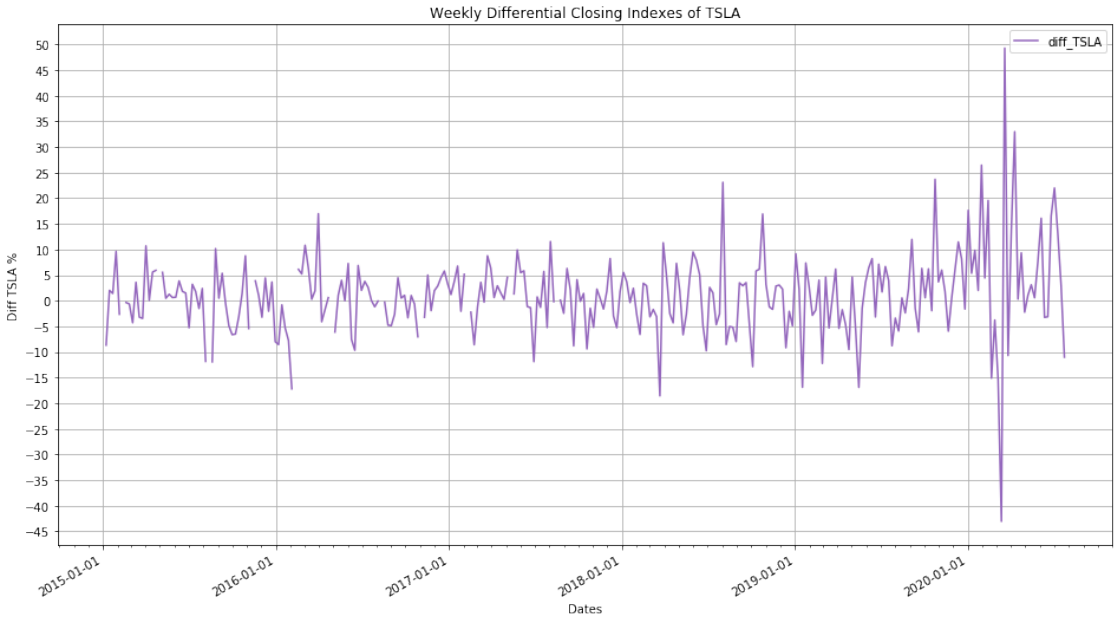
Date		
2016-01-28	112.690002	19.311812
2016-02-04	101.000000	-10.373593
2018-04-26	176.070007	10.257377
2018-07-26	171.649994	-21.080463
2019-01-31	170.490005	13.342646
2020-03-05	170.240005	-11.222356
2020-03-12	146.960007	-13.674811
2020-05-14	229.970001	12.125790



Date	GOOG	diff_GOOG
2015-07-16	662.099976	18.185715
2015-10-22	712.950012	10.945990
2018-02-01	1048.579956	-10.373181
2018-10-04	1081.219971	-10.119289
2020-03-05	1215.410034	-12.340968

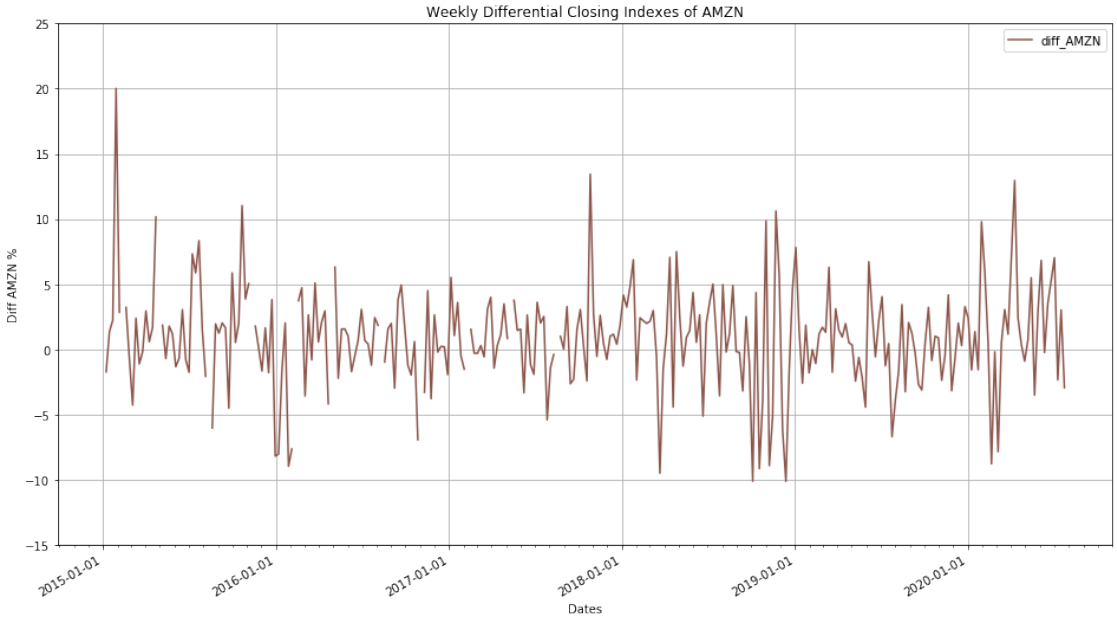


Date	NFLX	diff_NFLX
2015-01-15	58.468571	26.227485
2015-04-16	79.668571	17.292724
2015-07-16	111.500000	13.624787
2015-07-30	123.709999	15.530442
2015-10-15	97.959999	-11.131275
2016-01-21	91.150002	-15.398178
2016-04-14	96.769997	-11.746470
2016-05-19	100.199997	10.718229
2016-10-13	121.870003	22.482415
2017-07-13	183.860001	15.817323
2017-12-28	205.050003	10.099870
2018-01-18	261.299988	20.137925
2018-03-01	321.160004	10.220330
2018-04-12	334.519989	10.159046
2018-07-12	375.130005	-10.395316
2018-10-04	325.890015	-13.568486
2018-10-11	364.700012	11.908925
2018-10-18	301.829987	-17.238833
2018-11-08	286.730011	-12.448852
2019-01-03	319.959991	19.539710
2019-07-18	317.940002	-12.277894
2020-04-09	426.750000	14.989762

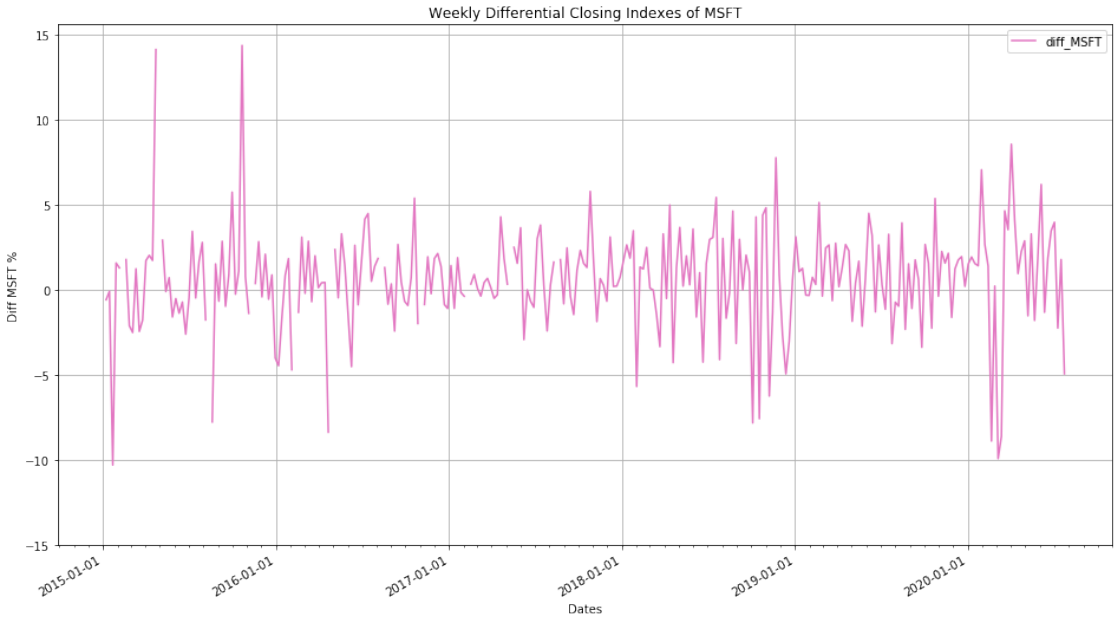


Date	TSLA	diff_TSLA
2015-04-02	207.669998	10.704197
2015-08-06	238.169998	-11.831343
2015-08-20	224.839996	-11.913811
2015-08-27	247.690002	10.162785
2016-02-04	143.669998	-17.183536
2016-03-03	208.720001	10.820859
2016-03-31	265.420013	16.981804
2017-06-29	327.089996	-11.892575
2017-08-03	363.529999	11.549904
2018-03-22	257.779999	-18.560642
2018-03-29	286.940002	11.311973
2018-08-02	370.339996	23.101981
2018-10-04	256.880005	-12.862953
2018-10-25	337.320007	16.922013
2019-01-17	287.589996	-16.893511
2019-02-28	276.239990	-12.232319
2019-05-16	192.729996	-16.908817
2019-09-05	247.100006	11.972093
2019-10-24	315.010010	23.688558
2019-12-12	393.149994	11.468665
2020-01-02	492.140015	17.643973
2020-01-30	734.700012	26.456570
2020-02-13	917.419983	19.566267
2020-02-20	778.799988	-15.109764
2020-03-05	634.229980	-15.379589

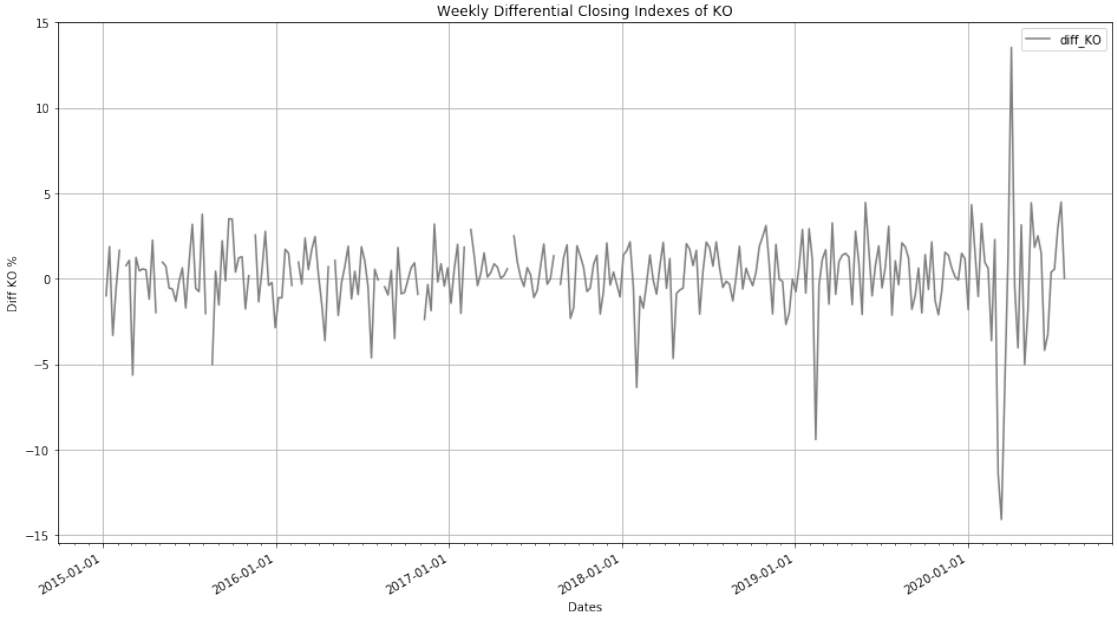
2020-03-12	361.220001	-43.045896
2020-03-19	539.250000	49.285753
2020-03-26	481.559998	-10.698192
2020-04-02	548.840027	13.971266
2020-04-09	729.830017	32.976820
2020-06-04	1025.050049	16.092464
2020-06-25	1119.630005	16.524955
2020-07-02	1365.880005	21.993873
2020-07-09	1546.010010	13.187835
2020-07-23	1417.000000	-11.010906



Date	AMZN	diff_AMZN
2015-01-29	364.750000	20.019083
2015-04-23	429.369995	10.151362
2015-10-22	617.099976	11.035132
2017-10-26	1103.680054	13.441129
2018-10-04	1755.250000	-10.114403
2018-11-22	1677.750000	10.616261
2018-12-13	1495.079956	-10.126602
2020-04-09	2307.679932	12.955454

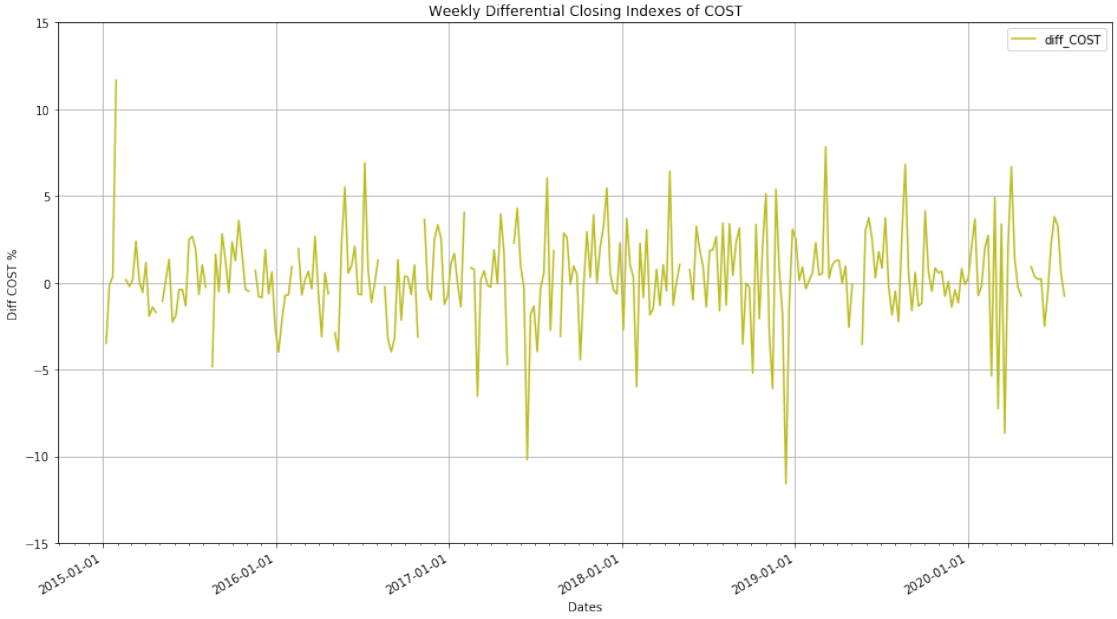



```
                MSFT  diff_MSFT
Date
2015-01-22  36.775524 -10.300503
2015-04-23  44.113792  14.119567
2015-10-22  49.175350  14.364394
```

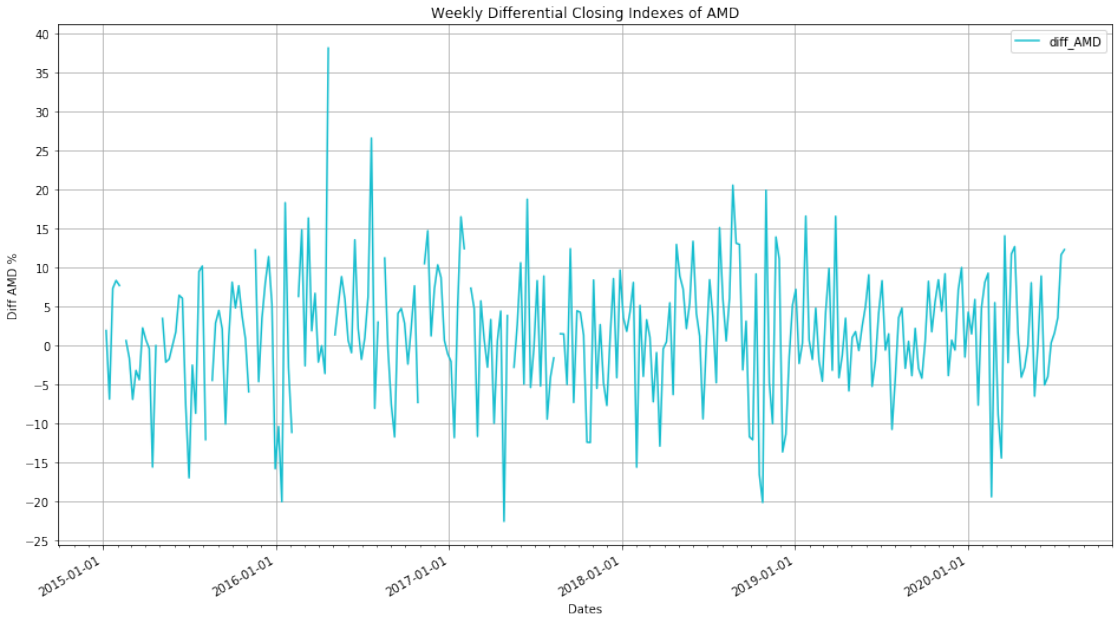



```
                KO    diff_KO
```

```
Date
2020-03-05  51.290131 -11.388316
2020-03-12  44.059799 -14.096925
2020-04-02  47.389473  13.532772
```

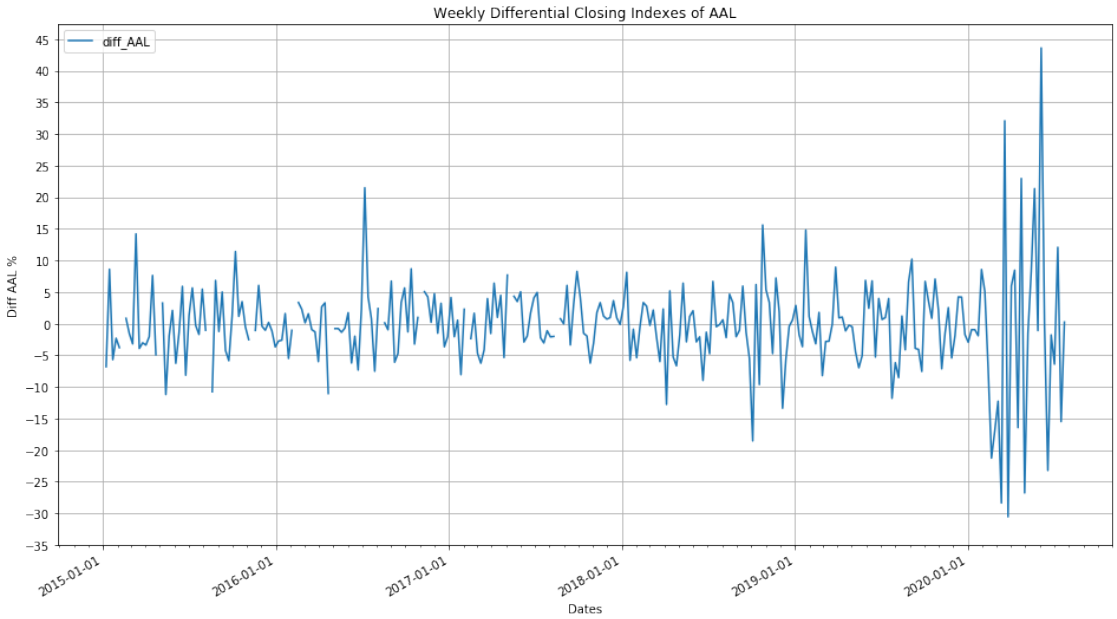



```
                 COST  diff_COST
Date
2015-01-29  136.853317  11.674532
2017-06-15  158.208633 -10.194322
2018-12-13  198.577652 -11.584730
```



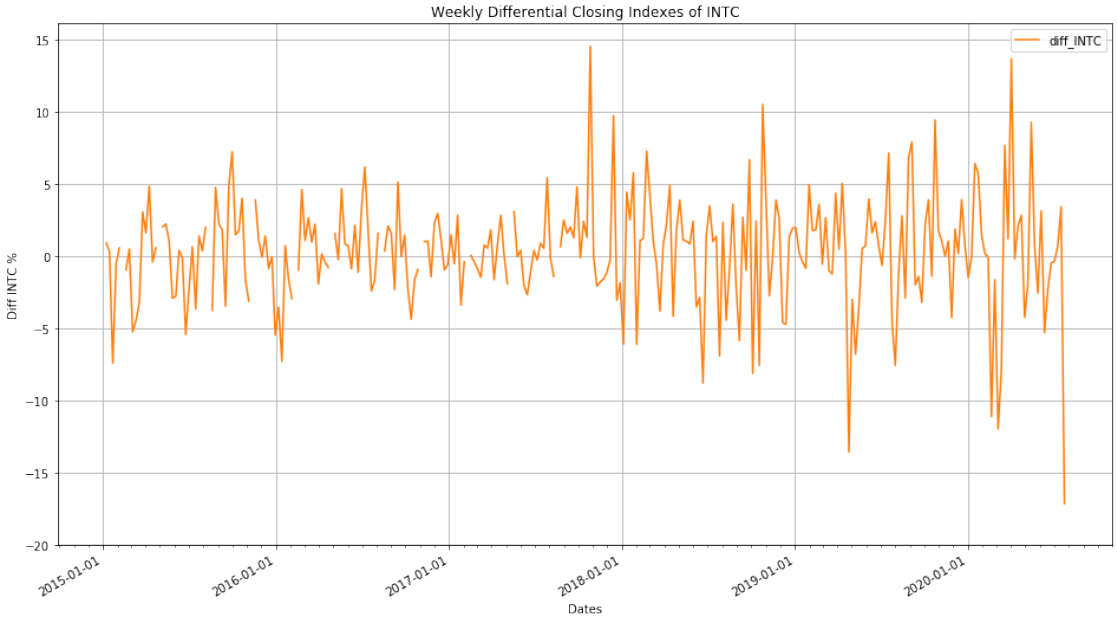
Date	AMD	diff_AMD
2015-04-16	2.280000	-15.555558
2015-07-02	2.010000	-16.942152
2015-07-30	2.160000	10.204084
2015-08-06	1.900000	-12.037042
2015-09-17	1.700000	-10.052907
2015-11-19	2.380000	12.264162
2015-12-17	2.830000	11.417322
2015-12-31	2.510000	-15.771813
2016-01-07	2.250000	-10.358565
2016-01-14	1.800000	-20.000002
2016-01-21	2.130000	18.333343
2016-02-04	1.840000	-11.111107
2016-02-25	2.320000	14.851483
2016-03-10	2.630000	16.371687
2016-04-21	3.730000	38.148146
2016-06-16	5.020000	13.574658
2016-07-21	6.850000	26.617377
2016-08-18	7.430000	11.227545
2016-09-08	6.040000	-11.695909
2016-11-10	7.670000	10.518732
2016-11-17	8.800000	14.732726
2016-12-08	10.550000	10.355646
2017-01-12	9.880000	-11.785712
2017-01-26	12.060000	16.521739
2017-02-02	13.560000	12.437811

```
2017-03-02  13.220000 -11.631014
2017-04-27  10.390000 -22.520504
2017-06-01  12.380000  10.634500
2017-06-15  13.980000  18.776542
2017-09-14  13.740000  12.438621
2017-10-19  12.330000 -12.366736
2017-10-26  10.800000 -12.408757
2018-02-01  11.600000 -15.574959
2018-03-22   9.810000 -12.877440
2018-04-26  10.970000  12.976315
2018-05-31  15.670000  13.386400
2018-07-26  18.480000  15.140190
2018-08-23  25.200001  20.574169
2018-08-30  28.510000  13.134918
2018-09-06  32.209999  12.977898
2018-09-27  28.430000 -11.680641
2018-10-04  25.000000 -12.064721
2018-10-18  22.790001 -16.520141
2018-10-25  18.209999 -20.096541
2018-11-01  21.840000  19.934109
2018-11-22  21.340000  13.934867
2018-11-29  23.709999  11.105899
2018-12-06  20.480000 -13.622943
2018-12-13  18.160000 -11.328124
2019-01-24  23.090000  16.616167
2019-03-28  29.020000  16.593014
2019-07-25  30.450001 -10.729991
2019-12-19  46.540001  10.023645
2020-02-20  47.490002 -19.371816
2020-03-12  39.119999 -14.398253
2020-03-19  44.630001  14.084873
2020-04-02  48.790001  11.749888
2020-04-09  54.990002  12.707523
2020-07-16  61.790001  11.655224
2020-07-23  69.400002  12.315910
```



Date	AAL	diff_AAL
2015-03-12	51.392273	14.222393
2015-05-14	40.965961	-11.198215
2015-08-20	37.046940	-10.751201
2015-10-08	41.767197	11.421990
2016-04-21	35.177475	-11.027591
2016-07-07	33.107258	21.516758
2018-04-05	44.334946	-12.759403
2018-10-04	31.021378	-18.530924
2018-10-25	34.426765	15.622948
2018-12-06	33.803402	-13.366963
2019-01-24	35.761734	14.818326
2019-07-25	30.193247	-11.795326
2019-09-05	29.554626	10.222223
2020-02-20	22.309999	-21.249560
2020-02-27	18.530001	-16.943070
2020-03-05	16.260000	-12.250407
2020-03-12	11.650000	-28.351787
2020-03-19	15.390000	32.103012
2020-03-26	10.690000	-30.539316
2020-04-16	10.270000	-16.436123
2020-04-23	12.630000	22.979548
2020-04-30	9.250000	-26.761679
2020-05-21	11.980000	21.377910
2020-06-04	17.020000	43.628691
2020-06-18	13.040000	-23.203767

```
2020-07-09  13.440000  12.093410
2020-07-16  11.360000 -15.476190
```




```
                INTC  diff_INTC
Date
2017-10-26  43.728912  14.541442
2018-10-25  44.976070  10.513916
2019-04-25  49.315586 -13.555855
2020-02-20  59.314945 -11.116075
2020-03-05  51.369823 -11.963195
2020-04-02  58.648708  13.685429
2020-07-23  50.590000 -17.133496
```

[ ]: