In [1]:

**import** **pandas** **as** **pd**

**import** **numpy** **as** **np**

**import** **matplotlib.pyplot** **as** **plt**

%matplotlib inline

In [2]:

data = pd.read\_csv('data\_stocks.csv')

In [3]:

data.head()

Out[3]:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **DATE** | **SP500** | **NASDAQ.AAL** | **NASDAQ.AAPL** | **NASDAQ.ADBE** | **NASDAQ.ADI** | **NASDAQ.ADP** | **NASDAQ.ADSK** | **NASDAQ.AKAM** | **NASDAQ.ALXN** | **...** | **NYSE.WYN** | **NYSE.XEC** | **NYSE.XEL** | **NYSE.XL** | **NYSE.XOM** | **NYSE.XRX** | **NYSE.XYL** | **NYSE.YUM** | **NYSE.ZBH** | **NYSE.ZTS** |
| **0** | 1491226200 | 2363.6101 | 42.3300 | 143.6800 | 129.6300 | 82.040 | 102.2300 | 85.2200 | 59.760 | 121.52 | ... | 84.370 | 119.035 | 44.40 | 39.88 | 82.03 | 7.36 | 50.22 | 63.86 | 122.000 | 53.350 |
| **1** | 1491226260 | 2364.1001 | 42.3600 | 143.7000 | 130.3200 | 82.080 | 102.1400 | 85.6500 | 59.840 | 121.48 | ... | 84.370 | 119.035 | 44.11 | 39.88 | 82.03 | 7.38 | 50.22 | 63.74 | 121.770 | 53.350 |
| **2** | 1491226320 | 2362.6799 | 42.3100 | 143.6901 | 130.2250 | 82.030 | 102.2125 | 85.5100 | 59.795 | 121.93 | ... | 84.585 | 119.260 | 44.09 | 39.98 | 82.02 | 7.36 | 50.12 | 63.75 | 121.700 | 53.365 |
| **3** | 1491226380 | 2364.3101 | 42.3700 | 143.6400 | 130.0729 | 82.000 | 102.1400 | 85.4872 | 59.620 | 121.44 | ... | 84.460 | 119.260 | 44.25 | 39.99 | 82.02 | 7.35 | 50.16 | 63.88 | 121.700 | 53.380 |
| **4** | 1491226440 | 2364.8501 | 42.5378 | 143.6600 | 129.8800 | 82.035 | 102.0600 | 85.7001 | 59.620 | 121.60 | ... | 84.470 | 119.610 | 44.11 | 39.96 | 82.03 | 7.36 | 50.20 | 63.91 | 121.695 | 53.240 |

5 rows × 502 columns

In [4]:

data.columns.tolist()[0]

Out[4]:

'DATE'

In [5]:

data.set\_index(data.columns.tolist()[0],inplace=**True**)

In [6]:

data.head()

Out[6]:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **SP500** | **NASDAQ.AAL** | **NASDAQ.AAPL** | **NASDAQ.ADBE** | **NASDAQ.ADI** | **NASDAQ.ADP** | **NASDAQ.ADSK** | **NASDAQ.AKAM** | **NASDAQ.ALXN** | **NASDAQ.AMAT** | **...** | **NYSE.WYN** | **NYSE.XEC** | **NYSE.XEL** | **NYSE.XL** | **NYSE.XOM** | **NYSE.XRX** | **NYSE.XYL** | **NYSE.YUM** | **NYSE.ZBH** | **NYSE.ZTS** |
| **DATE** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1491226200** | 2363.6101 | 42.3300 | 143.6800 | 129.6300 | 82.040 | 102.2300 | 85.2200 | 59.760 | 121.52 | 38.99 | ... | 84.370 | 119.035 | 44.40 | 39.88 | 82.03 | 7.36 | 50.22 | 63.86 | 122.000 | 53.350 |
| **1491226260** | 2364.1001 | 42.3600 | 143.7000 | 130.3200 | 82.080 | 102.1400 | 85.6500 | 59.840 | 121.48 | 39.01 | ... | 84.370 | 119.035 | 44.11 | 39.88 | 82.03 | 7.38 | 50.22 | 63.74 | 121.770 | 53.350 |
| **1491226320** | 2362.6799 | 42.3100 | 143.6901 | 130.2250 | 82.030 | 102.2125 | 85.5100 | 59.795 | 121.93 | 38.91 | ... | 84.585 | 119.260 | 44.09 | 39.98 | 82.02 | 7.36 | 50.12 | 63.75 | 121.700 | 53.365 |
| **1491226380** | 2364.3101 | 42.3700 | 143.6400 | 130.0729 | 82.000 | 102.1400 | 85.4872 | 59.620 | 121.44 | 38.84 | ... | 84.460 | 119.260 | 44.25 | 39.99 | 82.02 | 7.35 | 50.16 | 63.88 | 121.700 | 53.380 |
| **1491226440** | 2364.8501 | 42.5378 | 143.6600 | 129.8800 | 82.035 | 102.0600 | 85.7001 | 59.620 | 121.60 | 38.93 | ... | 84.470 | 119.610 | 44.11 | 39.96 | 82.03 | 7.36 | 50.20 | 63.91 | 121.695 | 53.240 |

5 rows × 501 columns

In [7]:

data.drop('SP500', axis= 1, inplace= **True**)

In [8]:

**from** **sklearn** **import** cluster, datasets **as** dt

In [9]:

n\_clusters = 5

In [10]:

np.random.seed(0)

k\_Means= cluster.KMeans(n\_clusters=n\_clusters, random\_state= 25)

k\_Means.fit(data)

print('Over')

Over

In [11]:

k\_Means.labels\_

Out[11]:

array([4, 4, 4, ..., 3, 3, 3])

In [12]:

len(k\_Means.labels\_)

Out[12]:

41266

In [13]:

k\_Means.cluster\_centers\_

Out[13]:

array([[ 51.72587735, 151.06200533, 146.92356499, ..., 74.45093304,

125.61319585, 62.33830194],

[ 45.09849074, 151.07393826, 135.6250932 , ..., 68.61520574,

118.89374393, 58.85615306],

[ 49.26457875, 148.64841274, 141.54205293, ..., 73.30668488,

124.86423224, 62.56597652],

[ 46.2734716 , 160.11520883, 150.24007608, ..., 76.00650235,

112.7923117 , 61.19815181],

[ 43.77358132, 142.98696747, 130.74934798, ..., 64.62368362,

120.79942516, 53.63660808]])

In [14]:

data['Cluster\_No'] = k\_Means.labels\_

In [15]:

data.head()

Out[15]:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **NASDAQ.AAL** | **NASDAQ.AAPL** | **NASDAQ.ADBE** | **NASDAQ.ADI** | **NASDAQ.ADP** | **NASDAQ.ADSK** | **NASDAQ.AKAM** | **NASDAQ.ALXN** | **NASDAQ.AMAT** | **NASDAQ.AMD** | **...** | **NYSE.XEC** | **NYSE.XEL** | **NYSE.XL** | **NYSE.XOM** | **NYSE.XRX** | **NYSE.XYL** | **NYSE.YUM** | **NYSE.ZBH** | **NYSE.ZTS** | **Cluster\_No** |
| **DATE** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **1491226200** | 42.3300 | 143.6800 | 129.6300 | 82.040 | 102.2300 | 85.2200 | 59.760 | 121.52 | 38.99 | 14.61 | ... | 119.035 | 44.40 | 39.88 | 82.03 | 7.36 | 50.22 | 63.86 | 122.000 | 53.350 | 4 |
| **1491226260** | 42.3600 | 143.7000 | 130.3200 | 82.080 | 102.1400 | 85.6500 | 59.840 | 121.48 | 39.01 | 14.71 | ... | 119.035 | 44.11 | 39.88 | 82.03 | 7.38 | 50.22 | 63.74 | 121.770 | 53.350 | 4 |
| **1491226320** | 42.3100 | 143.6901 | 130.2250 | 82.030 | 102.2125 | 85.5100 | 59.795 | 121.93 | 38.91 | 14.64 | ... | 119.260 | 44.09 | 39.98 | 82.02 | 7.36 | 50.12 | 63.75 | 121.700 | 53.365 | 4 |
| **1491226380** | 42.3700 | 143.6400 | 130.0729 | 82.000 | 102.1400 | 85.4872 | 59.620 | 121.44 | 38.84 | 14.63 | ... | 119.260 | 44.25 | 39.99 | 82.02 | 7.35 | 50.16 | 63.88 | 121.700 | 53.380 | 4 |
| **1491226440** | 42.5378 | 143.6600 | 129.8800 | 82.035 | 102.0600 | 85.7001 | 59.620 | 121.60 | 38.93 | 14.67 | ... | 119.610 | 44.11 | 39.96 | 82.03 | 7.36 | 50.20 | 63.91 | 121.695 | 53.240 | 4 |

5 rows × 501 columns

**Problem 1:¶**

There are various stocks for which we have collected a data set, which all stocks are apparently similar in performance

In [16]:

companyMean = data.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(data.columns)-1]

companyName = data.columns.tolist()[:len(data.columns)-1]

companyStd= data.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(data.columns)-1]

companyData = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [17]:

companyData.sort\_values('cStd', ascending=**False**)

Out[17]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Company** | **cMean** | **cStd** |
| **91** | NASDAQ.PCLN | 1867.722306 | 78.341014 |
| **163** | NYSE.AZO | 598.324633 | 76.696839 |
| **65** | NASDAQ.ISRG | 901.320301 | 66.399068 |
| **199** | NYSE.CMG | 417.670807 | 63.347703 |
| **96** | NASDAQ.REGN | 457.676948 | 46.809471 |
| **52** | NASDAQ.GOOGL | 941.369665 | 43.450668 |
| **51** | NASDAQ.GOOG | 922.131875 | 42.227541 |
| **11** | NASDAQ.AMZN | 968.747188 | 38.966212 |
| **87** | NASDAQ.ORLY | 224.415988 | 27.904559 |
| **21** | NASDAQ.CHTR | 348.719389 | 25.256789 |
| **110** | NASDAQ.ULTA | 274.241713 | 25.112028 |
| **84** | NASDAQ.NVDA | 141.419298 | 25.003991 |
| **164** | NYSE.BA | 202.534020 | 22.736921 |
| **169** | NYSE.BCR | 306.158251 | 22.395830 |
| **208** | NYSE.COO | 228.707758 | 20.094502 |
| **175** | NYSE.BLK | 409.701703 | 19.228577 |
| **124** | NYSE.AAP | 123.296657 | 19.017181 |
| **280** | NYSE.GWW | 180.151759 | 18.426612 |
| **37** | NASDAQ.EQIX | 432.201844 | 17.370001 |
| **410** | NYSE.PXD | 161.082634 | 17.154405 |
| **447** | NYSE.TDG | 263.070526 | 17.060790 |
| **114** | NASDAQ.VRTX | 131.767415 | 16.472286 |
| **176** | NYSE.BLL | 51.297688 | 15.874231 |
| **15** | NASDAQ.BIIB | 272.957640 | 13.763828 |
| **428** | NYSE.SHW | 338.119248 | 13.586502 |
| **263** | NYSE.FL | 56.910060 | 13.280365 |
| **13** | NASDAQ.AVGO | 238.598238 | 12.824722 |
| **130** | NYSE.ADS | 244.854817 | 12.653201 |
| **81** | NASDAQ.NFLX | 160.850341 | 12.570000 |
| **162** | NYSE.AYI | 183.341470 | 12.518941 |
| **...** | ... | ... | ... |
| **485** | NYSE.WMB | 30.102558 | 0.927698 |
| **63** | NASDAQ.INTC | 35.489340 | 0.915785 |
| **53** | NASDAQ.GRMN | 51.344658 | 0.915121 |
| **489** | NYSE.WY | 33.248472 | 0.907998 |
| **320** | NYSE.KMI | 19.757229 | 0.906750 |
| **423** | NYSE.RSG | 63.788602 | 0.896241 |
| **245** | NYSE.EMR | 59.387318 | 0.887733 |
| **338** | NYSE.LUK | 25.528150 | 0.872308 |
| **62** | NASDAQ.INFO | 46.288736 | 0.850310 |
| **327** | NYSE.L | 47.246764 | 0.816147 |
| **359** | NYSE.MON | 116.911295 | 0.809861 |
| **292** | NYSE.HPE | 17.744404 | 0.763962 |
| **470** | NYSE.USB | 51.863284 | 0.760711 |
| **402** | NYSE.PPL | 38.609105 | 0.734429 |
| **165** | NYSE.BAC | 23.701408 | 0.674120 |
| **194** | NYSE.CHK | 4.990837 | 0.654819 |
| **293** | NYSE.HPQ | 18.534090 | 0.648805 |
| **203** | NYSE.CNP | 28.212311 | 0.647618 |
| **389** | NYSE.PFE | 33.269778 | 0.593905 |
| **317** | NYSE.KEY | 18.160407 | 0.577986 |
| **85** | NASDAQ.NWS | 13.715717 | 0.556995 |
| **86** | NASDAQ.NWSA | 13.316591 | 0.532835 |
| **101** | NASDAQ.SPLS | 9.666552 | 0.506469 |
| **297** | NYSE.HST | 18.272814 | 0.424607 |
| **89** | NASDAQ.PBCT | 17.362005 | 0.420245 |
| **414** | NYSE.RF | 14.260065 | 0.378946 |
| **488** | NYSE.WU | 19.272765 | 0.362158 |
| **56** | NASDAQ.HBAN | 13.051218 | 0.358746 |
| **133** | NYSE.AES | 11.347591 | 0.281469 |
| **257** | NYSE.F | 11.139757 | 0.262379 |

500 rows × 3 columns

In [18]:

df\_1 = data.loc[data.Cluster\_No == 0,:]

df\_2 = data.loc[data.Cluster\_No == 1,:]

df\_3 = data.loc[data.Cluster\_No == 2,:]

df\_4 = data.loc[data.Cluster\_No == 3,:]

df\_5 = data.loc[data.Cluster\_No == 4,:]

In [19]:

companyMean = df\_1.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(df\_1.columns)-1]

companyName = df\_1.columns.tolist()[:len(df\_1.columns)-1]

companyStd= df\_1.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(df\_1.columns)-1]

companydf\_1 = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [20]:

companydf\_1.shape

Out[20]:

(500, 3)

In [21]:

companydf\_1.head()

Out[21]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Company** | **cMean** | **cStd** |
| **0** | NASDAQ.AAL | 51.725877 | 1.430593 |
| **1** | NASDAQ.AAPL | 151.062005 | 4.324873 |
| **2** | NASDAQ.ADBE | 146.923565 | 2.093385 |
| **3** | NASDAQ.ADI | 79.556993 | 1.031093 |
| **4** | NASDAQ.ADP | 107.410306 | 5.979057 |

In [22]:

pd.Series(k\_Means.labels\_).value\_counts()

Out[22]:

2 12029

0 8905

4 6976

1 6709

3 6647

dtype: int64

In [23]:

companyData.sort\_values('cStd', ascending=**False**, inplace= **True**)

In [24]:

companydf\_1.sort\_values('cStd', ascending=**False**, inplace= **True**)

In [25]:

companyData.head()

Out[25]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Company** | **cMean** | **cStd** |
| **91** | NASDAQ.PCLN | 1867.722306 | 78.341014 |
| **163** | NYSE.AZO | 598.324633 | 76.696839 |
| **65** | NASDAQ.ISRG | 901.320301 | 66.399068 |
| **199** | NYSE.CMG | 417.670807 | 63.347703 |
| **96** | NASDAQ.REGN | 457.676948 | 46.809471 |

In [26]:

companydf\_1.head()

Out[26]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Company** | **cMean** | **cStd** |
| **91** | NASDAQ.PCLN | 1992.227976 | 41.105572 |
| **199** | NYSE.CMG | 364.615179 | 25.146872 |
| **11** | NASDAQ.AMZN | 1010.326946 | 22.047485 |
| **21** | NASDAQ.CHTR | 358.311887 | 19.844799 |
| **52** | NASDAQ.GOOGL | 963.180313 | 18.290805 |

In [27]:

companydf\_1.set\_index('Company', inplace=**True**)

In [28]:

companyData.set\_index('Company', inplace=**True**)

In [29]:

companyMean = df\_2.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(df\_2.columns)-1]

companyName = df\_2.columns.tolist()[:len(df\_2.columns)-1]

companyStd= df\_2.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(df\_2.columns)-1]

companydf\_2 = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [30]:

companyMean = df\_3.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(df\_3.columns)-1]

companyName = df\_3.columns.tolist()[:len(df\_3.columns)-1]

companyStd= df\_3.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(df\_3.columns)-1]

companydf\_3 = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [31]:

companyMean = df\_4.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(df\_4.columns)-1]

companyName = df\_4.columns.tolist()[:len(df\_4.columns)-1]

companyStd= df\_4.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(df\_4.columns)-1]

companydf\_4 = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [32]:

companyMean = df\_5.apply(np.mean, axis= 0)

companyM = companyMean.tolist()[:len(df\_5.columns)-1]

companyName = df\_5.columns.tolist()[:len(df\_5.columns)-1]

companyStd= df\_5.apply(np.std, axis= 0)

companyStdD= companyStd.tolist()[:len(df\_5.columns)-1]

companydf\_5 = pd.DataFrame({'Company':companyName , 'cMean':companyM, 'cStd': companyStdD})

In [33]:

companydf\_2.set\_index('Company', inplace=**True**)

companydf\_3.set\_index('Company', inplace=**True**)

companydf\_4.set\_index('Company', inplace=**True**)

companydf\_5.set\_index('Company', inplace=**True**)

In [34]:

companyData['Cluster\_1\_Mean'] = companydf\_1.cMean

companyData['Cluster\_2\_Mean'] = companydf\_2.cMean

companyData['Cluster\_3\_Mean'] = companydf\_3.cMean

companyData['Cluster\_4\_Mean'] = companydf\_4.cMean

companyData['Cluster\_5\_Mean'] = companydf\_5.cMean

In [35]:

companyData['Cluster\_1\_Std'] = companydf\_1.cStd

companyData['Cluster\_2\_Std'] = companydf\_2.cStd

companyData['Cluster\_3\_Std'] = companydf\_3.cStd

companyData['Cluster\_4\_Std'] = companydf\_4.cStd

companyData['Cluster\_5\_Std'] = companydf\_5.cStd

In [36]:

companyData.head()

Out[36]:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **cMean** | **cStd** | **Cluster\_1\_Mean** | **Cluster\_2\_Mean** | **Cluster\_3\_Mean** | **Cluster\_4\_Mean** | **Cluster\_5\_Mean** | **Cluster\_1\_Std** | **Cluster\_2\_Std** | **Cluster\_3\_Std** | **Cluster\_4\_Std** | **Cluster\_5\_Std** |
| **Company** |  |  |  |  |  |  |  |  |  |  |  |  |
| **NASDAQ.PCLN** | 1867.722306 | 78.341014 | 1992.227976 | 1848.808425 | 1861.036039 | 1831.703246 | 1772.828103 | 41.105572 | 37.515946 | 26.179982 | 29.979041 | 17.485453 |
| **NYSE.AZO** | 598.324633 | 76.696839 | 516.246706 | 687.793454 | 590.770040 | 521.993111 | 702.812512 | 16.607688 | 14.098184 | 26.866113 | 8.289924 | 8.821414 |
| **NASDAQ.ISRG** | 901.320301 | 66.399068 | 942.628129 | 850.114460 | 924.396511 | 976.244296 | 786.654340 | 9.697559 | 13.179252 | 16.860209 | 19.677591 | 28.612941 |
| **NYSE.CMG** | 417.670807 | 63.347703 | 364.615179 | 482.611264 | 449.560705 | 316.192173 | 464.646114 | 25.146872 | 7.184968 | 26.506990 | 9.434518 | 12.778987 |
| **NASDAQ.REGN** | 457.676948 | 46.809471 | 496.393374 | 430.232695 | 483.194189 | 475.714862 | 373.460891 | 16.965731 | 23.184139 | 21.629249 | 8.629385 | 6.040844 |

In [37]:

companyData['lift\_1'] = pd.Series(companyData.cStd/companyData.Cluster\_1\_Std)

In [38]:

companyData.sort\_values('lift\_1', inplace=**True**, ascending=**False**)

companyData.lift\_1.head()

Out[38]:

Company

NYSE.BHI inf

NASDAQ.WFM 34.848157

NYSE.BLL 25.800289

NYSE.BCR 16.166194

NASDAQ.SPLS 14.218695

Name: lift\_1, dtype: float64

In [39]:

companyData['lift\_2'] = pd.Series(companyData.cStd/companyData.Cluster\_2\_Std)

companyData['lift\_3'] = pd.Series(companyData.cStd/companyData.Cluster\_3\_Std)

companyData['lift\_4'] = pd.Series(companyData.cStd/companyData.Cluster\_4\_Std)

companyData['lift\_5'] = pd.Series(companyData.cStd/companyData.Cluster\_5\_Std)

In [40]:

companyData.loc['NASDAQ.ISRG']

Out[40]:

cMean 901.320301

cStd 66.399068

Cluster\_1\_Mean 942.628129

Cluster\_2\_Mean 850.114460

Cluster\_3\_Mean 924.396511

Cluster\_4\_Mean 976.244296

Cluster\_5\_Mean 786.654340

Cluster\_1\_Std 9.697559

Cluster\_2\_Std 13.179252

Cluster\_3\_Std 16.860209

Cluster\_4\_Std 19.677591

Cluster\_5\_Std 28.612941

lift\_1 6.846988

lift\_2 5.038152

lift\_3 3.938211

lift\_4 3.374349

lift\_5 2.320596

Name: NASDAQ.ISRG, dtype: float64

In [41]:

companyData.loc['NYSE.ROP']

Out[41]:

cMean 226.125881

cStd 9.239688

Cluster\_1\_Mean 234.816249

Cluster\_2\_Mean 220.412763

Cluster\_3\_Mean 230.114628

Cluster\_4\_Mean 230.905701

Cluster\_5\_Mean 209.094555

Cluster\_1\_Std 1.622447

Cluster\_2\_Std 2.230649

Cluster\_3\_Std 2.774568

Cluster\_4\_Std 1.811958

Cluster\_5\_Std 3.660084

lift\_1 5.694909

lift\_2 4.142153

lift\_3 3.330135

lift\_4 5.099283

lift\_5 2.524447

Name: NYSE.ROP, dtype: float64

In [42]:

companyLifts = companyData[['lift\_1','lift\_2','lift\_3','lift\_4','lift\_5']]

In [43]:

max = companyLifts.loc[companyLifts.lift\_1 != np.inf].max()

companyLifts.replace(np.inf, max, inplace=**True**)

C:\Anaconda3\lib\site-packages\ipykernel\_launcher.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy

In [44]:

companyLifts.head()

print(np.std(companyLifts.lift\_1))

print(np.mean(companyLifts.lift\_1))

2.769257065856375

2.6237397287442903

In [45]:

k\_Means= cluster.KMeans(n\_clusters = 30, random\_state= 25, max\_iter=200)

k\_Means.fit(companyLifts)

Out[45]:

KMeans(algorithm='auto', copy\_x=True, init='k-means++', max\_iter=200,

n\_clusters=30, n\_init=10, n\_jobs=1, precompute\_distances='auto',

random\_state=25, tol=0.0001, verbose=0)

In [46]:

companyLifts['Cluster\_No'] = k\_Means.labels\_

C:\Anaconda3\lib\site-packages\ipykernel\_launcher.py:1: 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: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy

"""Entry point for launching an IPython kernel.

In [47]:

companyLifts.head()

Out[47]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **lift\_1** | **lift\_2** | **lift\_3** | **lift\_4** | **lift\_5** | **Cluster\_No** |
| **Company** |  |  |  |  |  |  |
| **NYSE.BHI** | 34.848157 | 11.158152 | 2.150664 | 1.669370e+13 | 9.714463 | 1 |
| **NASDAQ.WFM** | 34.848157 | 12.926240 | 0.993825 | 3.319598e+01 | 1.598287 | 15 |
| **NYSE.BLL** | 25.800289 | 0.954455 | 21.275511 | 2.757203e+01 | 12.416564 | 11 |
| **NYSE.BCR** | 16.166194 | 25.007256 | 8.046074 | 2.837327e+01 | 1.009736 | 10 |
| **NASDAQ.SPLS** | 14.218695 | 1.337905 | 1.207683 | 2.683804e+01 | 1.820849 | 18 |

In [48]:

pd.Series(k\_Means.labels\_).value\_counts()

Out[48]:

0 98

27 73

16 67

8 62

22 53

12 42

19 41

29 12

23 10

21 8

20 8

28 7

24 2

25 2

26 2

6 1

1 1

2 1

3 1

4 1

5 1

10 1

7 1

11 1

13 1

15 1

18 1

14 1

dtype: int64

In [147]:

companyCluster\_1 = companyLifts.loc[companyLifts.Cluster\_No == 0,:]

companyCluster\_27 = companyLifts.loc[companyLifts.Cluster\_No == 27,:]

companyCluster\_16 = companyLifts.loc[companyLifts.Cluster\_No == 16,:]

companyCluster\_8 = companyLifts.loc[companyLifts.Cluster\_No == 8,:]

companyCluster\_22 = companyLifts.loc[companyLifts.Cluster\_No == 22,:]

companyCluster\_12 = companyLifts.loc[companyLifts.Cluster\_No == 12,:]

companyCluster\_19 = companyLifts.loc[companyLifts.Cluster\_No == 19,:]

companyCluster\_29 = companyLifts.loc[companyLifts.Cluster\_No == 29,:]

companyCluster\_23 = companyLifts.loc[companyLifts.Cluster\_No == 23,:]

writer = pd.ExcelWriter('companyLiftCluster.xlsx')

companyCluster\_1.to\_excel(writer,'Sheet1')

companyCluster\_27.to\_excel(writer,'Sheet2')

companyCluster\_16.to\_excel(writer,'Sheet3')

companyCluster\_8.to\_excel(writer,'Sheet4')

companyCluster\_22.to\_excel(writer,'Sheet5')

companyCluster\_12.to\_excel(writer,'Sheet6')

companyCluster\_19.to\_excel(writer,'Sheet7')

companyCluster\_29.to\_excel(writer,'Sheet8')

companyCluster\_23.to\_excel(writer,'Sheet9')

writer.save()

In [152]:

companyData.loc['NASDAQ.WYNN']

Out[152]:

cMean 128.505151

cStd 6.975419

Cluster\_1\_Mean 131.644940

Cluster\_2\_Mean 124.095096

Cluster\_3\_Mean 132.623519

Cluster\_4\_Mean 133.423110

Cluster\_5\_Mean 116.950926

Cluster\_1\_Std 3.519493

Cluster\_2\_Std 1.971915

Cluster\_3\_Std 4.146181

Cluster\_4\_Std 3.581380

Cluster\_5\_Std 3.054034

lift\_1 1.981939

lift\_2 3.537384

lift\_3 1.682372

lift\_4 1.947690

lift\_5 2.284002

Name: NASDAQ.WYNN, dtype: float64

1.There are various stocks for which we have collected a data set, which all stocks are apparently similar in performance

After doing the above analysis, I am grouping the companies which are apparently similar in performance. Group 1 : NYSE.AIZ,NYSE.CRM,NASDAQ.CSCO,NASDAQ.JBHT,NYSE.LUK,NYSE.DFS,NYSE.ETR,NYSE.BF.B,NYSE.SLG,NYSE.CCI,NYSE.WRK,NYSE.PSA,NYSE.UPS,NYSE.IVZ,NYSE.DTE,NYSE.ARNC,NASDAQ.AAL,NYSE.ES,NYSE.MPC,NASDAQ.BIIB,NYSE.PNC,NYSE.VMC,NYSE.BMY,NYSE.LNC,NASDAQ.FITB,NYSE.D,NASDAQ.SNPS,NYSE.PEP,NYSE.COP,NYSE.APA,NYSE.DOV,NASDAQ.WYNN,NYSE.LEN,NASDAQ.XLNX,NYSE.WMB,NYSE.LYB,NYSE.AIG,NASDAQ.DLTR,NYSE.MLM,NYSE.MS,NYSE.PSX,NYSE.MTB,NYSE.GM,NASDAQ.FISV,NYSE.CLX,NYSE.UAA,NYSE.PPL,NYSE.ALB,NYSE.EOG,NYSE.CMI,NYSE.EIX,NYSE.AVB,NYSE.AES,NASDAQ.PAYX,NASDAQ.EBAY,NYSE.SYY,NYSE.HRL,NYSE.PWR,NYSE.GGP,NYSE.TIF,NYSE.CXO,NYSE.DAL,NYSE.SWK,NYSE.GIS,NYSE.CNP,NYSE.UA,NYSE.WY,NYSE.SRE,NASDAQ.NWS,NYSE.UNM,NYSE.FTV,NYSE.PRGO,NYSE.FRT,NASDAQ.SYMC,NYSE.OKE,NASDAQ.NWSA,NYSE.COTY,NYSE.HD,NYSE.EVHC,NYSE.EFX,NYSE.PCG,NASDAQ.MU,NYSE.DHR,NYSE.DRI,NYSE.DG,NYSE.HPQ,NASDAQ.IDXX,NYSE.PEG,NYSE.WMT,NASDAQ.CSX,NYSE.NEM,NASDAQ.NTAP,NYSE.AIV,NASDAQ.ILMN,NYSE.TGT,NASDAQ.WDC,NYSE.NLSN,NYSE.CVX

Group 2: NYSE.AYI,NYSE.SCHW,NASDAQ.MDLZ,NASDAQ.PCAR,NYSE.RSG,NYSE.VLO,NYSE.JWN,NYSE.PRU,NASDAQ.QRVO,NYSE.CBG,NASDAQ.PBCT,NYSE.WM,NASDAQ.AMD,NYSE.ALLE,NYSE.TMK,NYSE.ARE,NYSE.PFE,NYSE.PFG,NYSE.TXT,NYSE.HCN,NYSE.HBI,NYSE.PH,NYSE.IFF,NYSE.MON,NASDAQ.MCHP,NYSE.CMA,NYSE.URI,NASDAQ.INCY,NYSE.LLY,NYSE.VTR,NASDAQ.ADI,NYSE.SPG,NYSE.CFG,NYSE.CVS,NYSE.MRK,NYSE.BAC,NYSE.RF,NYSE.CBS,NYSE.DLR,NYSE.,TI,NYSE.EMR,NYSE.DD,NASDAQ.DISH,NYSE.NUE,NYSE.WFC,NYSE.HST,NASDAQ.HBAN,NYSE.GPS,NYSE.USB,NYSE.DOW,NASDAQ.QCOM,NASDAQ.VRSK,NYSE.REG,NY,E.CF,NYSE.TSN,NASDAQ.SWKS,NYSE.JEC,NYSE.EXR,NYSE.OXY,NASDAQ.GRMN,NASDAQ.TXN,NYSE.GS,NYSE.KEY,NYSE.WU,NYSE.COF,NASDAQ.INTC,NYSE.L,NYSE.UNP,NYSE.SEE,NYSE.PPG,NYSE.F,NYSE.KSS,NYSE.NSC

Group 3: NASDAQ.ROST,NYSE.HSY,NYSE.COH,NYSE.DLPH,NASDAQ.CELG,NYSE.DE,NYSE.SIG,NYSE.DXC,NASDAQ.ATVI,NASDAQ.ADSK,NYSE.AMT,NYSE.NKE,NYSE.NWL,NYSE.SYF,NYSE.HRS,NYSE.TMO,NYSE.EL,NYSE.TDG,NASDAQ.TRIP,NASDAQ.AVGO,NYSE.MAC,NASDAQ.CERN,NYSE.CMS,NYSE.PX,NASDAQ.HSIC,NYSE,MCK,NASDAQ.EQIX,NASDAQ.XRAY,NASDAQ.AMGN,NYSE.AMG,NASDAQ.FOX,NYSE.HUM,NASDAQ.FOXA,NASDAQ.CHRW,NYSE.AEP,NASDAQ.TSCO,NASDAQ.CTXS,NYSE.,KI,NYSE.AGN,NYSE.ALL,NYSE.LNT,NASDAQ.PCLN,NYSE.RL,NYSE.KO,NYSE.XEL,NYSE.MCO,NYSE.DUK,NASDAQ.GILD,NASDAQ.AMZN,NYSE.KMI,NYSE.K,NY,E.AJG,NYSE.UTX,NYSE.VAR,NYSE.FE,NYSE.WEC,NASDAQ.AAPL,NYSE.AEE,NASDAQ.MYL,NYSE.KORS,NYSE.FMC,NYSE.AMP,NYSE.FCX,NASDAQ.CINF,NYSE.UAL,NYSE.LEG,NASDAQ.ADP

Group 4: NYSE.FL,NYSE.AET,NYSE.MCD,NYSE.CAG,NASDAQ.ISRG,NYSE.APC,NYSE.RHT,NYSE.HRB,NYSE.STZ,NYSE.AXP,NYSE.ROP,NASDAQ.CBOE,NYSE.YUM,NYSE.KSU,NYSE.CI,NYSE.EW,NYSE.NFX,NYSE.NOC,NASDAQ.NVDA,NYSE.FTI,NYSE.GWW,NASDAQ.WLTW,NYSE.XEC,NYSE.BLK,NYSE.CCL,NYSE.M,NYSE.STT,NYSE.IT,NYSE.ICE,NYSE.DVN,NYSE.ANTM,NYSE.C,NASDAQ.ADBE,NYSE.LMT,NYSE.SPGI,NYSE.GE,NYSE.UNH,NYSE.BK,NYSE.ABT,NASDAQ.MNST,NYSE.BAX,NYSE,MMC,NYSE.BDX,NASDAQ.REGN,NYSE.NBL,NASDAQ.EA,NYSE.NI,NYSE.NEE,NASDAQ.ETFC,NYSE.ESS,NASDAQ.VRSN,NYSE.HP,NYSE.KMB,NYSE.PVH,NYSE.TAP,NYSE.XYL,NYSE.RCL,NASDAQ.NDAQ,NASDAQ.PDCO,NASDAQ.FB,NASDAQ.CHTR,NYSE.IPG

Group 5: NYSE.TJX,NASDAQ.COST,NYSE.CL,NYSE.JNJ,NASDAQ.MAR,NYSE.AME,NYSE.COG,NASDAQ.GOOG,NASDAQ.GOOGL,NASDAQ.AKAM,NYSE.APD,NASDAQ.ALXN,NYSE.ADM,NYSE.DIS,NYSE.LLL,NYSE.PM,NASDAQ.WBA,NYSE.EQT,NYSE.O,NASDAQ.EXPD,NYSE.MSI,NYSE.KIM,NASDAQ.AMAT,NASDAQ.FFIV,NYSE.BRK.B,NYSE.JNPR,NYSE.AWK,NYSE.IP,NYSE.MET,NYSE.BBT,NYSE.CHD,NYSE.BEN,NYSE.EXC,NYSE.HCP,NASDAQ.FLIR,NYSE.ABC,NASDAQ.HOLX,NYSE.VFC,NYSE.ADS,NYSE.WHR,NASDAQ.NAVI,NYSE.PNR,NYSE.ZBH,NYSE.T,NYSE.AN,NYSE.TWX,NASDAQ.GT,NYSE.HCA,NYSE.UHS,NYSE.RHI,NASDAQ.CA,NASDAQ.NTRS,NYSE.VZ

Group 6: NYSE.LB,NYSE.ECL,NYSE.CTL,NASDAQ.FAST,NYSE.HES,NYSE.RJF,NYSE.XOM,NYSE.LVLT,NYSE.CHK,NYSE.DHI,NYSE.ROK,NASDAQ.ESRX,NYSE.KMX,NYSE,BWA,NYSE.JPM,NYSE.BXP,NYSE.SNA,NASDAQ.EXPE,NYSE.EMN,NYSE.HAL,NASDAQ.ZION,NASDAQ.CMCSA,NYSE.DPS,NYSE.SYK,NYSE.TEL,NASDAQ.KHC,NYSE.ED,NYSE.FBHS,NYSE.PHM,NYSE.MAA,NYSE.MHK,NYSE.TRV,NYSE.HPE,NASDAQ.LRCX,NYSE.XL,NYSE.ITW,NYSE.GD,NYSE.PNW,NYSE.SHW,NYSE.GLW,NASDAQ.KLAC,NYSE.IR

Group 7: NYSE.MAS,NYSE.MRO,NYSE.CPB,NYSE.CSRA,NASDAQ.SRCL,NYSE.DGX,NYSE.CNC,NYSE.LOW,NYSE.ABBV,NYSE.FDX,NYSE.UDR,NYSE.CB,NYSE.OMC,NYSE.M,C,NYSE.ACN,NYSE.LH,NASDAQ.CME,NYSE.BSX,NYSE.CAT,NYSE.SCG,NYSE.WAT,NASDAQ.MSFT,NYSE.APH,NYSE.PLD,NYSE.AFL,NASDAQ.SBUX,NYSE.MDT,N,SE.SO,NYSE.PXD,NYSE.MMM,NYSE.EQR,NYSE.ALK,NASDAQ.HAS,NYSE.CAH,NYSE.PG,NYSE.IRM,NYSE.LUV,NASDAQ.CTAS,NASDAQ.NFLX,NYSE.ETN,NYSE.J,I

Group 8: NYSE.ZTS,NASDAQ.CTSH,NYSE.RTN,NYSE.MOS,NASDAQ.VIAB,NYSE.FLR,NYSE.IBM,NYSE.GPN,NASDAQ.DISCA,NASDAQ.DISCK,NYSE.DVA,NASDAQ.SNI

Group 9: NYSE.AVY,NYSE.SLB,NYSE.NOV,NYSE.HOG,NASDAQ.ORLY,NASDAQ.TROW,NYSE.RRC,NASDAQ.LKQ,NASDAQ.STX,NYSE.FLS

Attaching the companyListCluster.xslx sheet containing the list of all the companies in separate sheets which are similar in performance.

In [51]:

companyData.reset\_index(inplace=**True**)

**Problem 2:¶**

1. How many Unique patterns that exist in the historical stock data set, based on fluctuations in price.

In [52]:

**import** **seaborn** **as** **sns**

plt.figure(2, figsize=(40, 15.2))

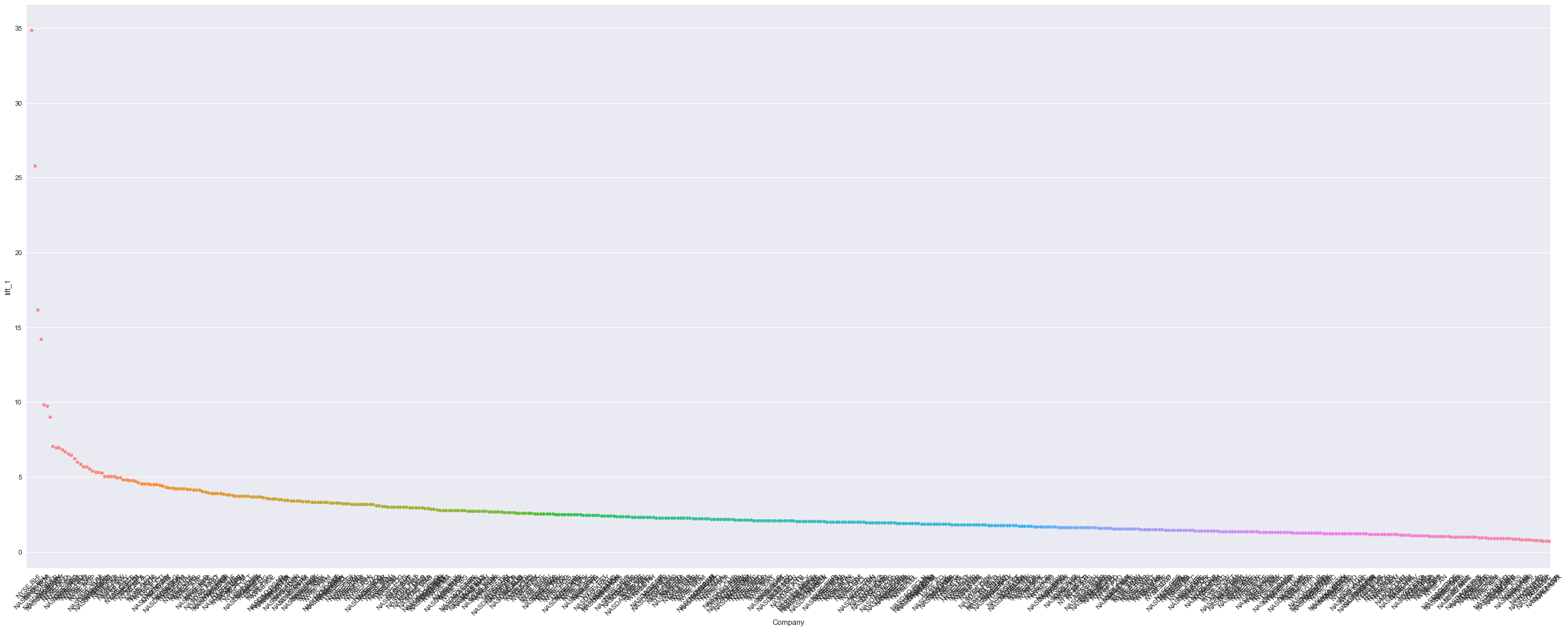
g = sns.swarmplot(x="Company", y="lift\_1", data=companyData);

*#g.set\_xticklabels(rotation=90)*

plt.xticks(rotation = 45)

*#plt.scatter(companyData.index, companyData.lift\_1)*

plt.show()



In [53]:

**import** **seaborn** **as** **sns**

plt.figure(2, figsize=(40, 15.2))

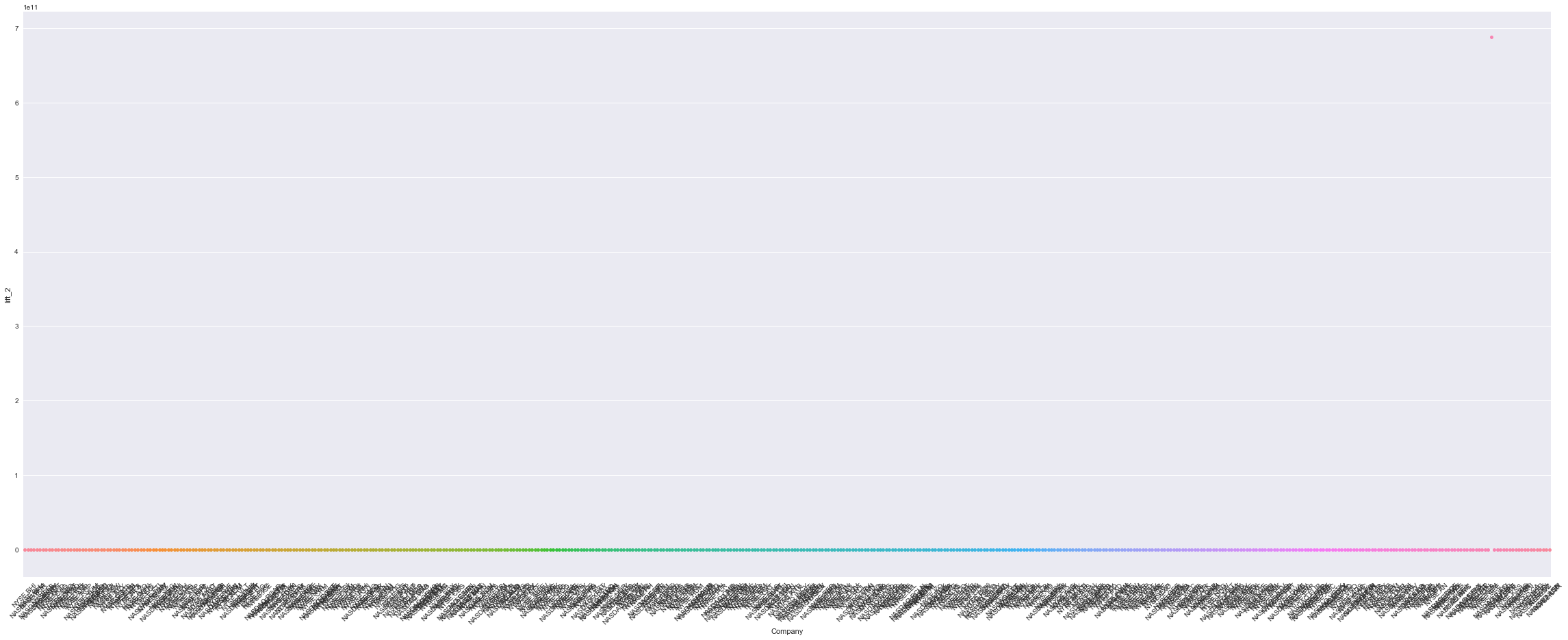
g = sns.swarmplot(x="Company", y="lift\_2", data=companyData);

*#g.set\_xticklabels(rotation=90)*

plt.xticks(rotation = 45)

*#plt.scatter(companyData.index, companyData.lift\_1)*

plt.show()



In [54]:

**import** **seaborn** **as** **sns**

plt.figure(2, figsize=(40, 15.2))

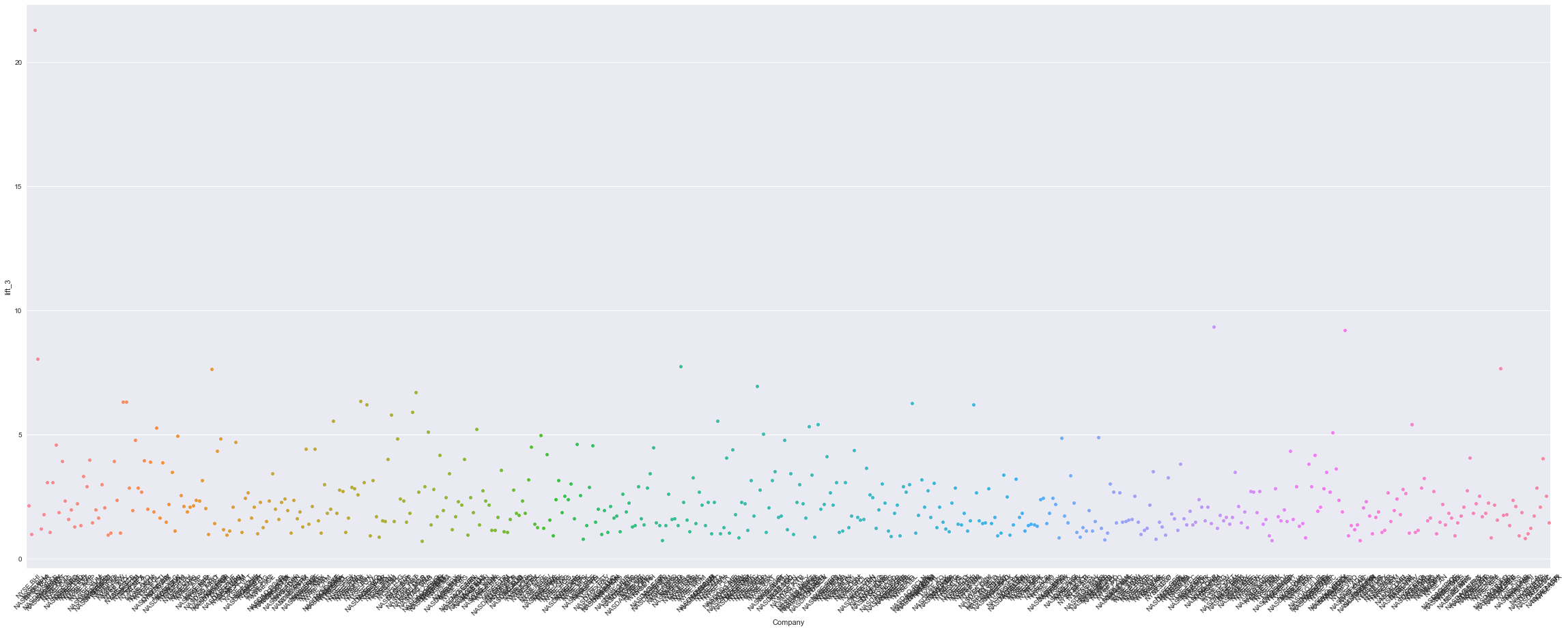
g = sns.swarmplot(x="Company", y="lift\_3", data=companyData);

*#g.set\_xticklabels(rotation=90)*

plt.xticks(rotation = 45)

*#plt.scatter(companyData.index, companyData.lift\_1)*

plt.show()



In [55]:

**import** **seaborn** **as** **sns**

plt.figure(2, figsize=(40, 15.2))

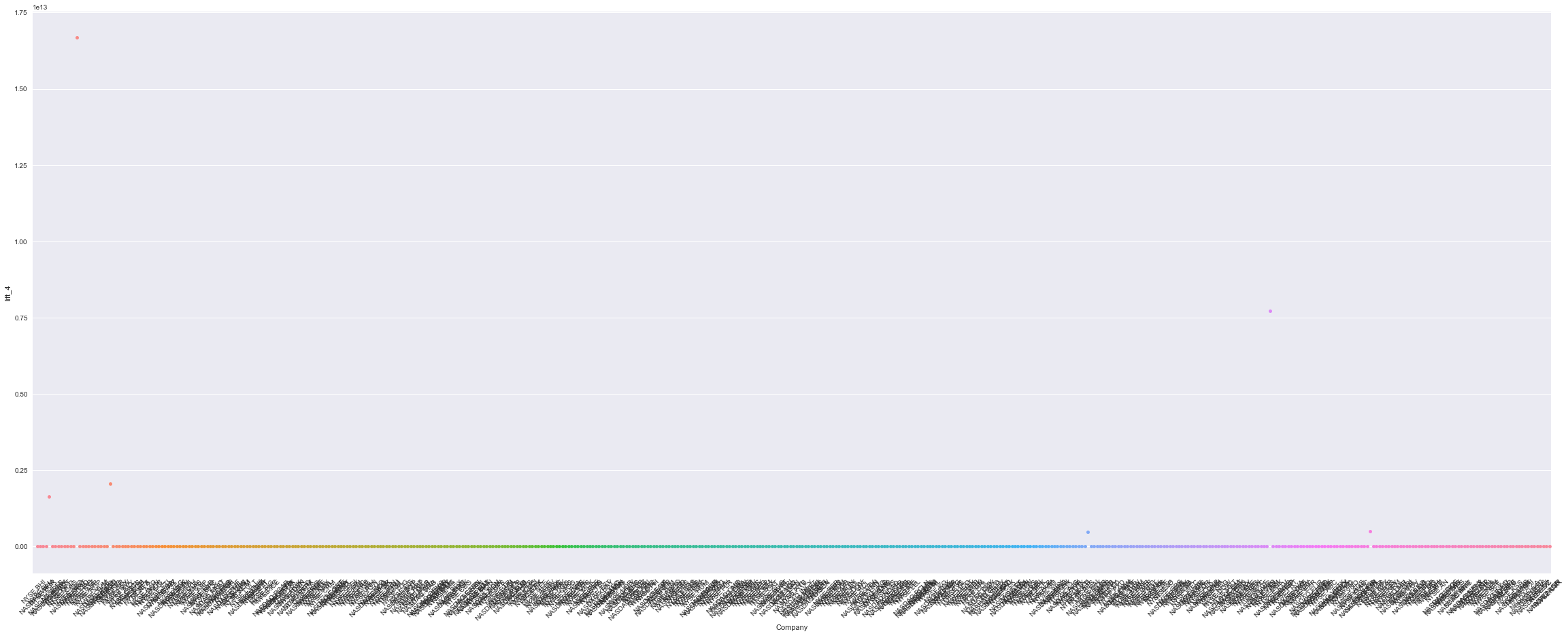
g = sns.swarmplot(x="Company", y="lift\_4", data=companyData);

*#g.set\_xticklabels(rotation=90)*

plt.xticks(rotation = 45)

*#plt.scatter(companyData.index, companyData.lift\_1)*

plt.show()



In [56]:

**import** **seaborn** **as** **sns**

plt.figure(2, figsize=(40, 15.2))

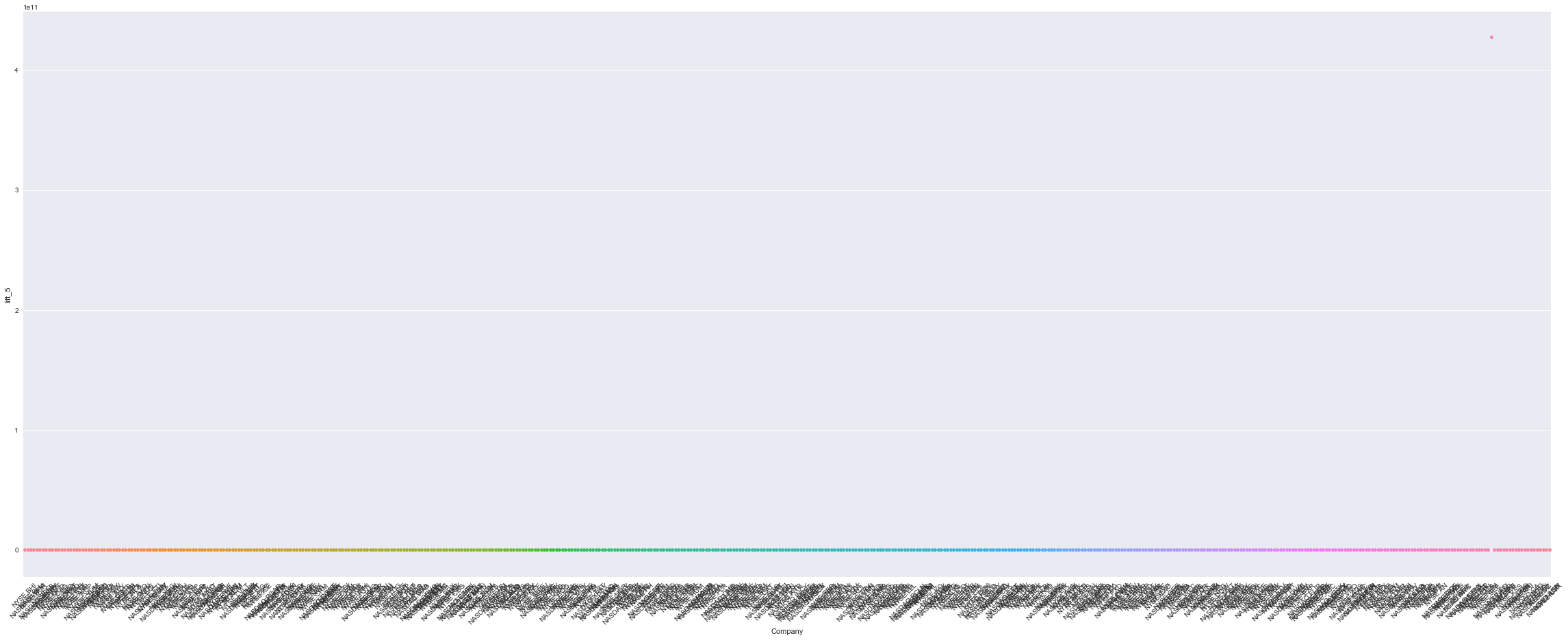
g = sns.swarmplot(x="Company", y="lift\_5", data=companyData);

*#g.set\_xticklabels(rotation=90)*

plt.xticks(rotation = 45)

*#plt.scatter(companyData.index, companyData.lift\_1)*

plt.show()



As per the graph displaying above the lifts in the standard deviation against all the companies, Lift\_2 & Lift\_5 is the most unique pattern that existed in historical stock data set, based on fluctuations in price, since the lift remains consistence after clustering.

**Problem 3:¶**

Identify which all stocks are moving together and which all stocks are different from each other.

In [57]:

companyData.set\_index('Company', inplace= **True**)

In [58]:

data.drop('Cluster\_No',axis = 1, inplace=**True**)

In [59]:

**import** **seaborn** **as** **sns**

f, ax = plt.subplots(figsize=(100, 100))

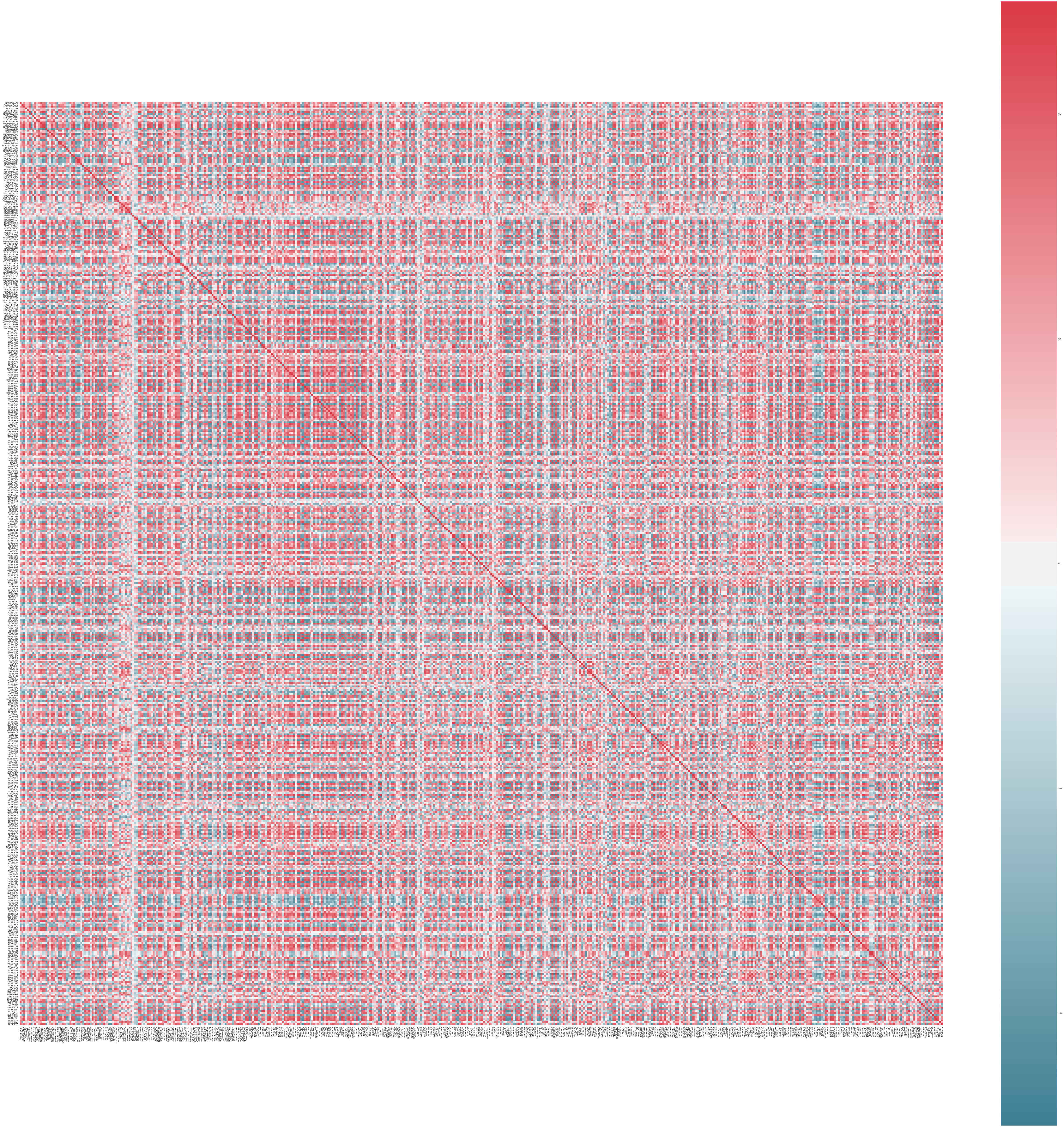
corr = data.corr()

sns.heatmap(corr, mask=np.zeros\_like(corr, dtype=np.bool), cmap=sns.diverging\_palette(220, 10, as\_cmap=**True**),

square=**True**, ax=ax)

Out[59]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x1dd0bf84160>



In [60]:

corrMatrix = data.corr()

In [61]:

corrMatrix

Out[61]:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **NASDAQ.AAL** | **NASDAQ.AAPL** | **NASDAQ.ADBE** | **NASDAQ.ADI** | **NASDAQ.ADP** | **NASDAQ.ADSK** | **NASDAQ.AKAM** | **NASDAQ.ALXN** | **NASDAQ.AMAT** | **NASDAQ.AMD** | **...** | **NYSE.WYN** | **NYSE.XEC** | **NYSE.XEL** | **NYSE.XL** | **NYSE.XOM** | **NYSE.XRX** | **NYSE.XYL** | **NYSE.YUM** | **NYSE.ZBH** | **NYSE.ZTS** |
| **NASDAQ.AAL** | 1.000000 | 0.082065 | 0.542213 | 0.209446 | 0.245801 | 0.610881 | -0.441142 | 0.009581 | 0.681863 | 0.272518 | ... | 0.798624 | -0.744085 | 0.284370 | 0.810955 | -0.070950 | 0.522341 | 0.395730 | 0.642336 | 0.588241 | 0.753567 |
| **NASDAQ.AAPL** | 0.082065 | 1.000000 | 0.714578 | 0.264269 | 0.265641 | 0.645233 | -0.712522 | 0.326712 | 0.545541 | -0.238091 | ... | 0.306664 | -0.250959 | 0.738655 | 0.266994 | -0.671488 | 0.341687 | 0.693062 | 0.627531 | -0.580055 | 0.442600 |
| **NASDAQ.ADBE** | 0.542213 | 0.714578 | 1.000000 | 0.259282 | 0.476496 | 0.872353 | -0.759604 | 0.488717 | 0.733609 | 0.201821 | ... | 0.718851 | -0.792876 | 0.850709 | 0.722161 | -0.758967 | 0.806615 | 0.911106 | 0.939185 | -0.067235 | 0.776052 |
| **NASDAQ.ADI** | 0.209446 | 0.264269 | 0.259282 | 1.000000 | -0.085074 | 0.470756 | -0.323603 | -0.409747 | 0.504019 | 0.042083 | ... | 0.203224 | -0.160317 | 0.308948 | 0.050697 | -0.024428 | -0.017967 | 0.067258 | 0.261173 | 0.130786 | 0.351893 |
| **NASDAQ.ADP** | 0.245801 | 0.265641 | 0.476496 | -0.085074 | 1.000000 | 0.325630 | -0.229119 | 0.572143 | 0.099934 | 0.447506 | ... | 0.321414 | -0.438662 | 0.443773 | 0.347293 | -0.463670 | 0.589464 | 0.634191 | 0.418591 | -0.201805 | 0.160213 |
| **NASDAQ.ADSK** | 0.610881 | 0.645233 | 0.872353 | 0.470756 | 0.325630 | 1.000000 | -0.845720 | 0.102522 | 0.871658 | -0.027431 | ... | 0.789021 | -0.717767 | 0.824764 | 0.671072 | -0.531723 | 0.577194 | 0.701701 | 0.915175 | 0.016689 | 0.884658 |
| **NASDAQ.AKAM** | -0.441142 | -0.712522 | -0.759604 | -0.323603 | -0.229119 | -0.845720 | 1.000000 | -0.163074 | -0.702694 | 0.234894 | ... | -0.633208 | 0.608156 | -0.779460 | -0.525856 | 0.515614 | -0.542393 | -0.668413 | -0.863792 | 0.212780 | -0.815995 |
| **NASDAQ.ALXN** | 0.009581 | 0.326712 | 0.488717 | -0.409747 | 0.572143 | 0.102522 | -0.163074 | 1.000000 | -0.037657 | 0.370871 | ... | 0.184740 | -0.341883 | 0.286281 | 0.324473 | -0.566261 | 0.696062 | 0.679090 | 0.343294 | -0.259731 | 0.071510 |
| **NASDAQ.AMAT** | 0.681863 | 0.545541 | 0.733609 | 0.504019 | 0.099934 | 0.871658 | -0.702694 | -0.037657 | 1.000000 | -0.029178 | ... | 0.783305 | -0.587720 | 0.592717 | 0.671377 | -0.313532 | 0.371468 | 0.485314 | 0.770276 | 0.236575 | 0.855434 |
| **NASDAQ.AMD** | 0.272518 | -0.238091 | 0.201821 | 0.042083 | 0.447506 | -0.027431 | 0.234894 | 0.370871 | -0.029178 | 1.000000 | ... | 0.092926 | -0.389991 | -0.024601 | 0.244269 | -0.128342 | 0.450787 | 0.261115 | 0.083384 | 0.417055 | -0.043194 |
| **NASDAQ.AMGN** | 0.528900 | 0.131275 | 0.644373 | -0.142261 | 0.520941 | 0.332389 | -0.210959 | 0.726371 | 0.265674 | 0.662632 | ... | 0.493936 | -0.728216 | 0.325763 | 0.647509 | -0.442814 | 0.846687 | 0.710567 | 0.559293 | 0.308449 | 0.391030 |
| **NASDAQ.AMZN** | 0.803319 | 0.370111 | 0.730719 | 0.386619 | 0.252712 | 0.828757 | -0.648854 | 0.059345 | 0.897693 | 0.170978 | ... | 0.879967 | -0.733852 | 0.530139 | 0.804398 | -0.203979 | 0.533584 | 0.503878 | 0.793529 | 0.419472 | 0.872246 |
| **NASDAQ.ATVI** | 0.593574 | 0.704358 | 0.961857 | 0.321411 | 0.421174 | 0.923135 | -0.850687 | 0.375481 | 0.802540 | 0.093668 | ... | 0.797973 | -0.815299 | 0.874434 | 0.723013 | -0.652308 | 0.763749 | 0.857080 | 0.972927 | -0.025171 | 0.875018 |
| **NASDAQ.AVGO** | 0.660926 | 0.671936 | 0.906947 | 0.354330 | 0.377840 | 0.904435 | -0.800303 | 0.260828 | 0.874700 | 0.148718 | ... | 0.782207 | -0.775669 | 0.797486 | 0.736186 | -0.550863 | 0.661323 | 0.775354 | 0.924221 | 0.074611 | 0.848088 |
| **NASDAQ.BBBY** | -0.639629 | -0.531976 | -0.919636 | -0.117133 | -0.501317 | -0.764401 | 0.710620 | -0.550564 | -0.615979 | -0.335640 | ... | -0.707349 | 0.855164 | -0.685915 | -0.795003 | 0.663575 | -0.891690 | -0.878635 | -0.894123 | -0.056490 | -0.745852 |
| **NASDAQ.BIIB** | 0.096628 | 0.279693 | 0.546833 | -0.199210 | 0.672531 | 0.162129 | -0.066038 | 0.847354 | -0.005919 | 0.613656 | ... | 0.166353 | -0.421515 | 0.361652 | 0.293722 | -0.641143 | 0.689869 | 0.707914 | 0.357933 | -0.135797 | 0.039560 |
| **NASDAQ.CA** | 0.468258 | -0.166888 | 0.365780 | -0.045582 | -0.100914 | 0.186955 | -0.009380 | 0.261685 | 0.235930 | 0.455738 | ... | 0.440784 | -0.530320 | 0.029439 | 0.564468 | -0.114891 | 0.475533 | 0.234487 | 0.331574 | 0.579098 | 0.368573 |
| **NASDAQ.CBOE** | 0.481325 | 0.637467 | 0.931362 | 0.104149 | 0.564974 | 0.757830 | -0.719942 | 0.602586 | 0.552340 | 0.237157 | ... | 0.610682 | -0.807648 | 0.825180 | 0.657556 | -0.817511 | 0.884094 | 0.963673 | 0.892653 | -0.153966 | 0.672903 |
| **NASDAQ.CELG** | 0.404745 | 0.133267 | 0.607426 | -0.170338 | 0.616948 | 0.267719 | -0.159542 | 0.788970 | 0.134300 | 0.701299 | ... | 0.398342 | -0.652483 | 0.309339 | 0.569328 | -0.468697 | 0.845987 | 0.704244 | 0.485283 | 0.182084 | 0.270159 |
| **NASDAQ.CERN** | 0.567155 | 0.314661 | 0.584149 | 0.281608 | -0.091642 | 0.714164 | -0.632066 | -0.051358 | 0.770590 | -0.162259 | ... | 0.791314 | -0.618241 | 0.498324 | 0.572474 | -0.167171 | 0.353743 | 0.346992 | 0.716088 | 0.343205 | 0.876411 |
| **NASDAQ.CHRW** | -0.681113 | -0.514152 | -0.751792 | -0.224910 | -0.326741 | -0.852274 | 0.818748 | -0.147326 | -0.789063 | -0.001086 | ... | -0.842976 | 0.703115 | -0.657406 | -0.779165 | 0.344204 | -0.569777 | -0.603112 | -0.856862 | -0.100767 | -0.870382 |
| **NASDAQ.CHTR** | 0.102313 | 0.671804 | 0.715841 | -0.044414 | 0.747106 | 0.483826 | -0.462811 | 0.647658 | 0.228392 | 0.190528 | ... | 0.273708 | -0.471300 | 0.761129 | 0.345595 | -0.839133 | 0.659235 | 0.861522 | 0.607525 | -0.531004 | 0.255031 |
| **NASDAQ.CINF** | 0.211496 | 0.546814 | 0.694139 | -0.146740 | 0.771811 | 0.439522 | -0.431250 | 0.743543 | 0.174519 | 0.311028 | ... | 0.334465 | -0.580188 | 0.673951 | 0.470896 | -0.707502 | 0.783041 | 0.885925 | 0.608500 | -0.383688 | 0.276460 |
| **NASDAQ.CMCSA** | 0.354914 | 0.586400 | 0.636358 | 0.403285 | 0.248106 | 0.775026 | -0.678326 | -0.071559 | 0.690549 | -0.237192 | ... | 0.612354 | -0.531054 | 0.817212 | 0.446533 | -0.436081 | 0.335207 | 0.515056 | 0.733720 | -0.153318 | 0.684714 |
| **NASDAQ.CME** | 0.418846 | 0.281176 | 0.671714 | 0.003393 | 0.467620 | 0.529594 | -0.530086 | 0.491585 | 0.272507 | 0.276108 | ... | 0.517939 | -0.825558 | 0.671232 | 0.513693 | -0.551538 | 0.843126 | 0.771472 | 0.733242 | 0.015593 | 0.559707 |
| **NASDAQ.COST** | -0.409006 | -0.255595 | -0.661589 | 0.037644 | -0.486985 | -0.363024 | 0.342572 | -0.706319 | -0.227249 | -0.596613 | ... | -0.352271 | 0.642316 | -0.365612 | -0.568532 | 0.542557 | -0.856123 | -0.723782 | -0.546114 | -0.110309 | -0.353265 |
| **NASDAQ.CSCO** | -0.583706 | -0.268124 | -0.635736 | -0.290632 | -0.349626 | -0.724334 | 0.633150 | -0.032668 | -0.507281 | -0.195809 | ... | -0.519406 | 0.751296 | -0.633788 | -0.549587 | 0.488107 | -0.620022 | -0.585948 | -0.714597 | -0.108373 | -0.643026 |
| **NASDAQ.CSX** | 0.662241 | -0.065778 | 0.227743 | 0.274731 | -0.280747 | 0.425963 | -0.301676 | -0.359973 | 0.582505 | -0.164188 | ... | 0.648793 | -0.423651 | 0.074682 | 0.511769 | 0.154909 | 0.088312 | -0.020575 | 0.393468 | 0.559190 | 0.665189 |
| **NASDAQ.CTAS** | 0.351587 | 0.511664 | 0.806725 | 0.097947 | 0.723890 | 0.631028 | -0.525570 | 0.598902 | 0.424585 | 0.410769 | ... | 0.523210 | -0.690963 | 0.759851 | 0.563145 | -0.681620 | 0.810165 | 0.860151 | 0.736841 | -0.114124 | 0.478395 |
| **NASDAQ.CTSH** | 0.639932 | 0.693353 | 0.943406 | 0.273389 | 0.433268 | 0.910171 | -0.875456 | 0.372475 | 0.785383 | 0.081983 | ... | 0.789409 | -0.815568 | 0.834715 | 0.778150 | -0.618865 | 0.776839 | 0.859383 | 0.972541 | -0.033438 | 0.870623 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **NYSE.USB** | 0.523509 | 0.297190 | 0.514964 | -0.059792 | 0.514439 | 0.462715 | -0.408208 | 0.405590 | 0.401440 | 0.180411 | ... | 0.579231 | -0.556924 | 0.369836 | 0.636815 | -0.200882 | 0.564458 | 0.605176 | 0.564803 | 0.071026 | 0.490646 |
| **NYSE.UTX** | 0.698371 | 0.068743 | 0.356371 | 0.243140 | -0.172957 | 0.525459 | -0.414264 | -0.141042 | 0.673330 | -0.137863 | ... | 0.784558 | -0.450007 | 0.124054 | 0.647913 | 0.157504 | 0.211910 | 0.116581 | 0.491635 | 0.526279 | 0.771706 |
| **NYSE.V** | 0.385453 | 0.781096 | 0.951404 | 0.140149 | 0.581320 | 0.793225 | -0.728715 | 0.585401 | 0.608903 | 0.149188 | ... | 0.588776 | -0.692584 | 0.880952 | 0.611833 | -0.842236 | 0.783261 | 0.955558 | 0.875789 | -0.282328 | 0.632928 |
| **NYSE.VAR** | 0.717439 | 0.308028 | 0.755542 | 0.367813 | 0.059239 | 0.766144 | -0.675438 | 0.106316 | 0.757070 | 0.173648 | ... | 0.767912 | -0.860564 | 0.608186 | 0.709656 | -0.365920 | 0.668394 | 0.582883 | 0.839560 | 0.420198 | 0.889419 |
| **NYSE.VFC** | 0.174225 | 0.559299 | 0.719638 | -0.136562 | 0.756445 | 0.440166 | -0.445181 | 0.776055 | 0.166578 | 0.323447 | ... | 0.321056 | -0.549509 | 0.680778 | 0.412349 | -0.795935 | 0.786999 | 0.893556 | 0.617875 | -0.383619 | 0.276358 |
| **NYSE.VLO** | 0.349937 | 0.107434 | 0.434775 | -0.193338 | 0.479212 | 0.146945 | -0.155561 | 0.717214 | 0.057029 | 0.509289 | ... | 0.282984 | -0.473028 | 0.134327 | 0.409068 | -0.276266 | 0.691940 | 0.599961 | 0.354575 | 0.152084 | 0.211013 |
| **NYSE.VMC** | 0.382945 | -0.360217 | -0.179305 | 0.202237 | -0.425880 | 0.072111 | -0.020381 | -0.443334 | 0.298031 | -0.156235 | ... | 0.392046 | -0.085136 | -0.286377 | 0.161582 | 0.626857 | -0.210183 | -0.391977 | 0.008988 | 0.673300 | 0.367287 |
| **NYSE.VNO** | -0.267094 | -0.792875 | -0.880266 | -0.056424 | -0.579779 | -0.704094 | 0.694804 | -0.639707 | -0.540998 | -0.102945 | ... | -0.497756 | 0.571718 | -0.808145 | -0.560560 | 0.804892 | -0.725078 | -0.913845 | -0.787570 | 0.370004 | -0.532490 |
| **NYSE.VTR** | 0.494899 | -0.061758 | 0.460024 | 0.357921 | 0.024149 | 0.520487 | -0.386324 | -0.028758 | 0.402414 | 0.313765 | ... | 0.545210 | -0.697421 | 0.425809 | 0.438598 | -0.144274 | 0.543516 | 0.313447 | 0.553377 | 0.526283 | 0.626431 |
| **NYSE.VZ** | -0.586295 | 0.211194 | -0.126482 | -0.140572 | 0.407075 | -0.242402 | 0.116539 | 0.219061 | -0.483655 | -0.025662 | ... | -0.496739 | 0.287427 | 0.198988 | -0.532794 | -0.272052 | -0.080968 | 0.118792 | -0.208376 | -0.684011 | -0.505063 |
| **NYSE.WAT** | 0.688989 | 0.388563 | 0.719185 | 0.263736 | 0.062264 | 0.800411 | -0.704766 | 0.072101 | 0.795380 | -0.083679 | ... | 0.869618 | -0.772875 | 0.633231 | 0.734603 | -0.330323 | 0.543341 | 0.541817 | 0.846420 | 0.274112 | 0.939908 |
| **NYSE.WEC** | 0.211689 | 0.666012 | 0.783377 | 0.234131 | 0.441161 | 0.722203 | -0.695861 | 0.348303 | 0.472195 | 0.001706 | ... | 0.492028 | -0.648108 | 0.962946 | 0.398255 | -0.686485 | 0.648638 | 0.794351 | 0.790929 | -0.313096 | 0.574044 |
| **NYSE.WFC** | 0.205066 | -0.577962 | -0.317679 | -0.246926 | -0.104686 | -0.361404 | 0.388399 | 0.041246 | -0.195488 | 0.278082 | ... | -0.015781 | 0.019817 | -0.621775 | 0.143707 | 0.503575 | -0.015256 | -0.251901 | -0.273270 | 0.548228 | -0.136084 |
| **NYSE.WHR** | 0.665643 | -0.259964 | 0.102225 | 0.125038 | -0.282936 | 0.273835 | -0.126800 | -0.269368 | 0.515859 | -0.018901 | ... | 0.645325 | -0.398225 | -0.075949 | 0.542212 | 0.375416 | 0.090828 | -0.117444 | 0.266785 | 0.758154 | 0.575343 |
| **NYSE.WM** | 0.275277 | 0.484494 | 0.754316 | -0.064980 | 0.641030 | 0.450667 | -0.370322 | 0.706451 | 0.287723 | 0.363571 | ... | 0.369985 | -0.635512 | 0.668644 | 0.474759 | -0.777773 | 0.796307 | 0.864431 | 0.641050 | -0.167319 | 0.335875 |
| **NYSE.WMB** | 0.231421 | -0.059534 | 0.015326 | -0.273027 | 0.338342 | -0.111130 | 0.187978 | 0.370132 | -0.001081 | 0.337470 | ... | 0.040194 | 0.090273 | -0.353284 | 0.242302 | 0.133902 | 0.114721 | 0.104699 | -0.114823 | 0.149021 | -0.135316 |
| **NYSE.WMT** | 0.335280 | 0.761047 | 0.684392 | 0.164550 | 0.490919 | 0.766906 | -0.748331 | 0.223714 | 0.594354 | -0.227044 | ... | 0.567559 | -0.436091 | 0.780630 | 0.443228 | -0.549231 | 0.404601 | 0.678175 | 0.727428 | -0.394405 | 0.587326 |
| **NYSE.WRK** | 0.836602 | 0.198634 | 0.719864 | 0.163785 | 0.375851 | 0.668367 | -0.487602 | 0.332558 | 0.693023 | 0.343900 | ... | 0.850729 | -0.854719 | 0.461421 | 0.855927 | -0.288706 | 0.735308 | 0.627625 | 0.770008 | 0.467109 | 0.790217 |
| **NYSE.WU** | -0.409519 | -0.227968 | -0.470321 | -0.237112 | 0.197022 | -0.517072 | 0.636646 | 0.027380 | -0.442589 | 0.086381 | ... | -0.407927 | 0.589443 | -0.433912 | -0.440772 | 0.282068 | -0.452254 | -0.352716 | -0.586255 | -0.130390 | -0.635135 |
| **NYSE.WY** | -0.166749 | -0.739510 | -0.668333 | -0.060394 | -0.228522 | -0.575179 | 0.753206 | -0.387581 | -0.412708 | 0.202026 | ... | -0.335840 | 0.442275 | -0.648917 | -0.345225 | 0.718794 | -0.494419 | -0.681559 | -0.670210 | 0.480670 | -0.521397 |
| **NYSE.WYN** | 0.798624 | 0.306664 | 0.718851 | 0.203224 | 0.321414 | 0.789021 | -0.633208 | 0.184740 | 0.783305 | 0.092926 | ... | 1.000000 | -0.770048 | 0.555088 | 0.846707 | -0.251412 | 0.605548 | 0.548014 | 0.825260 | 0.369747 | 0.904186 |
| **NYSE.XEC** | -0.744085 | -0.250959 | -0.792876 | -0.160317 | -0.438662 | -0.717767 | 0.608156 | -0.341883 | -0.587720 | -0.389991 | ... | -0.770048 | 1.000000 | -0.664045 | -0.785750 | 0.476910 | -0.867222 | -0.737225 | -0.862418 | -0.322979 | -0.792090 |
| **NYSE.XEL** | 0.284370 | 0.738655 | 0.850709 | 0.308948 | 0.443773 | 0.824764 | -0.779460 | 0.286281 | 0.592717 | -0.024601 | ... | 0.555088 | -0.664045 | 1.000000 | 0.453725 | -0.753178 | 0.636184 | 0.811981 | 0.855050 | -0.310343 | 0.660476 |
| **NYSE.XL** | 0.810955 | 0.266994 | 0.722161 | 0.050697 | 0.347293 | 0.671072 | -0.525856 | 0.324473 | 0.671377 | 0.244269 | ... | 0.846707 | -0.785750 | 0.453725 | 1.000000 | -0.308922 | 0.717401 | 0.610465 | 0.763547 | 0.339366 | 0.765026 |
| **NYSE.XOM** | -0.070950 | -0.671488 | -0.758967 | -0.024428 | -0.463670 | -0.531723 | 0.515614 | -0.566261 | -0.313532 | -0.128342 | ... | -0.251412 | 0.476910 | -0.753178 | -0.308922 | 1.000000 | -0.623126 | -0.798461 | -0.634492 | 0.458027 | -0.337984 |
| **NYSE.XRX** | 0.522341 | 0.341687 | 0.806615 | -0.017967 | 0.589464 | 0.577194 | -0.542393 | 0.696062 | 0.371468 | 0.450787 | ... | 0.605548 | -0.867222 | 0.636184 | 0.717401 | -0.623126 | 1.000000 | 0.866193 | 0.785073 | 0.081512 | 0.588426 |
| **NYSE.XYL** | 0.395730 | 0.693062 | 0.911106 | 0.067258 | 0.634191 | 0.701701 | -0.668413 | 0.679090 | 0.485314 | 0.261115 | ... | 0.548014 | -0.737225 | 0.811981 | 0.610465 | -0.798461 | 0.866193 | 1.000000 | 0.842826 | -0.246085 | 0.581894 |
| **NYSE.YUM** | 0.642336 | 0.627531 | 0.939185 | 0.261173 | 0.418591 | 0.915175 | -0.863792 | 0.343294 | 0.770276 | 0.083384 | ... | 0.825260 | -0.862418 | 0.855050 | 0.763547 | -0.634492 | 0.785073 | 0.842826 | 1.000000 | 0.018217 | 0.899693 |
| **NYSE.ZBH** | 0.588241 | -0.580055 | -0.067235 | 0.130786 | -0.201805 | 0.016689 | 0.212780 | -0.259731 | 0.236575 | 0.417055 | ... | 0.369747 | -0.322979 | -0.310343 | 0.339366 | 0.458027 | 0.081512 | -0.246085 | 0.018217 | 1.000000 | 0.271767 |
| **NYSE.ZTS** | 0.753567 | 0.442600 | 0.776052 | 0.351893 | 0.160213 | 0.884658 | -0.815995 | 0.071510 | 0.855434 | -0.043194 | ... | 0.904186 | -0.792090 | 0.660476 | 0.765026 | -0.337984 | 0.588426 | 0.581894 | 0.899693 | 0.271767 | 1.000000 |

500 rows × 500 columns

In [127]:

corr = corrMatrix

c1 = corr.unstack()

sorted = c1.sort\_values(ascending = **False**)

stdf = pd.DataFrame({'sorted':sorted})

In [128]:

sortedCorr = stdf.sort\_values('sorted', ascending=**True**)

In [142]:

sortCorr = stdf.sorted

sortCorr = sortCorr.reset\_index()

The stocks which are moving together i.e. they are closely related

In [144]:

sortCorr.loc[(sortCorr.sorted > 0.7) & (sortCorr.sorted < 1.0)]

Out[144]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **level\_0** | **level\_1** | **sorted** |
| **500** | NASDAQ.GOOG | NASDAQ.GOOGL | 0.998352 |
| **501** | NASDAQ.GOOGL | NASDAQ.GOOG | 0.998352 |
| **502** | NASDAQ.DISCA | NASDAQ.DISCK | 0.996855 |
| **503** | NASDAQ.DISCK | NASDAQ.DISCA | 0.996855 |
| **504** | NASDAQ.NWSA | NASDAQ.NWS | 0.996612 |
| **505** | NASDAQ.NWS | NASDAQ.NWSA | 0.996612 |
| **506** | NASDAQ.FOXA | NASDAQ.FOX | 0.996182 |
| **507** | NASDAQ.FOX | NASDAQ.FOXA | 0.996182 |
| **508** | NYSE.DD | NYSE.DOW | 0.986025 |
| **509** | NYSE.DOW | NYSE.DD | 0.986025 |
| **510** | NYSE.UA | NYSE.UAA | 0.983613 |
| **511** | NYSE.UAA | NYSE.UA | 0.983613 |
| **512** | NYSE.DVN | NYSE.MRO | 0.983269 |
| **513** | NYSE.MRO | NYSE.DVN | 0.983269 |
| **514** | NYSE.CMS | NYSE.ES | 0.981819 |
| **515** | NYSE.ES | NYSE.CMS | 0.981819 |
| **516** | NYSE.DTE | NYSE.CMS | 0.981701 |
| **517** | NYSE.CMS | NYSE.DTE | 0.981701 |
| **518** | NYSE.LMT | NYSE.RTN | 0.981134 |
| **519** | NYSE.RTN | NYSE.LMT | 0.981134 |
| **520** | NYSE.MA | NYSE.V | 0.980574 |
| **521** | NYSE.V | NYSE.MA | 0.980574 |
| **522** | NYSE.RIG | NYSE.DVN | 0.980150 |
| **523** | NYSE.DVN | NYSE.RIG | 0.980150 |
| **524** | NASDAQ.ADBE | NYSE.MA | 0.979629 |
| **525** | NYSE.MA | NASDAQ.ADBE | 0.979629 |
| **526** | NASDAQ.ETFC | NYSE.BK | 0.978342 |
| **527** | NYSE.BK | NASDAQ.ETFC | 0.978342 |
| **528** | NYSE.DTE | NYSE.ES | 0.978332 |
| **529** | NYSE.ES | NYSE.DTE | 0.978332 |
| **...** | ... | ... | ... |
| **34018** | NYSE.MHK | NYSE.UNM | 0.700162 |
| **34019** | NYSE.UNM | NYSE.MHK | 0.700162 |
| **34020** | NYSE.MOS | NYSE.HCA | 0.700150 |
| **34021** | NYSE.HCA | NYSE.MOS | 0.700150 |
| **34022** | NASDAQ.ROST | NYSE.HOG | 0.700144 |
| **34023** | NYSE.HOG | NASDAQ.ROST | 0.700144 |
| **34024** | NYSE.SNA | NASDAQ.KHC | 0.700137 |
| **34025** | NASDAQ.KHC | NYSE.SNA | 0.700137 |
| **34026** | NYSE.AGN | NASDAQ.PBCT | 0.700136 |
| **34027** | NASDAQ.PBCT | NYSE.AGN | 0.700136 |
| **34028** | NYSE.AEP | NYSE.AME | 0.700134 |
| **34029** | NYSE.AME | NYSE.AEP | 0.700134 |
| **34030** | NYSE.CAT | NYSE.AEP | 0.700102 |
| **34031** | NYSE.AEP | NYSE.CAT | 0.700102 |
| **34032** | NASDAQ.EBAY | NYSE.YUM | 0.700083 |
| **34033** | NYSE.YUM | NASDAQ.EBAY | 0.700083 |
| **34034** | NYSE.CB | NYSE.PM | 0.700078 |
| **34035** | NYSE.PM | NYSE.CB | 0.700078 |
| **34036** | NYSE.MMM | NYSE.COH | 0.700066 |
| **34037** | NYSE.COH | NYSE.MMM | 0.700066 |
| **34038** | NYSE.NOC | NYSE.SWK | 0.700064 |
| **34039** | NYSE.SWK | NYSE.NOC | 0.700064 |
| **34040** | NYSE.FTV | NYSE.ICE | 0.700055 |
| **34041** | NYSE.ICE | NYSE.FTV | 0.700055 |
| **34042** | NYSE.DHI | NYSE.NEE | 0.700049 |
| **34043** | NYSE.NEE | NYSE.DHI | 0.700049 |
| **34044** | NYSE.IVZ | NYSE.ARE | 0.700049 |
| **34045** | NYSE.ARE | NYSE.IVZ | 0.700049 |
| **34046** | NYSE.ALB | NASDAQ.NVDA | 0.700005 |
| **34047** | NASDAQ.NVDA | NYSE.ALB | 0.700005 |

33548 rows × 3 columns

The stocks which are moving different to each other i.e. they are negatively co-related

In [145]:

sortCorr.loc[(sortCorr.sorted < -0.7)]

Out[145]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **level\_0** | **level\_1** | **sorted** |
| **231474** | NASDAQ.DISCK | NYSE.LLL | -0.700006 |
| **231475** | NYSE.LLL | NASDAQ.DISCK | -0.700006 |
| **231476** | NYSE.LNT | NYSE.M | -0.700007 |
| **231477** | NYSE.M | NYSE.LNT | -0.700007 |
| **231478** | NYSE.DPS | NYSE.BAX | -0.700029 |
| **231479** | NYSE.BAX | NYSE.DPS | -0.700029 |
| **231480** | NYSE.MMC | NYSE.GWW | -0.700033 |
| **231481** | NYSE.GWW | NYSE.MMC | -0.700033 |
| **231482** | NYSE.PNC | NASDAQ.DISCK | -0.700052 |
| **231483** | NASDAQ.DISCK | NYSE.PNC | -0.700052 |
| **231484** | NYSE.FMC | NYSE.JCI | -0.700073 |
| **231485** | NYSE.JCI | NYSE.FMC | -0.700073 |
| **231486** | NYSE.AMT | NYSE.UHS | -0.700079 |
| **231487** | NYSE.UHS | NYSE.AMT | -0.700079 |
| **231488** | NYSE.PHM | NYSE.LB | -0.700083 |
| **231489** | NYSE.LB | NYSE.PHM | -0.700083 |
| **231490** | NYSE.CBG | NYSE.BHI | -0.700088 |
| **231491** | NYSE.BHI | NYSE.CBG | -0.700088 |
| **231492** | NYSE.CXO | NYSE.CLX | -0.700119 |
| **231493** | NYSE.CLX | NYSE.CXO | -0.700119 |
| **231494** | NYSE.KR | NYSE.IVZ | -0.700146 |
| **231495** | NYSE.IVZ | NYSE.KR | -0.700146 |
| **231496** | NYSE.AFL | NYSE.COP | -0.700167 |
| **231497** | NYSE.COP | NYSE.AFL | -0.700167 |
| **231498** | NYSE.T | NYSE.IVZ | -0.700201 |
| **231499** | NYSE.IVZ | NYSE.T | -0.700201 |
| **231500** | NASDAQ.AKAM | NYSE.TDG | -0.700221 |
| **231501** | NYSE.TDG | NASDAQ.AKAM | -0.700221 |
| **231502** | NASDAQ.NDAQ | NYSE.UHS | -0.700238 |
| **231503** | NYSE.UHS | NASDAQ.NDAQ | -0.700238 |
| **...** | ... | ... | ... |
| **249970** | NASDAQ.ISRG | NYSE.APC | -0.957076 |
| **249971** | NYSE.APC | NASDAQ.ISRG | -0.957076 |
| **249972** | NYSE.CMG | NYSE.PVH | -0.957122 |
| **249973** | NYSE.PVH | NYSE.CMG | -0.957122 |
| **249974** | NYSE.YUM | NYSE.APC | -0.957417 |
| **249975** | NYSE.APC | NYSE.YUM | -0.957417 |
| **249976** | NYSE.NEE | NYSE.HP | -0.957630 |
| **249977** | NYSE.HP | NYSE.NEE | -0.957630 |
| **249978** | NYSE.NOC | NYSE.RRC | -0.957893 |
| **249979** | NYSE.RRC | NYSE.NOC | -0.957893 |
| **249980** | NYSE.MCD | NYSE.SLB | -0.959518 |
| **249981** | NYSE.SLB | NYSE.MCD | -0.959518 |
| **249982** | NYSE.ANTM | NYSE.APC | -0.959559 |
| **249983** | NYSE.APC | NYSE.ANTM | -0.959559 |
| **249984** | NYSE.BLK | NYSE.XEC | -0.962016 |
| **249985** | NYSE.XEC | NYSE.BLK | -0.962016 |
| **249986** | NYSE.NEE | NYSE.RRC | -0.962443 |
| **249987** | NYSE.RRC | NYSE.NEE | -0.962443 |
| **249988** | NYSE.HUM | NYSE.NBL | -0.963759 |
| **249989** | NYSE.NBL | NYSE.HUM | -0.963759 |
| **249990** | NYSE.MCD | NYSE.APC | -0.963995 |
| **249991** | NYSE.APC | NYSE.MCD | -0.963995 |
| **249992** | NYSE.MRO | NASDAQ.ISRG | -0.965143 |
| **249993** | NASDAQ.ISRG | NYSE.MRO | -0.965143 |
| **249994** | NASDAQ.ISRG | NYSE.SLB | -0.966186 |
| **249995** | NYSE.SLB | NASDAQ.ISRG | -0.966186 |
| **249996** | NYSE.KSU | NYSE.NFX | -0.967553 |
| **249997** | NYSE.NFX | NYSE.KSU | -0.967553 |
| **249998** | NYSE.APC | NYSE.AET | -0.976916 |
| **249999** | NYSE.AET | NYSE.APC | -0.976916 |