

stocks

June 10, 2020

```
[1]: import numpy as np
import pandas as pd
import gspread
from google.oauth2.service_account import Credentials
from gspread_pandas import Spread, Client
import seaborn as sns
import matplotlib.pyplot as plt
```

```
[2]: %matplotlib inline
```

```
[3]: url = 'https://en.wikipedia.org/wiki/List_of_S%26P_500_companies'
stocks_df = pd.read_html(url, header=0)[0]
stocks_df.head()
```

```
[3]: Symbol          Security SEC filings          GICS Sector \
0    MMM            3M Company      reports      Industrials
1    ABT  Abbott Laboratories      reports      Health Care
2    ABBV          AbbVie Inc.      reports      Health Care
3    ABMD          ABIOMED Inc      reports      Health Care
4    ACN      Accenture plc      reports  Information Technology

          GICS Sub Industry  Headquarters Location Date first added \
0    Industrial Conglomerates      St. Paul, Minnesota      1976-08-09
1    Health Care Equipment  North Chicago, Illinois      1964-03-31
2    Pharmaceuticals      North Chicago, Illinois      2012-12-31
3    Health Care Equipment  Danvers, Massachusetts      2018-05-31
4  IT Consulting & Other Services      Dublin, Ireland      2011-07-06

          CIK      Founded
0    66740      1902
1    1800      1888
2  1551152  2013 (1888)
3    815094      1981
4  1467373      1989
```

```
[4]: stocks_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```

RangeIndex: 505 entries, 0 to 504
Data columns (total 9 columns):
Symbol                505 non-null object
Security              505 non-null object
SEC filings           505 non-null object
GICS Sector           505 non-null object
GICS Sub Industry     505 non-null object
Headquarters Location 505 non-null object
Date first added      408 non-null object
CIK                   505 non-null int64
Founded               234 non-null object
dtypes: int64(1), object(8)
memory usage: 35.6+ KB

```

```
[5]: stocks_df[stocks_df['Security'].str.contains('Class')]
```

```
[5]:
```

	Symbol	Security	SEC filings	GICS Sector \
23	GOOGL	Alphabet Inc. (Class A)	reports	Communication Services
24	GOOG	Alphabet Inc. (Class C)	reports	Communication Services
147	DISCA	Discovery, Inc. (Class A)	reports	Communication Services
148	DISCK	Discovery, Inc. (Class C)	reports	Communication Services
203	FOXA	Fox Corporation (Class A)	reports	Communication Services
204	FOX	Fox Corporation (Class B)	reports	Communication Services
338	NWSA	News Corp. Class A	reports	Communication Services
339	NWS	News Corp. Class B	reports	Communication Services
456	UAA	Under Armour (Class A)	reports	Consumer Discretionary
457	UA	Under Armour (Class C)	reports	Consumer Discretionary

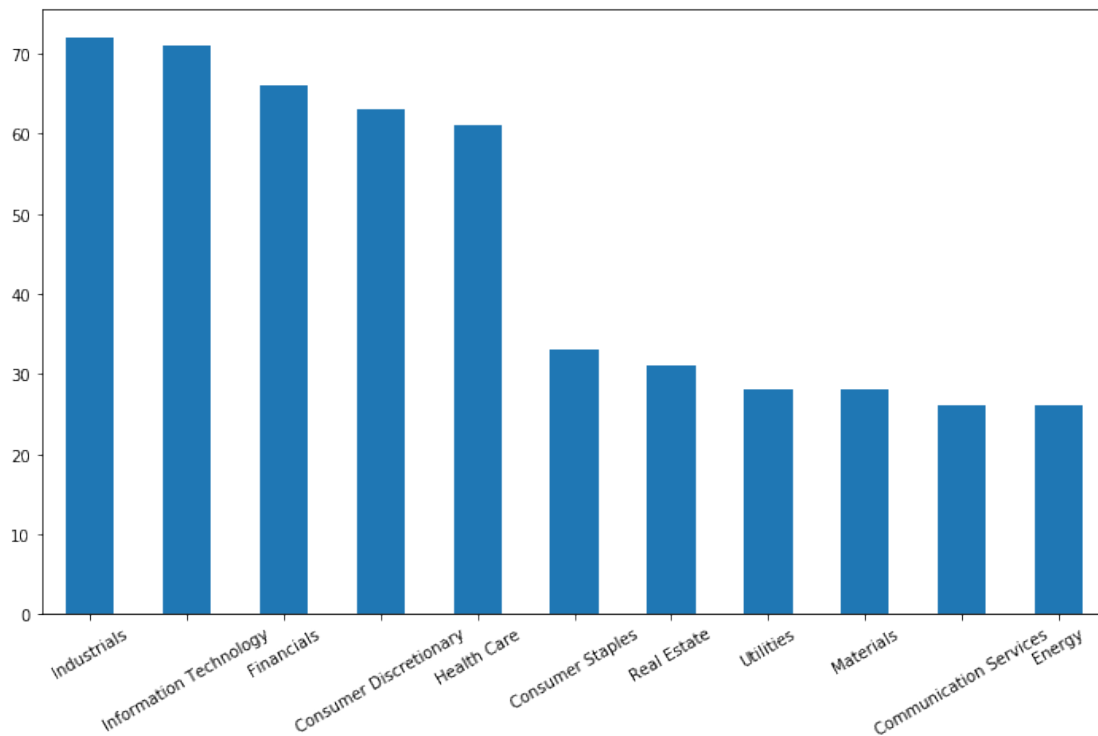
	GICS Sub Industry	Headquarters Location \
23	Interactive Media & Services	Mountain View, California
24	Interactive Media & Services	Mountain View, California
147	Broadcasting	Silver Spring, Maryland
148	Broadcasting	Silver Spring, Maryland
203	Movies & Entertainment	New York, New York
204	Movies & Entertainment	New York, New York
338	Publishing	New York, New York
339	Publishing	New York, New York
456	Apparel, Accessories & Luxury Goods	Baltimore, Maryland
457	Apparel, Accessories & Luxury Goods	Baltimore, Maryland

	Date first added	CIK	Founded
23	2014-04-03	1652044	1998
24	2006-04-03	1652044	1998
147	2010-03-01	1437107	NaN
148	2014-08-07	1437107	NaN
203	2013-07-01	1308161	NaN
204	2015-09-18	1308161	NaN

338	2013-08-01	1564708	NaN
339	2015-09-18	1564708	NaN
456	2014-05-01	1336917	NaN
457	2016-04-08	1336917	NaN

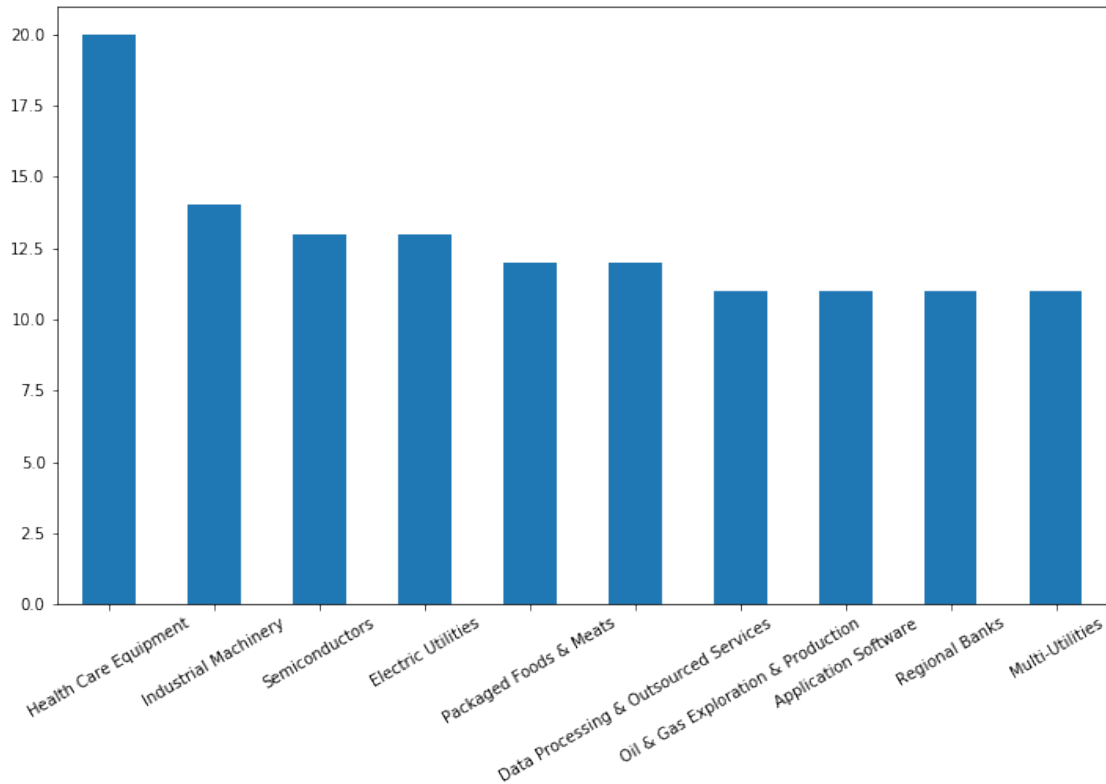
```
[6]: plt.figure(figsize=(12, 7))
stocks_df['GICS Sector'].value_counts().sort_values(ascending=False).plot(
    kind='bar', rot=30)
```

[6]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe064a5288>



```
[7]: plt.figure(figsize=(12, 7))
stocks_df['GICS Sub Industry'].value_counts().sort_values(
    ascending=False).head(10).plot(
    kind='bar', rot=30)
```

[7]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe065f8248>



```
[8]: stocks_df['Price_1_1'] = stocks_df['Symbol'].apply(
    lambda x: '=INDEX(GOOGLEFINANCE("'" + x + "'", "Price", "1/1/2020"),2,2)')
stocks_df['Price_3_23'] = stocks_df['Symbol'].apply(
    lambda x: '=INDEX(GOOGLEFINANCE("'" + x + "'", "Price", "3/23/2020"),2,2)')
stocks_df['Price_4_9'] = stocks_df['Symbol'].apply(
    lambda x: '=INDEX(GOOGLEFINANCE("'" + x + "'", "Price", "4/9/2020"),2,2)')
```

```
[9]: stocks_df['Shares'] = stocks_df['Symbol'].apply(lambda x: '=GOOGLEFINANCE("'" +
    x + "'", "shares")')
```

```
[10]: scope = [
    'https://spreadsheets.google.com/feeds',
    'https://www.googleapis.com/auth/drive'
]
credentials = Credentials.from_service_account_file(
    'credentials/credentials.json', scopes=scope)
```

```
[11]: client = Client(scope=scope, creds=credentials)
spread = Spread('stocks-data', client=client)
```

```
[12]: cols_to_keep = [
        'Symbol', 'Security', 'GICS Sector', 'GICS Sub Industry', 'Price_1_1',
        'Price_3_23', 'Price_4_9', 'Shares'
    ]
```

```
[13]: spread.df_to_sheet(stocks_df[cols_to_keep])
```

```
[14]: stocks_df = spread.sheet_to_df()
stocks_df.head()
```

```
[14]:
```

	Symbol	Security	GICS Sector \
index			
0	MMM	3M Company	Industrials
1	ABT	Abbott Laboratories	Health Care
2	ABBV	AbbVie Inc.	Health Care
3	ABMD	ABIOMED Inc	Health Care
4	ACN	Accenture plc	Information Technology

	GICS Sub Industry	Price_1_1	Price_3_23	Price_4_9 \
index				
0	Industrial Conglomerates	180	117.87	147.78
1	Health Care Equipment	86.95	62.82	86.04
2	Pharmaceuticals	89.55	64.5	79.75
3	Health Care Equipment	168.81	132.34	160.08
4	IT Consulting & Other Services	210.15	143.69	177.92

	Shares
index	
0	575196000
1	1768845000
2	1762342000
3	44957000
4	637027000

```
[15]: for index, values in stocks_df.iteritems():
        for i,v in values.items():
            if '#N/A' in v:
                stocks_df.drop(i, inplace=True)
```

```
[16]: stocks_df[['Price_1_1', 'Price_3_23', 'Price_4_9', 'Shares']] = stocks_df[['Price_1_1', 'Price_3_23', 'Price_4_9', 'Shares']]
        .apply(pd.to_numeric)
```

```
[17]: stocks_df['Marketcap_1_1'] = stocks_df['Price_1_1'] * stocks_df['Shares']
stocks_df['Marketcap_3_23'] = stocks_df['Price_3_23'] * stocks_df['Shares']
stocks_df['Marketcap_4_9'] = stocks_df['Price_4_9'] * stocks_df['Shares']
```

```
[18]: stocks_df['PercentageChange_3_23_1_1'] = (
        stocks_df['Price_3_23'] -
        stocks_df['Price_1_1']) / stocks_df['Price_1_1'] * 100
stocks_df['PercentageChange_4_9_3_23'] = (
        stocks_df['Price_4_9'] -
        stocks_df['Price_3_23']) / stocks_df['Price_3_23'] * 100
stocks_df['PercentageChange_4_9_1_1'] = (
        stocks_df['Price_4_9'] -
        stocks_df['Price_1_1']) / stocks_df['Price_1_1'] * 100
```

```
[19]: sum(stocks_df['Marketcap_3_23'] - stocks_df['Marketcap_1_1']) / 10**9
```

```
[19]: -8622.55383716952
```

```
[20]: sum(stocks_df['Marketcap_4_9'] - stocks_df['Marketcap_3_23']) / 10**9
```

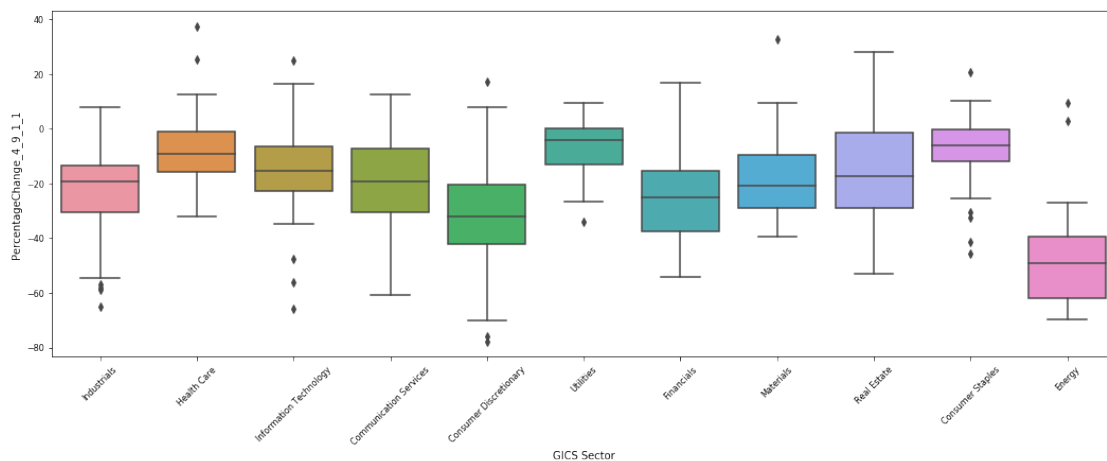
```
[20]: 4701.654194262481
```

```
[21]: sum(stocks_df['Marketcap_4_9'] - stocks_df['Marketcap_1_1']) / 10**9
```

```
[21]: -3920.89964290704
```

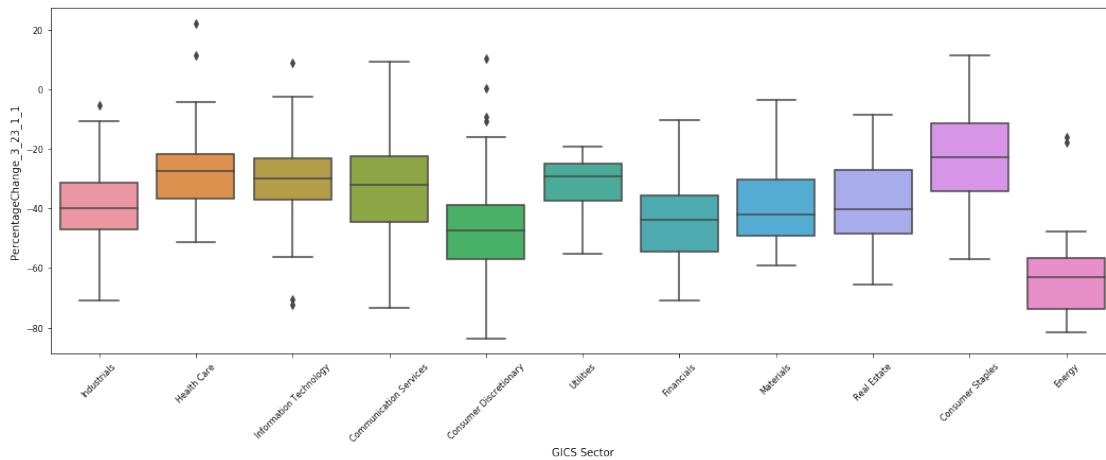
```
[22]: plt.figure(figsize=(18, 6))
plt.tick_params('both', labelsize='8')
plt.xticks(rotation=45)
sns.boxplot(
    x=stocks_df['GICS Sector'], y=stocks_df['PercentageChange_4_9_1_1'])
```

```
[22]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe087b1208>
```



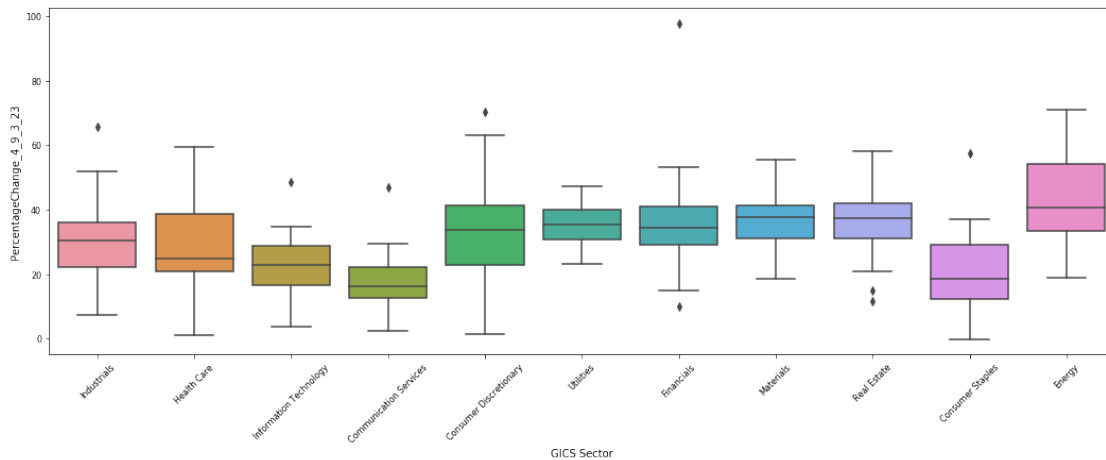
```
[23]: plt.figure(figsize=(18, 6))
plt.tick_params('both', labelsize='8')
plt.xticks(rotation=45)
sns.boxplot(
    x=stocks_df['GICS Sector'], y=stocks_df['PercentageChange_3_23_1_1'])
```

[23]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe08875c48>



```
[24]: plt.figure(figsize=(18, 6))
plt.tick_params('both', labelsize='8')
plt.xticks(rotation=45)
sns.boxplot(
    x=stocks_df['GICS Sector'], y=stocks_df['PercentageChange_4_9_3_23'])
```

[24]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe08d51c48>



```
[25]: plt.figure(figsize=(18, 6))
plt.tick_params('both', labelsize='8')
stocks_df.groupby('GICS Sub Industry').mean(
)['PercentageChange_4_9_1_1'].sort_values().plot.bar()
```

[25]: <matplotlib.axes._subplots.AxesSubplot at 0x1fe08e63c48>

