Data In Motion Week 6 Pandas Challenge

By Fasina Inioluwa

Challenge Questions

1. What is the frequency of the dataset? (The time period between each row)
2. What is the data type of the index?
3. Set the index to a Datetime.
4. Change the frequency to monthly, sum the values and assign it to new variable called monthly.
5. You will notice that it filled the dataFrame with months that don't have any data with NaN. Let's drop these rows.
6. Good, now we have the monthly data. Now change the frequency to year and assign to a new variable called year.
7. Create your own question and answer it.

Importing modules

In [1]:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import datetime as dt
```

Loading in the data

In [2]:

```
url = 'https://raw.githubusercontent.com/datasets/investor-flow-of-funds-us/master/data/weekly.csv'
df = pd.read_csv(url)
```

In [3]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44 entries, 0 to 43
Data columns (total 9 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Date             44 non-null     object
 1   Total Equity     44 non-null     int64 
 2   Domestic Equity  44 non-null     int64 
 3   World Equity     44 non-null     int64 
 4   Hybrid           44 non-null     int64 
 5   Total Bond       44 non-null     int64 
 6   Taxable Bond     44 non-null     int64 
 7   Municipal Bond   44 non-null     int64 
 8   Total            44 non-null     int64 
dtypes: int64(8), object(1)
memory usage: 3.2+ KB
```

In [4]:

```
df.describe()
```

Out[4]:

	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
count	44.000000	44.000000	44.000000	44.000000	44.000000	44.000000	44.000000	44.000000
mean	-161.727273	-1815.931818	1654.227273	684.227273	2452.613636	1931.977273	520.613636	2975.045455
std	4318.401639	3223.717184	1533.770151	670.325983	2729.245012	2574.644536	786.337321	6228.735692
min	-11156.000000	-9622.000000	-1533.000000	-1057.000000	-8193.000000	-8658.000000	-3369.000000	-9040.000000
25%	-1758.500000	-3385.750000	883.250000	249.500000	1524.500000	1089.000000	279.000000	-549.750000
50%	476.000000	-1774.000000	1563.500000	791.500000	2417.000000	1863.500000	668.000000	3948.000000
75%	1916.500000	-22.000000	2561.500000	1072.000000	3610.750000	3235.500000	855.500000	5844.500000
max	14817.000000	7995.000000	6821.000000	2888.000000	9766.000000	7311.000000	2455.000000	27471.000000

In [24]:

```
df.head()
```

Out[24]:

Date	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
2012-12-05	-7426	-6060	-1367	-74	5317	4210	1107	-2183
2012-12-12	-8783	-7520	-1263	123	1818	1598	219	-6842
2012-12-19	-5496	-5470	-26	-73	103	3472	-3369	-5466
2012-12-26	-4451	-4076	-375	550	2610	3333	-722	-1291
2013-01-02	-11156	-9622	-1533	-158	2383	2103	280	-8931

1. What is the frequency of the dataset? (The time period between each row)

In [6]:

```
frequency = (pd.to_datetime(df["Date"])).diff()
frequency.value_counts()
```

Out[6]:

```
7 days      37
21 days      2
56 days      1
43 days      1
6 days       1
448 days     1
Name: Date, dtype: int64
```

In [14]:

```
frequency.median().days
```

Out[14]:

```
7
```

The frequency of the dataset is 7 days

2. What is the data type of the index?

In [7]:

```
index_dtype = df.index.dtype
print("The data type of the dataframe index is {}".format(index_dtype))
```

```
The data type of the dataframe index is int64
```

3. Set the index to a Datetime.

In [8]:

```
df["Date"] = pd.to_datetime(df["Date"])
df.set_index("Date", inplace=True)
df.head()
```

Out[8]:

Date	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
2012-12-05	-7426	-6060	-1367	-74	5317	4210	1107	-2183
2012-12-12	-8783	-7520	-1263	123	1818	1598	219	-6842
2012-12-19	-5496	-5470	-26	-73	103	3472	-3369	-5466
2012-12-26	-4451	-4076	-375	550	2610	3333	-722	-1291
2013-01-02	-11156	-9622	-1533	-158	2383	2103	280	-8931

4. Change the frequency to monthly, sum the values and assign it to new variable called monthly.

In [9]:

```
monthly = df.resample("M").sum()
monthly.head(10)
```

Out[9]:

Date	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
2012-12-31	-26156	-23126	-3031	526	9848	12613	-2765	-15782
2013-01-31	3661	-1627	5288	2730	12149	9414	2735	18540
2013-02-28	0	0	0	0	0	0	0	0
2013-03-31	0	0	0	0	0	0	0	0
2013-04-30	0	0	0	0	0	0	0	0
2013-05-31	0	0	0	0	0	0	0	0
2013-06-30	0	0	0	0	0	0	0	0
2013-07-31	0	0	0	0	0	0	0	0
2013-08-31	0	0	0	0	0	0	0	0
2013-09-30	0	0	0	0	0	0	0	0

5. You will notice that it filled the dataFrame with months that don't have any data with NaN. Let's drop these rows.

The months without data were filled with "0", so df.dropna() didn't work

In [10]:

```
filter = monthly["Total Equity"] != 0
monthly = monthly[filter]
monthly
```

Out[10]:

Date	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
2012-12-31	-26156	-23126	-3031	526	9848	12613	-2765	-15782
2013-01-31	3661	-1627	5288	2730	12149	9414	2735	18540
2014-04-30	10842	1048	9794	4931	8493	7193	1300	24267
2014-05-31	-2203	-8720	6518	3172	13767	10192	3576	14736
2014-06-30	2319	-6546	8865	4588	9715	7551	2163	16621
2014-07-31	-7051	-11128	4078	2666	7506	7026	481	3122
2014-08-31	1943	-5508	7452	1885	1897	-1013	2910	5723
2014-09-30	-2767	-6596	3829	1599	3984	2479	1504	2816
2014-11-30	-2753	-7239	4485	729	14528	11566	2962	12502
2015-01-31	3471	-1164	4635	1729	7368	2762	4606	12569
2015-02-28	5508	3509	1999	1752	9099	7443	1656	16359
2015-03-31	5691	-8176	13867	2829	9138	7267	1870	17657
2015-04-30	379	-4628	5007	970	423	514	-91	1772

6. Good, now we have the monthly data. Now change the frequency to year and assign to a new variable called year.

In [20]:

```
year = df.resample("Y").sum()
year
```

Out[20]:

Date	Total Equity	Domestic Equity	World Equity	Hybrid	Total Bond	Taxable Bond	Municipal Bond	Total
2012-12-31	-26156	-23126	-3031	526	9848	12613	-2765	-15782
2013-12-31	3661	-1627	5288	2730	12149	9414	2735	18540
2014-12-31	330	-44689	45021	19570	59890	44994	14896	79787
2015-12-31	15049	-10459	25508	7280	26028	17986	8041	48357

7. What is the correlation between equity and bonds?

In [35]:

```
df["Total Bond"].corr(df["Total Equity"])
```

Out[35]:

```
0.2957113783444713
```

In [37]:

```
df["Total Bond"].corr(df["Total Equity"], method = "spearman")
```

Out[37]:

```
0.12276250880902043
```

In [38]:

```
df["Total Bond"].corr(df["Total Equity"], method="kendall")
```

Out[38]:

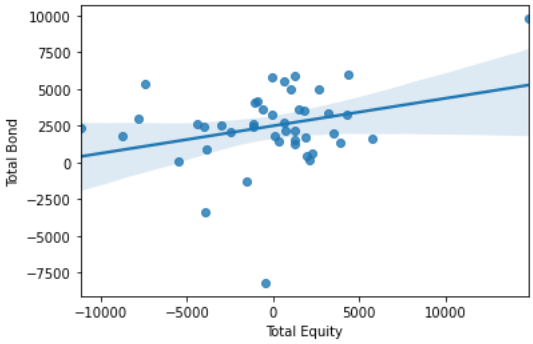
```
0.07610993657505284
```

In [34]:

```
sns.regplot(y="Total Bond", x="Total Equity", data = df)
```

Out[34]:

```
<AxesSubplot:xlabel='Total Equity', ylabel='Total Bond'>
```



The business has a slighly positive equity-bond correlation, indicating a low equity market volatility