Assignment4a

May 29, 2019

1 Did Snowfall and Rainfall Effect Temperature of United States of America (1961 - 1990)

1.1 Assignment 4

Before working on this assignment please read these instructions fully. In the submission area, you will notice that you can click the link to **Preview the Grading** for each step of the assignment. This is the criteria that will be used for peer grading. Please familiarize yourself with the criteria before beginning the assignment.

This assignment requires that you to find **at least** two datasets on the web which are related, and that you visualize these datasets to answer a question with the broad topic of **weather phenomena** (see below) for the region of **Ann Arbor, Michigan, United States**, or **United States** more broadly.

You can merge these datasets with data from different regions if you like! For instance, you might want to compare **Ann Arbor, Michigan, United States** to Ann Arbor, USA. In that case at least one source file must be about **Ann Arbor, Michigan, United States**.

You are welcome to choose datasets at your discretion, but keep in mind **they will be shared with your peers**, so choose appropriate datasets. Sensitive, confidential, illicit, and proprietary materials are not good choices for datasets for this assignment. You are welcome to upload datasets of your own as well, and link to them using a third party repository such as github, bitbucket, pastebin, etc. Please be aware of the Coursera terms of service with respect to intellectual property.

Also, you are welcome to preserve data in its original language, but for the purposes of grading you should provide english translations. You are welcome to provide multiple visuals in different languages if you would like!

As this assignment is for the whole course, you must incorporate principles discussed in the first week, such as having as high data-ink ratio (Tufte) and aligning with Cairo's principles of truth, beauty, function, and insight.

Here are the assignment instructions:

- State the region and the domain category that your data sets are about (e.g., **Ann Arbor, Michigan, United States** and **weather phenomena**).
- You must state a question about the domain category and region that you identified as being interesting.
- You must provide at least two links to available datasets. These could be links to files such as CSV or Excel files, or links to websites which might have data in tabular form, such as Wikipedia pages.

- You must upload an image which addresses the research question you stated. In addition to addressing the question, this visual should follow Cairo's principles of truthfulness, functionality, beauty, and insightfulness.
- You must contribute a short (1-2 paragraph) written justification of how your visualization addresses your stated research question.

What do we mean by **weather phenomena**? For this category you might want to consider seasonal changes, natural disasters, or historical trends.

1.2 Tips

- Wikipedia is an excellent source of data, and I strongly encourage you to explore it for new data sources.
- Many governments run open data initiatives at the city, region, and country levels, and these are wonderful resources for localized data sources.
- Several international agencies, such as the United Nations, the World Bank, the Global Open Data Index are other great places to look for data.
- This assignment requires you to convert and clean datafiles. Check out the discussion forums for tips on how to do this from various sources, and share your successes with your fellow students!

1.3 Example

Looking for an example? Here's what our course assistant put together for the **Ann Arbor, MI, USA** area using **sports and athletics** as the topic. Example Solution File

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt

        %matplotlib notebook
```

```
In [2]: plt.style.use('seaborn-colorblind')
```

```
In [3]: Snowfall = pd.read_csv('UNdata_Snowfall.csv')
        Snowfall.head()
```

Out[3]:

	Country or Territory	Station Name	WMO Station Numb
0	UNITED STATES OF AMERICA	BARROW/W. POST W. ROGERS, AK	70026
1	UNITED STATES OF AMERICA	KOTZEBUE/RALPH WIEN, AK	70133
2	UNITED STATES OF AMERICA	BETTLES/FIELD AK	70174
3	UNITED STATES OF AMERICA	NOME, AK	70200
4	UNITED STATES OF AMERICA	BETHEL/BETHEL AIRPORT, AK	70219

	National Station Id Number	Period	Element-Statistic Qualifier Code
0	500546.0	1961-1990	NaN
1	505076.0	1961-1990	NaN
2	500761.0	1961-1990	NaN
3	506496.0	1961-1990	NaN

```
4                   500754.0  1961-1990                                  NaN

  Statistic Description Unit   Jan  Jan Footnotes  \
0          Median Value   cm   4.3            NaN
1          Median Value   cm  14.2            NaN
2          Median Value   cm  26.4            NaN
3          Median Value   cm  17.3            NaN
4          Median Value   cm  14.2            NaN

                    ...                   Oct  Oct Footnotes   Nov  \
0                   ...                  14.0            NaN   6.6
1                   ...                  15.2            NaN  19.0
2                   ...                  28.7            NaN  27.4
3                   ...                  10.4            NaN  27.4
4                   ...                   9.4            NaN  18.8

   Nov Footnotes   Dec  Dec Footnotes  Annual  Annual Footnotes  \
0            NaN   5.1            NaN -9999.9               2.0
1            NaN  16.5            NaN -9999.9               2.0
2            NaN  36.3            NaN -9999.9               2.0
3            NaN  22.9            NaN -9999.9               2.0
4            NaN  17.3            NaN -9999.9               2.0

   Annual NCDC Computed Value  Annual NCDC Computed Value Footnotes
0                     -9999.9                                  2.0
1                     -9999.9                                  2.0
2                     -9999.9                                  2.0
3                     -9999.9                                  2.0
4                     -9999.9                                  2.0

[5 rows x 36 columns]
```

```
In [4]: Temperature = pd.read_excel('Temp_1961_1990.xls')
        Temperature.head()
```

Out[4]:

	tas	\tYear	Month	Country	ISO3	ISO2
0	-5.21460	1961	1	USA	NaN	NaN
1	-3.21400	1961	2	USA	NaN	NaN
2	-0.15080	1961	3	USA	NaN	NaN
3	4.52072	1961	4	USA	NaN	NaN
4	11.86120	1961	5	USA	NaN	NaN

```
In [5]: Rainfall = pd.read_excel('Rain_1961_1990.xls')
        Rainfall.head()
```

Out[5]:

	pr	\tYear	Month	Country	ISO3	ISO2
0	29.8744	1961	1	USA	NaN	NaN
1	52.7065	1961	2	USA	NaN	NaN
2	59.5436	1961	3	USA	NaN	NaN

```
3  48.6108    1961        4       USA     NaN     NaN
4  57.4379    1961        5       USA     NaN     NaN
```

```
In [6]: Jan = Snowfall['Jan'].mean()
        Feb = Snowfall['Feb'].mean()
        Mar = Snowfall['Mar'].mean()
        Apr = Snowfall['Apr'].mean()
        May = Snowfall['May'].mean()
        Jun = Snowfall['Jun'].mean()
        Jul = Snowfall['Jul'].mean()
        Aug = Snowfall['Aug'].mean()
        Sep = Snowfall['Sep'].mean()
        Oct = Snowfall['Oct'].mean()
        Nov = Snowfall['Nov'].mean()
        Dec = Snowfall['Dec'].mean()
```

```
In [7]: Weather = pd.DataFrame({'Snowfall': [Jan, Feb, Mar, Apr, May, Jun, Jul, Aug
        Weather
```

```
Out[7]:          Snowfall
        0   32910.929764
        1   32099.560367
        2   32814.367087
        3   35263.127507
        4   35805.531129
        5   30106.764987
        6   29014.876535
        7   28906.219423
        8   32462.127717
        9   35576.560892
        10  34939.955171
        11  33215.316745
```

```
In [8]: Monthly_Temp = Temperature.groupby(' Month').mean()['tas']
        Jan = Monthly_Temp.iloc[0]
        Feb = Monthly_Temp.iloc[1]
        Mar = Monthly_Temp.iloc[2]
        Apr = Monthly_Temp.iloc[3]
        May = Monthly_Temp.iloc[4]
        Jun = Monthly_Temp.iloc[5]
        Jul = Monthly_Temp.iloc[6]
        Aug = Monthly_Temp.iloc[7]
        Sep = Monthly_Temp.iloc[8]
        Oct = Monthly_Temp.iloc[9]
        Nov = Monthly_Temp.iloc[10]
        Dec = Monthly_Temp.iloc[11]
```

```
In [9]: Weather['Temperature'] = [Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct,
        Weather
```

Out[9]:

	Snowfall	Temperature
0	32910.929764	-6.266277
1	32099.560367	-4.228557
2	32814.367087	0.161376
3	35263.127507	5.987872
4	35805.531129	12.186650
5	30106.764987	17.210970
6	29014.876535	20.030330
7	28906.219423	18.935290
8	32462.127717	14.488357
9	35576.560892	7.789156
10	34939.955171	0.468967
11	33215.316745	-4.694447

In [10]:
```
Monthly_Rain = Rainfall.groupby(' Month').mean()['pr']
Monthly_Rain.iloc[0]
Jan = Monthly_Rain.iloc[0]
Feb = Monthly_Rain.iloc[1]
Mar = Monthly_Rain.iloc[2]
Apr = Monthly_Rain.iloc[3]
May = Monthly_Rain.iloc[4]
Jun = Monthly_Rain.iloc[5]
Jul = Monthly_Rain.iloc[6]
Aug = Monthly_Rain.iloc[7]
Sep = Monthly_Rain.iloc[8]
Oct = Monthly_Rain.iloc[9]
Nov = Monthly_Rain.iloc[10]
Dec = Monthly_Rain.iloc[11]
```

In [11]:
```
Weather['Rainfall'] = [Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, N
Weather
```

Out[11]:

	Snowfall	Temperature	Rainfall
0	32910.929764	-6.266277	44.567710
1	32099.560367	-4.228557	41.915983
2	32814.367087	0.161376	51.627430
3	35263.127507	5.987872	49.433157
4	35805.531129	12.186650	60.396267
5	30106.764987	17.210970	64.473557
6	29014.876535	20.030330	65.537783
7	28906.219423	18.935290	65.038193
8	32462.127717	14.488357	61.432697
9	35576.560892	7.789156	49.504067
10	34939.955171	0.468967	50.642860
11	33215.316745	-4.694447	50.472827

In [12]:
```
Weather['Snowfall'] = Weather['Snowfall']/1000
Weather
```

Out[12]:

	Snowfall	Temperature	Rainfall
0	32.910930	-6.266277	44.567710
1	32.099560	-4.228557	41.915983
2	32.814367	0.161376	51.627430
3	35.263128	5.987872	49.433157
4	35.805531	12.186650	60.396267
5	30.106765	17.210970	64.473557
6	29.014877	20.030330	65.537783
7	28.906219	18.935290	65.038193
8	32.462128	14.488357	61.432697
9	35.576561	7.789156	49.504067
10	34.939955	0.468967	50.642860
11	33.215317	-4.694447	50.472827

In [13]:
```
Weather.index = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', '
Weather
```

Out[13]:

	Snowfall	Temperature	Rainfall
Jan	32.910930	-6.266277	44.567710
Feb	32.099560	-4.228557	41.915983
Mar	32.814367	0.161376	51.627430
Apr	35.263128	5.987872	49.433157
May	35.805531	12.186650	60.396267
Jun	30.106765	17.210970	64.473557
Jul	29.014877	20.030330	65.537783
Aug	28.906219	18.935290	65.038193
Sep	32.462128	14.488357	61.432697
Oct	35.576561	7.789156	49.504067
Nov	34.939955	0.468967	50.642860
Dec	33.215317	-4.694447	50.472827

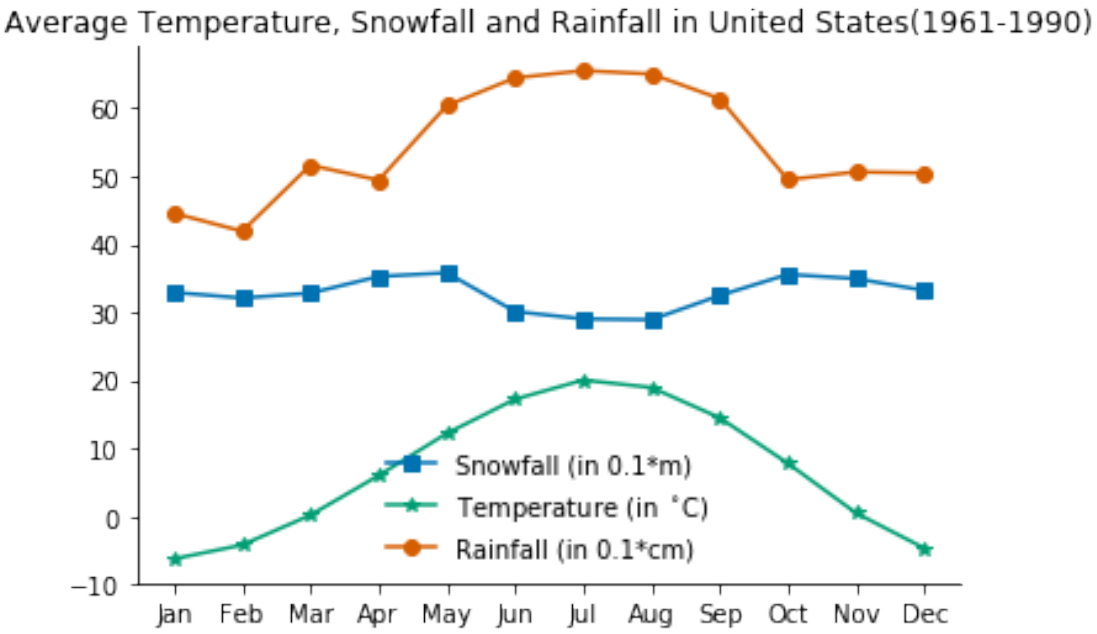
In [14]:
```
x = Weather.index.tolist()
y1 = Weather['Snowfall'].tolist()
y2 = Weather['Temperature'].tolist()
y3 = Weather['Rainfall'].tolist()

plt.plot(x, y1, '-s', label = "Snowfall (in 0.1*m)")
plt.plot(x, y2, '-*', label = "Temperature (in $\,^{\circ}\mathrm{C}$)")
plt.plot(x, y3, '-o', label = "Rainfall (in 0.1*cm)")

plt.title('Average Temperature, Snowfall and Rainfall in United States(196

plt.legend(frameon=False)
[plt.gca().spines[loc].set_visible(False) for loc in ['top', 'right']]

plt.show()
```


Average Temperature, Snowfall and Rainfall in United States(1961-1990)
60
50
40
30
20
10
0
−10
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Snowfall (in 0.1*m)
Temperature (in ˚C)
Rainfall (in 0.1*cm)