Project #1: Explore Weather Trend

Date: Sep03-2019
Name: Sami Adham
Program: Data Analysis Nanodegree

Project Task:

The project is to show the similarities and differences between my city and global by using visualization chart. Also, to prepare a PDF documents explain all the steps, results and comments.

Goals:

1. How to extract data from database using SQL query
2. How to use statistical equation to answer real-life question. Such as comparison between nearest city avg. temperature and global avg. temperature.
3. How to use chart visualization to represent your result.

Tools:

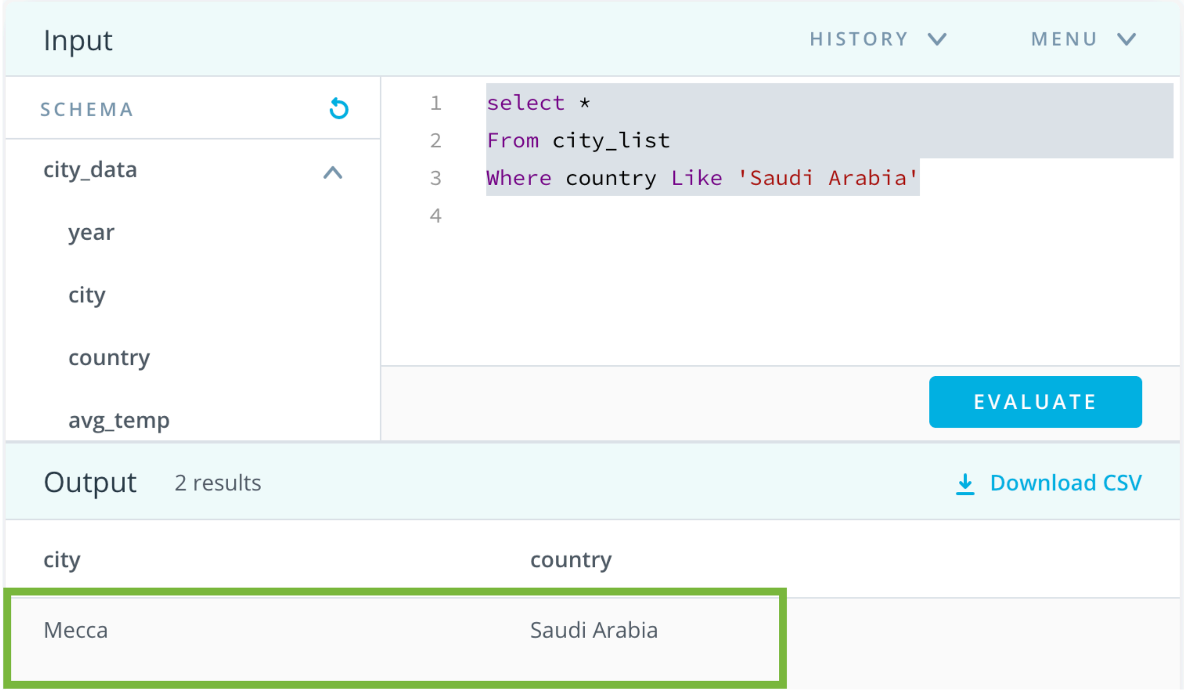
1. SQL query to extract data from UDACITY remote server and download it as .CSV format.
2. I preferred using python through Jupyter notebook (ANACONDA) to take a chance to practice in this first project for future projects.

Project Steps:

- Step#1: Extract Data from database using SQL query
- Step#2: Analyze data to solve problem by Python (Equations, Methods, Libraries, Charts)
- Step#3: Observation and result

Step#1: Extract Data from database using SQL query:

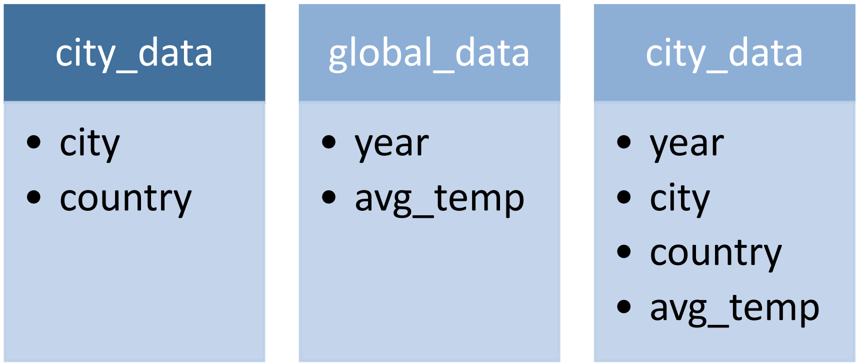
- First of all, we must find nearest city from my home which is JEDDAH. So, I used the below SQL query, then I found ‘Mecca’ which is typed differently than what I know which is ‘Makkah’.



```sql
select *
From city_list
Where country Like 'Saudi Arabia'
```

city	country
Mecca	Saudi Arabia

- Secondly, I want to extract 'Mecca' avg. temp and global avg. temp from schema. But I faced a problem as below figure:

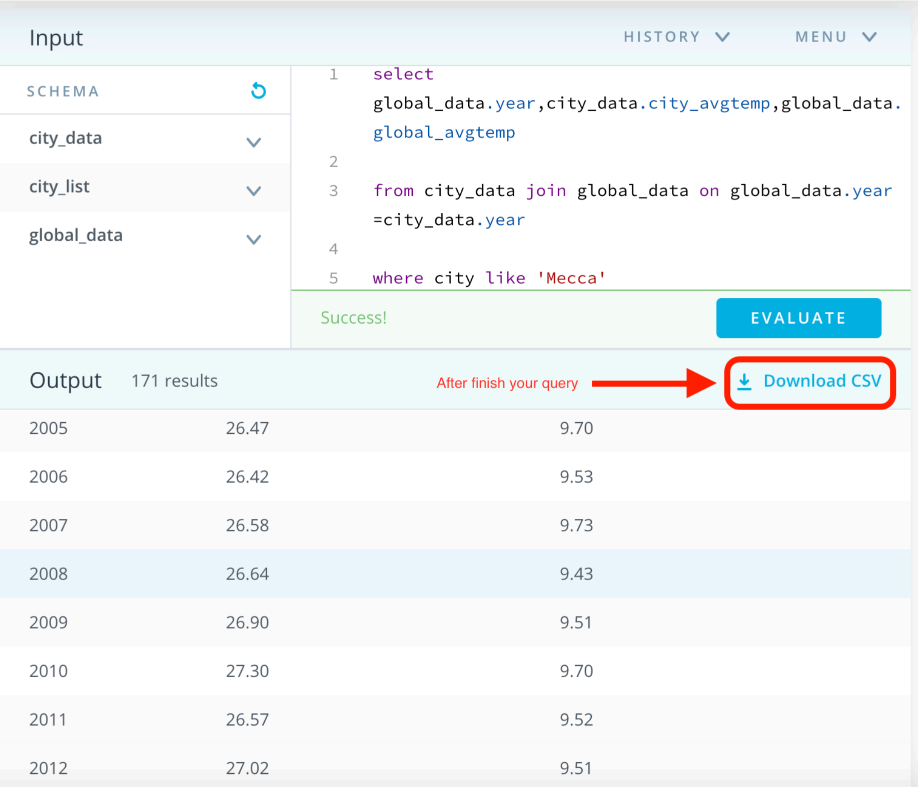


city_data	global_data	city_data
• city • country	• year • avg_temp	• year • city • country • avg_temp

As shown in the above Schema there is duplicate so the solution is to rename some elements in tables as following commands:

```sql
ALTER TABLE city_data RENAME COLUMN avg_temp to
city_avgtemp;
ALTER TABLE global_data RENAME COLUMN avg_temp to
global_avgtemp;
```

Now, we can join Table and extract Year, city_avgtemp , and global_avgtemp as below SQL query:



```
select
global_data.year,city_data.city_avgtemp,global_data.global_avgtemp

from city_data join global_data on global_data.year =city_data.year

where city like 'Mecca'
```

After finish your query

2005	26.47	9.70
2006	26.42	9.53
2007	26.58	9.73
2008	26.64	9.43
2009	26.90	9.51
2010	27.30	9.70
2011	26.57	9.52
2012	27.02	9.51

Finally, I download my SQL join result as CSV file.

Step#2: Analyze data to solve problem by Python

In this step I need to find the average moving data to plot it. I used python libraries to obtain moving average. That is why I imported numpy, pandas, and matplotlib. Also, I need to use DataFrame.rolling(), mean() and dropna() as well. To find moving average data as desired.

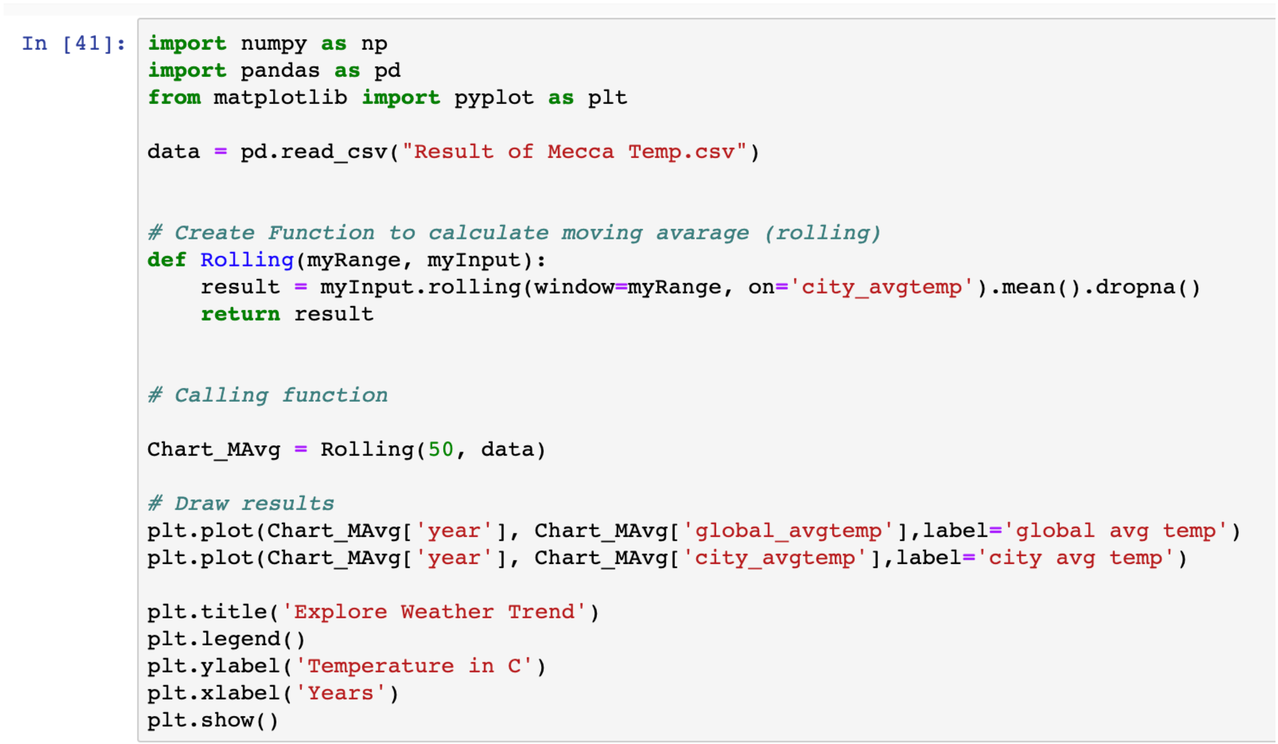
1. Rolling syntax as following:

```
DataFrame.rolling(self, window, min_periods=None, center=False, win_type=None, on=None, axis=0, closed=None)
```

In our case we need the following:

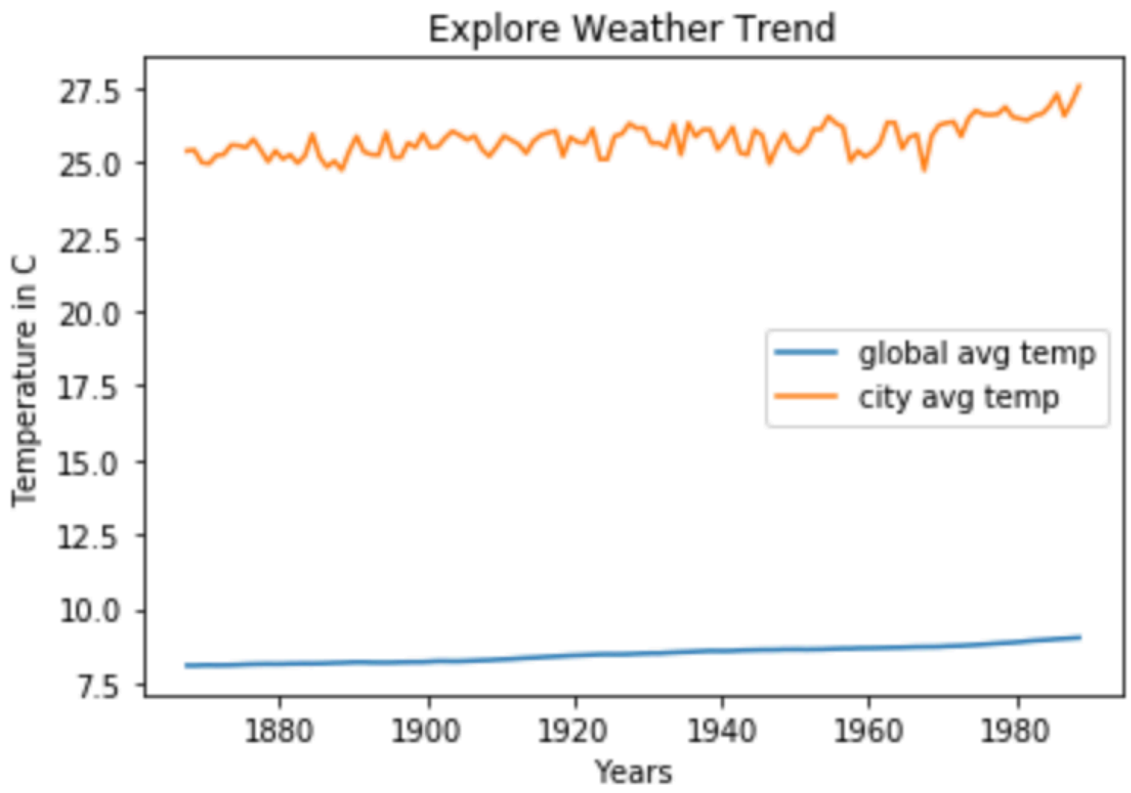
Window: The range of observation among the years. [in my case: **myRange**]
on: To specify which column we should apply this rule. [in my case: **city_avgtemp]**

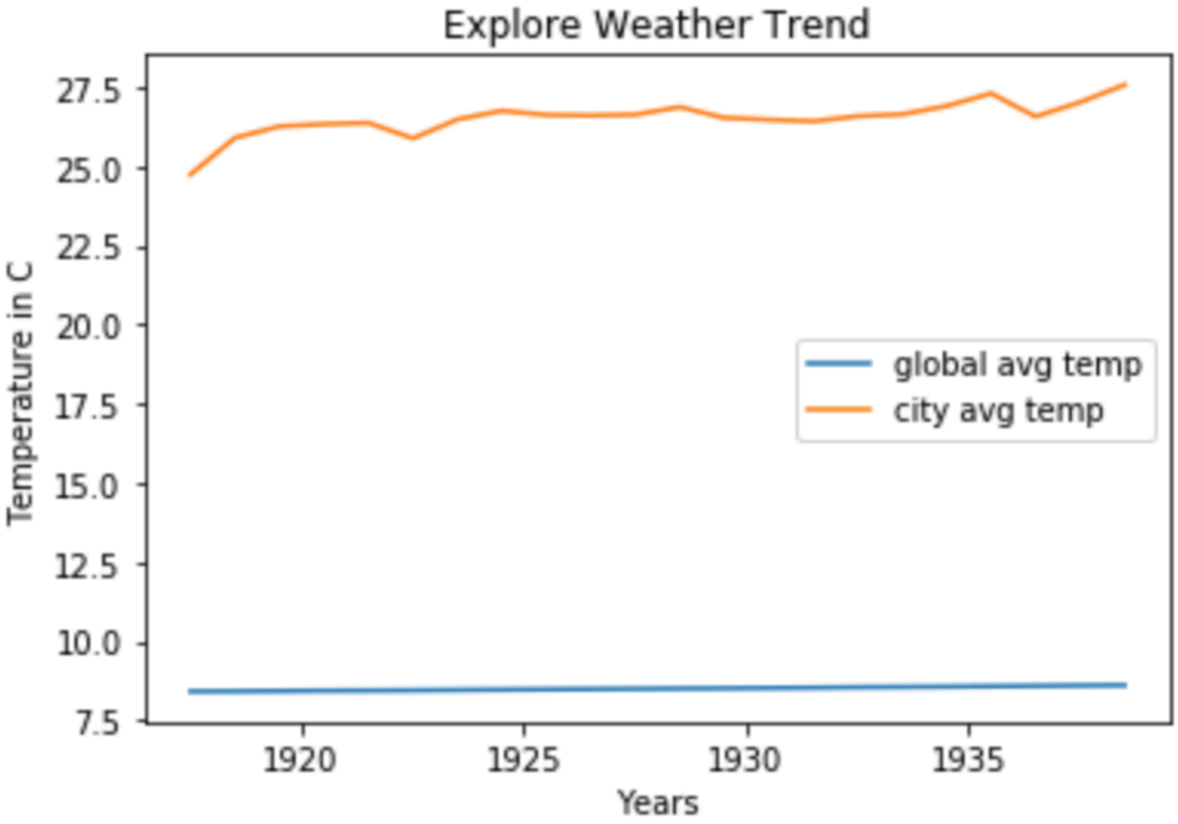
2. Mean() to find average mean for a rage of year [I name it: myRage]
3. Dropna() to remove unnecessary data which affect the quality of chart.
4. The following code is snapshot of my python code:



```
In [41]: import numpy as np
         import pandas as pd
         from matplotlib import pyplot as plt

         data = pd.read_csv("Result of Mecca Temp.csv")


         # Create Function to calculate moving avarage (rolling)
         def Rolling(myRange, myInput):
             result = myInput.rolling(window=myRange, on='city_avgtemp').mean().dropna()
             return result


         # Calling function

         Chart_MAvg = Rolling(50, data)

         # Draw results
         plt.plot(Chart_MAvg['year'], Chart_MAvg['global_avgtemp'],label='global avg temp')
         plt.plot(Chart_MAvg['year'], Chart_MAvg['city_avgtemp'],label='city avg temp')

         plt.title('Explore Weather Trend')
         plt.legend()
         plt.ylabel('Temperature in C')
         plt.xlabel('Years')
         plt.show()
```

5. The following chart is snapshot of my results: First chart with range 50 years and the second one with range of 150 years.





Step#3: Observation and result

1. As long as I decrease my range of years such as 50 years in the first line chart it will become distorted. In other hand, as long as I increase my range such as 150 years it will become smooth. That because we increase our population to calculate mean.
2. The X-Axis changed depends on ranges for example when I used 50 years range the X-Axis become larger that when we used 150 Years which become shorter.
3. There is big difference between my country 'Mecca' average temperature and global temperature.
4. There is a slight increase in global average temperature.
5. The city average temp trend is increasing over the years.

References:

https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.rolling.html