EXPLORE WEATHER TRENDS

Udacity DAND Project 1

What tools did you use for each step?

The data used in this analysis was collected from database using SQL and processed further using three python libraries(Pandas,Numpy,MATPLOTLIB) in Jupyter Notebook.

STEP 1: SQL EXTRACTION OF DATA FROM DB

Temperature Database: There are three tables in the database:

- **city_list** - This contains a list of cities and countries in the database. Look through them in order to find the city nearest to you.
- **city_data** - This contains the average temperatures for each city by year (°C).
- **global_data** - This contains the average global temperatures by year (°C).

In order to simplify ,the two tables(relations) joined the two relations **city_data** and **global_data** using **INNER JOIN** on Year. Here in both relations we have **avg_temp name** column in order to ease the process of comparison of temperatures for both the columns we have renamed the column to **city_avg_temp** and **global_avg_temp** using **ALTER** command.All queries used are list out in next slides as well as SQLQueries.txt file provided in the submitted folder.

Query 1: *To list out all cities in 'India'*

SELECT *

FROM city_list

WHERE country like 'India' ;

It returns 22 cities in which Delhi is the most nearby city to Gurgaon.So I am considering Delhi as my local city.

Query 2 : *In order to join **avg_temp** data for both city and global level separately.We need to rename the columns.*

ALTER TABLE global_data RENAME COLUMN avg_temp to global_avg_temp;

ALTER TABLE city_data RENAME COLUMN avg_temp to city_avg_temp;

These query will rename the column '**avg_temp**' to distinguish the data for city and global level.

Query 1 :Resultant Output.

Output 22 results

Download CSV

city	country
Agra	India
Ahmadabad	India
Allahabad	India
Amritsar	India

Query 3: *To download the data we are using for comparison using Join Query.*

SELECT global_data.year, global_data.global_avg_temp, city_avg_temp

FROM global_data INNER JOIN city_data

ON global_data.year=city_data.year

WHERE city like 'Delhi';

Saved the results of this query in AvgTempYearlyData.csv provide in submission folder.

STEP 2: Import Libraries ,Open up & Read CSV

Importing the liberaries required for this project :

In [100]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Importing the data!

In [101]:

```
ds=pd.read_csv("AvgTempYearlyData.csv") #for importing datasets containing
yearly average Temperature data for global and local city (Delhi)
```

STEP 3: How did you calculate the moving average?

The function 'simplifiedRollingMean' is used in order to reduce noise in data.

Defining Function 'simplifiedRollingMean' Caculation for the Global data and Local data.

Moving averages are used to smooth out data and to make it easier to observe long term trends and not get lost in daily fluctuations.

In [102]:

```
def simplifiedRollingMean (windowRolling, ds_i):
    ds_o = ds_i.rolling(window = windowRolling, center =False, on =
"year").mean().dropna()
    return ds_o
```

Where ds_o and ds_i are the outputted and inputted the dataset frame respectively,windowRolling is the size of window ,the function .dropna() is used to remove the rows containing ‘NaN’as a result of running this function.

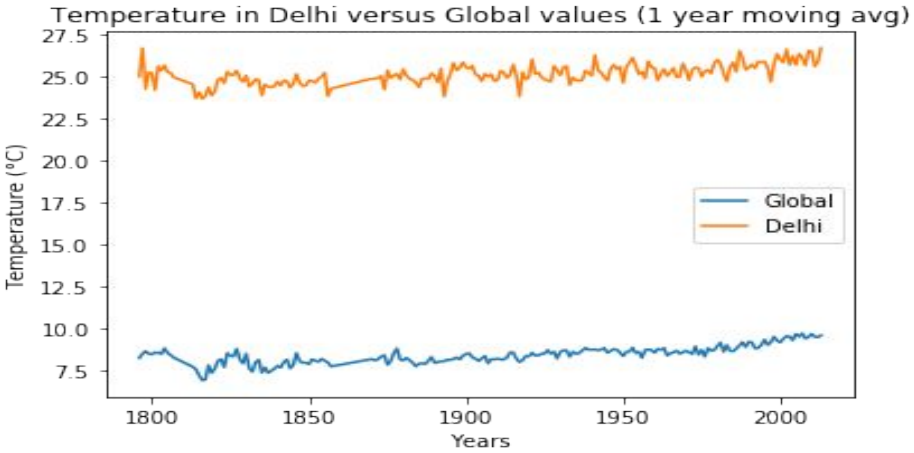
Doing calculation using 'ds_movingAvg' in order to create multiple plots by just changing rollingWindow Value.

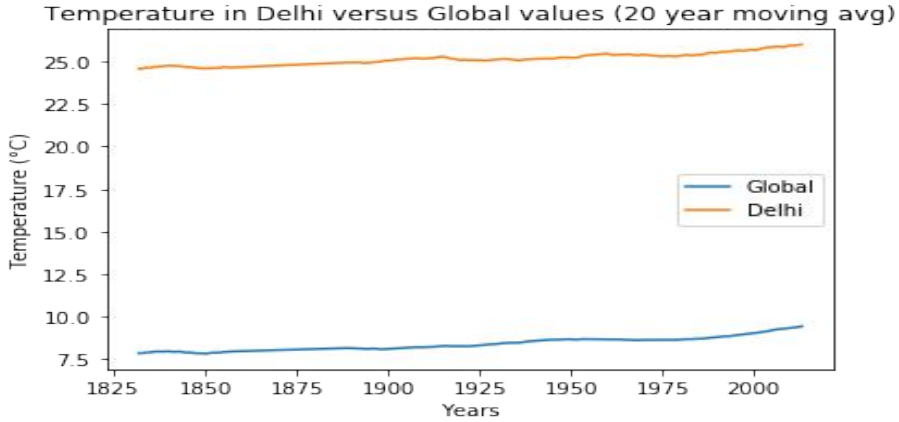
In [107]:

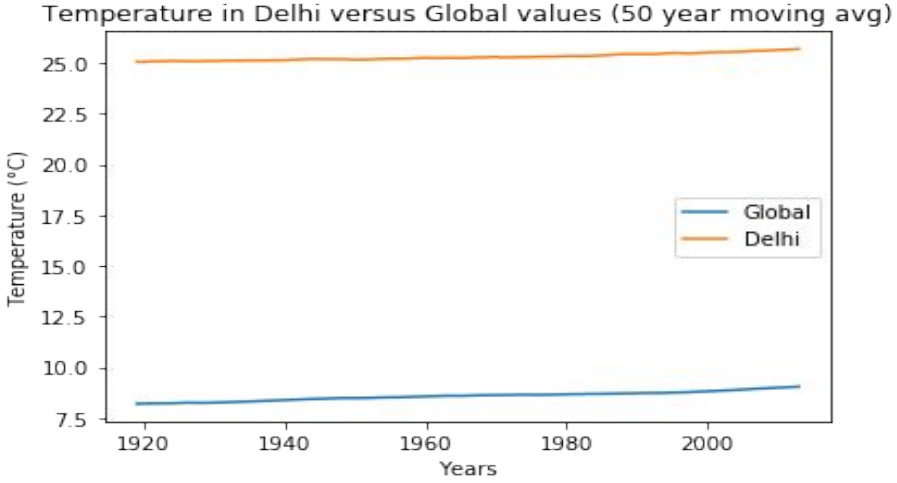
```
rollingWindow = 100
ds_movingAvg = simplifiedRollingMean(rollingWindow, ds)
```

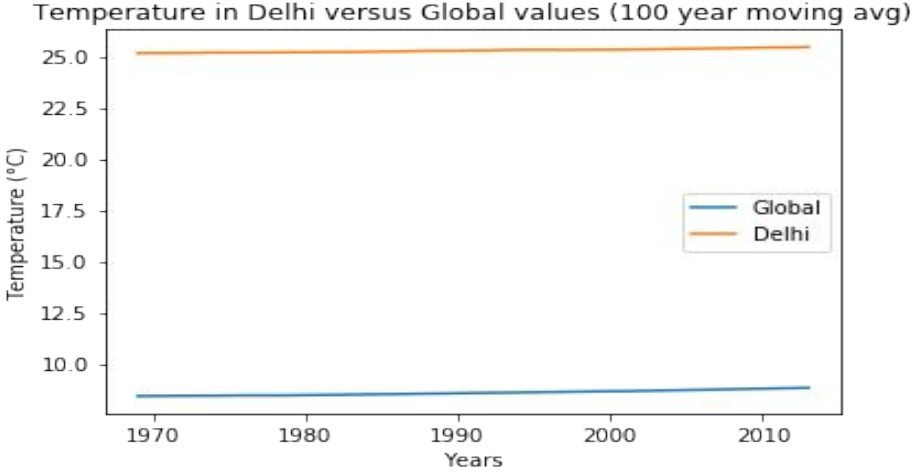
By using different values for rollingWindow we can produce charts for different moving year average. See the figures on next page.After cleaning and transforming the data it was plotted on using Matplotlib.

Graphs: Plotted Matplotlib graphs using different year for moving Average









Key Observations

Observation 1:

- The Global Temperature varies between ***7.5 to 9.61 Degree Celsius*** whereas Delhi temperature varies between ***23.8 to 25.71 Degree Celsius***.
- If comparing global and Delhi temperature trends then Delhi city is hotter than Global in Average Temperature trends.

Observation 2:

- Delhi and Global Temperature have similar kind of trends. During early years both trends seems to have ups and downs then approx . around 1993 the moving average temperature starts to increase at steady rate.

Caclulating Correlation Coefficient Between Year and Temperature(°C) for Global Temperature Trends :

In [121]:

```
np.corrcoef(ds.global_avg_temp,ds.year)
```

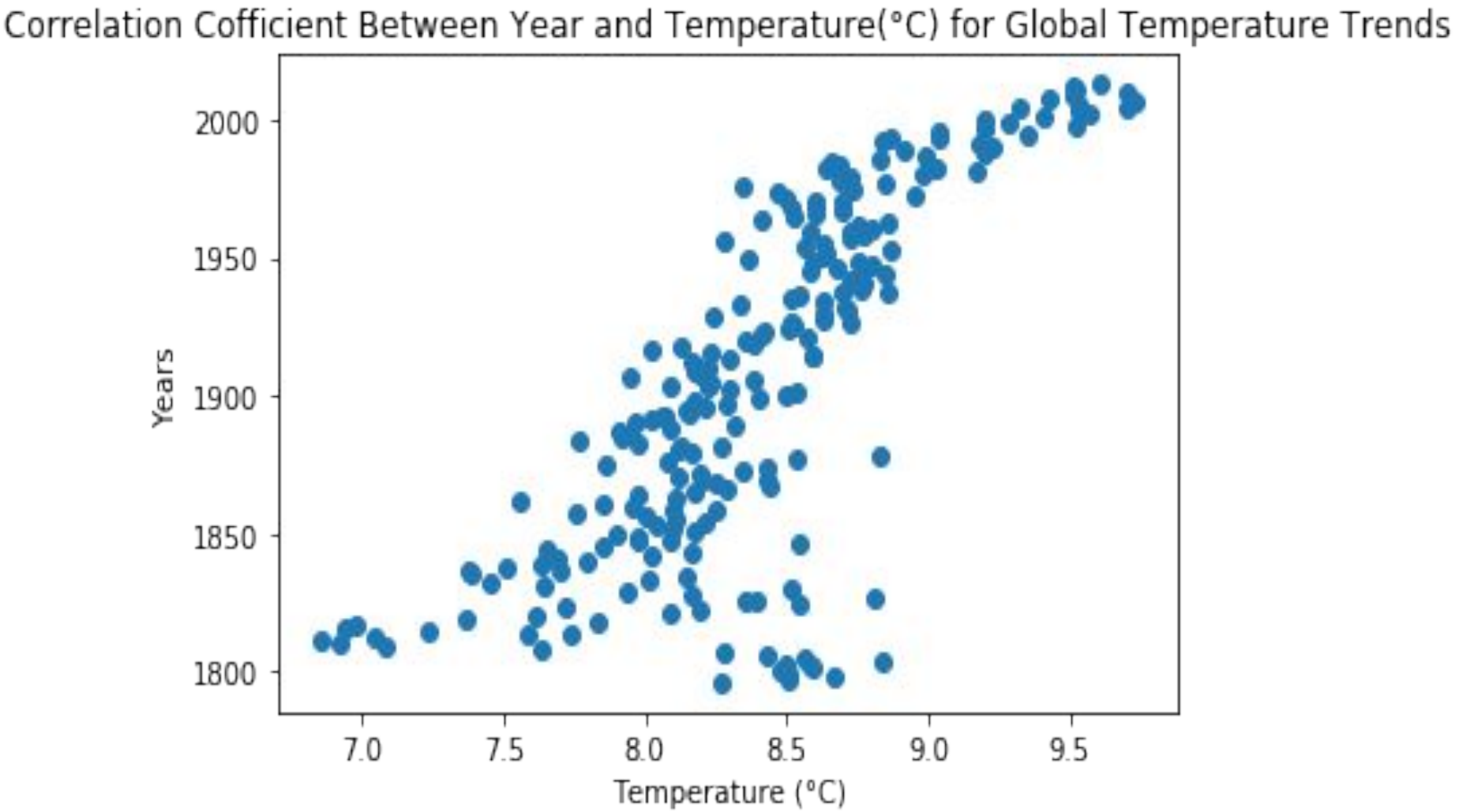
Out[121]:

```
array([[1.        , 0.76526748],
       [0.76526748, 1.        ]])
```

Draw a scatter plot between Year and Temperature(°C) for Global Temperature Trends

In [122]:

```
plt.scatter(ds.global_avg_temp,ds.year)
plt.ylabel("Years")
plt.xlabel("Temperature (°C)")
plt.title("Correlation Cofficient Between Year and Temperature(°C) for Global Temperature Trends")
plt.show()
```


Correlation Cofficient Between Year and Temperature(°C) for Global Temperature Trends
2000
1950
1900
1850
1800
Years
7.0
7.5
8.0
8.5
9.0
9.5
Temperature (°C)

Key Observations

Observation 3:

- The difference between year 1970 to 2010 is around 1 degree globally.(increasing)
- The difference between year 1970 to 2010 is more than 1 degree in delhi.(increasing)

Observation 4:

- The correlation scatter plot shows a strong positive correlation (0.76) between year and Global Average Temperature Trends. The relationship between year and global_avg_temp has a very strong positive linear relationship since the value is close to +1. So it means as the year passby , there is an increase in global Average Temperature.

Conclusion

In conclusion , there is evident that the temperature of globe has been rising at exponential rate over the past century.This could be profound effect on global ecosystem as there is dependency on temperature in order to function normally.