Exploring Weather Trends

To analyse local and global temperature data and compare the temperature trends where I live to overall global temperature trends.

Project 1 – Udacity DAND
By Irene Cheung

Extract data from SQL Query

Export the temperature data for the world and the closest big city to where I live, with filename 'yearlyavgtemp.csv'

```sql
--Find city in Hong Kong
SELECT *
FROM city_list
WHERE country like 'Hong Kong';
-- Returns one city, Shenzhen

-- Rename columns for joining
ALTER TABLE global_data RENAME COLUMN avg_temp to global_avg_temp;
ALTER TABLE city_data RENAME COLUMN avg_temp to city_avg_temp;

-- Download the joined tables
SELECT global_data.year, global_data.global_avg_temp, city_data.city_avg_temp
FROM global_data INNER JOIN city_data
ON global_data.year=city_data.year
WHERE city like 'Shenzhen';
-- Saved as yearlyavgtemp.csv
```

I lived in Hong Kong, SQL query return city as Shenzhen. In fact, Shenzhen is not a city of Hong Kong. It should be in China. Since no city in HK, so I picked Shenzhen as city in this project.

Method 1 — Calculate Moving Average by Python

DAND: Project 1 Exploring Weather Trends

In this project, I will analyze local and global temperature data and compare the temperature trends where I live to overall global temperature trends.

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

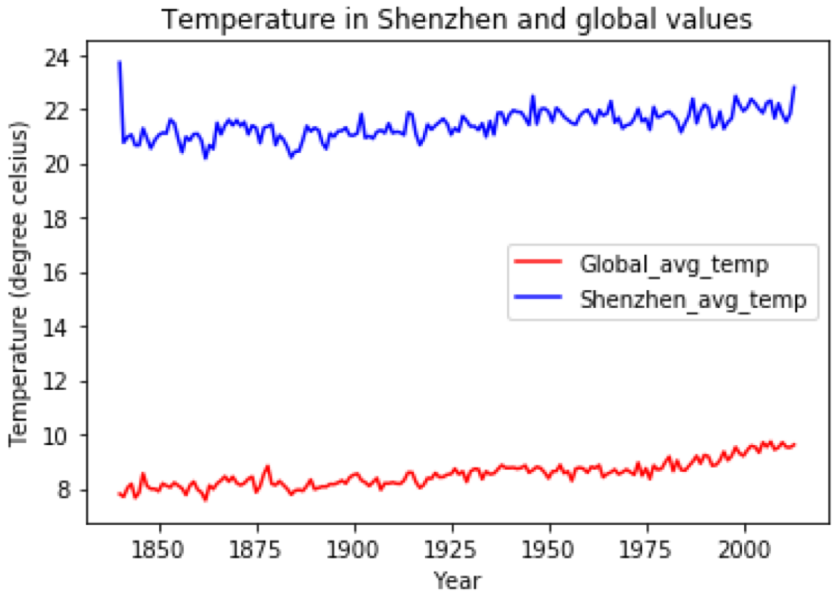
In [2]:
```
# Import dataset
df = pd.read_csv("/Users/irene/Desktop/DAFD/Project_1/yearlyavgtemp.csv")
```

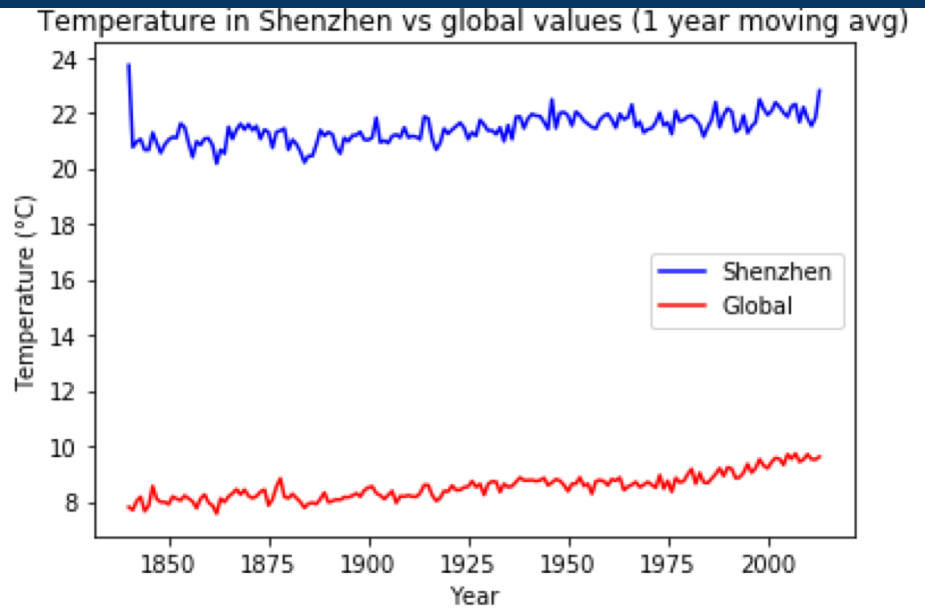
In [3]:
```
# Case1_Calculate the moving averages =1
def simplifiedRollingMean(windowRolling, df_i):
    df_o = df_i.rolling(window = windowRolling, center=False, on = "year").mean().dropna()
    return df_o

# Calculation
rollingWindow = 1
df_movingAverage = simplifiedRollingMean(rollingWindow, df)
```

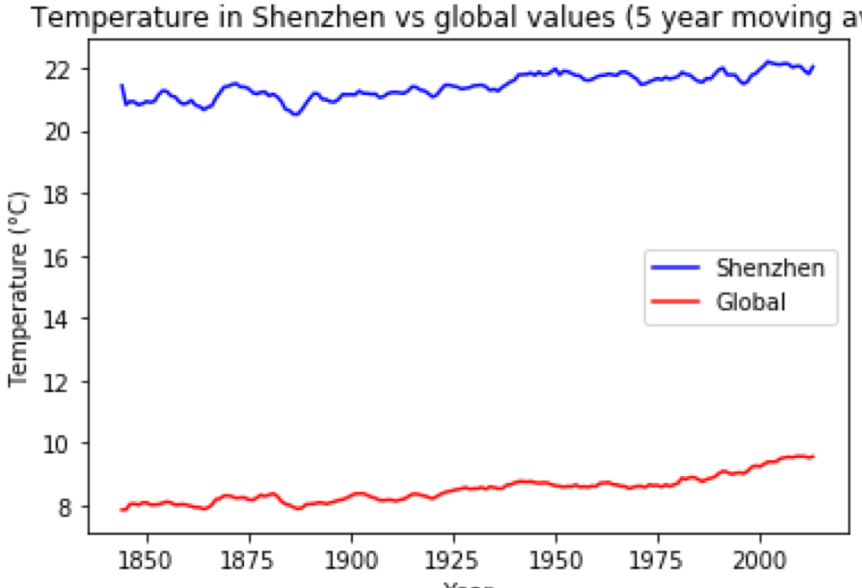
In [4]:
```
# Draw graph in matplotlib
plt.plot(df_movingAverage['year'], df_movingAverage['city_avg_temp'], label='Shenzhen',color='blue')
plt.plot(df_movingAverage['year'], df_movingAverage['global_avg_temp'], label='Global', color='red')
plt.legend()
plt.xlabel("Year")
plt.ylabel("Temperature (°C)")
plt.title("Temperature in Shenzhen vs global values ({} year moving avg)".format(rollingWindow))
plt.show()
```

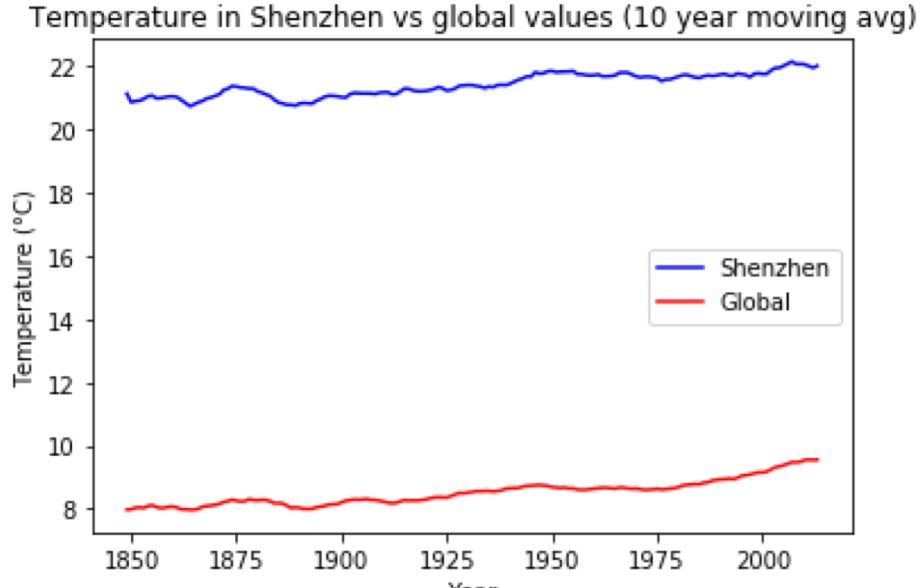
- Create a line chart in Python.
- import pandas, numpy, matplotlib.pyplot libraries
- Import dataset, namely “yearlyavgtemp.csv”
- Computes moving averages with ”Pandas.Rolling()”. To use this rolling function due to reduce the noise in the data.
- Compare the results of 1 year, 5 year and 10 years moving average with visualization.

Temperature in Shenzhen & Global in 1840-2013 with moving average by Python

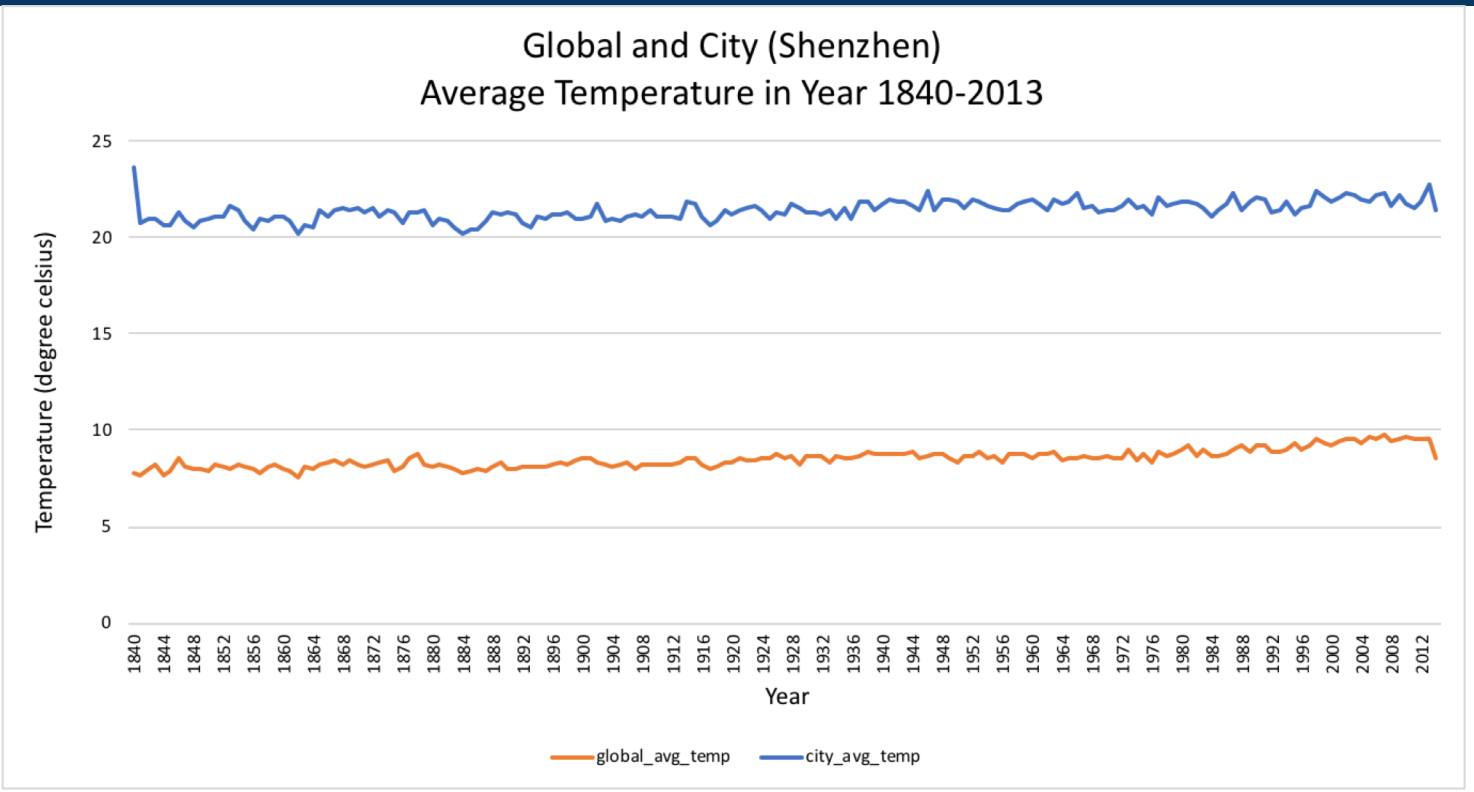








Method 2—Calculate Moving Average by Excel



	global_avg_temp	city_avg_temp
Mean	8.52	21.43
Standard Error	0.04	0.04
Median	8.52	21.43
Mode	8.18	21.47
Standard Deviation	0.47	0.51
Sample Variance	0.22	0.26
Range	2.17	3.54
Minimum	7.56	20.17
Maximum	9.73	23.71
Count	174	174

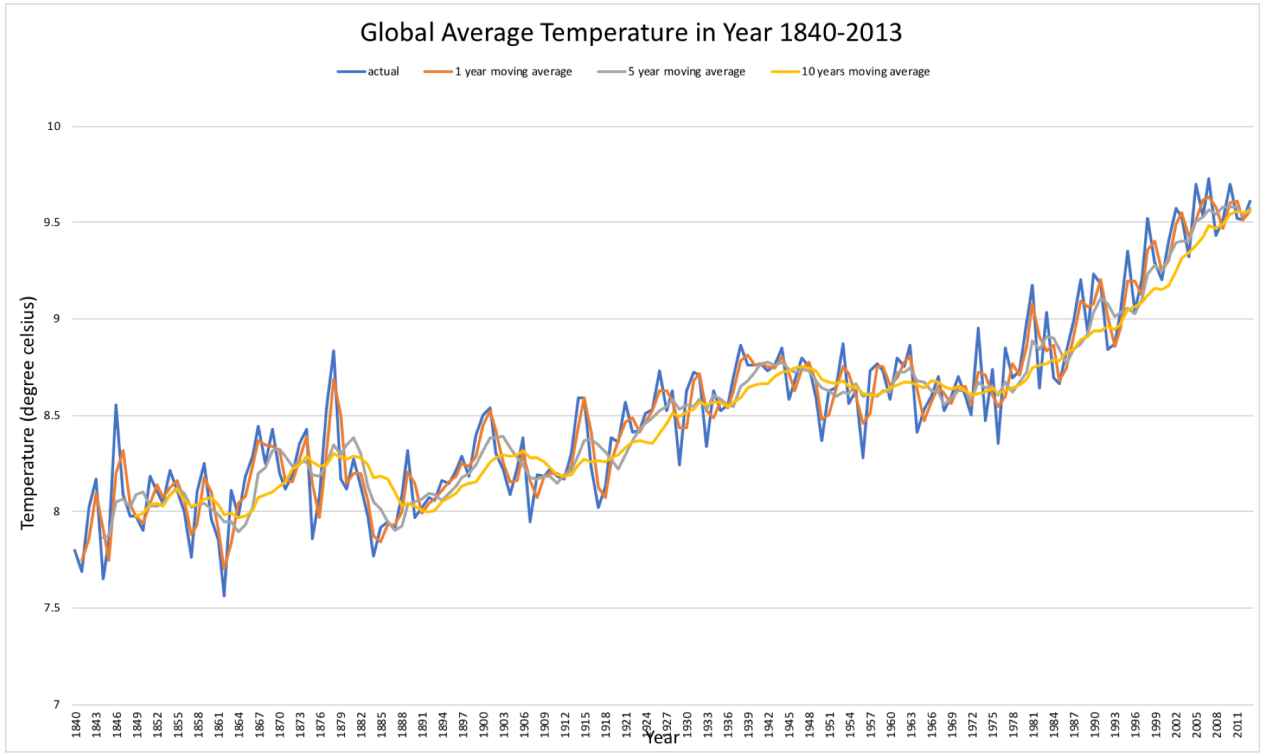
Steps:

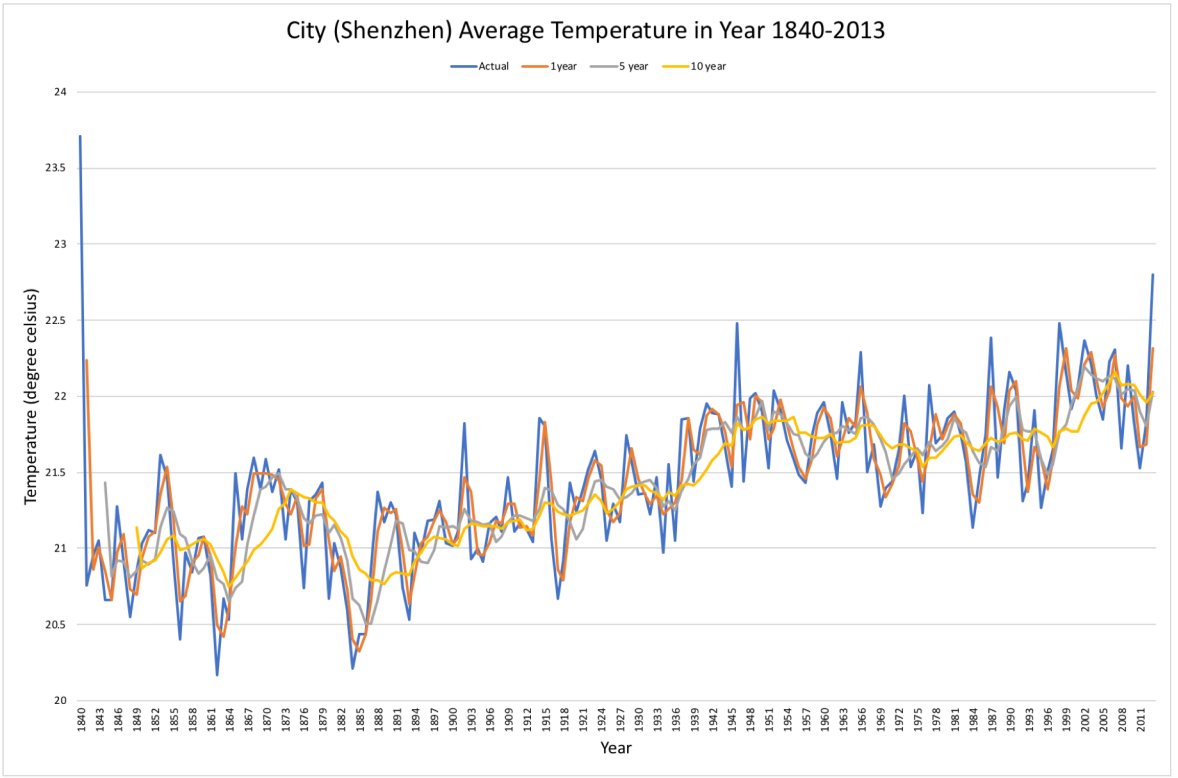
1. Go to the main interface of Excel. Click the **Data** > **Data Analysis.**
2. In the popping up Data Analysis dialog box, click to **Moving Average** in the **Analysis Tools** box, and click the **OK** button.
3. Put cursor into the **Input Range** box, and select the range to calculate the moving averages.
4. In the **Interval** box, enter the interval to calculate moving averages based on.
5. Put cursor into the **Output Range** box, and then select the range to output the moving averages.
6. Check the **Chart Output** option, and click the **OK** button.

Observations

- We found that the average temperature in both global and city (Shenzhen) are steadily increased with time during years of 1840-2013.
- The average global temperature is 8.52 ± 0.47°C, with median and mode are 8.52 °C and 8.18 °C, respectively.
- The minimum and maximum global temperature are 7.56 °C and 9.73 °C, with the range of 2.17 °C .
- In contrast, the average city (Shenzhen) temperature is 21.43 ± 0.51 °C, with median and mode are 21.43 °C and 21.47 °C, respectively.
- The minimum and maximum city (Shenzhen) temperature are 20.17 °C and 23.71 °C, with the range of 3.54 °C.
- Shenzhen's climate is sub-tropical, tending towards temperate for nearly half the year. During winter, November and December, there are pleasant breezes, plenty of sunshine and comfortable temperatures with 16-19 °C. It is not uncommon for temperatures to drop below 10 °C. During summer, May to August, are hot and humid. Afternoon temperatures often exceed 31 °C whereas at night, temperatures generally remain around 26 °C with high humidity.
- Compared with global average, Shenzhen is significant hotter as they are located in Southeast Asia. There is a clear tendency of temperature increase.

Global and City Moving Average (Shenzhen) Temperature in Year 1840-2013 by Excel





Calculate the Moving Average and Key considerations to Visualize the Trends

- A moving average is commonly used with time series data to smooth out short-term fluctuations and highlight longer-term trends or cycles.
- The blue line indicates the individual years, the orange, grey and yellow line are the 1-year, 5-years and 10-years moving average. As an example for 5-years moving average, the smoothed value for 1844 is calculated as the sum of temperatures 1840-1844, divided by the length of the smoothing interval (5 years). The meaning of the expression *data smoothing* should be apparent from this illustration. Smoothing reduces the variance of the original data; the longer the smoothing interval, the greater the reduction.
- For the time interval 1940-2013 the average temperatures in global and city are increased regularly. The most serious consequence of smoothing or filtering, like 10-years moving average, is the shift of peaks and troughs in the smoothed curve, relative to the original data. This shows how more and more details are lost as the length of the smoothing interval is increased.
- In this case, 5-years moving average is the best for illustration, as it keeps certain details without too much lost.