Project1- Explore Weather Trend

https://github.com/porschebest/udacity-data-analyst-nanodegree/tree/master/T1-1-Explore-Weather-Trends

Intro

In this project, I **analyzed the local and global temperature data using SQL and Excel**. The data is based on the Database provided by Udacity. Then I compared the temperature trends in Taipei and nearby city to overall global temperature trends.

Part1- Extract Data with SQL

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 (ºC).
- global_data - This contains the average global temperatures by year (ºC).

Step1- Write a Query to check available cities

First of all, I'll checked the **city_list** in order to know what city is available in the database nearby where my current city: Taipei.

As a results, I run the following Query using the **WHERE** clause:

```
select *
from city_list
WHERE country = 'Taiwan'
```

There are 3 results: (city_list_taiwan.csv)

	city	country
1	Kaohsiung	Taiwan
2	Taichung	Taiwan
3	Taipei	Taiwan

Step2- Write a Query to extract Temperature data in Taipei, Kaohsing, Taichung

After I know that Taipei's historical data is in the database, I write a Query to extract Taipei's data from the database.

For Taipei: (city_data_taipei.csv)

```
SELECT year, avg_temp
FROM city_data
WHERE city = 'Taipei';
```

Same for Kaohsiung: (city_data_ Kaohsiung.csv)

```
SELECT year, avg_temp
FROM city_data
WHERE city = 'Kaohsiung';
```

Same for Taichung: (city_data_taichung.csv)

```
SELECT year, avg_temp
FROM city_data
WHERE city = 'Taichung';
```

Step3- Write a Query to extract global temperature data from the database

After I got the data for all the nearby cities, I extracted the global from the following query:

```
SELECT year, avg_temp
FROM global_data;
```

After I extracted the global data, I found that there is a mismatch between global data and the city data, the city data only contains data from 1841-2013. However, global data contains data from 1750-2015. As results I run the following Query: (global_data_1841-2013.csv)

```
SELECT year, avg_temp
FROM global_data
WHERE year > 1840 AND year < 2014;
```

After I have both global and local temperature data, I started to calculate “moving average” as a indicator of the Global Weather Trend.

Part2- Analyze Data with Python

Step1- Extract data from csv

My first step is to extract the data from the csv, then use a “for loop” and “split” function to extract the data

```
def extract_csv(filepath='',year=[],temp=[]):
    """
    This function extracct the csv from the input filepath
    """
    #   open global weather data
    f = open(filepath)
    #   Extract data from csv to arrays
    for line in f.readlines():
        if(line.split(',')[0] != 'year'):
            temp1 = line.split(',')[1]
            temp1 = temp1.replace('\n','')
            year1 = line.split(',')[0]
            year1 = year1.replace("\'","")
            temp.append(temp1)
            year.append(year1)
    print("extract csv success")
```

Step2- Calculate moving average

The second step is to calculate moving average. The way I do it is to use a “while loop” to run through all the data and calculate the moving average in a 10 year basis.

```
def moving_average(year=[],temp=[],mov_avg_data=[],mov_avg_temp=[]):
    """
    This function calculate the moving average in a 10 year basis
    """
    i = 0
    avg_temp = 0
    last_num = len(year) - 9
    while i < last_num:
        avg_temp = (float(temp[i]) + float(temp[i+1]) + float(temp[i+2]) +
float(temp[i+3])
        + float(temp[i+4]) + float(temp[i+5]) + float(temp[i+6]) +
float(temp[i+7]) + float(temp[i+8]) + float(temp[i+9]))/9
        avg_temp = round(avg_temp, 2)
```

```
        mov_avg_temp.append(avg_temp)
        mov_avg_data.append('\"'+year[i+9]+'\"'+','+'\"'+str(avg_temp)+'\"'+','+'\n')
        i += 1
    print("calculate moving average success")
```

Step3- Save the results in a new csv

After calculating the moving average, I saved all the results in a new csv.

```
def save_csv(filepath, mov_avg_data=[]):
    """
    This function save the moving average in a new csv
    """
    f = open(filepath, 'w')
    f.writelines(mov_avg_data)
    f.close()
    print("save csv success")
```

Step4- Transform the data from string back to int or float

This is a step before starting to plot everything. This function transform the year list back to int and temp list back to float in order to better plot and show the results.

```
#   transform data in the lists from string to int and float
def string_to_int_and_float(year=[], mov_avg_temp=[],year_out=[],temp_out=[]):
    """
    This function transform the year list back to int and temp list back to
float
    """
    for item in year:
        if int(item) >= 1850:
            year_out.append(int(item))
    for item in mov_avg_temp:
        temp_out.append(float(item))
    print('string to int and float success')
    print('--end process data--')
```

Step5- plot the result

I used the matplotlib library to plot the results using Python.

In order to plot all 4 lines, I use the subplot function.

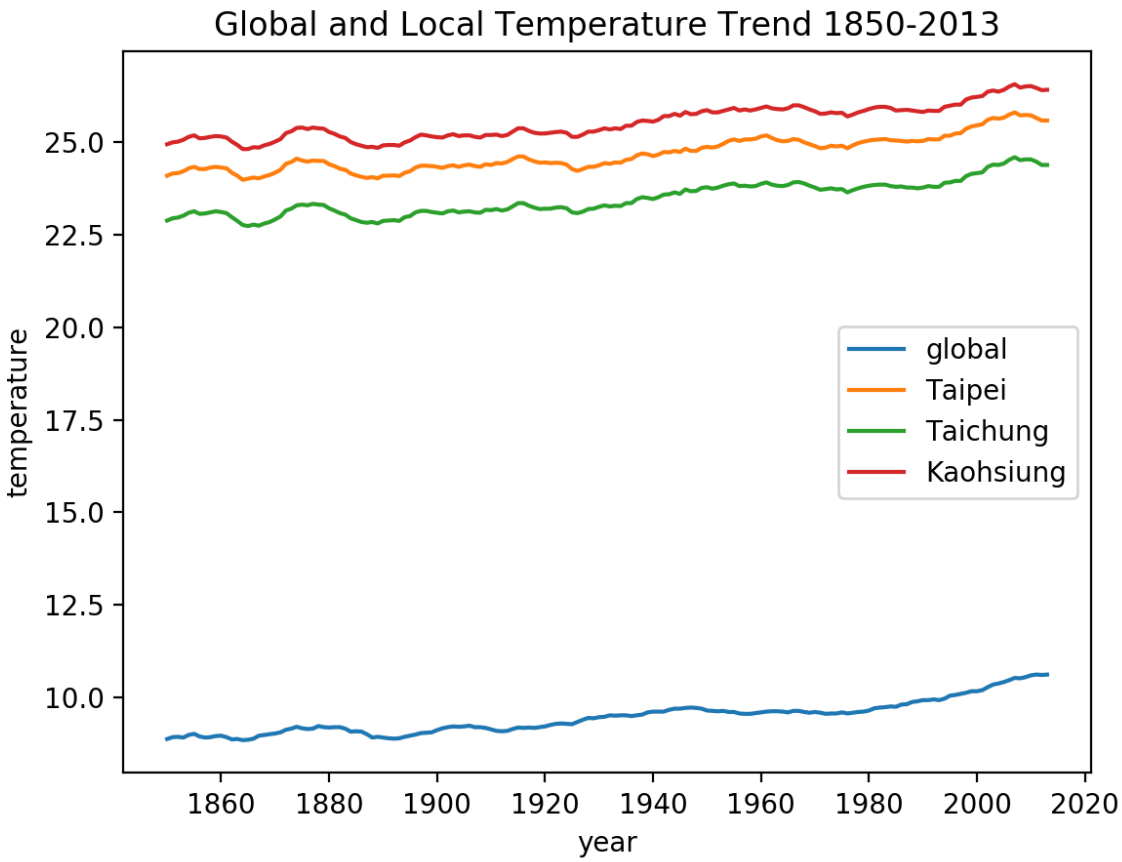
```
plt.ylabel('temperature')
plt.xlabel('year')
plt.title('Global and Local Temperature Trend 1850-2013')
plt.legend() # show the label
plt.show()
fig.savefig('Global-and-Local-Temperature-Trend-1850-2013.png')fig, ax = plt.subplots()
line_global = ax.plot(year_int_global, temp_float_global, label='global')
line_tpe = ax.plot(year_int_tpe, temp_float_tpe, label='Taipei')
line_txg = ax.plot(year_int_txg, temp_float_txg, label='Taichung')
line_khh = ax.plot(year_int_khh, temp_float_khh, label='Kaohsiung')

plt.ylabel('temperature')
plt.xlabel('year')
plt.title('Global and Local Temperature Trend 1850-2013')
plt.legend() # show the label
plt.show()
fig.savefig('Global-and-Local-Temperature-Trend-1850-2013.png')
```

The complete script can be found in:

https://github.com/porschebest/udacity-data-analyst-nanodegree/blob/master/T1-1-Explore-Weather-Trends/weather.py

Part3- Interpret the results according to the chart



This line chart shows the moving average of global and local (Taipei, Taichung, Kaohsiung) temperature in the past 163 years(1850-2013).

The following are some observations based on the chart:

Similarities

- Both global and local temperature has increased in the past 163 years.
- Local temperature has the same trend in the past 163 years.

Differences

- Local trend has more obvious peak and valley compared to global trend.
- Between 1860~1870, global data had slightly increased while local data had dropped with a valley.
- Around 2005, local temperature dropped while global temperature increased. There might be somewhere else in the world has more drastically change or local temperature

behave weird compare to the globe.

Other

- Though Taipei's latitude is larger than Taichung, Taichung has cooler temperature compared to Taipei. This may be the cause of “Urban heat island effect” that results in higher temperature in Taipei.