Project 1 – Explore Weather Trends

Ashwini Hegde

Contents

1. Data Extraction

I used SQL to extract the data from the database schema. PFB the codes for the same

- Query 1: To get distinct city values in the data.
    - **select distinct city from city_data;**

- Query 2: Extracted data for New Delhi and exported the results in CSV format.
    - **select year, avg_temp as avg_temp_delhi from city_data where city = 'New Delhi';**

- Query 3: Extracted data for Global and exported the results in CSV format.
    - **select * from global_data;**

Finally, combined dataset for global and New Delhi in python

2. Data Cleaning

- EDA – New Delhi

```
In [2]: delhi = pd.read_csv('datafile/delhi.csv')
        print(delhi.year.min(), delhi.year.max())
        print(delhi.avg_temp_delhi.min(), delhi.avg_temp_delhi.max())
        print(delhi.info())
        delhi.head()
```

```
1796 2013
23.7 26.71
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 218 entries, 0 to 217
Data columns (total 2 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   year            218 non-null    int64
 1   avg_temp_delhi  201 non-null    float64
dtypes: float64(1), int64(1)
memory usage: 3.5 KB
None
```

Out[2]:

	year	avg_temp_delhi
0	1796	25.03
1	1797	26.71
2	1798	24.29
3	1799	25.28
4	1800	25.21

As observed, data has total 218 years but avg_temp_delhi is available only for 201 years. So, 17 datapoints is missing. We have data from 1796 to 2013. Minimum temperature is 23.7 and maximum temperature is 26.71.

- EDA – Global Temperature

```
In [3]: global_temp = pd.read_csv('datafile/global.csv')
        print(global_temp.year.min(), global_temp.year.max())
        print(global_temp.avg_temp_global.min(), global_temp.avg_temp_global.max())
        print(global_temp.info())
        global_temp.head()
```

```
1750 2015
5.78 9.83
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 266 entries, 0 to 265
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   year             266 non-null    int64
 1   avg_temp_global  266 non-null    float64
dtypes: float64(1), int64(1)
memory usage: 4.3 KB
None
```

Out[3]:

	year	avg_temp_global
0	1750	8.72
1	1751	7.98
2	1752	5.78
3	1753	8.39
4	1754	8.47

As observed, global data has total 266 years. There are no missing values. We have data from 1750 to 2015. Minimum temperature is 5.78 and maximum temperature is 9.83.

- Combining datasets

In [4]:
```
df = pd.merge(global_temp, delhi, on = 'year', how = 'outer')
print(df.info())
df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 266 entries, 0 to 265
Data columns (total 3 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   year             266 non-null    int64  
 1   avg_temp_global  266 non-null    float64
 2   avg_temp_delhi   201 non-null    float64
dtypes: float64(2), int64(1)
memory usage: 8.3 KB
None
```

Out[4]:

	year	avg_temp_global	avg_temp_delhi
0	1750	8.72	NaN
1	1751	7.98	NaN
2	1752	5.78	NaN
3	1753	8.39	NaN
4	1754	8.47	NaN

- Filtering cases where we have data available for both city and global.

In [6]:
```
df.dropna(inplace = True)
print(df.info())
df.head()
```

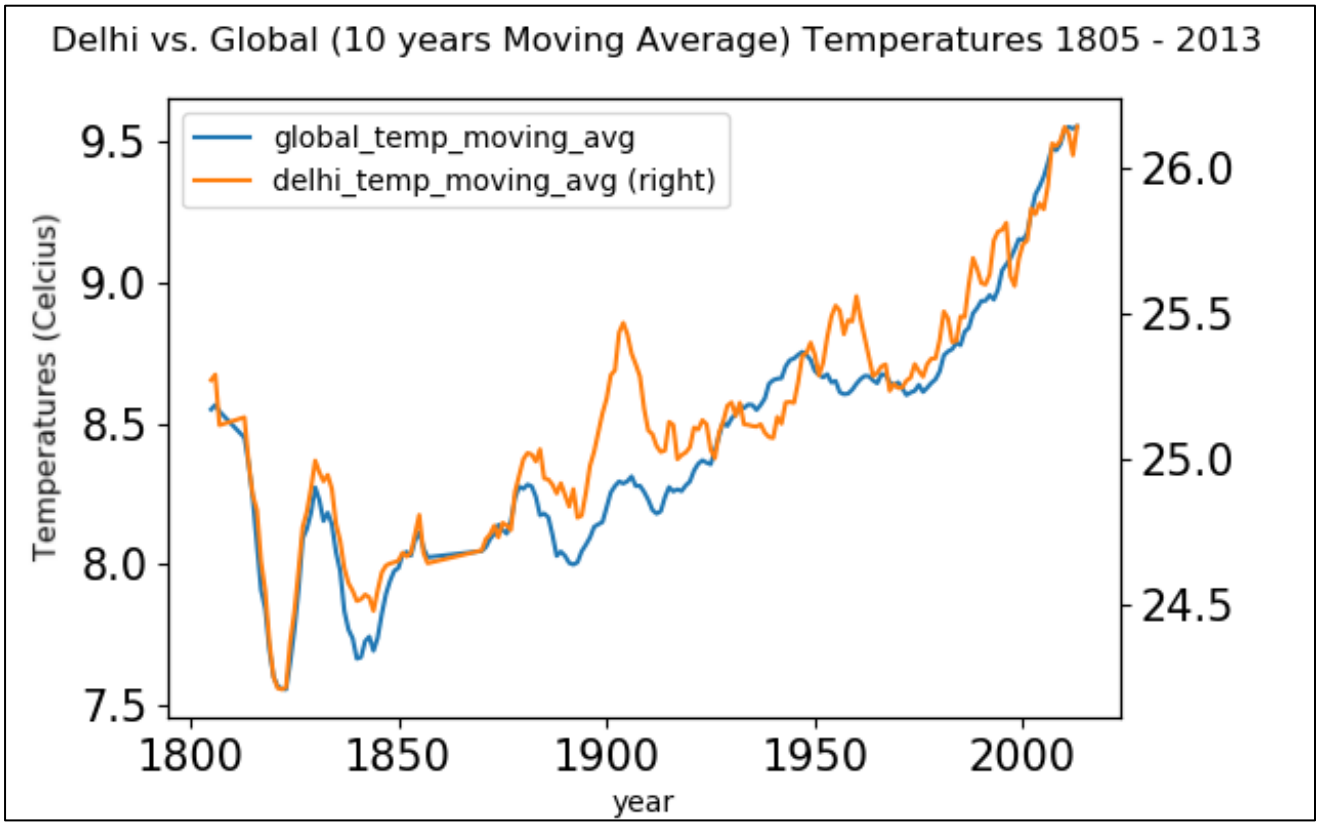
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 201 entries, 46 to 263
Data columns (total 3 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   year             201 non-null    int64  
 1   avg_temp_global  201 non-null    float64
 2   avg_temp_delhi   201 non-null    float64
dtypes: float64(2), int64(1)
memory usage: 6.3 KB
None
```

Out[6]:

	year	avg_temp_global	avg_temp_delhi
46	1796	8.27	25.03
47	1797	8.51	26.71
48	1798	8.67	24.29
49	1799	8.51	25.28
50	1800	8.48	25.21

3. Data Visualisation

- Moving Average Plot



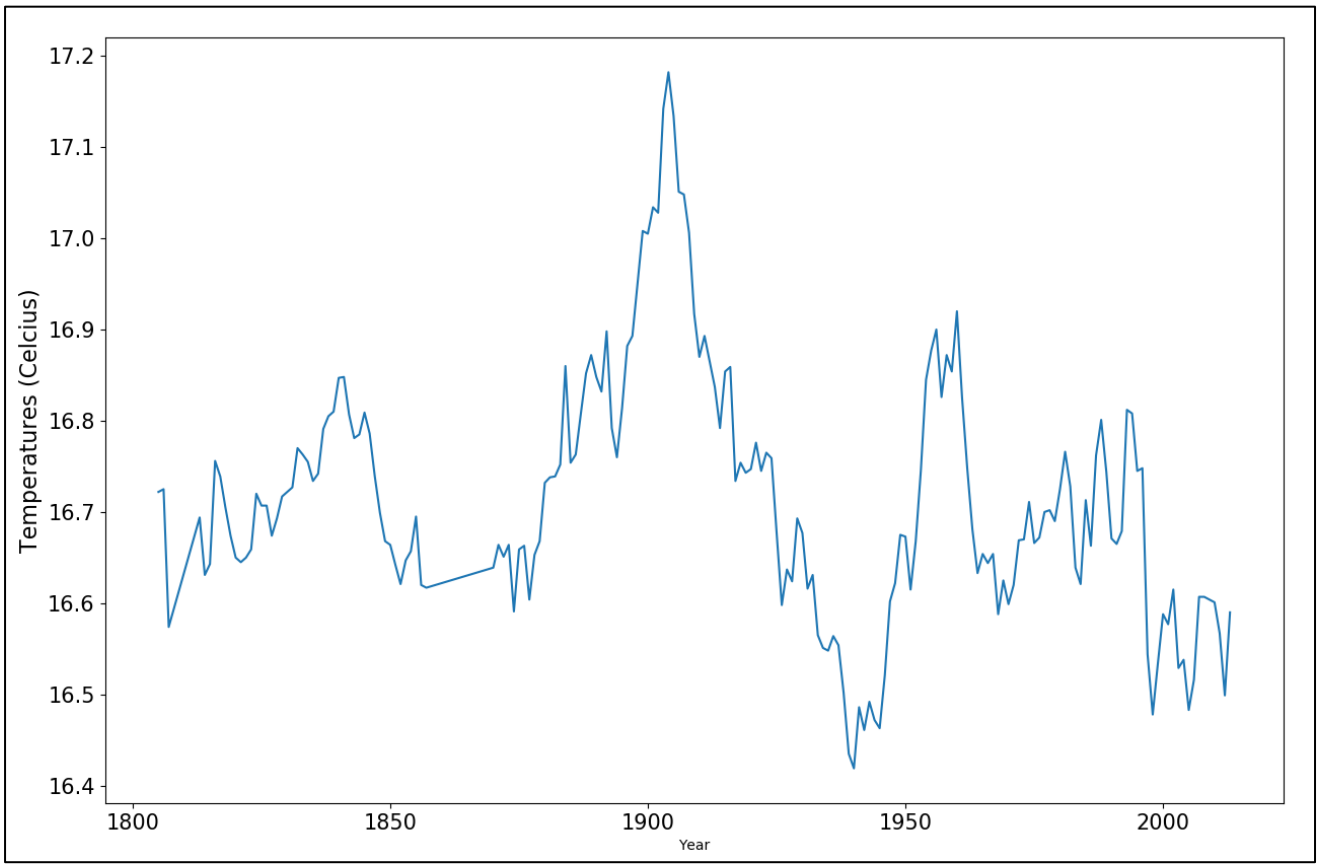
Line graph with the moving average temperature values on Y-axis and the year on X-axis for Global and New Delhi.

Calculating 10 year moving average

In [6]:
```
N = 10
df['global_temp_moving_avg'] = df.iloc[:,1].rolling(window=N).mean()
df['delhi_temp_moving_avg'] = df.iloc[:,2].rolling(window=N).mean()
df['difference_in_temperature'] = df['avg_temp_delhi'] - df['avg_temp_global']
df['difference_in_moving_average'] = df['delhi_temp_moving_avg'] - df['global_temp_moving_avg']
```

In [8]:
```
ax = df.plot(x = 'year', y = 'global_temp_moving_avg', label='global_temp_moving_avg')
ax2 = df.plot(x = 'year', y = 'delhi_temp_moving_avg', secondary_y=True, label='delhi_temp_moving_avg' , ax=ax)
plt.savefig('moving_average.png', dpi=1600)
plt.tight_layout()
plt.show()
```

- Difference between Moving Average Plot




```
In [9]: plt.figure(figsize=[15,10])
        plt.plot(df['year'], df['difference_in_moving_average'],label='difference_in_moving_average')
        plt.rc('xtick', labelsize=15)
        plt.rc('ytick', labelsize=15)
        plt.savefig('difference_in_moving_average.png', dpi=400)
        x = np.std(df['difference_in_moving_average'])
        print(x)
        plt.legend(loc=4)

        0.13514949033754778
```

4. Observations

- Global temperatures are very less compared to Delhi.
- There has been a steady rise in temperature for both Global temperature and New Delhi.
- Global and New Delhi recorded their lowest temperature around 1820.
- Moving average temperature difference between Global temperatures and New Delhi temperatures is highest around 1900.