

UDACITY Data Analysis Nanodegree

Project 1 - Exploring Weather Trends

MAY 31, 2020

Neil Nahar

New Delhi, 201301

Overview

In the project, I have examined local temperature of New Delhi, India in accordance with the global temperature data and compared them. I had been provided with a database on Udacity portal from where I have to extract, manipulate and visualize the data.

Goals

- 1) Extraction of data from the database and exporting to CSV file
- 2) Making a chart visualization.
- 3) Observation based on chart

Tools Utilized

- 1) SQL - To obtain the data from the database
- 2) Python - For calculating moving average and plotting the line chart
- 3) Anaconda - Jupyter Notebook - For writing python code and making observations

Step 1 - Extraction of Data from provided Database

```
SELECT *  
FROM city_list  
WHERE country = 'India'
```

As the nearest city to my location is New Delhi, therefore I selected and obtained the data for the same along with the Global data using SQL statements as follows:

```
SELECT c.year "Year", c.avg_temp "New Delhi", g.avg_temp "Global"  
FROM global_data g  
JOIN city_data c ON c.year = g.year  
WHERE c.city = 'New Delhi'
```

The data was obtained in a CSV file which was named results.csv and the same was used for further analysis.

Step 2 - Python Code for Making Line Chart

I have selected Python for data processing and data visualization for completing the task and I have used pandas library for data processing and matplotlib for data visualization. All operations were performed in Jupyter Notebooks to create the document

1) Importing the libraries

```
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

2) Importing the extracted Data Set

```
demo = pd.read_csv("results.csv")
```

3) Getting insights from the data

```
demo.head()
```

	Year	New Delhi	Global
0	1796	25.03	8.27
1	1797	26.71	8.51
2	1798	24.29	8.67
3	1799	25.28	8.51
4	1800	25.21	8.48

```
demo.columns.unique()
```

Removing Whitespace from Column names –

```
demo.columns = ['Year','NewDelhi','Global']
demo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 218 entries, 0 to 217
Data columns (total 3 columns):
#   Column      Non-Null Count  Dtype
---  ---
0    Year        218 non-null    int64
1   NewDelhi    201 non-null    float64
2    Global      218 non-null    float64
dtypes: float64(2), int64(1)
memory usage: 5.2 KB
```

```
demo.describe()
```

	Year	NewDelhi	Global
count	218.000000	201.000000	218.000000
mean	1904.500000	25.166269	8.403532
std	63.075352	0.594003	0.548662
min	1796.000000	23.700000	6.860000
25%	1850.250000	24.800000	8.092500
50%	1904.500000	25.140000	8.415000
75%	1958.750000	25.550000	8.727500
max	2013.000000	26.710000	9.730000

Removing NA values using dropna function –

```
demo.dropna(axis=0, how='any', thresh=None, subset=None, inplace=True)
```

4) Calculating the Moving Average using Rolling and Mean function

Using Window=10 sets a 10-year window for the moving average. This value was used because it will smoothed the result chart optimally for further analysis among a set of other values.

```
demo['NewDelhi'] = demo.rolling(window=10)['NewDelhi'].mean()
```

```
demo['Global'] = demo.rolling(window=10)['Global'].mean()
```

5) Arranging the Parameters in the Graph

```
params = { 'legend.fontsize': 24, 'figure.figsize': (18, 10), 'axes.labelsize': 20, 'axes.titlesize': 20, 'xtick.labelsize': 'x-large', 'ytick.labelsize': 'x-large' }
```

```
plt.rcParams.update(params)
```

```
plt.title("\nLine Chart of New Delhi vs. Global Temperature")
```

```
plt.ylabel("Temperature")
```

```
plt.xlabel("Year")
```

```
plt.xticks(np.arange(1800, 2013, step=10))
```

```
plt.plot(demo.Year, demo.NewDelhi, linewidth=4, color='Red', ls = 'dashdot', label='New Delhi')
```

```
plt.plot(demo.Year, demo.Global, linewidth=4, color='green', ls = 'dashdot', label='Global')
```

```
plt.grid(True)
```

```
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))
```

```
plt.show()
```

6) Plotting the Line Chart

```
plt.plot(demo.Year, demo.NewDelhi, linewidth=4, color='Red', ls = '—', label='New Delhi')
```

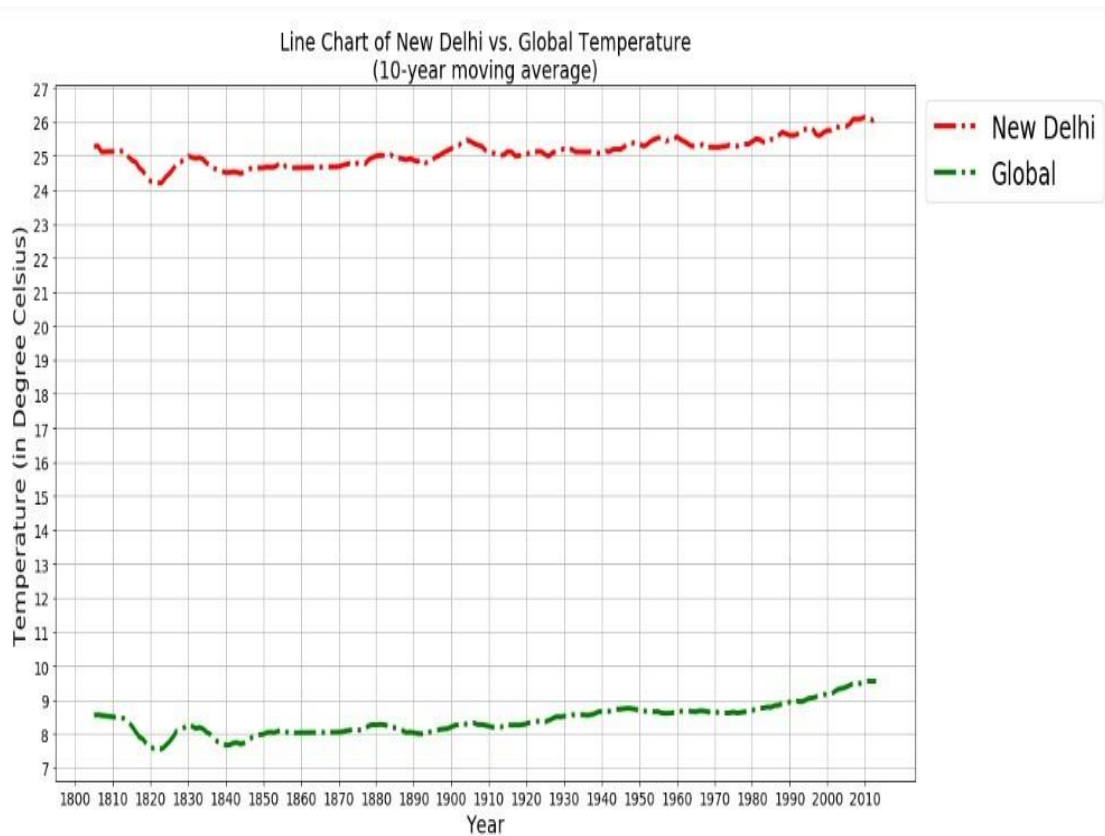
```
plt.plot(demo.Year, demo.Global, linewidth=4, label='Global')
```

```
plt.grid(True)
```

Legend:

```
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))
```

```
plt.show()
```



Observations (Result) –

- 1) When Comparing to Global average, my city's temperature is significantly higher i.e. on an average, it is almost 17 degrees hotter than global average.
- 2) However, after examining the graph, it is evident that the temperature of my city as well as the Global temperature have increased steadily throughout the years.
- 3) The world is getting hotter every passing year.
- 4) The temperature of my city has increased by 3 degrees while the global temperature has also increased by an equivalent amount. This is evident from the Maximum and Minimum temperature recordings.
- 5) The ratio of the temperatures of my city and global average is also nearly equal to 3 degrees i.e. my city's average temperature is almost 3 times the global average.
- 6) The Correlation coefficient between my city's and Global temperature is 0.76. Therefore, we can conclude it is positive but weak relationship and thus the temperature of my city cannot be estimated based on Global temperature.

References

<https://www.livescience.com/37003-global-warming.html>

https://web.stanford.edu/~moore/Boon_To_Man.html