

EXPLORING GLOBAL AND LOCAL TEMPERATURE DATA:

Tool used: SQL, Python, Jupyter Notebook

Steps taken to extract the data:

Checking database to search for the cities in my country, India:

Input

HISTORY ▾

MENU ▾

SCHEMA ↻

city_data ▾

city_list ▾

global_data ▾

1 SELECT *

2 FROM city_list

3 WHERE country='India'

Success!

EVALUATE

Output 22 results

Download CSV

Nagpur	India
New Delhi	India
Patna	India
Pune	India
Rajkot	India
Ranchi	India
Surat	India
Vadodara	India
Varanasi	India

^ Menu Expand

Extracting data from 'Pune' city and downloading results into CSV

Input

HISTORY ▾

MENU ▾

SCHEMA

city_data

year

city

country

avg_temp

1 SELECT *

2 FROM city_data

3 WHERE city='Pune'

Success!

EVALUATE

Output

218 results

[Download CSV](#)

year	city	country	avg_temp
1796	Pune	India	24.39
1797	Pune	India	25.17
1798	Pune	India	24.05
1799	Pune	India	24.68
1800	Pune	India	24.67
1801	Pune	India	23.94
1802	Pune	India	25.18
1803	Pune	India	24.95

^ Menu

□ Expand

Extracting global data and saving into CSV:

Input

HISTORY ▾

MENU ▾

SCHEMA

↺

avg_temp

city_list ▾

global_data ▴

year

avg_temp

1

SELECT *

2

FROM global_data

Success!

EVALUATE

Output

266 results

[Download CSV](#)

year	avg_temp
1750	8.72
1751	7.98
1752	5.78
1753	8.39
1754	8.47
1755	8.36
1756	8.85
1757	9.02

^ Menu

Expand

Steps taken to analyse the data:

Reading the data:

```
In [43]: import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

Reading the data

```
In [9]: pune_df = pd.read_csv('pune_data.csv')
pune_df.head()
```

	year	city	country	avg_temp
0	1796	Pune	India	24.39
1	1797	Pune	India	25.17
2	1798	Pune	India	24.05
3	1799	Pune	India	24.68
4	1800	Pune	India	24.67

```
In [11]: global_df = pd.read_csv('global_data.csv')
global_df.head()
```

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

Dealing with NULL values:

```
In [29]: #dealing with NULL values
pune_df.info()
#we see that there are 7 NULL values in the avg_temp column
pune_df.mean()
pune_df.fillna(pune_df.mean(), inplace=True)

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 218 entries, 0 to 217
Data columns (total 4 columns):
year      218 non-null int64
city      218 non-null object
country   218 non-null object
avg_temp  211 non-null float64
dtypes: float64(1), int64(1), object(2)
memory usage: 6.9+ KB

In [33]: pune_df.info()
#we have succesfully filled the null values with mean of the data

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 218 entries, 0 to 217
Data columns (total 4 columns):
year      218 non-null int64
city      218 non-null object
country   218 non-null object
avg_temp  218 non-null float64
dtypes: float64(1), int64(1), object(2)
memory usage: 6.9+ KB
```

Plotting normal Global and Local Data:

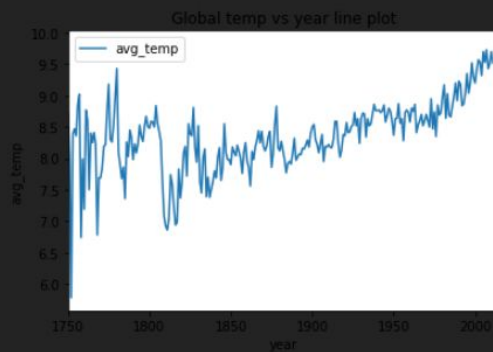
Plotting normal global data

```
In [32]: global_df.info()  
#hence no null values
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 266 entries, 0 to 265  
Data columns (total 2 columns):  
year      266 non-null int64  
avg_temp  266 non-null float64  
dtypes: float64(1), int64(1)  
memory usage: 4.3 KB
```

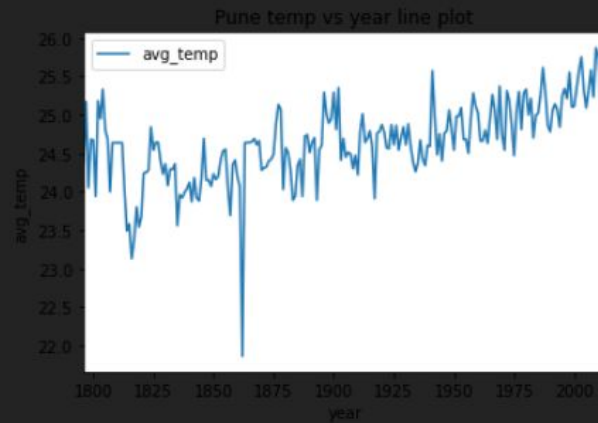
```
In [37]: global_line_plot = global_df.plot(x="year", y="avg_temp", kind="line")  
global_line_plot.set_xlabel("year")  
global_line_plot.set_ylabel("avg_temp")  
global_line_plot.set_title("Global temp vs year line plot")
```

```
Text(0.5, 1.0, 'Global temp vs year line plot')
```

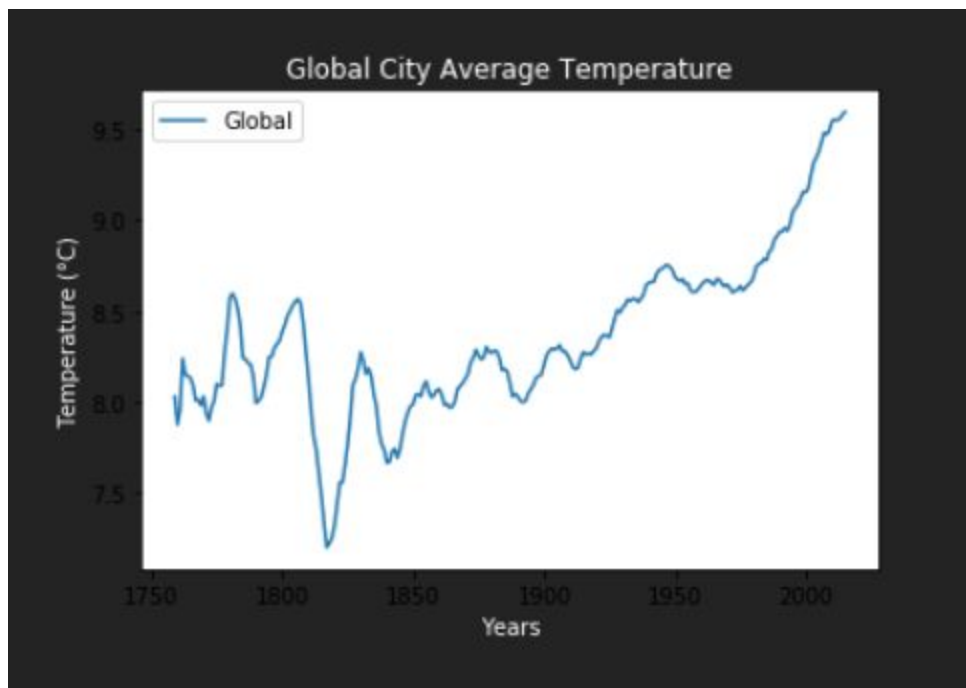


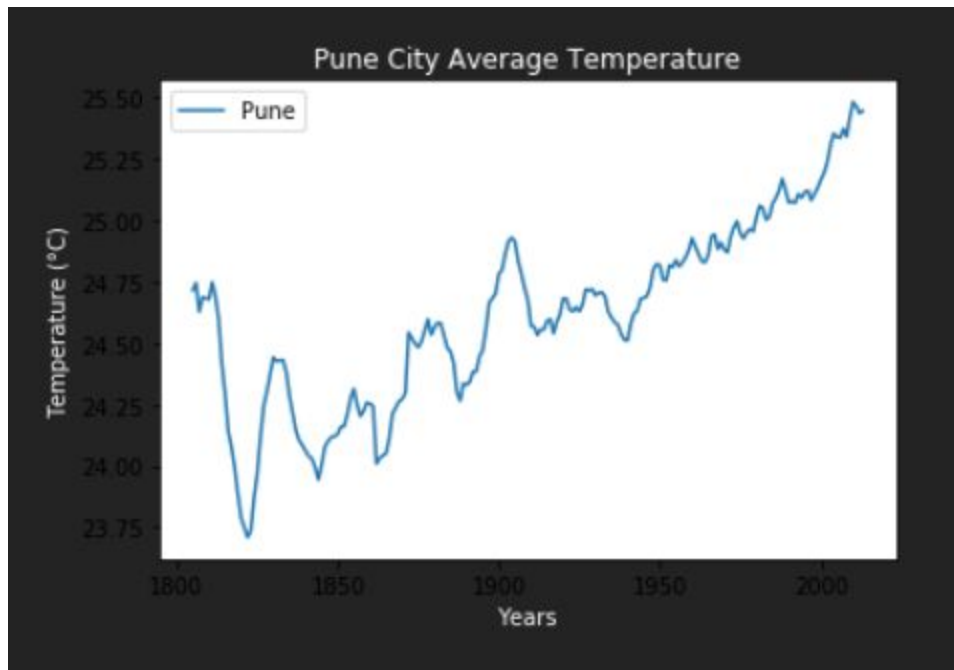
```
In [41]: pune_line_plot = pune_df.plot(x="year", y="avg_temp", kind="line")
pune_line_plot.set_xlabel("year")
pune_line_plot.set_ylabel("avg_temp")
pune_line_plot.set_title("Pune temp vs year line plot")
```

```
Text(0.5, 1.0, 'Pune temp vs year line plot')
```



Calculating and plotting moving average data:





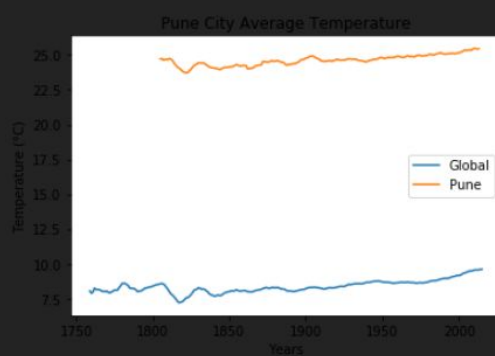
Performing Moving averages for both data

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 [44]: global_mv_avg = global_df['avg_temp'].rolling(10).mean()
local_mv_avg = pune_df['avg_temp'].rolling(10).mean() #rolling avg of 10 years at a time
```

Plotting rolling average data to find trend

```
In [45]: plt.plot(global_df['year'],global_mv_avg,label='Global')
plt.plot(pune_df['year'],local_mv_avg,label='Pune')
plt.legend()
plt.xlabel("Years")
plt.ylabel("Temperature (°C)")
plt.title("Pune City Average Temperature")
plt.show()
```



Observations:

- Post 1840s, global temperature seems to be increasing exponentially, indicating significant global warming effects
- In Pune, this increase of temperature is seen from 1940s, almost 100 years after global increase
- Pune is hotter than the average global temperature.
- Mean temp of Pune is 24.63 whereas globally it is 8.36
- In Pune, max change in temp was between 1825-1830 and 1890-1910 (both around 0.75 degree)
- Globally max temp rise in a short span was between 1825-1830 which was around 0.75 degrees