

Project 1

EXPLORE WEATHER TRENDS

NAME: PAMOLI DUTTA

DESCRIBING THE SOLUTION FLOW:

This project has four parts, as indicated in the instructions:

- i. Extracting the data
- ii. Opening up the CSV
- iii. Creating a line chart
- iv. Making Observations

For each of the tasks and the questions associated with them, the individual solutions would include:

- i. Name of the tools and code used to achieve the task (if applicable)
- ii. Explanation of the code and the methodology used (if applicable)
- iii. Output/ conclusion (if any)

- **EXTRACTING THE DATA:**

- Write a SQL query to extract the city level data. Export to CSV.
- Write a SQL query to extract the global data. Export to CSV.

SOLUTION:

Tool used:

SQL workspace provided with the project

Code:

```
/*Exploring the structure of the city_list*/
```

```
SELECT * FROM city_list  
LIMIT 3;
```

```
/*Looking for the list of cities included from my coutry*/
```

```
SELECT * FROM city_list  
WHERE UPPER(Country)='INDIA';
```

```
/*City 'Hyderabad' is included*/
```

```
/*Exploring the structure of the city_data*/
```

```
SELECT * FROM city_data  
LIMIT 3;
```

```
/*Extracting relevant fields only for city 'Hyderabad'*/
```

```
SELECT year, avg_temp
FROM city_data
WHERE UPPER(city)='HYDERABAD' AND UPPER(country)='INDIA';
```

/*Extracting data for global temp*/

```
SELECT * FROM global_data
WHERE year BETWEEN 1796 AND 2013;
```

Explanation:

- First, a query is written to show the first 3 rows in *city_list* data, just to understand the structure of the data
- The list only contains *country* and *city*, so next I searched for the list of cities included in the database for my country, *India*. In WHERE statement, the UPPER function is used to avoid omission of city due to case sensitivity.
- The list of cities include *Hyderabad*, my current city of residence, so next the temperature information are extracted for Hyderabad only. Along with *city*, an AND condition has been used on *country* also so that information on any other city also named Hyderabad but belonging to a country other than India (e.g. Pakistan) are excluded.
- The global data has more time points than the data corresponding to city. Therefore, I've extracted global data only for the period for which the city data exists (1796 to 2013), so that a comparison is viable

- OPENING UP THE CSV:

- After the above queries were executed, the resulting datasets, one containing average temperature of Hyderabad for 1796-2013 and another containing the same at a global level, are downloaded in the local system in .csv format
- However, as the .csv are opened, a filter on the *avg_temp* field in city data reveals some missing values, which I would impute in the next step

- CREATING A LINE CHART:

- Create a line chart that compares your city's temperatures with the global temperatures. Make sure to plot the *moving average* rather than the yearly averages in order to smooth out the lines

SOLUTION:

The solution to this part comprises three steps:

- Step 1: Missing value imputation for city data
- Step 2: Calculating the moving average
- Step 3: Plotting the MA values in a line chart

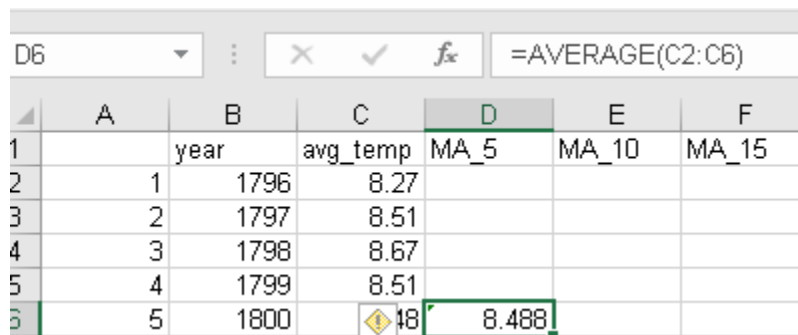
Tool used:

- MS Excel (For computing moving averages)
- R (For missing value imputation and creating line charts)

Methodology to compute the moving average:

I've used three periods (5, 10 and 15) to calculate the moving averages. Assuming data starts from Cell_2, the moving average for the data point in Cell_2 was calculated using the formula:
AVERAGE(Cell_2:Cell_(period of MA+1)).

For city data, the moving averages were calculated on the imputed dataset.



	A	B	C	D	E	F
1		year	avg_temp	MA_5	MA_10	MA_15
2		1	1796	8.27		
3		2	1797	8.51		
4		3	1798	8.67		
5		4	1799	8.51		
6		5	1800	8.488		

Code for imputation and creating line chart:

```
#Reading the data files
```

```
hyd_data=as.data.frame(read_csv("G:/AppData/InfoOne_sasmix/promomix1/dutta  
pa5/Udacity/Hyderabad_weather_data.csv"))  
global_data=as.data.frame(read_csv("G:/AppData/InfoOne_sasmix/promomix1/dut  
tapa5/Udacity/Global_weather_data.csv"))
```

```
#Installing and attaching library for multiple imputation
```

```
install.packages('mice')
```

```
library(mice)
```

```
mice=mice(hyd_data,method="norm.predict")
```

```
imputed_hyd_data=complete(mice)
```

```
#Exporting imputed dataset to compute moving average in excel
```

```
write.csv(data.matrix(imputed_hyd_data),"G:/AppData/InfoOne_sasmix/promomix  
1/duttapa5/Udacity/Imputed_Hyderabad_weather_data_1.csv")
```

```
#Importing the excel sheets consisting moving average values
```

```
hyd_ma_data=as.data.frame(read_csv("G:/AppData/InfoOne_sasmix/promomix1/d  
uttapa5/Udacity/Imputed_Hyderabad_weather_data.csv"))
```

```
global_ma_data=as.data.frame(read_csv("G:/AppData/InfoOne_sasmix/promomix  
1/duttapa5/Udacity/Global_weather_data.csv"))
```

```
#Creating 5-year MA plot
```

```
library(plotrix)
```

```
plot(hyd_ma_data$MA_5,main="Temperature Comparison: Global vs.
```

```
Hyderabad",xlab="5-Year MA",ylab="Temperature",type="o",col="red")
```

```
color.axis(side=2,col="red")
```

```
par(new=TRUE)
```

```
plot(global_ma_data$MA_5,axes=FALSE,xlab="5-Year
```

```
MA",ylab="Temperature",type="o",col="blue")
```

```
axis(side=4)
```

```
color.axis(side=4,col="blue")
```

```
legend("topleft",legend=c("Hyderabad","Global"),pch=c(16,16),col=c("red","blue"
```

```
))
```

```
#Creating 10-year MA plot
```

```
plot(hyd_ma_data$MA_10,main="Temperature Comparison: Global vs.
```

```
Hyderabad",xlab="10-Year MA",ylab="Temperature",type="o",col="red")
```

```

color.axis(side=2,col="red")
par(new=TRUE)
plot(global_ma_data$MA_10,axes=FALSE,xlab="10-Year
MA",ylab="Temperature",type="o",col="blue")
axis(side=4)
color.axis(side=4,col="blue")
legend("topleft",legend=c("Hyderabad","Global"),pch=c(16,16),col=c("red","blue"
))

```

```

#Creating 15-year MA plot
plot(hyd_ma_data$MA_15,main="Temperature Comparison: Global vs.
Hyderabad",xlab="15-Year MA",ylab="Temperature",type="o",col="red")
color.axis(side=2,col="red")
par(new=TRUE)
plot(global_ma_data$MA_15,axes=FALSE,xlab="15-Year
MA",ylab="Temperature",type="o",col="blue")
axis(side=4)
color.axis(side=4,col="blue")
legend("topleft",legend=c("Hyderabad","Global"),pch=c(16,16),col=c("red","blue"
))

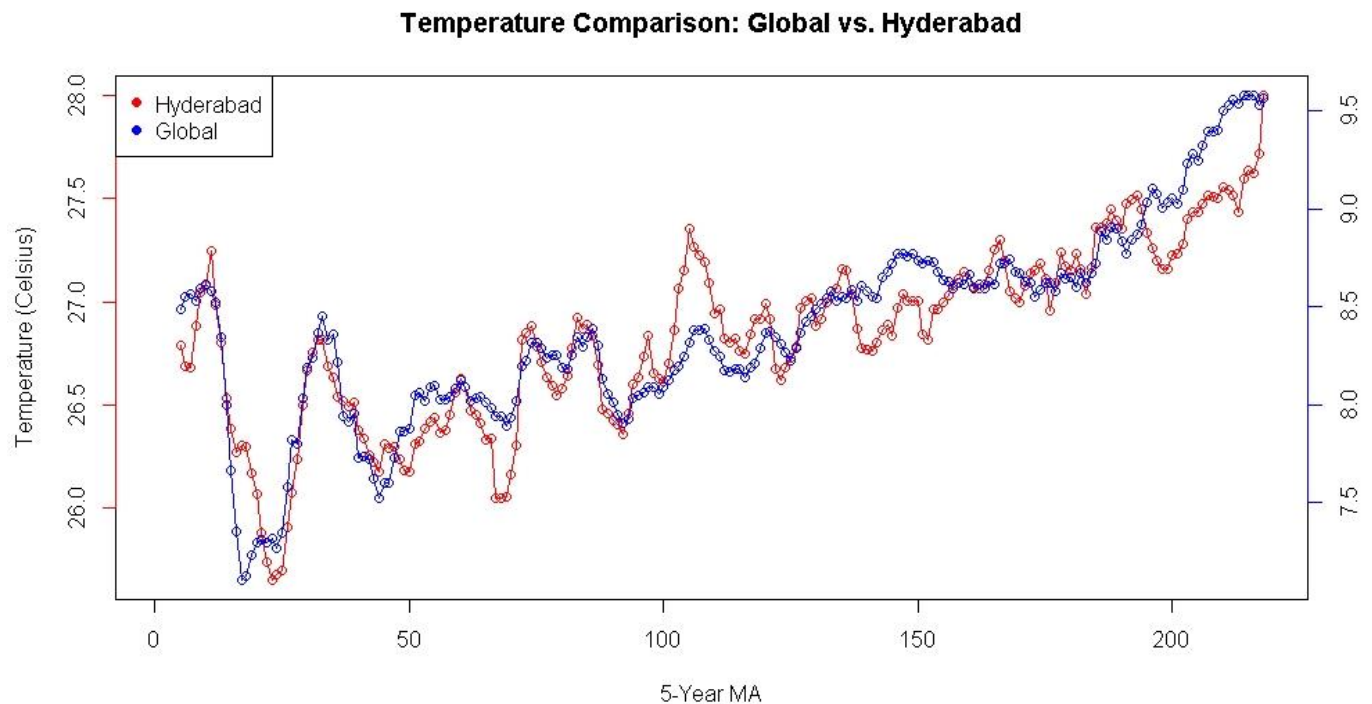
```

Explanation:

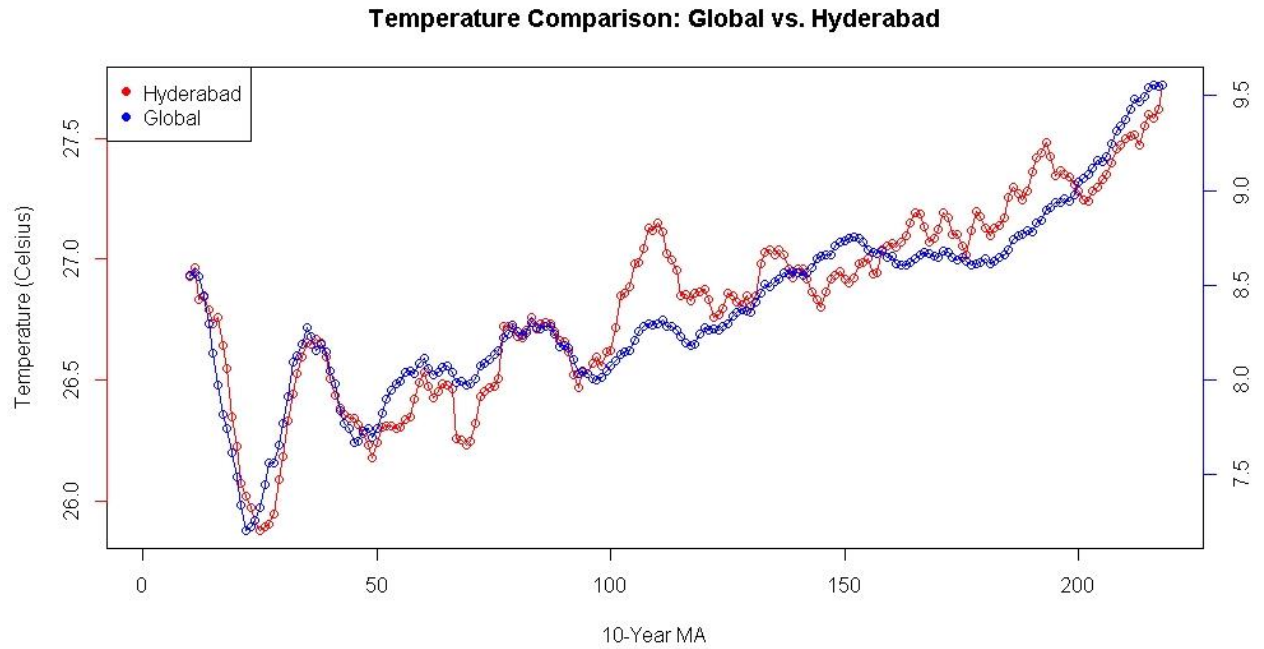
- First, the csv dataset are read into R and transformed into data frames for computational and representational conveniences
- Next, the missing values in city level data are imputed. Since the temperature has an increasing trend that a simple mean imputation would not be able to capture, method of Multiple Imputation by Chained Equations is used
- The imputed datasets are exported to Excel. Further, 5, 10 and 15 year MA have been computed to check the extent of smoothening in each

- case and to finally identify a period such that the trend is neither over-smoothed nor very erratic
- After calculation, the MAs are again imported to R and plotted for each of the 5, 10 and 15 year period

Output:

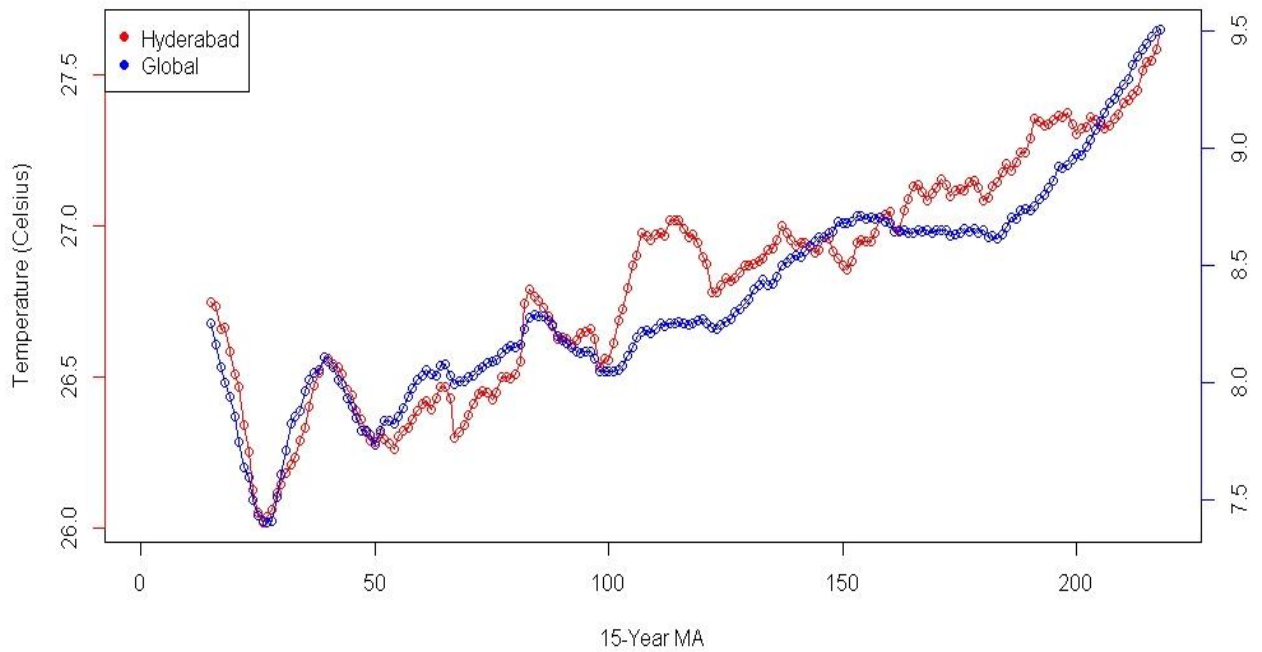


This chart shows 5 year moving averages. Though an increasing trend is clearly visible here, lot of erratic peaks are also present, so I'll plot the 10-year MAs next to smoothen out the roughness.



The 10 year MA plot looks reasonably smoothened. However, I'll plot the 15-year MAs finally to see how far that smoothenes the trend line.

Temperature Comparison: Global vs. Hyderabad



As expected, the global MA almost is smoothened to a straight line here.

For comparison of trends, I'll use the 10-year MA going forward.

Question:

What were your key considerations when deciding how to visualize the trends?

- Both the points and the lines connecting them are plotted to visualize both the individual data points and the general trend.
- Since the global temperatures and the Hyderabad temperatures fall in different ranges, plotting them on a single axis would make trends look like two parallel lines. Therefore, a secondary axis is used for plotting global temperature.
- The axes are colored and labelled properly to avoid ambiguity
- Two different colors are used to demarcate the different trends and a legend is included to clearly identify the data points.

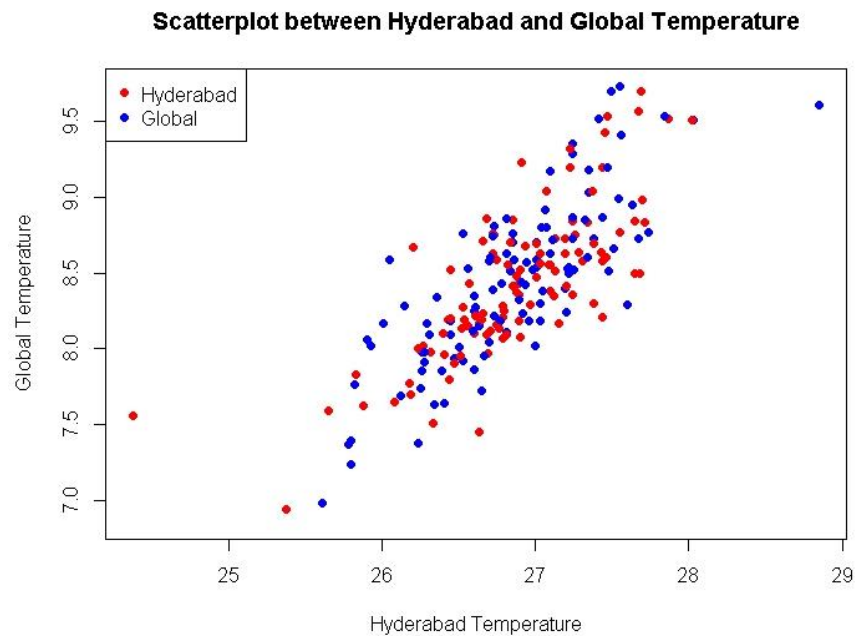
OBSERVATIONS:

- i. Both the lines show an increasing trend over the years, the individual highs and lows also coincide in most of the cases. So, one can say, Hyderabad's temperature changes are consistent with the global changes.
- ii. However, in general, Hyderabad's average temperature has remained quite higher than the global average. While the global temperature values lie within a scale of 7.5-9.5, the Hyderabad temperatures usually fall in the range of 26-27.5.
- iii. Though the trends are similar in both the cases, Hyderabad shows more frequent ups and downs, while the global temperature increases almost steadily.
- iv. The slope for the last few years looks higher than that in the initial years which means the increase in temperature has accelerated over

the last few years. However, a deeper look into the data would be required to further comment on this.

ADDITIONAL FINDINGS:

From the MA plots, we have seen both the global and Hyderabad temperature have an increasing trend. Now, to check whether they are associated or not, we plot a scatterplot as follows:



The scatterplot clearly exhibits a linear positive association between the temperatures.

Since the association is linear in nature, Pearson's correlation coefficient can be calculated to measure the extent of association.

The correlation coefficient is calculated using R, the results are as follows:

Pearson's product-moment correlation

```
data: hyd_data$avg_temp and global_data$avg_temp
t = 18.14, df = 209, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
0.7233730 0.8294847
sample estimates:
cor
0.782033
```

Based on the p-value the null hypothesis of correlation coefficient being 0 is rejected and as expected, there exists a high positive correlation of 0.78 between the Hyderabad and global temperatures.

R code for scatterplot and correlation coefficient:

```
#Creating Scatterplot
plot(hyd_data$avg_temp,global_data$avg_temp,xlab="Hyderabad
Temperature",main="Scatterplot between Hyderabad and Global
Temperature",ylab="Global Temperature",pch=16,col=c("red","blue"))
legend("topleft",legend=c("Hyderabad","Global"),pch=c(16,16),col=c("red","blue"
))

#Calculating correlation coefficients
cor.test(hyd_data$avg_temp,global_data$avg_temp,method="pearson")
```

Question:

Can you estimate the average temperature in your city based on the average global temperature?

- The scatterplot suggests there exists a linear relationship between the global and Hyderabad temperature. Therefore, we can estimate the temperature of Hyderabad based on city temperature using a linear regression as follows:

$$\text{Hyderabad_temperature} = \text{Intercept} + \text{Global_temperature} + \text{Error}$$

Where intercept captures the metrics we were unable to account for and the error term represents random fluctuations.

R code for regression:

```
#Running linear regression
model=lm(hyd_data$avg_temp~global_data$avg_temp)

#Prediction for existing data points
predict(model)
```