**Machine Learning**

**Data Distribution**

For data visualisation, Python allows use of Matplotlib. Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonot, Tkinter.

**Bar graph**

A bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally. Let us consider rainfall recorded in the different towns in mm for four cities in South Africa as below.

Rainfall= [140, 200, 120, 157]

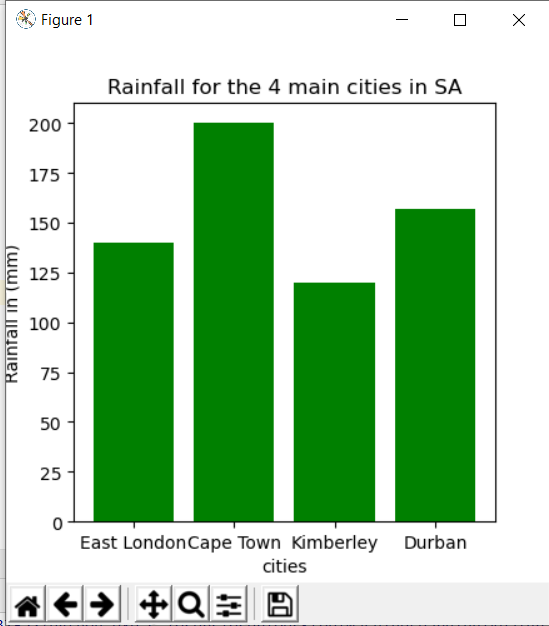
Cities=[“East London”, “Cape Town”, “Kimberley”, “Durban]

## For us to present the following diagrammatically, we need matplotlib.pyplot. Matplotlib. pyplot provides a MATLAB like plotting framework. The plt.bar function, however, takes a list of positions and values, the for x are then provided by plt.xticks().

## Try the following code and see how the graphs are presented.

**import** matplotlib.pyplot **as** plt  
cities=[**'East London'**, **'Cape Town'**, **'Kimberley'**, **'Durban'**]  
rainfall= [140, 200, 120, 157]  
x\_pos = [i **for** i, \_ **in** enumerate(cities)] *#labels on the x-axis  
#labeling and visuals*plt.bar(x\_pos, rainfall, color=**'green'**)  
plt.xlabel(**"cities"**)  
plt.ylabel(**"Rainfall in (mm)"**)  
plt.title(**"Rainfall for the 4 main cities in SA"**)  
plt.xticks(x\_pos, cities)  
plt.show()

You will have an output as show below



Take note that the image allows you to save it and you can use it for your presentations.

**Box Plots**

Boxplot is probably one of the most common type of graphic. It gives a nice summary of one or several numeric variables. The line that divides the box into 2 parts represents the median of the data. The end of the box shows the upper and lower quartiles. The extreme lines show the highest and lowest value excluding outliers. Remember we spoke of outliers at the beginning.

We are going to use testMarks data set for our boxplots.

Testmarks = [98,78, 68, 73, 72, 97, 88, 60, 94, 95, 80, 73, 82, 80, 99, 91, 74, 88, 70, 94, 86, 81, 100, 99, 84, 93, 94, 79]