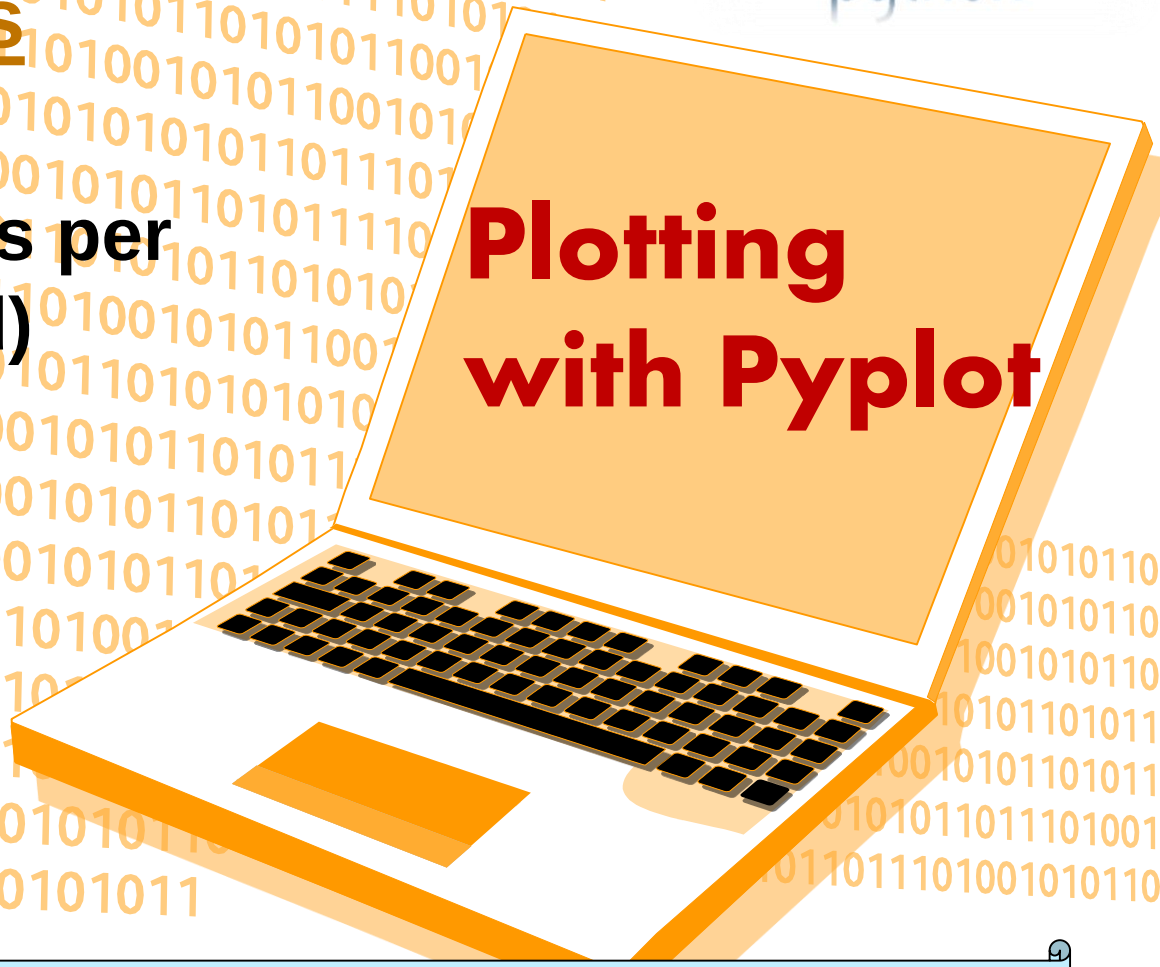
Chapter 6 :

Informatics Practices

Class XII (As per CBSE Board)

Plotting with Pyplot



New Syllabus 2019-20

Visit : python.mykvs.in for regular updates

Plotting with Pyplot

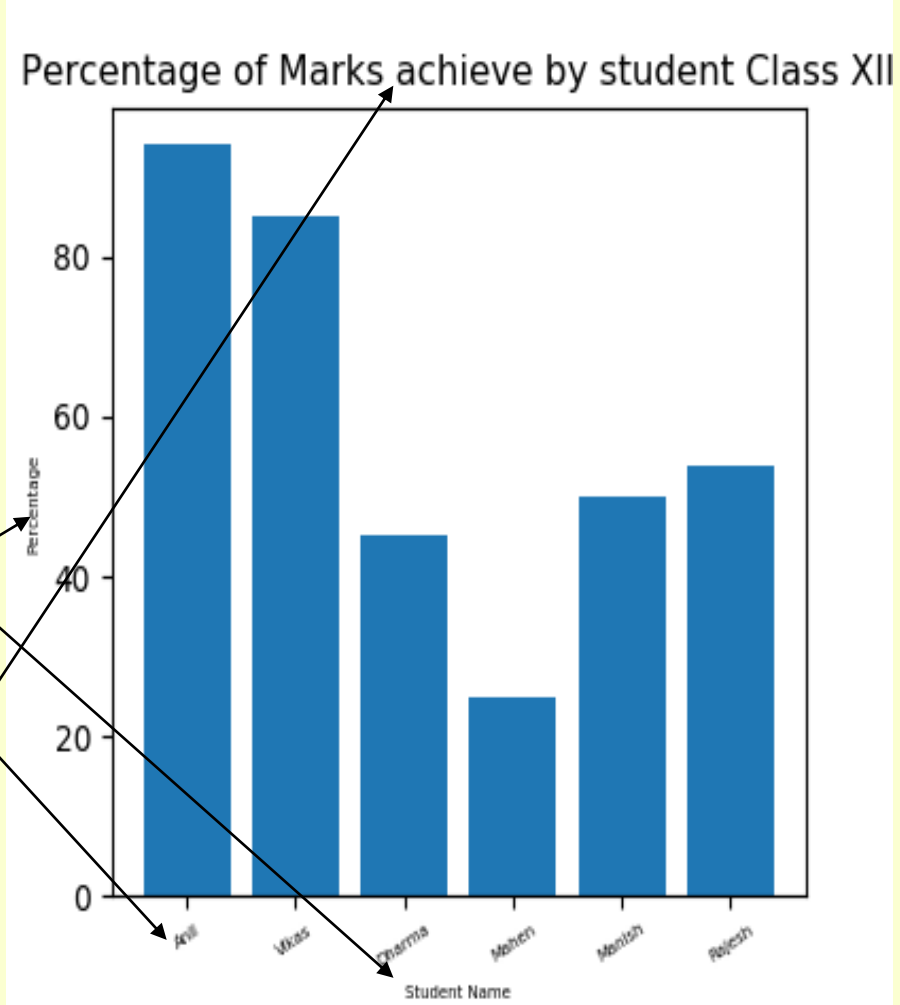
Matplotlib is the whole python package/ library used to create 2D graphs and plots by using python scripts. **pyplot** is a module in matplotlib, which supports a very wide variety of graphs and plots namely - histogram, bar charts, power spectra, error charts etc. It is used along with NumPy to provide an environment for MatLab.

Pyplot provides the state-machine interface to the plotting library in matplotlib.It means that figures and axes are implicitly and automatically created to achieve the desired plot. For example, calling plot from pyplot will automatically create the necessary figure and axes to achieve the desired plot. Setting a title will then automatically set that title to the current axes object.The pyplot interface is generally preferred for non-interactive plotting (i.e., scripting).

Plotting with Pyplot

Plot bar graphs
e.g program

```
import matplotlib.pyplot as plt
import numpy as np
label = ['Anil', 'Vikas', 'Dharma',
'Mahen', 'Manish', 'Rajesh']
per = [94,85,45,25,50,54]
index = np.arange(len(label))
plt.bar(index, per)
plt.xlabel('Student Name', fontsize=5)
plt.ylabel('Percentage', fontsize=5)
plt.xticks(index, label, fontsize=5,
rotation=30)
plt.title('Percentage of Marks achieve
by student Class XII')
plt.show()
```



Plotting with Pyplot

Histogram in Python –

There are various ways to create histogram in python pandas.One of them is using matplotlib python library.Using this library we can easily create histogram.We have to write just few statements to create histogram.

So install matplotlib library using following statements at command prompt.

>**pip install matplotlib**

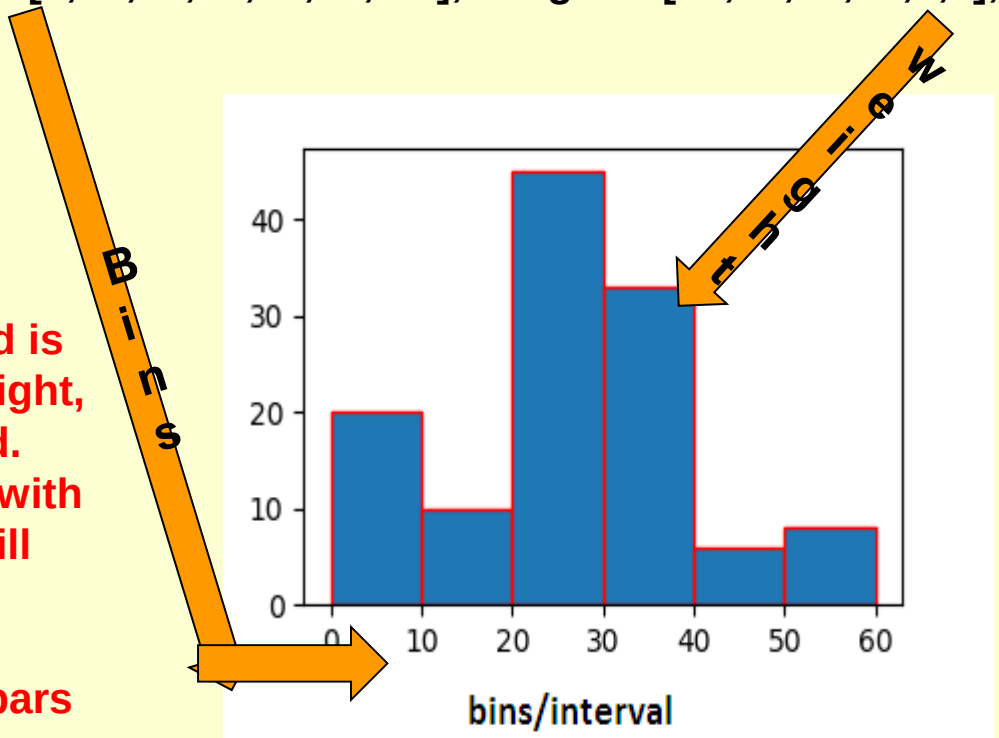
After installation we can create histogram.if pip does not work then copy the pip.exe file to the folder where we want to run the above command or move to the folder of pip.exe then write above command.

Plotting with Pyplot

Histogram in Python –

E.g. Program in python.Develop a python program with below code and execute it.

```
import numpy as np
import matplotlib.pyplot as plt
data = [1,11,21,31,41]
plt.hist([5,15,25,35,45, 55], bins=[0,10,20,30,40,50, 60], weights=[20,10,45,33,6,8],
edgecolor="red")
plt.show()
```



#first argument of hist() method is position (x,y Coordinate) of weight, where weight is to be displayed.
No of coordinates must match with No of weight otherwise error will generate
#Second argument is interval
#Third argument is weight for bars

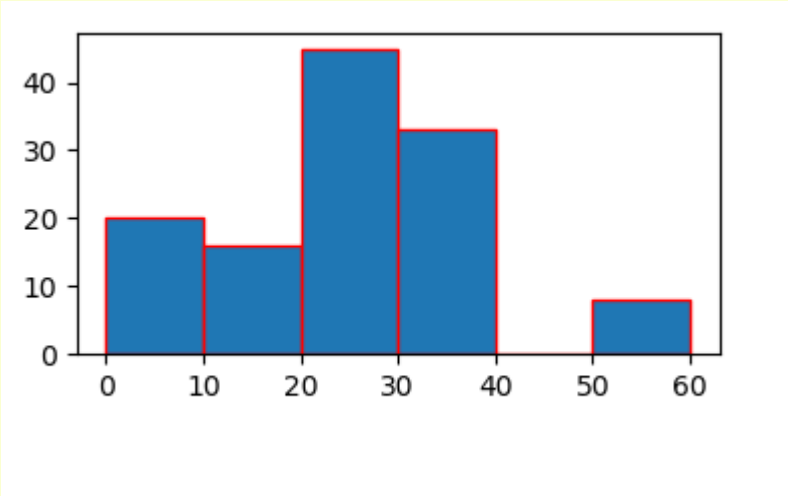
Plotting with Pyplot

Histogram in Python –

For better understading we develop the same program with minor change .

```
import numpy as np
import matplotlib.pyplot as plt
data = [1,11,21,31,41]
plt.hist([5,15,25,35,15, 55], bins=[0,10,20,30,40,50, 60], weights=[20,10,45,33,6,8],
edgecolor="red")
plt.show()
```

 # at interval(bin)40 to 50 no bar because we have not mentioned position from 40 to 50 in first argument(list) of hist method. Where as in interval 10 to 20 width is being Displayed as 16 (10+6 both weights are added) because 15 is twice In first argument.



Plotting with Pyplot

Histogram in Python –

By default bars of histogram is displayed in blue color but we can change it to other color with following code .

```
plt.hist([1,11,21,31,41, 51], bins=[0,10,20,30,40,50, 60], weights=[10,1,0,33,6,8], facecolor='y', edgecolor="red")
```

In above code we are passing ‘y’ as facecolor means yellow color to be displayed in bars.

To give a name to the histogram write below code before calling show()

```
plt.title("Histogram Heading")
```

he histogram can be saved by clicking on the Save button on the GUI. Also, the following code will save the histogram as a PNG image.

```
plt.savefig(“temp.png")
```

For x and y label below code can be written

```
plt.xlabel('Value')
plt.ylabel('Frequency')
```

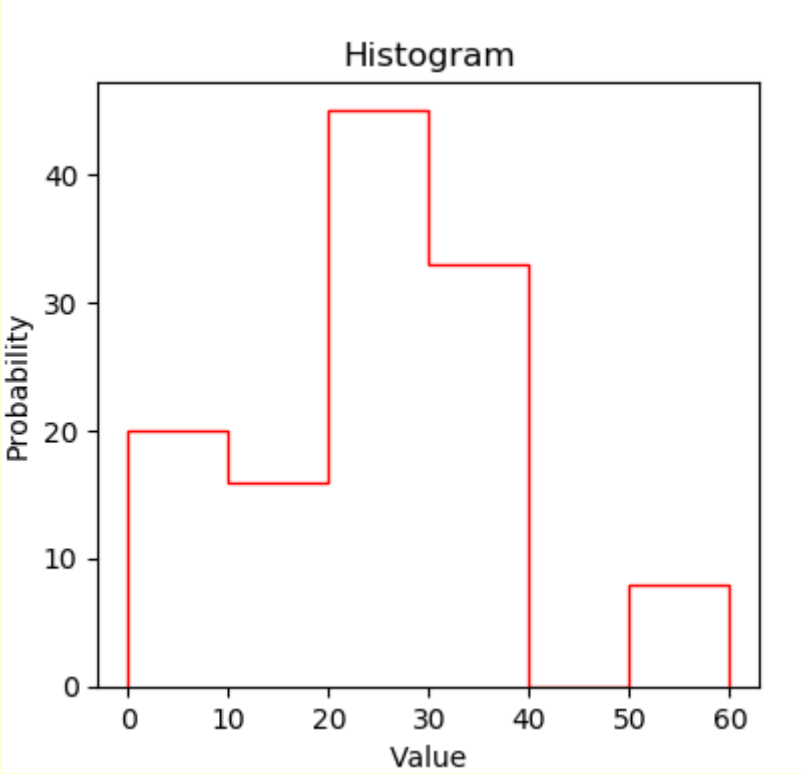
Plotting with Pyplot

Frequency polygons

If we just connect the top center points of each bins then we obtain relative frequency polygon.

e.g.program

```
import numpy as np
import matplotlib.pyplot as plt
data = [1,11,21,31,41]
plt.hist([5,15,25,35,15, 55],
bins=[0,10,20,30,40,50, 60],
weights=[20,10,45,33,6,8],
edgecolor="red",histtype='step')
#plt.hist(data, bins=20, histtype='step')
plt.xlabel('Value')
plt.ylabel('Probability')
plt.title('Histogram')
plt.show()
```



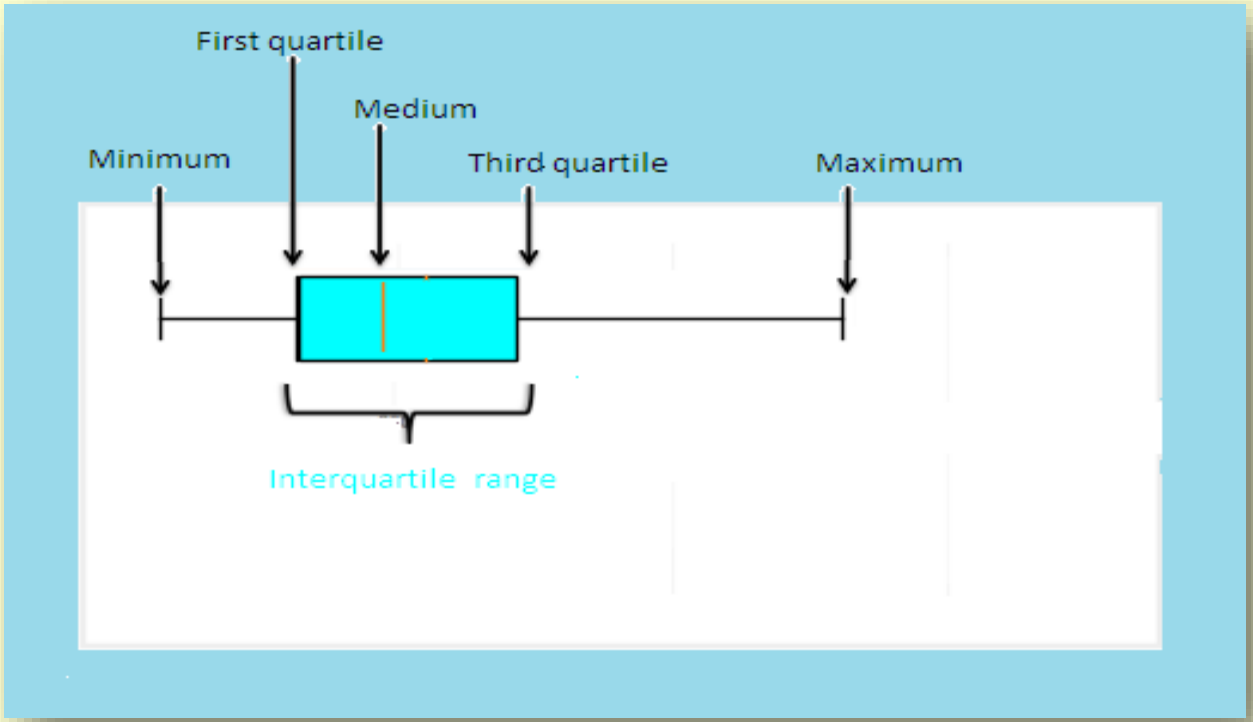
Plotting with Pyplot

Box Plots

A Box Plot is the visual representation of the statistical five number summary of a given data set.

A Five Number Summary includes:

- **Minimum**
- **First Quartile**
- **Median (Second Quartile)**
- **Third Quartile**
- **Maximum**

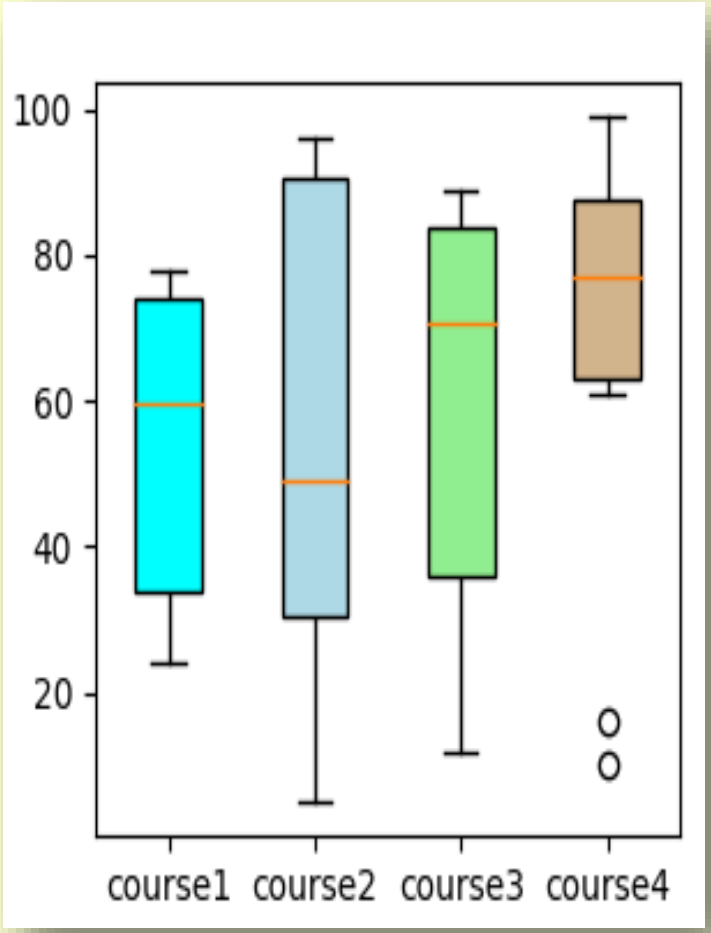


Plotting with Pyplot

Box Plots

e.g.program

```
import matplotlib.pyplot as plt
value1 = [72,76,24,40,57,62,75,78,31,32]
value2=[62,5,91,25,36,32,96,95,30,90]
value3=[23,89,12,78,72,89,25,69,68,86]
value4=[99,73,70,16,81,61,88,98,10,87]
box_plot_data=[value1,value2,value3,value4]
box=plt.boxplot(box_plot_data,vert=1,patch_ar
tist=True,labels=['course1','course2','course3','
course4'],
            )
 colors = ['cyan', 'lightblue', 'lightgreen', 'tan']
for patch, color in zip(box['boxes'], colors):
    patch.set_facecolor(color)
plt.show()
```



Note:- if vert=0 in boxplot() is set then horizontal box plots will be drawn

Plotting with Pyplot

Scatter plots

A scatter plot is a two-dimensional data visualization that uses dots to represent the values obtained for two different variables - one plotted along the x-axis and the other plotted along the y-axis.

e.g.program

```
import matplotlib.pyplot as plt

weight1=[93.3,67,62.3,43,71,71.8]
height1=[116.3,110.7,124.8,176.3,137.1,113.9]

plt.scatter(weight1,height1,c='b',marker='o')
plt.xlabel('weight', fontsize=16)
plt.ylabel('height', fontsize=16)
plt.title('scatter plot - height vs weight',fontsize=20)
plt.show()
```