Practical 2

21BCM074

KARAN ADWANI

Q. Matplotlib, seaborn, scipy and statsmodels/ Library for Probability and statistics in R/ Library for Probability and statistics in Matlab

Ans

Matploitlib is a Python Library used for plotting, this python library provides and objected-oriented APIs for integrating plots into applications.

* [Types Of Plots](https://www.edureka.co/blog/python-matplotlib-tutorial/#types)–  [Bar Graph](https://www.edureka.co/blog/python-matplotlib-tutorial/#Bargraph)  
  – [Histogram](https://www.edureka.co/blog/python-matplotlib-tutorial/#Histogram)  
  – [Scatter Plot](https://www.edureka.co/blog/python-matplotlib-tutorial/#Scatter)  
  – [Area Plot](https://www.edureka.co/blog/python-matplotlib-tutorial/#AreaPlot)  
  – [Pie Chart](https://www.edureka.co/blog/python-matplotlib-tutorial/#PiePlot)

Code:

**from** matplotlib **import** pyplot as plt

 #Plotting to our canvas

 plt.plot([1,2,3],[4,5,1])

 #Showing what we plotted

 plt.show()

x **=** [5,2,7]

y **=** [2,16,4]

plt.plot(x,y)

plt.title('Info')

plt.ylabel('Y axis')

plt.xlabel('X axis')

plt.show()

style.use('ggplot')

x **=** [5,8,10]

y **=** [12,16,6]

x2 **=** [6,9,11]

y2 **=** [6,15,7]

plt.plot(x,y,'g',label**=**'line one', linewidth**=**5)

plt.plot(x2,y2,'c',label**=**'line two',linewidth**=**5)

plt.title('Epic Info')

plt.ylabel('Y axis')

plt.xlabel('X axis')

plt.legend()

plt.grid(True,color**=**'k')

plt.show()

plt.bar([0.25,1.25,2.25,3.25,4.25],[50,40,70,80,20],

label**=**"BMW",width**=**.5)

plt.bar([.75,1.75,2.75,3.75,4.75],[80,20,20,50,60],

label**=**"Audi", color**=**'r',width**=**.5)

plt.legend()

plt.xlabel('Days')

plt.ylabel('Distance (kms)')

plt.title('Information')

plt.show()

population\_age **=** [22,55,62,45,21,22,34,42,42,4,2,102,95,85,55,110,120,70,65,55,111,115,80,75,65,54,44,43,42,48]

bins **=** [0,10,20,30,40,50,60,70,80,90,100]

plt.hist(population\_age, bins, histtype**=**'bar', rwidth**=**0.8)

plt.xlabel('age groups')

plt.ylabel('Number of people')

plt.title('Histogram')

plt.show()

x **=** [1,1.5,2,2.5,3,3.5,3.6]

y **=** [7.5,8,8.5,9,9.5,10,10.5]

x1**=**[8,8.5,9,9.5,10,10.5,11]

y1**=**[3,3.5,3.7,4,4.5,5,5.2]

plt.scatter(x,y, label**=**'high income low saving',color**=**'r')

plt.scatter(x1,y1,label**=**'low income high savings',color**=**'b')

plt.xlabel('saving\*100')

plt.ylabel('income\*1000')

plt.title('Scatter Plot')

plt.legend()

plt.show()

days **=** [1,2,3,4,5]

 sleeping **=**[7,8,6,11,7]

 eating **=** [2,3,4,3,2]

 working **=**[7,8,7,2,2]

 playing **=** [8,5,7,8,13]

 plt.plot([],[],color**=**'m', label**=**'Sleeping', linewidth**=**5)

 plt.plot([],[],color**=**'c', label**=**'Eating', linewidth**=**5)

 plt.plot([],[],color**=**'r', label**=**'Working', linewidth**=**5)

 plt.plot([],[],color**=**'k', label**=**'Playing', linewidth**=**5)

 plt.stackplot(days, sleeping,eating,working,playing, colors**=**['m','c','r','k'])

 plt.xlabel('x')

 plt.ylabel('y')

 plt.title('Stack Plot')

 plt.legend()

 plt.show()

days **=** [1,2,3,4,5]

sleeping **=**[7,8,6,11,7]

eating **=** [2,3,4,3,2]

working **=**[7,8,7,2,2]

playing **=** [8,5,7,8,13]

slices **=** [7,2,2,13]

activities **=** ['sleeping','eating','working','playing']

cols **=** ['c','m','r','b']

plt.pie(slices,

  labels**=**activities,

  colors**=**cols,

  startangle**=**90,

  shadow**=** True,

  explode**=**(0,0.1,0,0),

  autopct**=**'%1.1f%%')

plt.title('Pie Plot')

plt.show()