# Matplotlib

Pyplot

# Presented By

Pranab Bora

A Student of Rajiv Gandhi University, Itanagar

Import csv file for visualization

```
import matplotlib.pyplot as plt
import pandas as pd
cars=pd.read_csv(r'mtcars.csv')
Cars
```

Visualization in Line Plot

```
import matplotlib.pyplot as plt
y=cars['hp']
#hp column values
x=range(32)
#total rows
plt.plot(x,y)
```

```
import matplotlib.pyplot as plt
y1=cars['hp']
y2=cars['disp']
x=range(32)
#total rows
plt.plot(x,y1)
plt.plot(x,y2)
plt.legend(['hp','disp'])
```

```
import matplotlib.pyplot as plt
y1=cars['hp']
#hp column values
y2=cars['disp']
#disp column values
x=range(32)
#total rows
plt.stackplot(x,y1, alpha=1, colors="orange")
plt.stackplot(x,y2, alpha=0.5, colors="green")
plt.legend(['hp','disp'])
```

```
import matplotlib.pyplot as plt
y1=cars['hp']
#hp column values
y2=cars['disp']
#disp column values
x=range(32)
#total rows
plt.plot(x,y1, linewidth=2.0, color="c")
plt.stackplot(x,y1, alpha=1, colors="purple")
plt.plot(x,y2, linewidth=1.0, color="r")
plt.stackplot(x,y2, alpha=0.5, colors="black")
plt.legend(['hp','disp'])
```

Bar plot

```
y=cars['hp']
x=range(32)
x1=cars['model'].tolist()
fig=plt.figure(figsize=(30,20))
plt.bar(x1,y, color="y", alpha=0.6)
```

Bar plot horizontal

```
import matplotlib.pyplot as plt
import pandas as pd
cars=pd.read_csv(r"mtcars.csv")
y=cars['hp']
x=range(32)
x1=cars['model'].tolist()
fig=plt.figure(figsize=(30,15))
plt.barh(x1,y, color="purple", alpha=0.9)
```

## Line plot with grid and marker

```
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0,10,1)
y=2*x+5
fig=plt.figure(figsize=(10,5))
plt.plot(x,y, linewidth=2.0, linestyle=":", color="purple", marker="o", alpha=0.5)
plt.title("Test")
plt.xlabel("x axis")
plt.ylabel("y axis")
plt.legend(["line1"], loc="best")
plt.grid(True)
plt.show()
```

## Subplot

```
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0,10,1)
y=2*x+5
y1=3*x+10
###plt.subplot(row, col, number)
plt.subplot(1,2,1)
plt.plot(x,y, linewidth=2.0, linestyle=":", color="purple", marker="o", alpha=0.5)
plt.title("Test")
plt.grid(True)
plt.xlabel("x axis")
plt.ylabel("y axis")
plt.legend(["line1"], loc="best")

plt.subplot(1,2,2)
plt.plot(x,y1, linewidth=2.0, linestyle=":", color="purple", marker="o", alpha=0.5)
plt.title("Test1")
plt.grid(True)
plt.xlabel("x axis")
plt.ylabel("y axis")
plt.legend(["line1"], loc="best")
plt.show()
```

## Bars

```
data={'Apple':20, 'mango':15, 'Lemon':30, 'Oranges':10}
#data is dictionary
#list is listing the values
#data.keys() is keys of the list
#data.values() is values of the keys
name=list(data.keys())
values=list(data.values())
fig=plt.figure(figsize=(10,5))
plt.bar(name,values)
plt.show()
```

## Bar Horizontal

```
data={'Apple':20, 'mango':15, 'Lemon':30, 'Oranges':10}
name=list(data.keys())
values=list(data.values())
fig=plt.figure(figsize=(10,5))
plt.barh(name,values, color="orange")
plt.title("Horizontal Bars")
plt.xlabel("Fruits")
plt.ylabel("Quantities")
plt.show()
```

Scatter

```
a=[10,20,30,40,50,60,70,80]
b=[5,3,8,9,4,5,6,7]
x=[9,6,7,8,5,2,5,6]
plt.scatter(a,b)
plt.scatter(a,x)
plt.show()

a=[10,20,30,40,50,60,70,80]
b=[5,3,8,9,4,5,6,7]
x=[9,6,7,8,5,2,5,6]
plt.scatter(a,b,color="yellow", edgecolor="red", marker="o", s=300)
plt.scatter(a,x,color="green", marker="v", s=200)
plt.legend(["ab","ax"],loc="best")
plt.xlabel("a")
plt.ylabel("b and x")
plt.title("Scatter")
plt.savefig("scatter.png")                    #save the plot arear as png
plt.show()
```

## Histogram

```
a=[10,12,15,17,19,20,25,29,30,35,39,45,50,65,70,90]
plt.hist(a, bins=[0,20,40,60,80,100])
plt.show()
```

```
a=[10,12,15,17,19,20,25,29,30,35,39,45,50,65,70,90]
bins=[0,20,40,60,80,100]
plt.hist(a, bins)
#plt.hist(array, interval)
plt.show()
```

```
a=[10,12,15,17,19,20,25,29,30,35,39,45,50,65,70,90]
bins=[0,20,40,60,80,100]
plt.hist(a, bins, color="yellow", edgecolor="red")
#plt.hist(array, interval)
plt.xlabel("bins")
plt.ylabel("array values")
plt.title("Histogram")
plt.savefig("hist.png")
plt.grid(True)
plt.show()
```

## Box plot

```
total=[10,52,63,14,25,12,9,15,32]
plt.boxplot(total, showmeans=True)
plt.show()

#first sort the array
#find the mean q2
#in mean position split the array
#find first array mean q1
#find second array mean q3
#IQR=|q3-q1|
#min =q0
#max =q4
#box height is between q3 - q1

total=[10,52,63,14,25,12,9,15,32]
data =[52,41,32,15,45,74]
dis=[12,15,17,95,65,14,45]
var=list([total, data, dis])
plt.boxplot(var, showmeans=True)
plt.show()
```

Violin plot

```
total=[10,52,63,14,25,12,9,15,32]
data =[52,41,32,15,45,74]
dis=[12,15,17,95,65,14,45]
var=list([total, data, dis])
plt.violinplot(var, showmeans=True, showmedians=True)
plt.show()
```

Not much used in general
Areaplot()
Quiverplot()
Streamplot()

Pie plot

```
label=['dog', 'cat', 'lion', 'tiger']
size=[40,52,78,32]
plt.pie(size, labels=label)
plt.show()

label=['dog', 'cat', 'lion', 'tiger']
size=[40,52,78,32]
plt.pie(size, labels=label, startangle=90, autopct="%1.1f%%")
plt.show()

label=['dog', 'cat', 'lion', 'tiger']
size=[40,52,78,32]
plt.pie(size, labels=label, startangle=90, autopct="%1.1f%%", explode=(0.5,0.1,0,0))
plt.show()
```

Doughnut plot

```
#actually two pie plot to remove mid portion

label=['dog', 'cat', 'lion', 'tiger']
size=[40,52,78,32]
c=[5]
plt.pie(size, labels=label, radius=1.5, startangle=90)
plt.pie(c, radius=1, colors="yellow")
plt.show()

label=['dog', 'cat', 'lion', 'tiger']
size=[40,52,78,32]
c=[5]
plt.pie(size, labels=label, radius=1.5, startangle=90)
plt.pie(c, radius=1, colors="white")
plt.show()
```

## Area plot

```
import numpy as np
import matplotlib.pyplot as plt
x=np.arange(0,10,1)
y=2*x+5
fig=plt.figure(figsize=(10,5))
plt.stackplot(x,y, linewidth=2.0, linestyle=":", color="purple", alpha=0.5)
plt.title("Test")
plt.xlabel("x axis")
plt.ylabel("y axis")
plt.legend(["line1"], loc="best")
plt.grid(True)
plt.show()
```

# Thank you !!!!