

# DATA VISUALISATION IN PYTHON

## CHEATSHEET

### Why Is Data Visualisation an Important Concept ?

- Because it help us understand distribution, trend, relationship, comparison and composition of data values
- It helps decision makers to quickly examine large piles of data and discover the hidden patterns/insights

" BEAUTY OF AN ART LIES IN THE MESSAGE IT CONVEYS "

### WHAT IS REQUIRED TO MAKE VISUALISATION IN PYTHON ?

#### MATPLOTLIB

Python based plotting library offers matplotlib with a complete 2D support along with limited 3D graphics support. It is useful in producing publication quality figures in interactive environments across platforms.

#### SEABORN

Being based on matplotlib, seaborn offers various features such as built in themes, color palettes, functions and tools to visualize univariate, bivariate, linear regression, matrices of data, statistical time series etc which lets us to build complex visualizations.

### Sample Data Set Used For The VISUALISATION Show Below

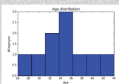
Year	Age	Sex	Smoker	Region	Sales
2010	24	M	Yes	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100
2010	24	M	No	US	100

#### Import Data Set:

```
import matplotlib.pyplot as plt
import pandas as pd
df=pd.read_excel("E:/Finsb.xlsx", "Sheet1")
```

#### Histogram

```
fig=plt.figure()
ax = fig.add_subplot(111)
ax.hist(df[Age], bins = 7) # Here you can
play with number of bins Labels and Title
plt.title("Age distribution")
plt.xlabel("Age")
plt.ylabel("Employee")
plt.show()
```



#### Box Plot

```
import matplotlib.pyplot as plt
import pandas as pd
fig=plt.figure()
ax = fig.add_subplot(111)
x.boxplot(df[Age])
plt.show()
```



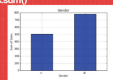
#### Violin Plot

```
import seaborn as sns
sns.violinplot(df[Age], df[Gender])
#Variable Plot
sns.despine()
```



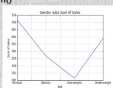
#### Bar Chart

```
var = df.groupby('Gender').Sales.sum()
#grouped sum of sales at
Gender level
fig = plt.figure()
ax1 = fig.add_subplot(111)
ax1.set_xlabel('Gender')
ax1.set_ylabel('Sum of Sales')
ax1.set_title('Gender wise Sum of Sales')
var.plot(kind='bar')
```



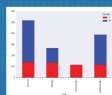
#### Line Chart

```
var = df.groupby('BMI').Sales.sum()
fig = plt.figure()
ax1 = fig.add_subplot(111)
ax1.set_xlabel('BMI')
ax1.set_ylabel('Sum of Sales')
ax1.set_title('BMI wise Sum of Sales')
var.plot(kind='line')
```



#### Stacked Column Chart

```
var = df.groupby('BMI','Gender').Sales.sum()
var.unstack().plot(kind='bar', stacked=True, color=['red', 'blue', 'grid=False])
```



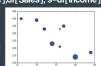
#### Scatter Plot

```
fig = plt.figure()
ax = fig.add_subplot(111)
ax.scatter(df[Age], df[Sales])
plt.show()
```



#### Bubble Plot

```
fig = plt.figure()
ax = fig.add_subplot(111)
ax.scatter(df[Age], df[Sales], s=df[Income])
plt.show()
```



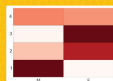
#### Pie Chart

```
var=df.groupby(['Gender']).sum().stack()
temp=var.unstack()
type(temp)
x_list = temp['Sales']
label_list = temp.index
pyplot.axis('equal') #The pie chart
is oval by default. To make it a
circle use pyplot.axis('equal')
plt.pie(x_list, labels=label_list, autopct='%1.1f%%')
plt.title("Pastafarianism expenses")
plt.show()
```



#### Heat Map

```
import numpy as np
data = np.random.rand(4,2)
rows = list('1234') #rows
categories_columns =
list('MF') #column categories
fig,ax=plt.subplots()
ax.pcolor(data,cmap=plt.cm.Reds,edgecolors='k')
ax.set_xticks(np.arange(0,2)+0.5)
ax.set_yticks(np.arange(0,4)+0.5)
ax.xaxis.tick_bottom()
ax.yaxis.tick_left()
ax.set_xticklabels(columns, minor=False, fontsize=20)
ax.set_yticklabels(rows, minor=False, fontsize=20)
plt.show()
```



To view the complete guide on data visualisation in python visit here : <http://bit.ly/1FJTkrF>