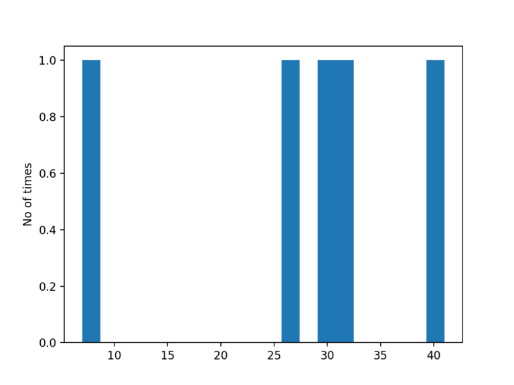
# Python

## Histogram

a **histogram represents** a frequency distribution by means of rectangles whose widths **represent** class intervals and whose areas are proportional to the corresponding frequencies: the height of each is the average frequency density for the interval.

import matplotlib.pyplot as plt  
import numpy as np  
import pandas as pdx = np.random.random\_integers(1, 100, 5)  
plt.hist(x, bins=20)  
plt.ylabel('No of times')  
plt.show()



## Subplotting

### With multi dimensional subplots

import matplotlib.pyplot as plt

import numpy as np

fig, ax = plt.subplots(2, 2)

x = np.linspace(0, 8, 1000)

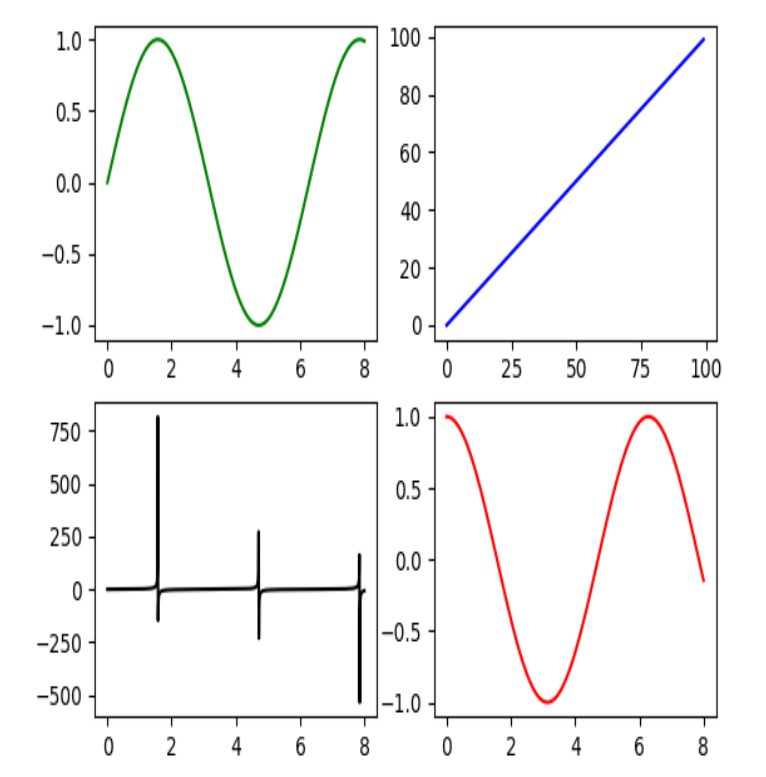
ax[0, 0].plot(x, np.sin(x), 'g') #row=0, col=0

ax[1, 0].plot(x, np.tan(x), 'k') #row=1, col=0

ax[0, 1].plot(range(100), 'b') #row=0, col=1

ax[1, 1].plot(x, np.cos(x), 'r') #row=1, col=1

fig.show()



### With single dimensional subplots

fig, ax = plt.subplots(1,2,figsize=(10,5))

ax[0].hist(cross\_tab['Total'], bins=50)

ax[0].set\_xlabel('Number of patches')

ax[0].set\_ylabel('Frequency')

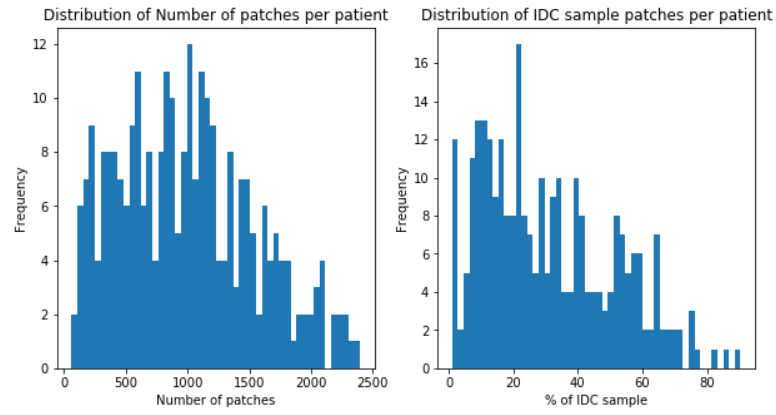
ax[0].set\_title('Distribution of Number of patches per patient')

ax[1].hist(cross\_tab['%\_1'], bins=50)

ax[1].set\_xlabel('% of IDC sample')

ax[1].set\_ylabel('Frequency')

ax[1].set\_title('Distribution of IDC sample patches per patient')

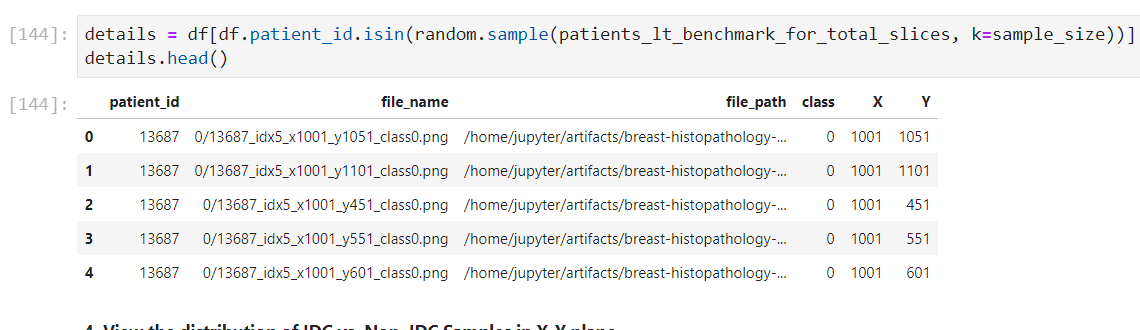


## Using ggplot in python

### Jitter plot

from plotnine import \*

For the following dataset



(ggplot(data=details)

+ aes(x='X', y='Y', color='class')

+ geom\_jitter()

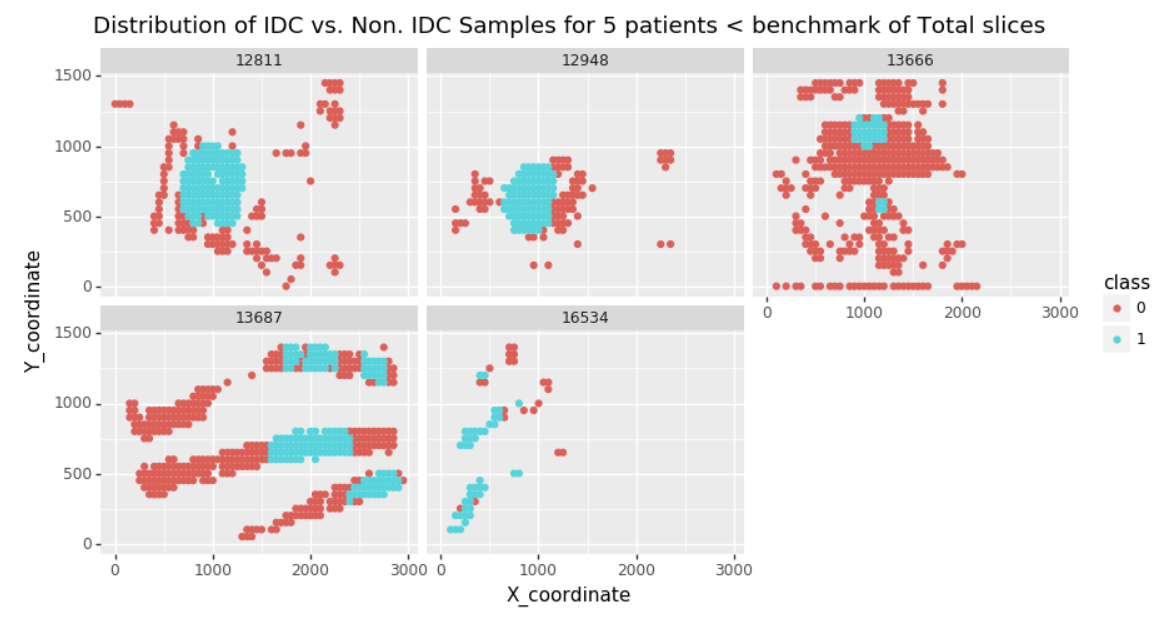
+ labs(x='X\_coordinate', y='Y\_coordinate')

+ facet\_wrap('patient\_id')

+ ggtitle('Distribution of IDC vs. Non. IDC Samples for {} patients < benchmark of Total slices'.format(sample\_size))

+ theme(figure\_size=(10,5))

)



### Histogram

(ggplot(data=df)

+ aes(x='original\_price',fill='search\_criteria')

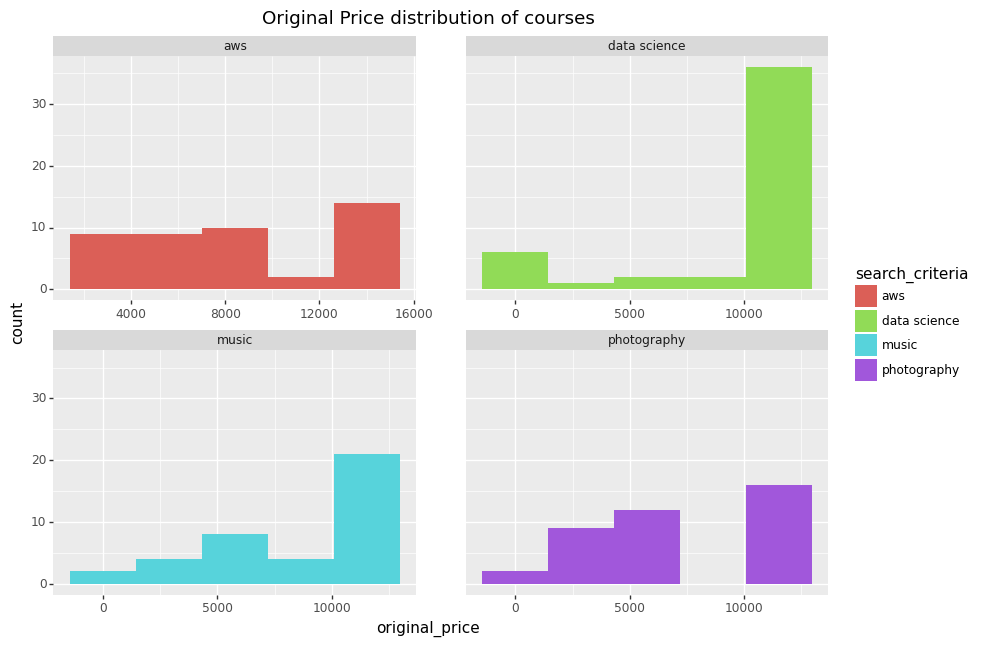
+ geom\_histogram()

+ facet\_wrap('search\_criteria',scales='free\_x')

+ ggtitle('Original Price distribution of courses')

+ theme(figure\_size=(10,7), panel\_spacing\_y=0.3, panel\_spacing\_x=0.5)

)



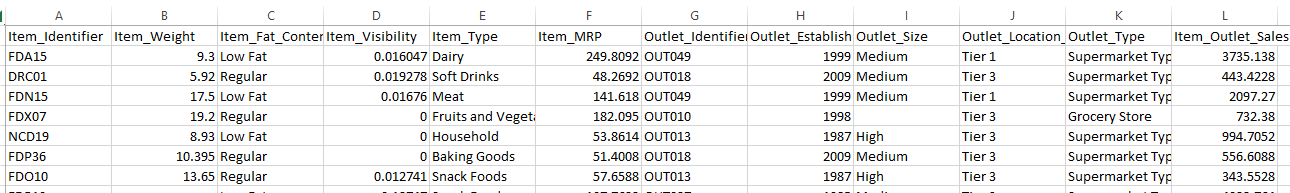
### Common options

1. Define graph colors. aes(x=<X axis column name>,y=<Y axis column name>,fill='search\_criteria')
2. Define type of graph: geom\_jitter(), geom\_histogram()
3. Facet wrapping: facet\_wrap(<discrete column name>,scales='free')
4. Giving chart title: ggtitle('Relationship of original price with duration')
5. Adjust figure size and spacing between each facets: theme(figure\_size=(10,7), panel\_spacing\_y=0.3, panel\_spacing\_x=0.5)
6. Rotate axis text: theme(axis\_text\_x=element\_text(size=7, rotation=90))
7. Giving labels other than default: labs(x='X\_coordinate', y='Y\_coordinate')
8. Put text in bar charts: geom\_text(aes(label='stat(count)'), stat='count', position='fill')
9. Put text in column charts: geom\_text(aes(label='y\_column'), position='identity',angle=90)

# R

## ggplot

Data:



Code:

ggplot(visualizations, aes(Item\_Visibility, Item\_MRP, color = Item\_Type)) +

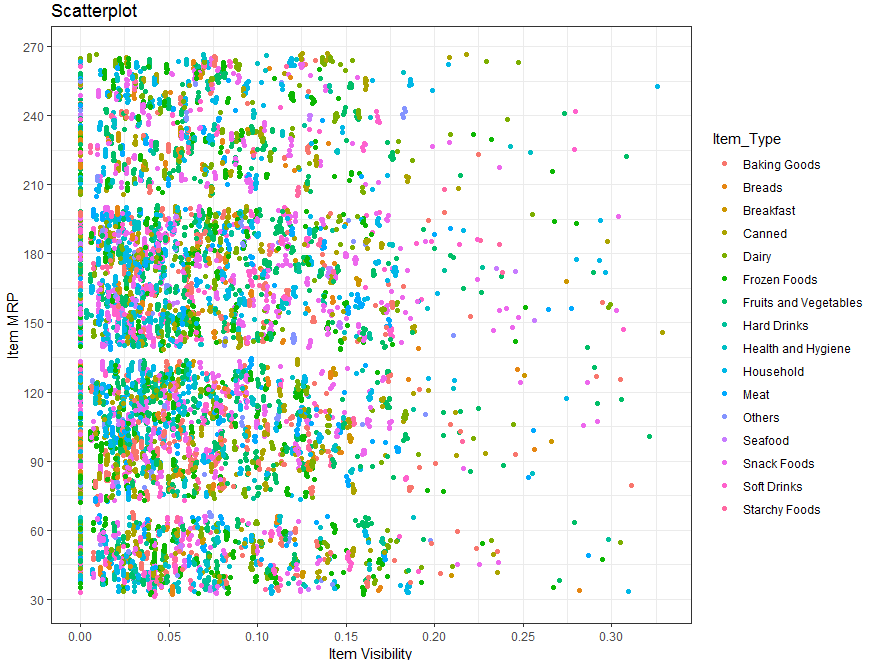
geom\_point() +

scale\_x\_continuous("Item Visibility", breaks = seq(0,0.35,0.05))+

scale\_y\_continuous("Item MRP", breaks = seq(0,270,by = 30))+

theme\_bw() + labs(title="Scatterplot")

Output:



### Facet wrapping

ggplot(visualizations, aes(Item\_Visibility, Item\_MRP)) + geom\_point(aes(color = Item\_Type)) +

scale\_x\_continuous("Item Visibility", breaks = seq(0,0.35,0.05))+

scale\_y\_continuous("Item MRP", breaks = seq(0,270,by = 30))+

theme\_bw() + labs(title="Scatterplot") + facet\_wrap( ~ Item\_Fat\_Content, scales=”free”)



### Multiple plots using ggplot

library(ggpubr)

# plot the curves

g1 <- ggplot(data=df\_pdf, aes(X, Y)) +

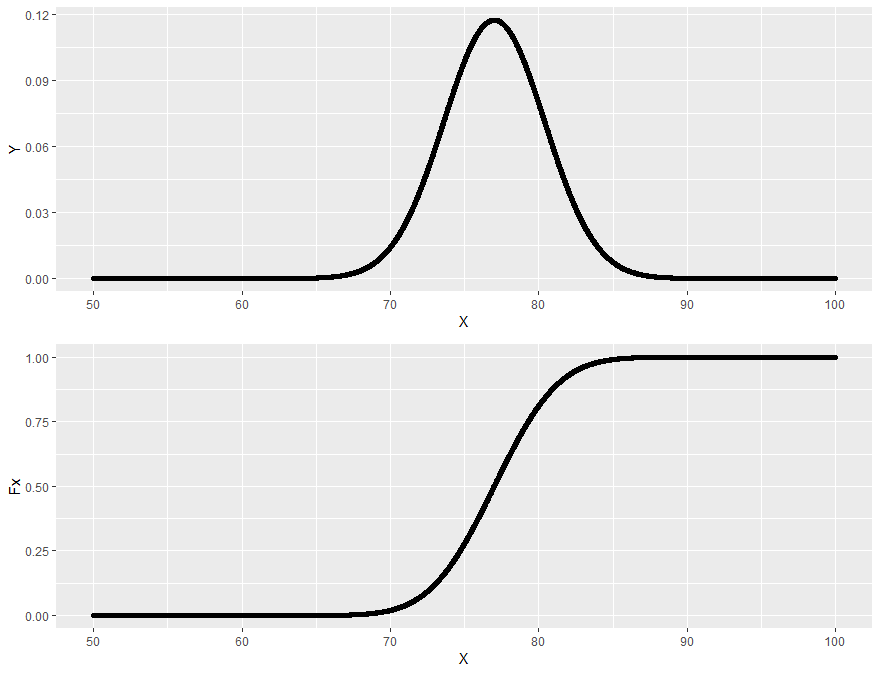
geom\_point()

print(g1)

g2 <- ggplot(data=df\_cdf, aes(X, Fx)) +

geom\_point()

ggarrange(g1,g2)



### Rotate X axis labels

g2 = ggplot(data=df) +

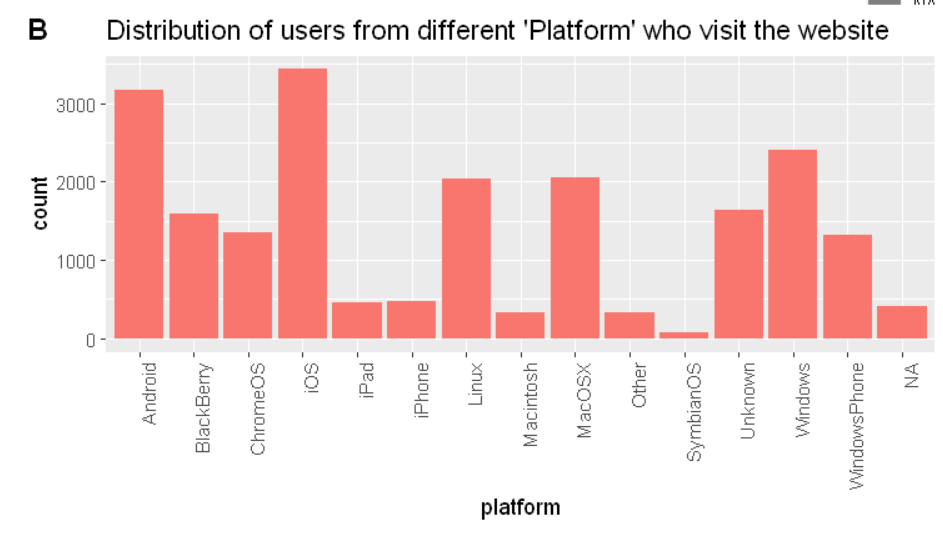
aes(platform,fill="red") +

geom\_bar() +

scale\_x\_discrete() +

theme(axis.text.x = element\_text(angle = 90, hjust = 1)) +

labs(title="Distribution of users from different 'Platform' who visit the website")



### Boxplots

## Plot area under the curve

Following are the features we can learn about:

1. Set graph sizes
2. Plot area under the curve with upper and lower x and y limits

## Given attributes

u = 77.0

sigma = 3.4

## Visualize the distribution

# Create a sequence of numbers between 50 and 100 incrementing by 0.01.

x <- seq(50, 100, by = .01)

# Choose the mean and standard deviation as sigma

#to give probability distribution at each point of x

y <- dnorm(x, mean = u, sd = sigma)

## probability that random variable is (c) between 81 and84

*## Let the random variable is in range [n1,n2]*

n1 = 81; n2= 84

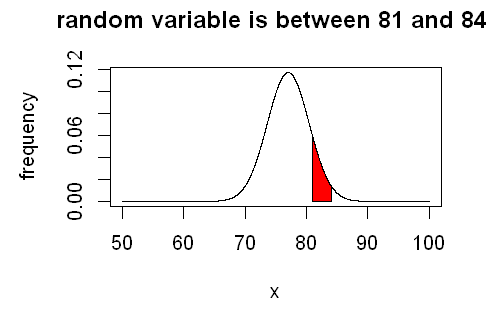
options(repr.plot.width=4, repr.plot.height=3)

plot(x,y, type='l',

main='random variable is between 81 and 84 ',

ylab='frequency')

polygon(x=c(n1,x[n1<=x & x<=n2],n2), y=c(0,y[n1<=x & x<=n2],0), col='red')



## Draw straight lines in a plot

no\_of\_trials = 15; no\_of\_success = 5

# Create a sequence of 15 numbers which denotes our maximum number of trials and each denotes number of successes

#which are incremented by 1.

x <- seq(0,no\_of\_trials,by = 1)

# Create the binomial distribution.size is the number of trials, prob= Probability of success

y <- dbinom(x=x, size=no\_of\_trials, prob=0.5)

# set the figure size

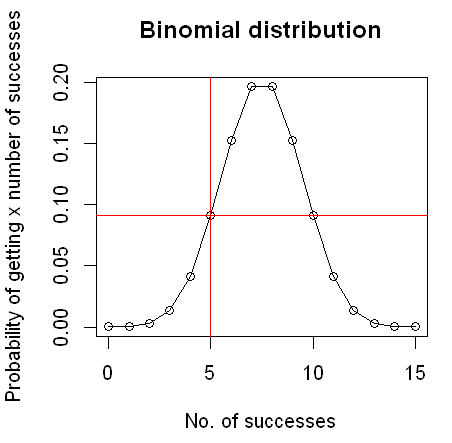
options(repr.plot.width=4, repr.plot.height=4)

plot(x,y, type='o',

main='Binomial distribution',

xlab='No. of successes',

ylab='Probability of getting x number of successes')



## Draw multiple lines in same graph with legends

options(repr.plot.width=5, repr.plot.height=5)

plot(x,y1,type="l",col="red", xlab='Amount in INR of claims filed', ylab='Probability of each agent')

lines(x,y2,col="blue")

legend(x=x[400000], y=y1[1], legend=c("Agent 1", "Agent 2"),

col=c("red", "blue"), lty=1:1, cex=0.8)

polygon(x=c(0,x[y2<=y1],x[y2>=y1][1]), y=c(0,y2[y2<=y1],0), col='blue')

