Note 14. Graphics

Introduction to Statistical Programming

Chanmin Kim

Department of Statistics
Sungkyunkwan University

2022 Spring

Creating Graphs

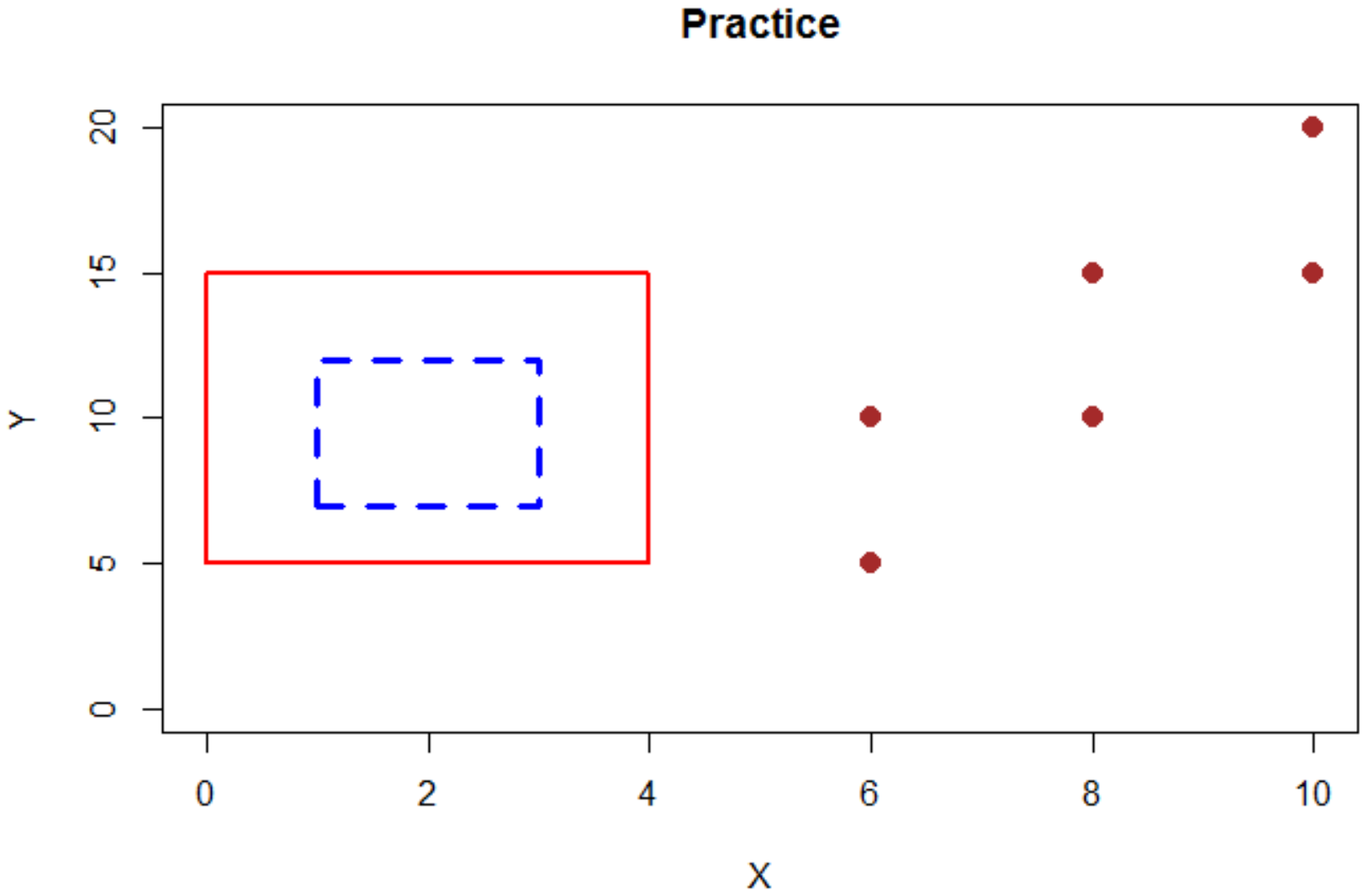
- `plot()`:
  - Basic function for creating graphs.
  - Generic function ⇒ Many different kinds of graphs.
  - Lines and points can be added to plots by `lines()` and `points()`, respectively.

```
> # Empty frame
> plot(c(0,10),c(0,20),type='n',xlab='X',ylab='Y',
+ main='Practice')

> # Add points
> # cex: size of points; pch: type of point
> points(c(6,8,10,6,8,10),c(5,10,15,10,15,20),col='brown',
+ cex=2,pch=20)

> # Add lines
> # col: color; lwd: width of lines
> lines(c(0,4,4,0,0),c(5,5,15,15,5),col='red',lwd=2)
> lines(c(1,3,3,1,1),c(7,7,12,12,7),col='blue',lwd=3,lty=2)
```

Creating Graphs



PLOT()

Arguments of plot()

Arguments	Description
main=' '	Title of plot
sub=' '	Subtitle of plot
xlab=' ', ylab=' '	Label of X & Y axes
ann=F	No labels for X & Y axes
tmag=2	Size of characters for title & labels.
axes=F	No X & Y axes
xlim=c(x1,x2), ylim=c(y1,y2)	upper and lower limits for X & Y axes

PLOT()

Graph type & line type in `plot()`

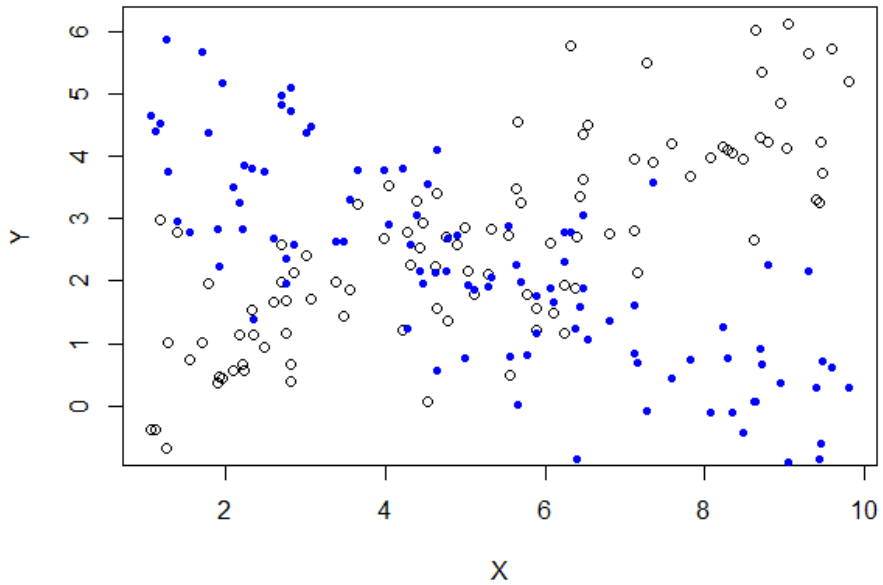
Graph Type	Description	Line Type	Description
`type='p'`	Points	`lty=0`	Blank
`type='l'`	Lines	`lty=1`	Solid (default)
`type='b'`	Both points and lines	`lty=2`	Dashed
`type='c'`	Lines part alone of 'b'	`lty=3`	Dotted
`type='o'`	'b' but overplotted	`lty=4`	Dot dash
`type='h'`	Histogram-like, vertical lines	`lty=5`	Long dash
`type='n'`	No plotting	`lty=6`	Two dash

- `col=' '`: Color.
- `pch=0`: Symbol of dots.
- `lwd=1`: Width of lines.
- `cex=1`: Size of dots.

Scatter Plot

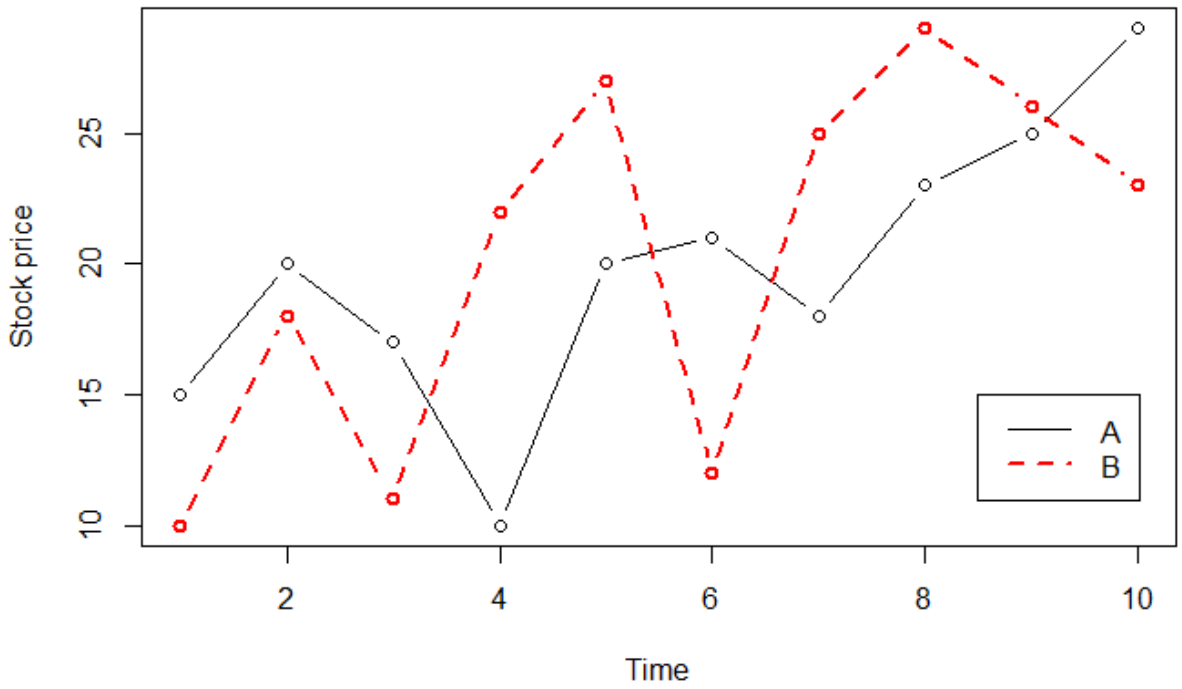
- Scatter plot: `plot()` function with two vectors for x & y axes.

```
> n = 100
> x = runif(n,1,10)
> y1 = 0.5*x + rnorm(n)
> y2 = 5 - 0.5*x + rnorm(n)
> plot(x,y1,xlab='X',ylab='Y')
> points(x,y2,col='blue',pch=20)
```



Line Graph

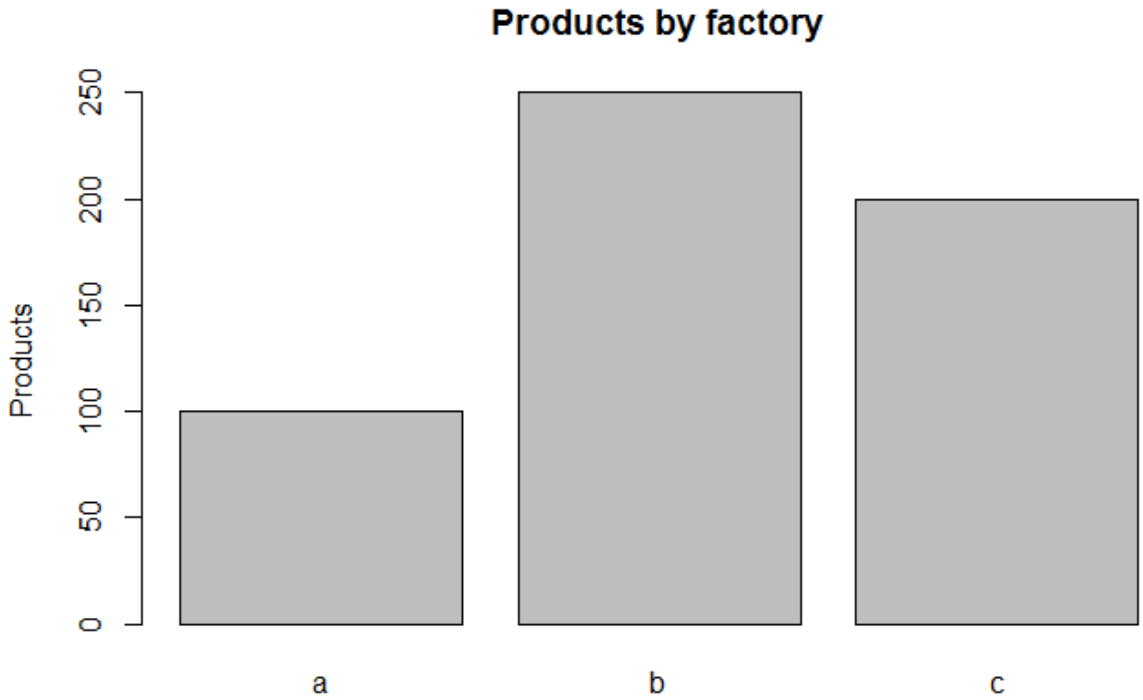
```
> Time <- 1:10
> stock1 <- c(15,20,17,10,20,21,18,23,25,29)
> stock2 <- c(10,18,11,22,27,12,25,29,26,23)
> plot(Time,stock1,type='b',xlab='Time',ylab='Stock price')
> lines(Time,stock2,type='b',col='red',lwd=2,lty=2)
> legend(8.5,15,c('A','B'),col=c('black','red'),lty=c(1,2),lwd=c(1,2))
```



Bar Chart

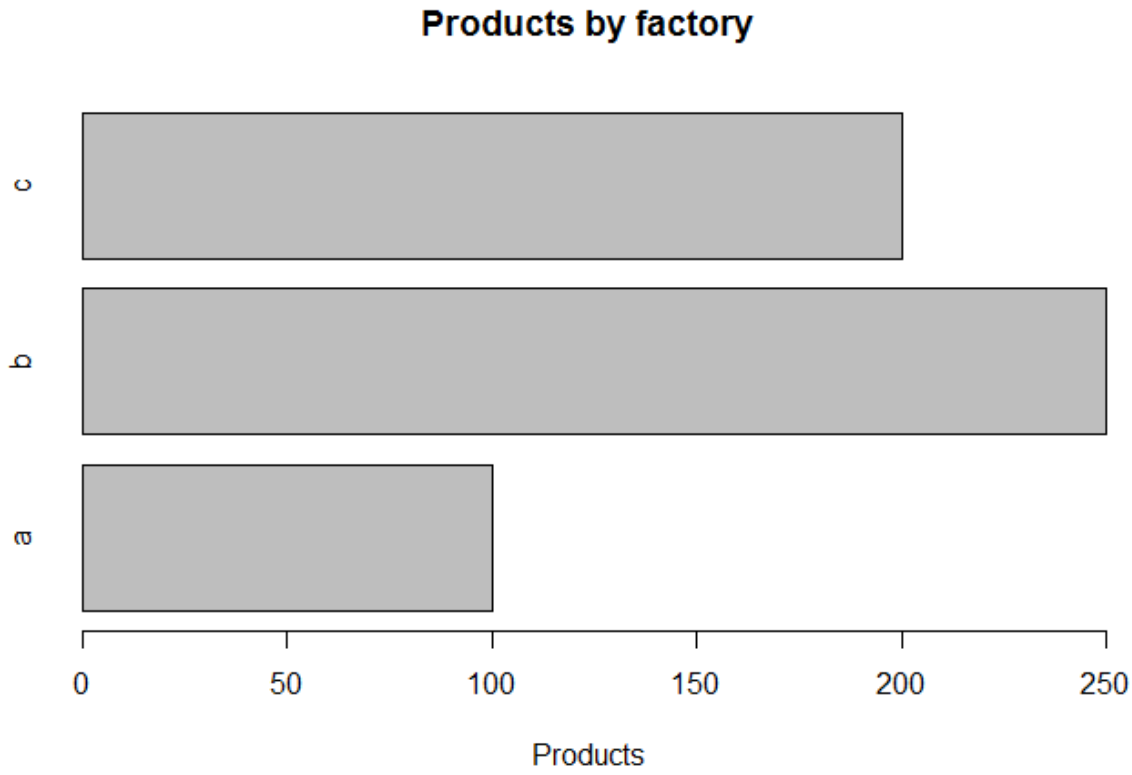
- Bar chart: `barplot()`.

```
> x <- c(100,250,200)
> barplot(x,names.arg=c('a','b','c'),ylab='Products',
+ main='Products by factory')
```



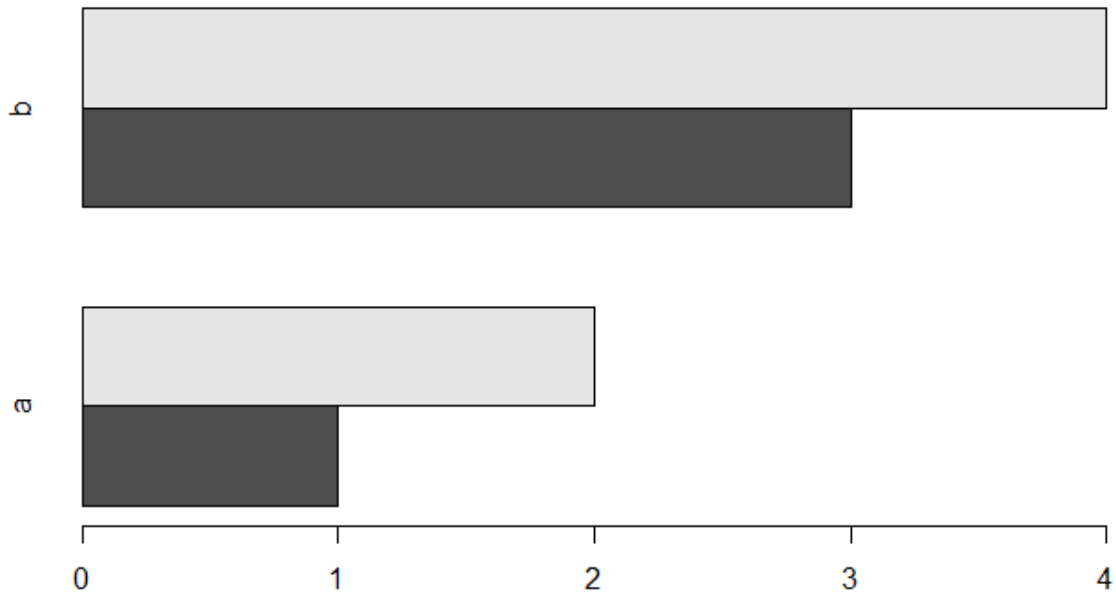
Bar Chart

```
> barplot(x,horiz=T,names.arg=c('a','b','c'),
+ xlab='Products',main='Products by factory')
```



Bar Chart

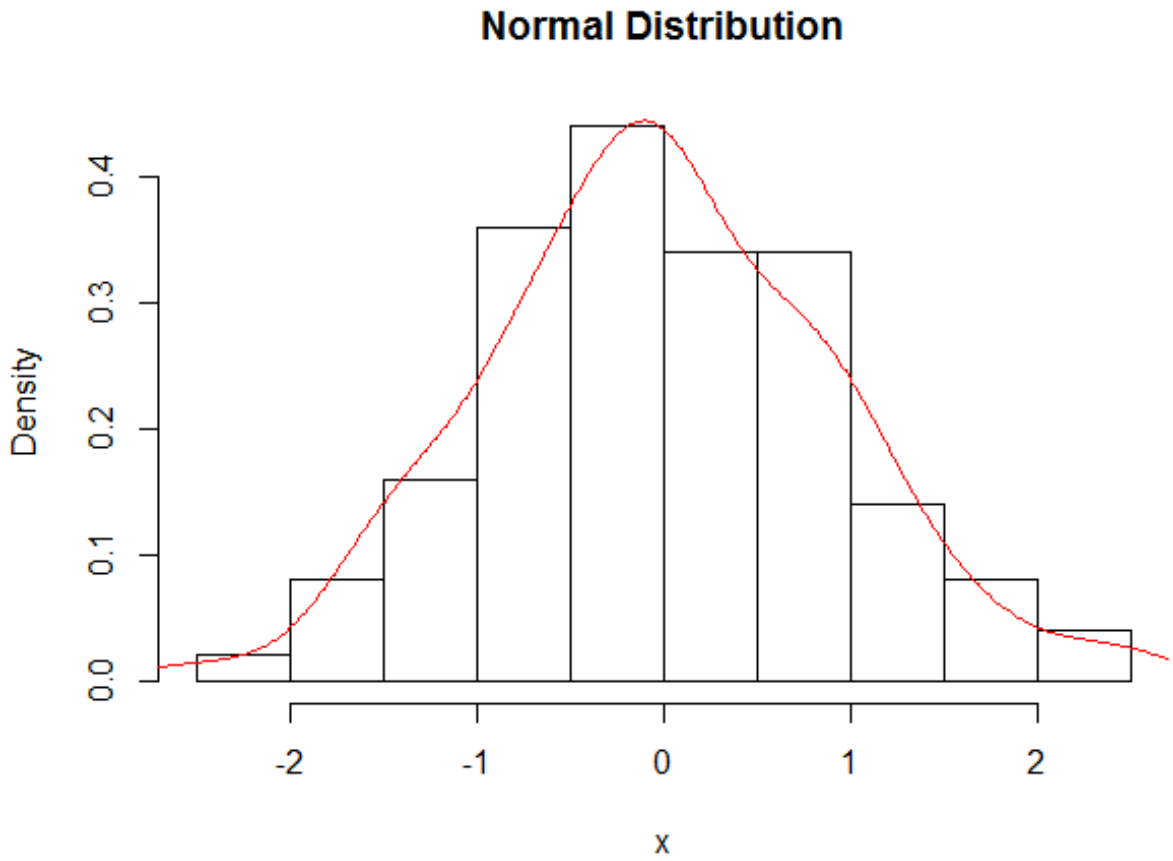
```
> x <- matrix(c(1,2,3,4),2,2)
> barplot(x,beside=T,horiz=T,names=c('a','b'))
```



Histogram

- `hist()`:

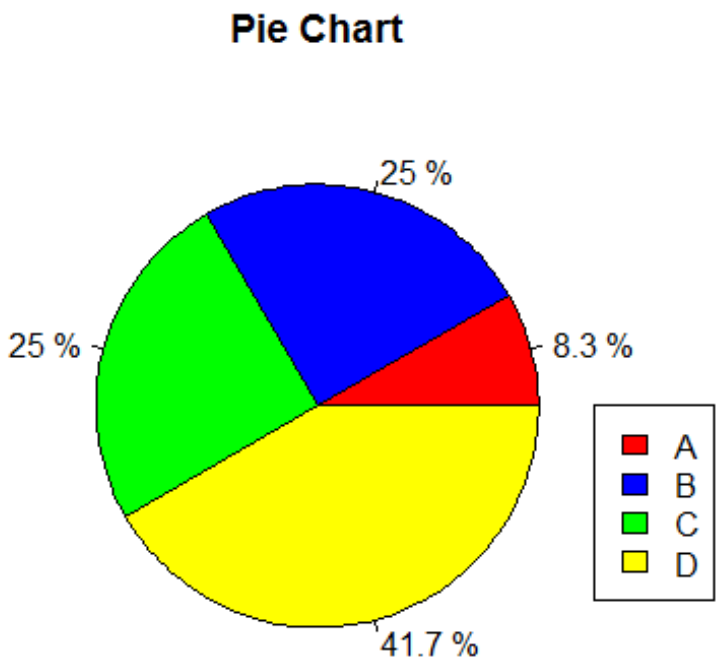
```
> x <- rnorm(100)
> hist(x,main='Normal Distribution',freq=F)
> lines(density(x),col='red')
```



Pie Chart

- pie():

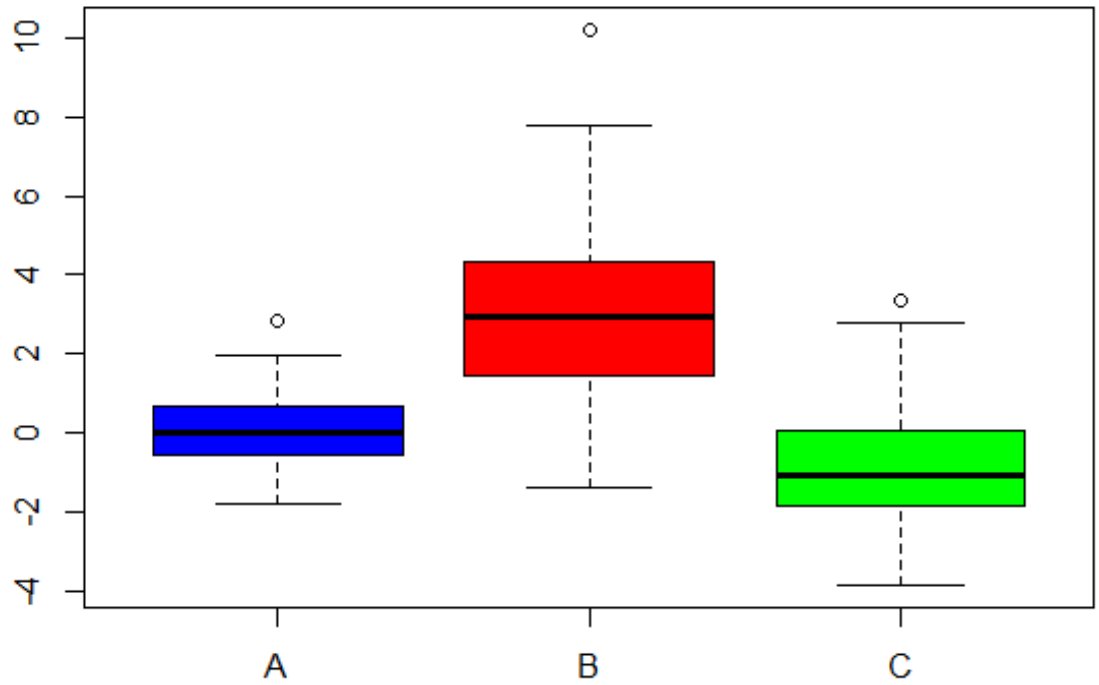
```
> x <- c(100,300,300,500)
> pct <- paste(round(x/sum(x)*100,1),'%')
> pie(x,labels=pct,col=c('red','blue','green','yellow'),
+ main='Pie Chart')
> legend(1,0,c('A','B','C','D'),
+ fill=c('red','blue','green','yellow'))
```



Box Plot

- `boxplot()`:

```
> x <- rnorm(100)
> y <- rnorm(100,mean=3,sd=2)
> z <- rnorm(100,mean=-1,sd=1.5)
> boxplot(x,y,z,names=c('A','B','C'),col=c('blue','red','green'))
```



Additional Issues

- `legend()`: Legend of graph for multiple classes.
- `text()`: Add characters to plots.
- `par(mfrow=c(p,q))`: If you perform this commend, you can draw multiple $(p \times q)$ plots on a single frame.

```
x <- cbind(rnorm(100),rnorm(100),rnorm(100))
par(mfrow=c(1,3))
for (i in 1:3) hist(x[,i])
```