Data Distribution and Graphic Representation Using R

Dr. Neena Mital
Department of Statistics
RAM LAL ANAND COLLEGE

Frequency tables and graphs

- **Contingency Tables**

>table(trees$Height)

>table(trees)

> table(trees$Height,trees$Girth)

- **Stem and leaf plot (A frequency plot)**
- **It records a number(the leaf) for each point(stem)**

>x<-c(2,3,16,23,14,12,4,13,2,0,0,0,6,28,31,14,4,8,2,5)

>stem(x) (data is in a sorted form looks like histogram and shows skewness)

>table(x); stem(trees, scale=2); hist(x);

In case of large numbers decimal point is mentioned on top

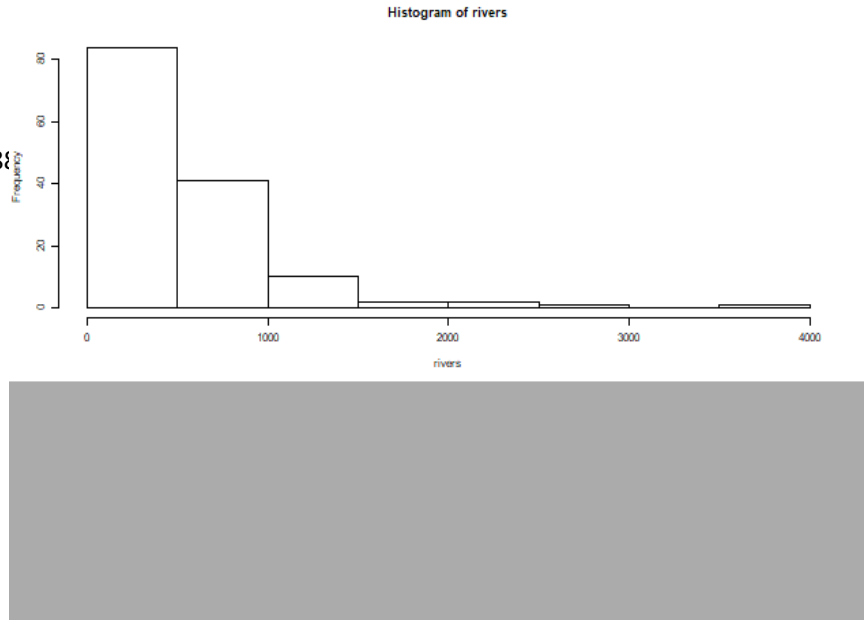
```
> data4
 [1] 23.0 17.0 12.5 11.0 17.0 12.0 14.5  9.0 11.0  9.0 12.5 14.5 17.0  8.0 21.0
> stem(data4)

  The decimal point is 1 digit(s) to the right of the |

  0 | 899
  1 | 11233
  1 | 55777
  2 | 13
```

```
> grass
   rich graze
1    12   mow
2    15   mow
3    17   mow
4    11   mow
5    15   mow
6     8 unmow
7     9 unmow
8     7 unmow
9     9 unmow
```

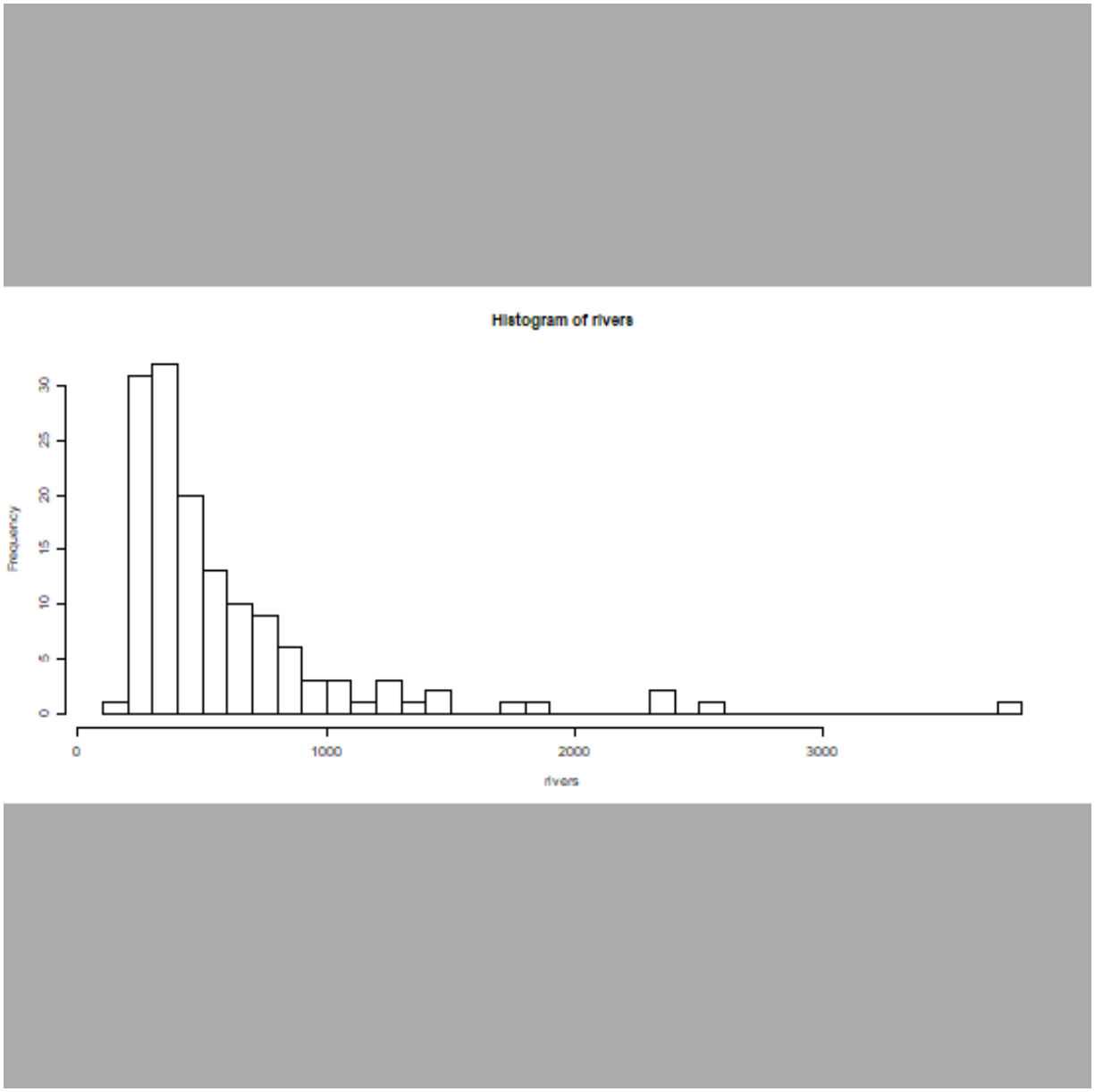
select a single treatment from the data with a conditional statement:

```
> with(grass, stem(rich[graze == 'mow']))

  The decimal point is at the |

  10 | 0
  12 | 0
  14 | 00
  16 | 0
```

```
> sort(rivers)
 [1]  135  202  210  210  215  217  230  230  233  237  246  250  250  250  255  259  260  260  265  268  270  276  280  280  280  281  286  290  291
300  300  300
 [33]  301  306  310  310  314  315  320  325  327  329  330  332  336  338  340  350  350  350  350  352  360  360  360  360  375  377  380  380  383
390  390  392
 [65]  407  410  411  420  420  424  425  430  431  435  444  445  450  460  460  4
570  600  600
 [97]  600  605  610  618  620  625  630  652  671  680  696  710  720  720  730  7
1038 1054 1100
[129] 1171 1205 1243 1270 1306 1450 1459 1770 1885 2315 2348 2533 3710
> stem(rivers)

  The decimal point is 2 digit(s) to the right of the |

   0 | 4
   2 | 011223334555566667778888899900001111223333344455555666688888
   4 | 11122233344556679001233344567
   6 | 000112233578012234468
   8 | 045790018
  10 | 04507
  12 | 1471
  14 | 56
  16 | 7
  18 | 9
  20 |
  22 | 25
  24 | 3
  26 |
  28 |
  30 |
  32 |
  34 |
  36 | 1
```



```
> stem(rivers,scale=2)
  The decimal point is 2 digit(s) to the right of the |
   1 | 4
   2 | 011223334555566667778888999
   3 | 000011112233333444555556666888889999
   4 | 111222333445566779
   5 | 001233344567
   6 | 000112233578
   7 | 012234468
   8 | 04579
   9 | 0018
  10 | 045
  11 | 07
  12 | 147
  13 | 1
  14 | 56
  15 |
  16 |
  17 | 7
  18 | 9
  19 |
  20 |
  21 |
  22 |
  23 | 25
  24 |
  25 | 3
  26 |
  27 |
  28 |
  29 |
  30 |
  31 |
  32 |
  33 |
  34 |
  35 |
  36 |
  37 | 1
```



Histogram: To study the distribution of continuous data

>hist(**x**, breaks = bin, freq = T, probability = !freq, right = T, density = NULL, angle = 45, col = NULL, border = NULL, main, xlim ,ylim , xlab, ylab, axes = T, plot = T, labels = F)

bin<-seq(initial, final, by=step) (breaks can be a no./seq/col e.g -3:3)

x: a vector of values

breaks: a vector giving the breakpoints between histogram cells,/a single number giving the number of cells

freq: logical; if TRUE, frequencies; if FALSE, probability densities are plotted (so that the histogram has a total area of one). (Defaults T good iff breaks are equidistant (and prob. is not given).

right :logical; if TRUE(default), the histogram cells are right-closed (left open) intervals.

density: 20(shading lines per inch; NULL no shading)

angle: in degrees the slope of shading lines(counter-clockwise).

col: "green" (NULL yields unfilled bars)

border: "darkgreen" (the color of the border around the bars)

main, xlab, ylab: to give title to main , x and y axis

xlim, ylim: the range of x and y values (xlim is not used to define the histogram (breaks), but only for plotting (when plot = TRUE). ylim<-c(start, end)

axes: logical. If TRUE (default), axes are draw if the plot is drawn.

plot: logical. If TRUE (default), a histogram is plotted, otherwise a list of breaks and counts is returned.

labels: logical. Additionally draw labels(freq on top of bars, if not FALSE(default)); char string or col. of char string may be added.

Continuous Frequency Table

>bin<- seq(initial, start, by=step)

>freq<- cut(x, breaks=bin)

- cut divides the range of x into intervals and codes the values in x according to which interval they fall.

>table(freq)

Gives the freq table with class intervals as named vector

- **par(mfrow=c(a,b))** gives **a** rows with **b** plots on each row

>**par(mfrow=c(1,2))**

Prac:1- Stem and leaf, freq. table and histogram

Aim: To compute A.M and S.D for the raw data. Plot a stem and leaf plot. Construct a frequency distribution with class intervals of width 5 and 10 and plot the histogram for each. Also compute A.M and S.D for each grouped data and compare the obtained statistics with raw data.

Exp. The following data in(1000 $) represents the net annual income of tax payers.

Data:

47, 55, 18, 24, 27, 41, 50, 38, 33, 29, 15, 77, 64, 22, 19, 35, 39, 41, 67, 55, 114, 77, 80, 34, 41, 48, 60, 30, 22, 28, 56, 64, 88, 104, 114, 39, 35, 18, 21, 30, 84, 55, 26, 105, 62, 30, 17, 23, 31, 28, 40, 57, 38, 29, 19, 46, 40, 49, 72, 70, 37, 39, 18, 22, 29, 52, 94, 86, 23, 26

```
>stem(x)
The decimal point is 1 digit(s) to the right of the |
1 | 5788899
2 | 122233466788999
3 | 00013455788999
4 | 001116789
5 | 0255567
6 | 02447
7 | 0277
8 | 0468
9 | 4
10|45
11|44
```

```
>bin<-seq(min(x)-5, max(x)+5, 5)
> hist(x,bin, freq=T, probability=F, right=T, density=20, col="black", main="Net annual Income" ,     xlab="Income", ylab="freq",ylim=c(0,12),axes=T)

>freq<- cut(x,bin)
>f5<-table(freq)
>ll<-bin[1:length(bin)-1]
> ll [1] 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110
>ul<-bin[2:length(bin)]
>ul [1] 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115
>x5<-(ll+ul)/2
>mean<- sum(f5*x5)/sum(f5)
mean [1] 45.64286
```

Scatter plots and line graphs

>plot(**x,y**, type="b",col="blue",lwd=1,lty=4,pch=5, main="My plot", xlab="x axis", ylab="y axis“, sub= “ “, cex= 4, xlim= c(), ylim=())

lwd: line width; lty: line type;

pch: pointer character; cex: char expansion (also for cex.main, cex.sub)

>x<- seq(0,10,0.2);y<-sqrt(x)

>lines(x,y)

>lines(x, 0.5*x)

>abline(h=1,v=4, col=c("darkred","green"), lty=c(1,4),lwd=c(4,6))

>text(8,2,”text”)

>grid(col=“red”)

>x<-rnorm(50); y<-rnorm(50)

> matrix(x, byrow=T, nrow=10) (to create a matrix of x)

>plot(x,y)

>lines(sort(x), sort(y), col=“blue”)

Plot level of the graph

High Level Plot Function

>plot() scatter plot

>hist(x) Histogram

>boxplot(x)

>pie(x)

>dotchart(), barplot()

>qqplot(),

>pairs(),

>curve() , symbols()

Low Level Plot Function

>lines()

>grid() text()

>legend(); title(); mtext()
extra text on the margins of plot

>abline()

>points(); axis(); axis.Date()

>box(), rect(),

>arrows(),

>segments()

Prac 2: Histogram and frequency polygon unequal classes

Aim: To construct a histogram of raw data with unequal classes and superimpose a frequency polygon.

Exp. Use the data on “rivers” available in R environment to construct a histogram of unequal classes and a frequency polygon.

```
>summary(rivers)
Min.    1st Qu.   Median   Mean    3rd Qu.   Max.
135.0   310.0     425.0    591.2   680.0     3710.0
>bin1<-seq(135,310,(310-135)/5)     (similarly define 5 bins for each quartile
>bin1                                                                range)
[1] 135 170 205 240 275 310
>hist(rivers, c(bin1,...),....)
>h<-hist(rivers)
>h
$breaks [1] 0 500 1000 1500 2000 2500 3000 3500 4000
$counts [1] 84 41 10 2 2 1 0 1
$mids [1] 250 750 1250 1750 2250 2750 3250 3750
> lines(c(0,h$mids,max(rivers)),c(0,h$counts,0))
```

Prac 3: Histogram and frequency polygon for grouped data

Aim: To construct a histogram of grouped data and superimpose a frequency polygon.

Exp.: Construct a histogram of grouped data and superimpose a frequency polygon on the histogram for the given frequency table:-

Data:

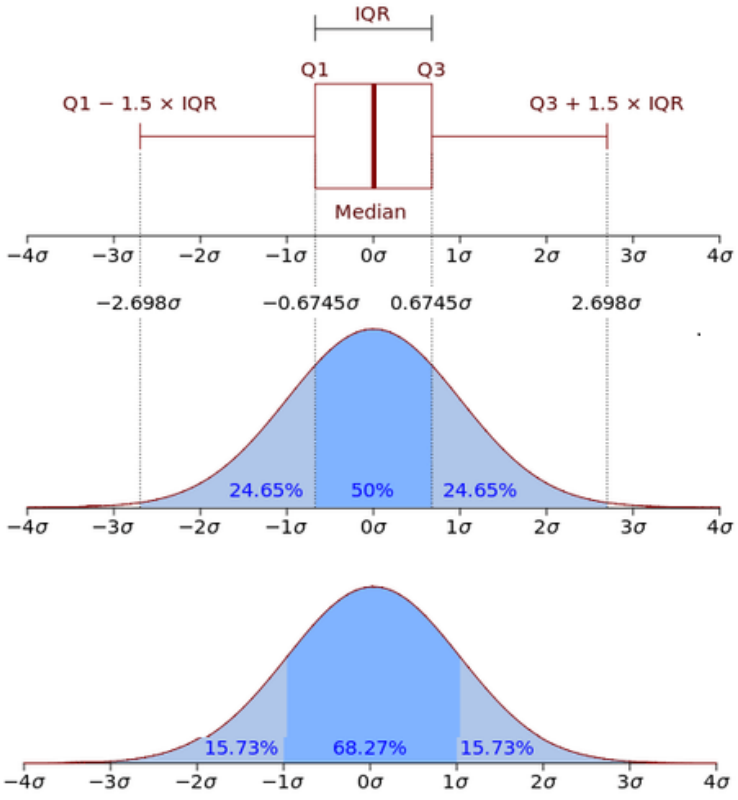
C.I.	0-10	10-20	20-30	30-40	40-50	50-60	60-70
Freq	4	8	11	15	12	6	3

```
>ll<-seq(0,60,10); ul<-seq(10,70,10); mid<-(ll+ul)/2
>f<-c(4,8,…)
>X<-rep(mid,f)
>h<-hist(X,…)
>lines(c(),c())
```

Box and Whisker Plot

- **A useful tool for studying data. It shows the median, quartiles and possible outliers.**
- **Five number summary of a very large univariate data sets**
- **Box is drawn from Q1 to Q3(IQR) middle 50% of data**

>boxplot(x, medcol="red",col="green", boxlty=0, whisklty=1 xlab=" ", staplewd=4, outpch=8, Outcex=3, boxwex=0.4, notch=F)



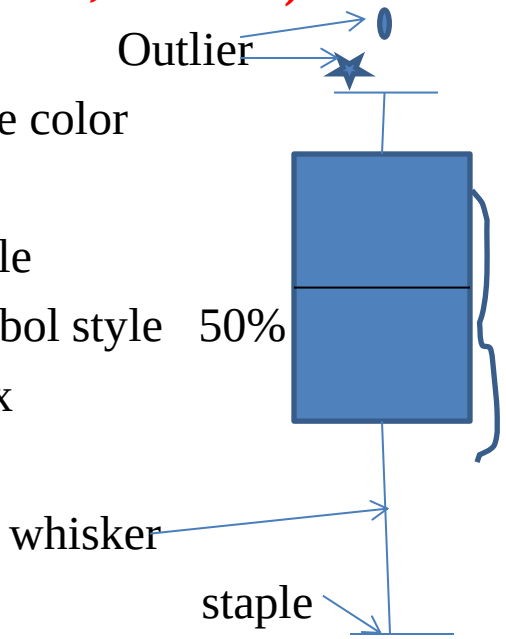
medcol: median line color

col: box color

boxlty: box line style

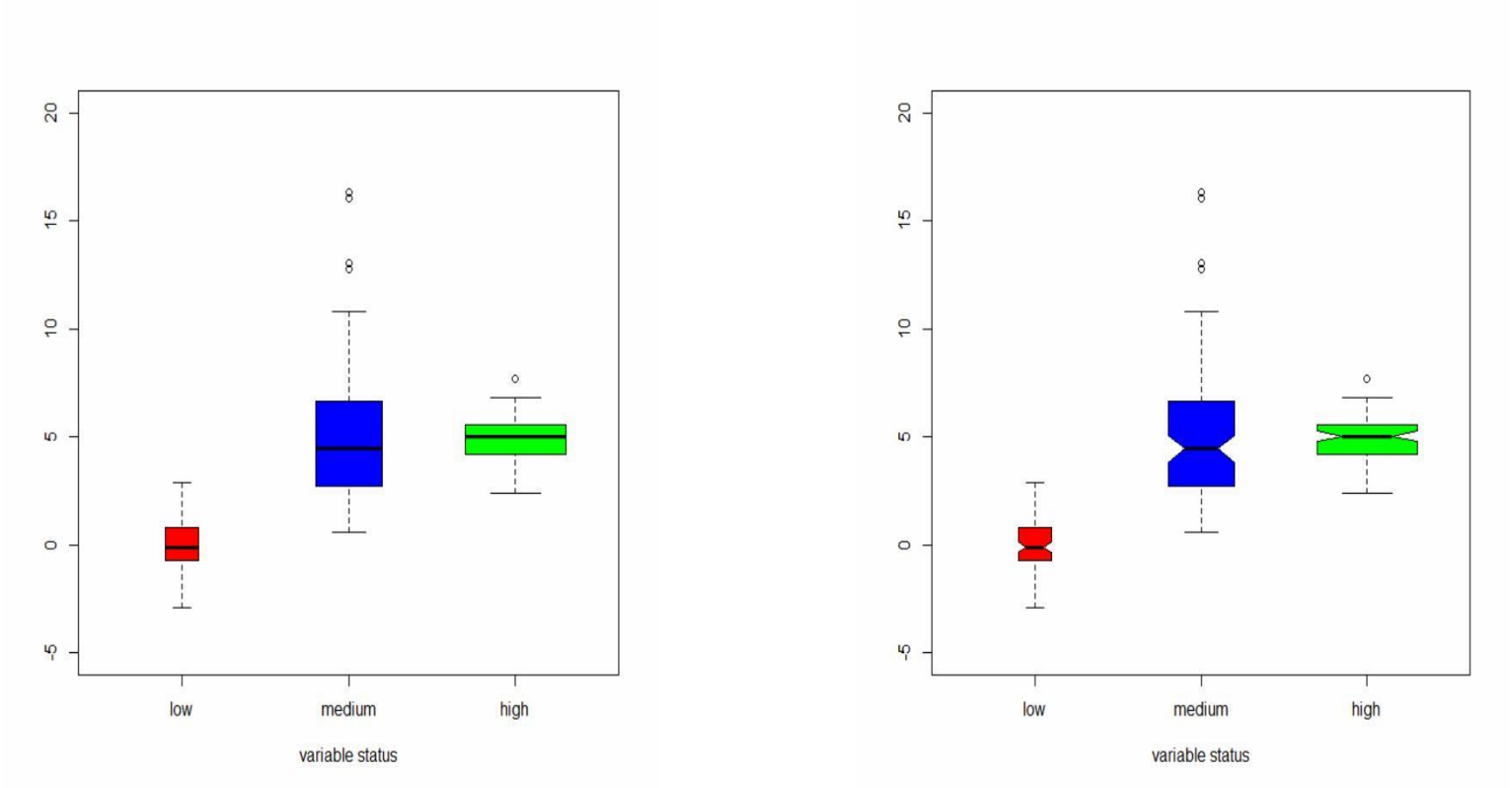
outpch: outlier symbol style

boxwex: size of box



- Longer whisker shows skewness on that side

```
>vblname<-c("low","medium", "high")
>boxplot(u1,u2,u3,names=vblname,boxwex=c(.2,.4,.6),col=c("red","blue","gr
een"), ylim=c(-5, 20),xlab="variable status", notch = TRUE)
```



Prac 4: Construction and interpretation of Box and whisker plot

Aim: Construct a boxplot for given data and for random variates.

Exp. Construct a boxplot for given data and for 100 random variates following standard Normal, Chi-square with 5df and Normal with specified mean=10 and sd=2. Also give summary and IQR for each data set and interpret the boxplot.

```
>a<-rnorm(100)
>b<-rchisq(100,df=5)
>c<-rnorm(100,mean=10,sd=2)
>IQR(a)
```

Pie chart: To display proportional data in angular form

>pie(x, labels=names(x),main=” “,cex.main=2,lty=c(1:length(x)),density=c( ), angle=c( ),col=c( ))

- labels: gives labels for slices
- clockwise: logical, indicating start of 1st slice
- init.angle: degrees starting angle of slice
- angle: slope of shading lines in slice
- col: >c(“gray40”, ”gray50”, …)
- borderlty:

>perx <-(x/sum)*100

>labx<-paste(names(x),perx)

Prac 5: To construct Pie chart

Aim: To construct pie charts and comparison of charts.

Exp.: The actual outlay on the public health sector in the 7th and 8th five year plan of India is shown below:

Heads	7thplan	8thplan
Agriculture and Allied services	30317	32025
Irrigation and flood control	16719	23771
Industries and minerals	30052	46889
Transport and communication	38804	80798
Energy	63615	122356
Services	39576	84088
Miscellaneous	3186	44173

Draw pie chart to show the relative importance attached to various heads in each plan and compare the expenditure done over the plans.

Prac 6: To construct Ogives and locate median graphically

Aim:To construct less than and more than Ogives and locate median graphically.

Experiment: Consider the following frequency distribution. Construct the less than and more than Ogives. Also compute the median of the grouped frequency distribution and compare it with median located graphically on the plot.

Data:

C.I	600-700	700-800	800-900	900-1000	1000-1100
Freq.	85	77	124	78	36

```
>f<- c(0,85,77,124,78,36,0); n<- length(f)
>less <- cumsum(f)
>more<- c()
>for(i in n:1)   { more[i] = sum(f[n:i])}
>ll<- seq(500,1100,100); ul<- seq(600,1200,100)
>plot(c(ul,ll), c(less,more), lines(c(ul,ll), c(less,more),ylim =c(0,500)))
>med<- locator(n=1,type="o");  med                    (type="o" / "l"  points are joined by line
$x [1] 830.3238                $y [1] 197.3357                "p"  only points
>abline(v=med$x)
```

Plotting functions and distributions

>curve(expr = sin, from = 0, to = 6 * pi) (101 points)

>curve(xˆ2 - 10 * x, from = 1, to = 10)

>data2 <- c(3, 5, 7, 5, 3, 2, 6, 8, 5, 6, 9, 4, 5, 7, 3, 4)

>plot(data2)

>hist(data2)

>plot(density(data2)) Individual density

>hist(data2,prob=T) Interval density

>lines(density(data2)) added to the hist

>hist(rnorm(100),prob=T)

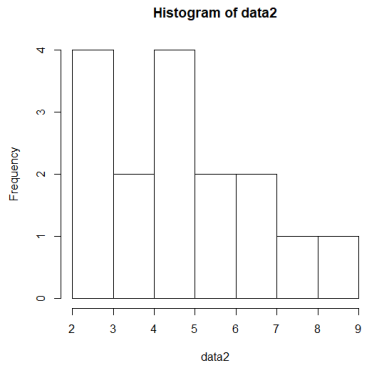
>curve(dnorm(x), add=T)

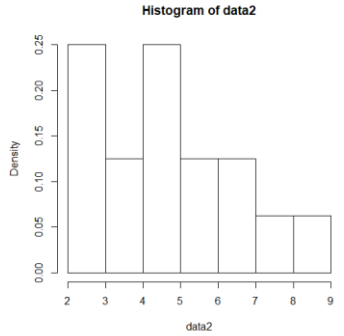
>lines(density(rnorm(100)),lty=1)

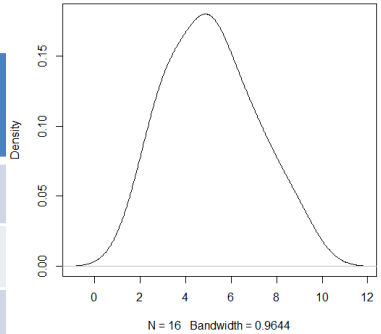
lines(density()) commands with
draw an idealized distribution
and see how your sample matches up.

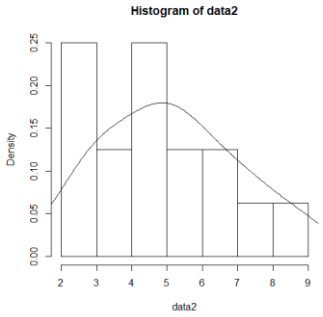
lty = “dotted” also works

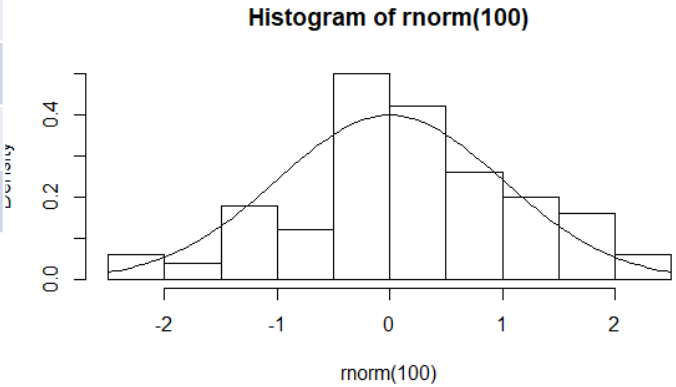
Value	Label
0	blank
1	Solid (default)
2	dashed
3	dotted
4	dotdash
5	longdash
6	twodash











Prac 7 Plotting of probability densities on a histogram

Exp: (a)Generate a random standard normal vector of size 100. Plot a frequency histogram

(b) Generate a random standard normal vector of size 100. Plot a density histogram and also plot a probability density function on it with mean and sd of the generated r.v vector.

>u<-rnorm(100) set.seed(12345) will generate same random variates every time

>hist(u,density=20,breaks=20,prob=T,xlab="X-variable", col="red",ylim=c(0,0.7),main="Normal curve with mean and sd of the generated r.v over histogram")

>m<-mean(u)

> std<-sd(u)

>curve(dnorm(x,m,std),col="darkblue",lwd=2,add=T)

(Perfect shape as based on pdf)

>data.norm<-rnorm(100,m,std)

>lines(density(data.norm))