PAGE NO.
DATE :

Module 5

Graphs and charts

Bar plot

R uses the function barplot() to create bar charts. A bar chart can represent data in rectangular bars of with length of the bar proportional to the value of the variable. If we supply a vector the plot will have bars with their height equal to the elements in the vector.

arguments used in barplots

1) height : a vector or matrix defining the bars which make up the plot. If height is a vector, the plot will be a series of rectangular bars. If height is a matrix and beside = false then each bar of the plot corresponds to the column height, If beside = true each column are juxtaposed to the bar tracked.

PAGE NO.
DATE :

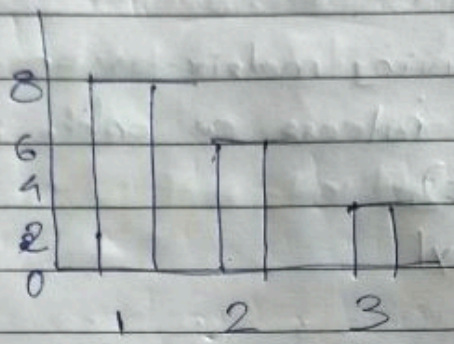
-) width : optional vector for bars widths
space : used to define the space between the bars can be one number or unique numbers as a vector.
) names.arg : used to name the individual bars if omitted the bars will names from attribute of height.
) legend.text : used to construct a legend of text for the barplot (sub arguments are
) las : which can be used to change the orientation of labels in the barplots). las = 1 will change vertical labels orientation las = 2 will change horizontal labels orientation las = 3 will change both labels).

Can be accepted in the height of bars also

Simple bar plot

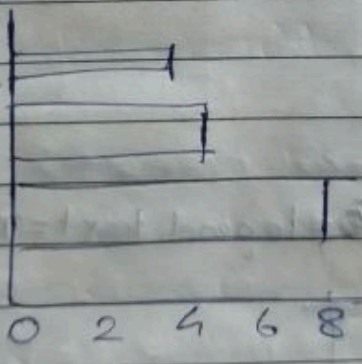
```
x = c(1,1,1,1,1,1,2,2,2,2,3,3,3,1,1,2,3,3)
y = table(x)
barplot(y)
```

output



II) horizontal box plot

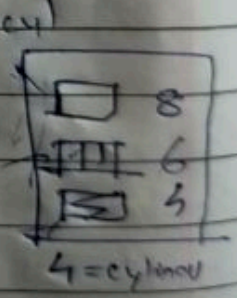
When horiz = true box plots orientation will change.

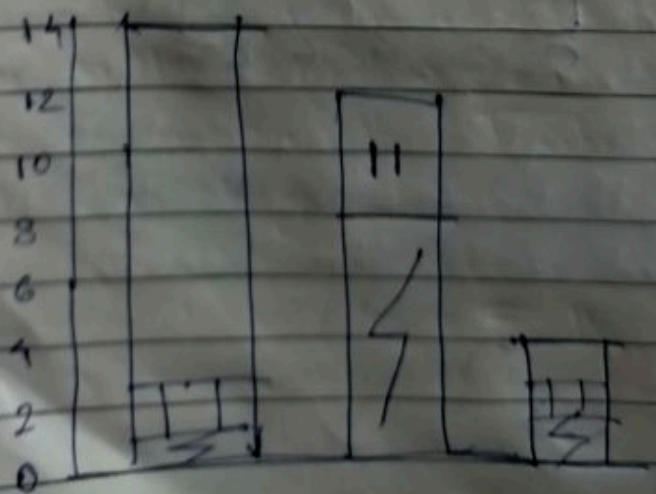


PAGE NO.
DATE :

III Stacked bar plot

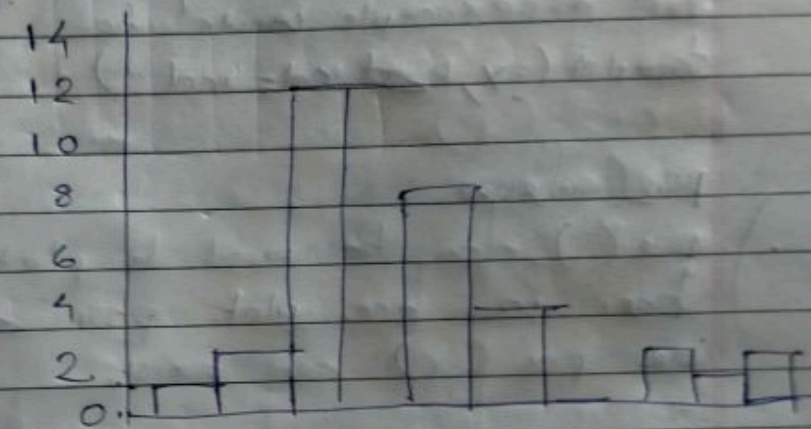
1) Created by giving matrix as an input.
Let take mtcars as data input. by
data(mtcars).
>mtcars$cyl
>mtcars$gear
>table(mtcars$cyl)
>table(mtcars$gear)
>Y=table(mtcars$cyl, mtcars$gear)
>Y
  3 4 5
4 1 8 2
6 2 4 1
8 12 0 2
>barplot(Y, legend.text=T)





IV grouped bar plot

- when you add beside = T the bar plot would change into grouped barplots.



- when you add horiz = T, the grouped bar plot would turn into horizontal grouped bar plot
- we will now refer to using y as our bar plot density : an argument that can be used to give shading lines in lines per inch, each increasing value will increase the shading intensity.
- angle : would change the slope of shading lines the values can be various.

PAGE NO.
DATE :

col : a vector of colors to give colours to each individual bar plot if needed the colors can have values ranging from and an example would be
barplot(y, col = "red")

par(mfrow
par() can be used to plot multiple bars in one plot. par(x, y..) or par (x) can exist.

different colors are given by using a vector

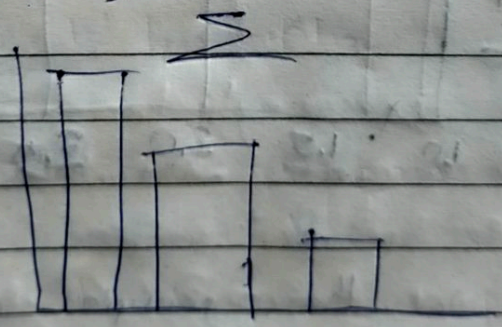
when
barplot(y, col = rainbow(x,) where x can be any number, the colors can randomly come in the bars.
adding to the rainbow S = i where i can be any number the intensity of the bars can change.

if we add borders = F the borders of the bar would not be present

1) main, sub: main is used to give overall title for the plot while sub is used to give a sub title at the bottom of the figure. main title will appear at the top.

2) xlab = labels for x axis, ylab = for y axis.
xlim, ylim : can be used to extend each axis to a preferred length.

barplot (y, main = expression(sumof)
then sigma (Σ) will appear at the top of the figure.



PAGE NO.
DATE :

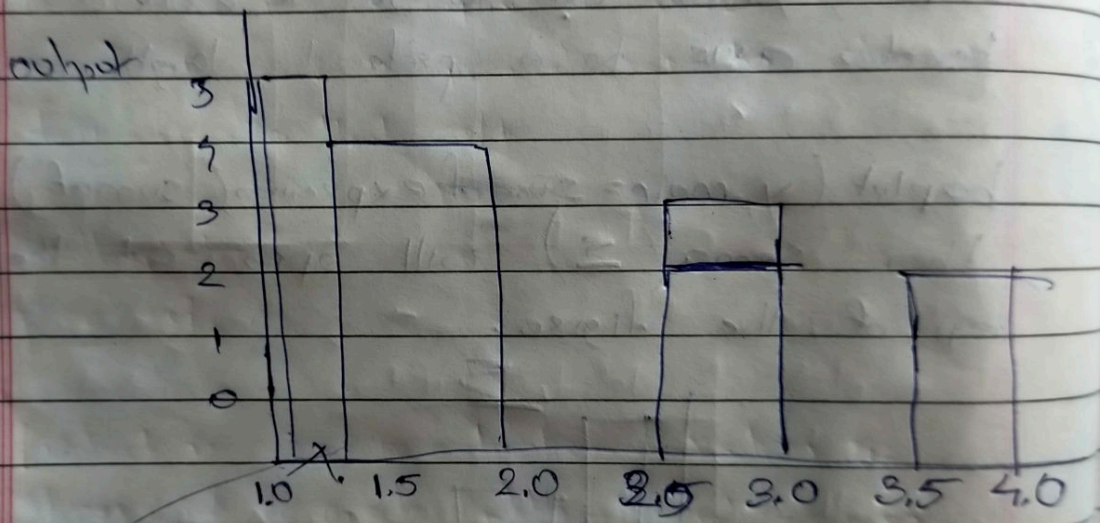
Histogram

It is used to plot quantitative data.

lets take x as an example

x = (1, 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 4, 4)

hist(x)



You can see the grouping of the data by using cut(x, 6)

the number of grouping of available datas and unavailable data in the end of the histogram.

cut(x, b, right = F)
will close the ~~be right~~ intervals.

Arguments in hist()

1) breaks: will cut the histogram into the number of part specified in breaks = 22.
2) argument like main, sub, xlab, ylab, xlim, ylim, color can be used in hist()
3) freq = FALSE can yield the probability distribution of the frequency as a histogram
4) density, ~~border~~ are other arguments that can be used in histogram

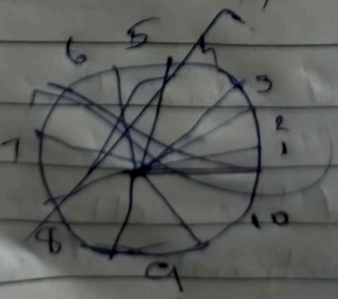
Return value of hist()

The hist() returns a list of 6 components
h = hist(temperature) they are:
>h
~~# breaks~~ breaks: places where the breaks occurs
counts -> the number of observations falling that cell.
density - density of the cells
mids :- the midpoint of cells
xname :- the x agument and
equidist: a logical value if the breaks are equally spaced or not.

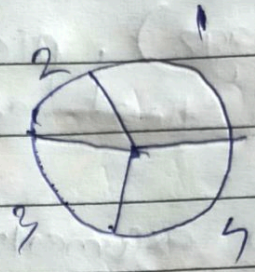
Breaks : helps us to define the no of cells to work in the histogram.
however the number is a suggestion
R calculates the best number of cells
keeping this suggestion in mind
vector can be an input to a break.
this can create unequal intervals in such case
area of cell is proportional to observation
falling under that cell.

~~Scatterplot()~~
piechart()

Piechart is a representation of values as
slices of a circle with different colors
it is created using pie.
x = c(1, 1, 1, 2, 2, 3, 3, 4, 4)
~~pie(x)~~ pie(table(x))
~~output~~



output



arguments in pie()

- main, sub has the same use in pie() as of the others ~~so does x!~~
- labels : used to name the slices of the pie chart
- edges → the circular outline can be modified into a polygonic shape.
  edges = x, where x can be any numeric value. edges=200, result in the circle shape
- radius : we can change the radius of the ~~circle~~ pie chat by this
  radius can ~~accept~~ form -1 : 1. (eg: radius = 1)
  default values is 1
- Clockwise : when ~~labelled~~ =T the logical indication ~~is~~ is the slices are drawn in clockwise direction. (default value = F).
- density, col, ~~border~~ others → argument that ~~can be~~ has the same purpose as in the ones before.

3D pie charts can be drawn by installing plotrix (a package), pie3D() can be used to create 3D pie charts.
explode = is an argument that can be used to "explode" the slices into different parts.

Scatterplot:

created by using plot() function. Scatterplot shows many points plotted in a cartesian plane.
There are 2 types plot
1) generic X:Y plot ('base')
2) default scatterplot function

generic X:Y plot (base)

plot the values in the plot() function can create scatter plot.
& contains numerical values as input will result in the y points created by according to the input in
(means x will be input will be y coordinate) while x will remain static)

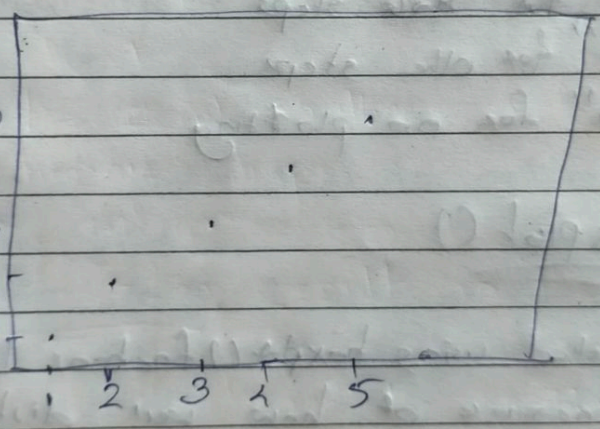
PAGE NO.
DATE :

solution has 1, 2, 3 etc in the X axis

Vectors as input can create a more normal version of the scatter plot

```
x<-6:5
y = 6:10
plot(x, y)
```

output :



Arguments in plot ()

xlab, ylab, main, sub, col, etc are also used in this function as well.

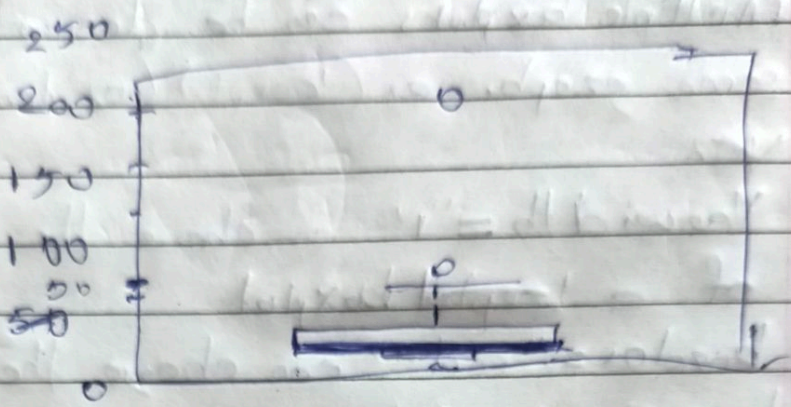
type argument : defines what type of plot should be drawn.

"p" for points
"l" for lines
"b" for both
"c" for the lines only alone of "b"
"o" for both overplotted
"h" for histogram like high density lines
"s" for stair steps
"S" for other steps
"n" for no plotting

Boxplot()

created using boxplot() function, they are a measure of how well distributed the data are.

eg: x = c(1,1,1,1,2,2,2,3,3,3,3,4,4,4,5,10,10,10,20,20,25,25,45,200)
boxplot(x)



outliers are data that seems like they don't belong to the boxplot at hand, outlies can be a mistake.

Median is the data that is colored in a pitch black color.
max is the highest value of the connected box plot while min is the smallest.
Range = max-min

arguments in the boxplot()

1) main, sub, xlab, ylab, colors used in the same way as before.
2) notch = "T" will create a notched shape from the median line.
3) horiz = "T" will create horizontal box plot

PAGE NO
DATE :

- Multiple boxplots can be drawn in the same graph. eg: boxplot(o2, temp, wind)
- varwidth = "T", can change the width of a ~~boxplot~~ boxplot
- border = "red" can change the boxplot's border's color.