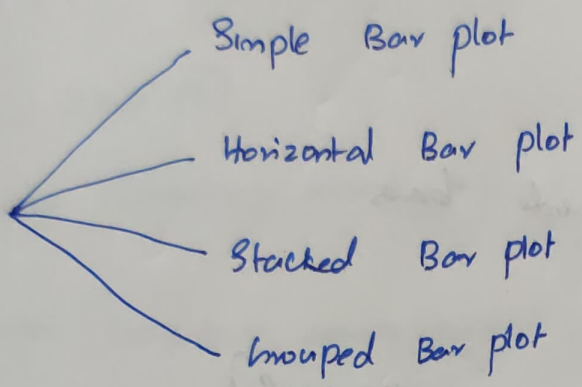
# 5: Graphs and Charts

## Bar plots in R




## R Bar plot

* Created by using → Barplot() function
* Input can be vector, matrix
* If we supply a vector, the plot will have bars with their heights equal to the elements in the vector/matrix

Eg: temp ← c(27, 26, 23, 24, 30)

### Argument used

+ main → used to give heading
+ xlim
+ ylim

xlab → x-axis name
ylab → y-axis name
col → Give colour to bar
horiz → TRUE
names.arg → name of each bar

Eg: temp = c (barplot temp,

main = "max Temp in a week",

xlab = "Degree Celcius",

ylab = "Dry",

col = "blue",

* density → line lines inside bars

* border → border to bars

density = 20, border = "red", col = "green")

* width = size of bars ie, width of bars [By default width = 1]

* space → space b/w bars

* x ← c (1,1,2,2,2,3,3,1,1,2,2,3,4,4,4)

table (x)

x

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| ↓ | ↓ | ↓ | ↓ |
| 4 | 5 | 3 | 3 |

plotting of Categorical Data

* x ← c (1,1,2,2,2,3,3,1,1,2,2,3,4,4,4)

y = table (x)

barplot (height = y, width = c(3,4,5,6))

* x ← c (1,1,2,2,2, 3,3, 1,1, 2,2, 3,4,4,4)

y = table (x)

barplot (height = y, space = 5)

* x ← c (1,1,2,2,2, 3,3, 1,1, 2,2, 3,4,4,4)

y = table (x)

barplot (height = y, names.arg = LETTERS [1:4])

barplot (height = y, names.arg = c ("student 1", "Student 2", "Student 3", "Student 4"))

* x = c (1,1,1,1,2,1,2,2,2,3,3,3,1,1,2,2,3,3)

y = table (x)

barplot (height = y, names.arg = c ("Student 1", "Student 2", "Student 3", legt.text = T))

legt.text → is a vector of text used to construct a legend for the plot if used to identify what each bar represent

Stacked Bar plots

Matrix is given as input

* data ("mtcars")

names (mtcars)

[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec"

"vs" "am" "gear" "carb"

>mtcars

>mtcars $cyl

[1]. . . .

>table (mtcars $cyl)

4 6 8

11 7 14

>table (mtcars $gear)

3 4 5

15 12 5

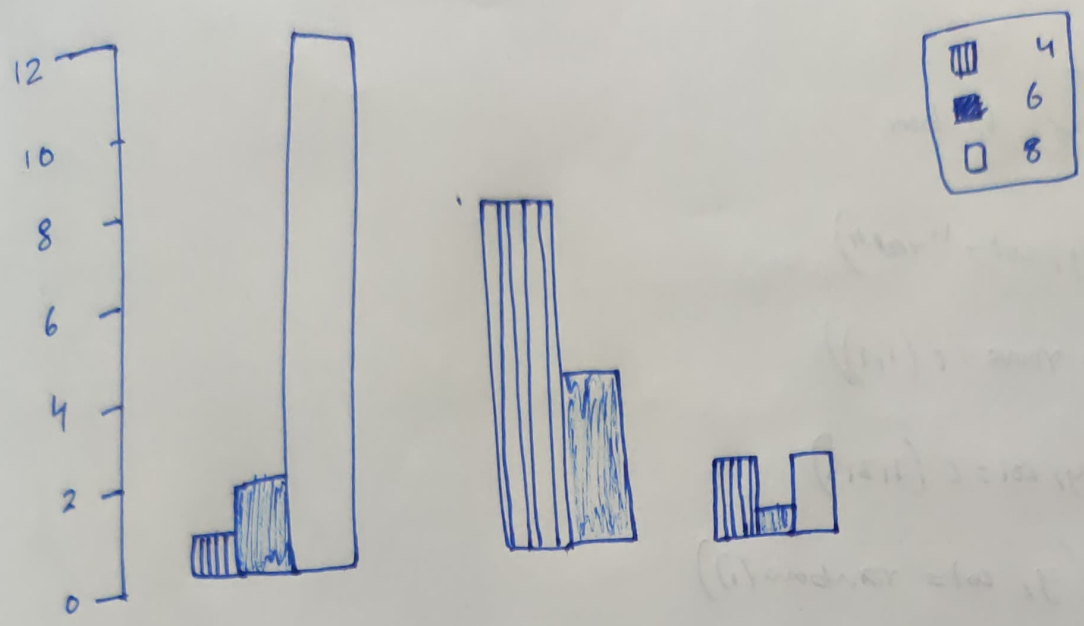
>table (mtcars $cyl, mtcars $gear)

| | 3 | 4 | 5 |
|---|---|---|---|
| 4 | 1 | 8 | 2 |
| 6 | 2 | 4 | 1 |
| 8 | 12 | 0 | 2 |

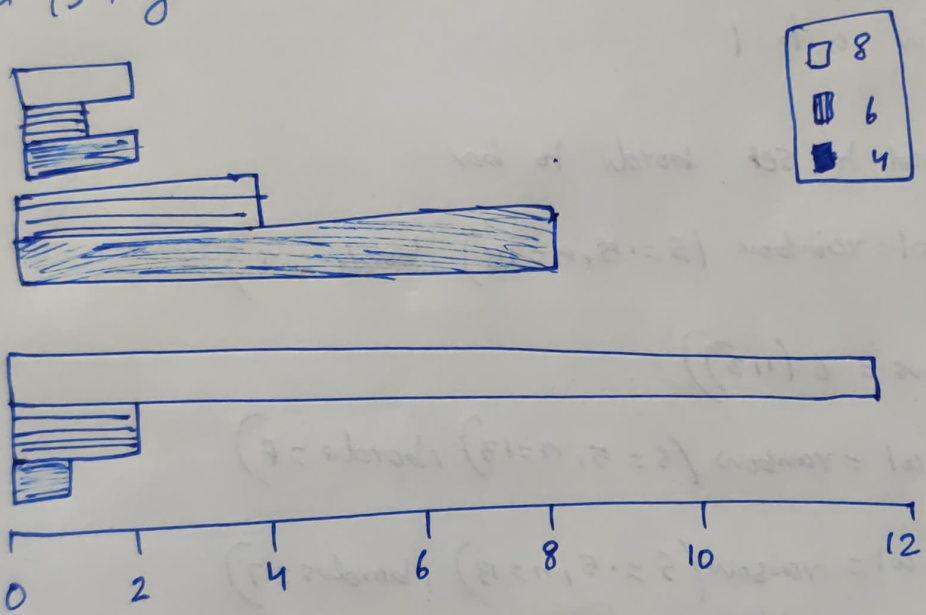
> y1 = table (mtcars $cyl, mtcars $gear)

>barplot (y1, legend.text = T)


>barplot (y1, legend.text = T, beside = T)




>barplot (y1, legend. text = T, beside = T, horiz = T)




* density :- used to give lines inside bars

Eg :- x=c(1,1,1,2,2,1,2,3,3,3,1)

y = table(x)

barplat (y, legend.text = T, lag=1, density = c(5,10,15))

* Angle :-
used to give angle to lines inside bar

* Colour :-

Give colour to bar

> barplot (y, col = "red")

> bar (mf rows = c(1,1))

> barplot (y, col = c(1,2,3))

> barplot (y, col = rainbow(1))

> barplot (y, col = rainbow(s = .2, n = 2))

s is b/w 0 to 1

* border :- used to set border to bar

> barplot (y, col = rainbow (s = .5, n = 15), border = T)

> bar (mf rows = c(1,2))

> barplot (y, col = rainbow (s = 5, n = 13), border = F)

> barplot (y, col = rainbow (s = .5, n = 13), borders = T)

> bar (mf row = c(1,1))

* main :- used to give heading to the particular barplot

Sub :- used to give heading at bottom

Eg: barplot (y, main = "header" Sub = "footer")

- barplot (y, main = expression (sum))
- xlim, ylim

barplot (y, ylim = c(0,10))

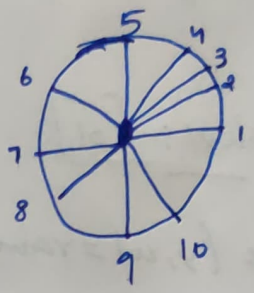
barplot (y, xlim = c(0,5))

## pie chart

Diagramatic representation of values

Eg: x = c(1,1,1,2,2,3,3,4,4,4)

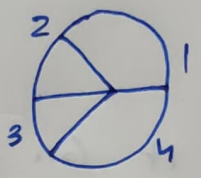
pie (x)




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

> y = table (x)

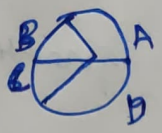
> pie (y)




> pie (y, main = "my first plot")

* x => a vector of non negative numerical quantities

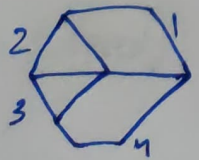
Eg1: > pie (y, labels = LETTERS [1:4])




Eg 2 > pie (y, labels = c("red", "blue", "green", "orange"))

* labels are names of each slices

* pie (y, edges = 10)



* pie(y, radius = .5)

* pie(y, clockwise = T)

* pie(y, density = c(10, 20, 30, 40))

Density : used to give shading to each slice.

colour :- [col]

* pie(y, col = rainbow(15))

* pie(y, col = 1:4)

Border :

used to set border it can be either T or F

pie(y, col = 1:4, border = F)

Histogram

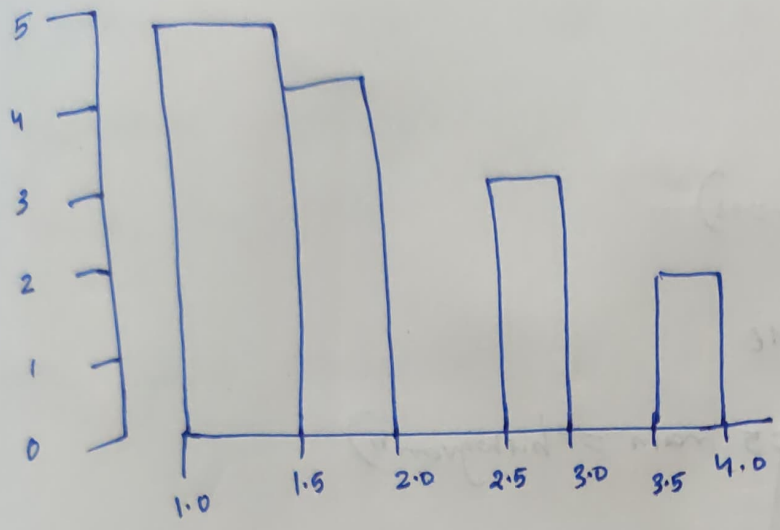
used to plot quantitive data

Function :- hist()

* inputs are vector inputs

> x <- c(1,1,1,1,1,2,2,2,2,3,3,3,4,4)

> hist(x)




* For viewing the grouping arrangement use the function cut()

```
>cut(x,6)

[1] [,] [,]

>data.frame(x,cut(x,6))

>data("cars")

>head(cars)
```

| | speed | dist |
|---|---|---|
| 1 | 4 | 2 |
| 2 | 4 | 10 |
| 3 | 7 | 4 |
| 4 | 7 | 22 |
| 5 | 8 | 16 |
| 6 | 9 | 10 |

```
>cars$speed

[1]

>hist(cars$speed)
```

Arguments

breaks:

>hist (cars $ speed, breaks = 22)

* main :- used to give title

>hist (cars$speed, breaks = 5, main = "histogram")

* xlab, ylab :-

>hist (cars $speed, xlab = "dist", ylab = "no of hrs")

Eg: >airquality

>head (airquality)

1 - - - -
2 - - - -
3 - - - -
4 - - - -
5 - - - -

>temp = airquality $ Temp

>hist (temp)

>str (airquality)

used to display

Str : Structure

* xlim, ylim :-

used to provide range of axes

* col :-

used to define color

* with the argument freq = FALSE we can get the probability distribution instead of the frequency

> hist (temp, freq = F)

Return value of hist()

Display the value in hist()

* breaks : places where the breaks occur
* counts : The no. of observations falling in that cell
* density : The density of cells
* mids :- The midpoint of cells
* xname : The x argument name
* equidist :- Logical value indicating if the breaks are equally spaced or not

Eg : > h = hist (temp)

> h

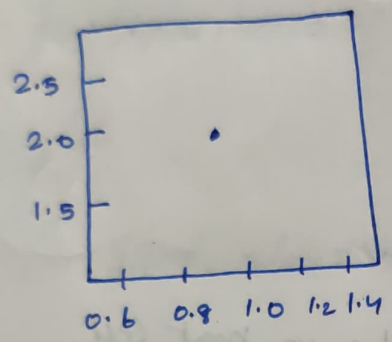
* with the breaks argument we can specify the no. of cells we want in the histogram
* we can also give breakpoints b/w the cells as a vector, this makes it possible to plot a histogram with unequal intervals

> hist (temp, border = "blue", breaks = c (55, 60, 70, 75, 80, 100))

# Scatter plot

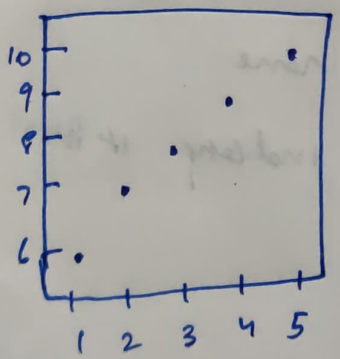
Created by using plot() function

>plot(2)




>plot ((5,6,7,8))

Eg > x=1:5
>y = 6:10
>plot(x,y)




Eg > av quality
>head (av quality)

> day = airquality $day

> temp = airquality $temp

> plot (day, temp)

Eg → x = 1:50

> y = sin(x)

> plot (x, y)

Eg) x = 1:10

> y = 21:30

> plot (x, y, main = "scatter point", xlab = "x-value", ylab = "y value", col = 1:10)

* type :-

Specifies what type of plot should be drawn. possible types are: "p" → for points

"l" → for lines

"b" → for both ie, combination of points & lines

"c" → for the lines part alone of b

"o" → for overplotted

"h" → for histogram like (for high-density) vertical lines

"s" → for staircase steps

"n" → no plotting

Eg : → plot (x, y, main = "scatter plot", type = "c")

Eg > x = 1:100

> y = sin(x)

> plot (x, y, type = "l")

Eg > x = seq (0,10,0,0.1)

> y = sin(x)

> plot t(x,y)

Boxplot:

used to plot quantitive Data

Eg: x = c(1,1,1,2,2, 1,1,3,3,3, 4,4,5,6,7,4,4,6,5,7,20,20,25,25
25,45,200)

boxplot (x)

* Boxplot can be used to identify median, range, quantile deviation and various other statistical measures

Eg: Str (airquality)

> barplot (airquality$ Ozone)

> boxplot (airquality$Ozone, main = "Ozone in parts per billion from 1300 to 1500 hrs at Roosevelt Island", xlab = "parts per billion", ylab = "Ozone", col = "orange", notch = T, horizontal = T)

For drawing multiple boxplot

```
> ozone = airquality$ozone
> temp = airquality$temp
  wind = airquality$wind
  boxplot(ozone, temp, wind)
```

For Changing the width of bars

```
>barplot(ozone, width = 1, border = "red")
```