5: Graphs and Charts

Bar Plot in R

- Simple Bar Plot
- Horizontal Bar Plot
- Stacked Bar Plot
- Grouped Bar Plot

R Bar Plot

* Created by using → barplot () function.

* Inputs can be Vector / Matrix.

* If we supply a vector, the plot will have bars with their hight equal to the element in the Vector

Eg: temp = c(27,26,23,24,30)
barplot(temp)

Argument used

* Main → used to give heading

xlab → x - axis name

ylab → y - axis name.

Col → Give colours to bar

horiz - True.

name .arg → name of each bar

```
Eg:- temp = c (
    barplot (temp
            main = "Max Temp in a week";

            xlab = ". Degree Celcius";

            ylab = "Day"

            col = "blue"
```

- * density -> Give lines inside bars
- * border -> border to bars.

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

- * width -> Size of bars
- * Space -> Space b/w bars
- * x <- c 1,1,2,2,2,3,3,1,1,2,2,3,4,4,4).

```
table (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, 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 [1:3])

barplot (height = y, names. arg = c("student 1", "student 2", "student3",
"students")

* 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",
"students 3"), legend. text = T)

legend. text -> is a Vector of text used to Construct
a legend for the plot ie, used
to identify what each bar represent

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

y = table (x)

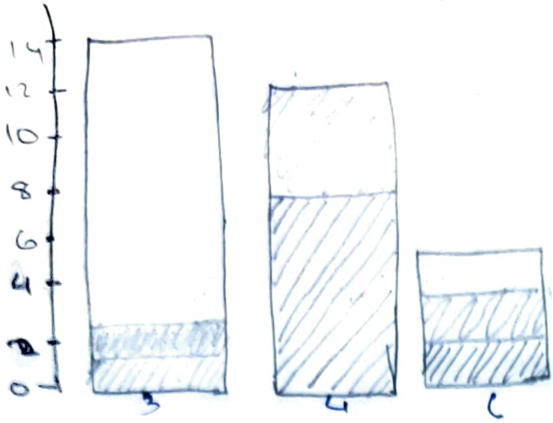
barplot (height = y, las = 1)

barplot (height = y, las = 2)

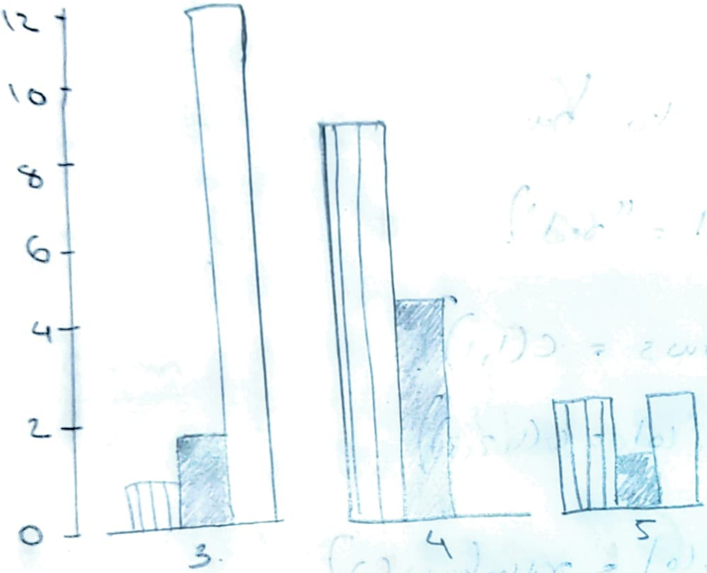
Stacked Bar Plots

Matrix is given as input

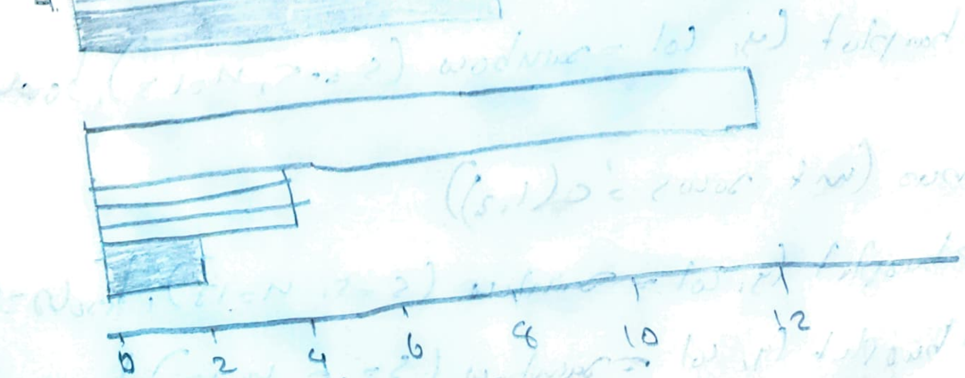
```
* > data (mt cars')

  > names (mt cars)

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

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

> mt cars

> mt cars $cyl

[1] . .   . .

> table (mt cars $ cyl)

  4   6   8
  11  7   14

> table (mt cars $ gear)

  3   4   5
  15  12  5

> table (mt cars $cyl , mt carts $ gears)

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

> y1 = table (mt cars $cyl , mtcarts $ 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)

barplot(y, legend.text = T, las = 1, density = c(5,10,15]

* Angle:

used to give angle to lines inside bar

* Colour :-

Give colours to bars

* barplot(y, col = "red").

* >par(mfrows = c(1,1)).

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

* >barplot(y, col = rainbow(7)).

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

S is b/w 0 to 1

* border:- used to set border to bars.

>barplot(y, col = rainbow(s=.5, n=15), border=4)

*>par(mfrows = c(1,2)).

>barplot(y, col = rainbow(s=.5, n=13), border=8)

>barplot(y, col = rainbow(s=.5, n=13) border=1).

>par(mfrows = c(1,1).

* main :- used to give heading to the Particular, bars Plot.

Sub :- used to give heading at bottom.

Eg:- barplot (y, main = "header" sub = "footer").

• barplot (y, main = expression (sum (i)

* x limit, ylimit

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

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

Pie chart

Diagrammatic 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 quantity.

Eg1) Pie (y, labels = letters [1:4])

Eg2) Pie (y, labels = c("red", "blue", "green", "orange")

* labels are names of each slices.

⊛ Pie (y, edges = 10)

⊛ Pie (y, radius = .5)

⊛ Pie (y, density = c(10, 20, 30, 40)

Density: used to give shading to each Slicl

<u>Colours</u> :- [col]

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

* Pie (y, col = 1:4)

<u>Border</u> :-

used to set border it can be either

T or F Pie (y, col = 1:4, border = F)

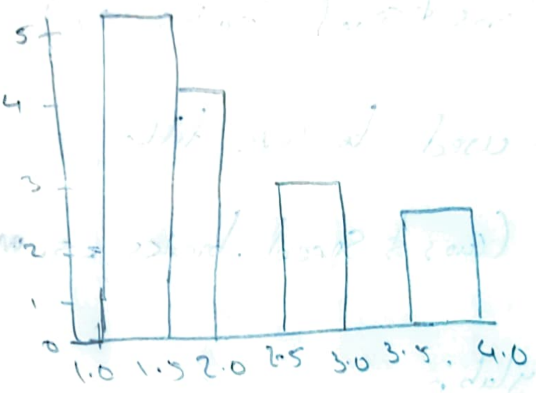
Histogram.

used to plot quantative 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
```

```
>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 times)
```

Eg :- >airquality

```
    >head airquality

  >temp = airquality $ Temp.
  >hist (temp)
  >str (airquality).
```

str :- used to display 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)

Eg: hist (temp, main= "maximum Daily Temperature",

xlab = "Temp in Degree , xlim = c(50,100)

col = rainbow (20) , freq = FALSE , las = T)

- border :- (T/F)
- density :

  > hist(temp , border = "blue" , density = 20)

Return Value of hist ()

Display the Value in hist ()

- breaks :- places where the breaks occur
- counts :- The no: of observation falling in that cell
- density :- The density of cells
- mids :- The midpoint of cells
- name :- The x argument name

* Equidist :- A 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 define breakpoints b/w the cells as a vector this make it possible to plot a histogram with unequal intervals

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

* 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 line part alone of b

"o" -> for overplotted

"h" -> for histogram (ike (or high - density) vertical lines

"s" -> for scalar steps.

"n" -> no plotting

Eg:- Plot (x,y, main = "Scatter Plot", type = "c").

Eg:-

```
>x = 1:100
>y = sin (x)
>plot (x,y, type = ~~B~~ ~~l~~ "D").
```

Eg:-

```
>x = Seq (0,10,0,0.1)
>y = sin (x)
>polot (x,y).
```

Boxplot :

used to plot quantitative Data

Eg: x = c(1,1,1,2,12,1,1,3,3,3,4,4,4,5,6,7,4,14,1

5,7,70,20,25,25,25,45,201)

boxplot (x)

* Boxplot can be used to identify median, inter quartile deviation and various other statistical measures.

boxplot (airquality $ ozone, main = "ozone in parts per billion from 1300 to 1500 hrs at roosevelt island, x lab = "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.

```
> boxplot (ozone , width = 1 , border = "red"
```