

Cargar un archivo

getwd()

Despliega el directorio actual de trabajo

```
> getwd()  
[1] "C:/Users/marze/OneDrive/Documentos"
```

setwd()

asigna el directorio actual de trabajo

```
> setwd("E:/clases2021_2/ECD/code")  
> getwd()  
[1] "E:/clases2021_2/ECD/code"
```

ls()

Lista las variables en el workspace

```
> ls()  
[1] "a"      "b"      "c"      "datos"  "food"  
[6] "mdat"   "mdatb"  "mydata" "paso"   "surtido"  
[11] "title"  "week1"  "week2"  "week3"  "x"  
[16] "xlist"  
>
```

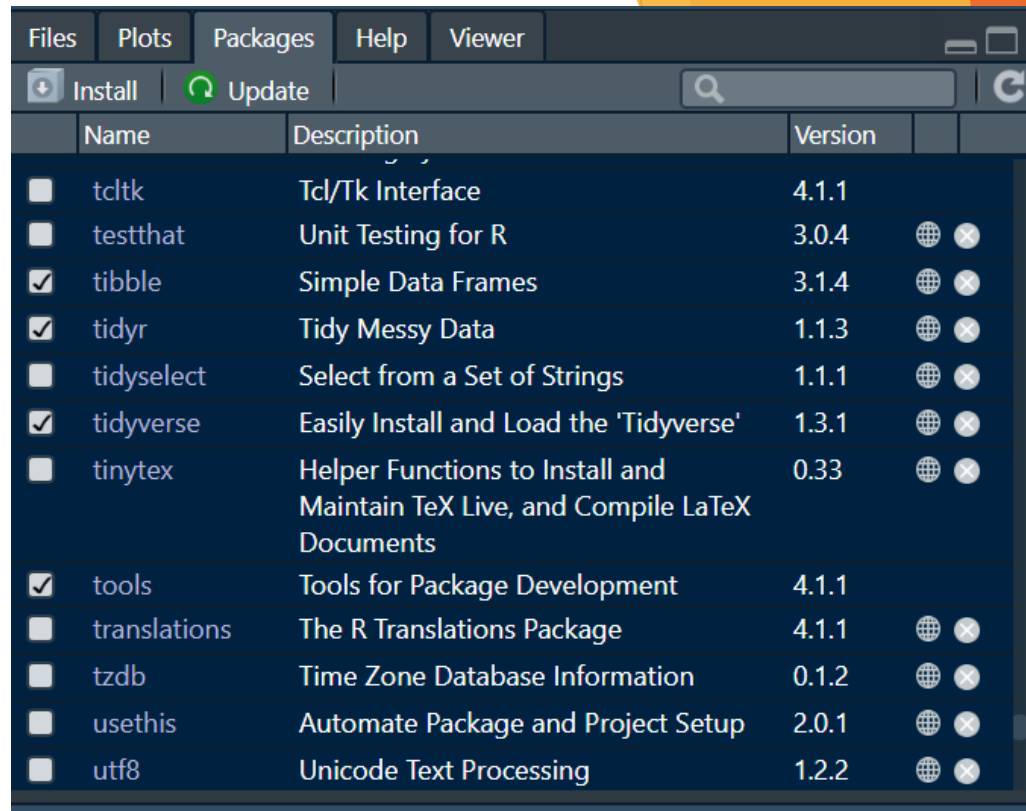
Packages

`install.packages()`

```
>  
> install.packages("tidyverse")
```

`library()`

```
>  
> library(tidyverse)
```



The screenshot shows the R Package Manager window. The 'Install' tab is selected. The table lists various packages with their names, descriptions, versions, and status (checked for installation). The 'Update' button is visible in the top right corner.

	Name	Description	Version		
<input type="checkbox"/>	tcltk	Tcl/Tk Interface	4.1.1		
<input type="checkbox"/>	testthat	Unit Testing for R	3.0.4	🌐	✕
<input checked="" type="checkbox"/>	tibble	Simple Data Frames	3.1.4	🌐	✕
<input checked="" type="checkbox"/>	tidyr	Tidy Messy Data	1.1.3	🌐	✕
<input type="checkbox"/>	tidyselect	Select from a Set of Strings	1.1.1	🌐	✕
<input checked="" type="checkbox"/>	tidyverse	Easily Install and Load the 'Tidyverse'	1.3.1	🌐	✕
<input type="checkbox"/>	tinytex	Helper Functions to Install and Maintain TeX Live, and Compile LaTeX Documents	0.33	🌐	✕
<input checked="" type="checkbox"/>	tools	Tools for Package Development	4.1.1		
<input type="checkbox"/>	translations	The R Translations Package	4.1.1	🌐	✕
<input type="checkbox"/>	tzdb	Time Zone Database Information	0.1.2	🌐	✕
<input type="checkbox"/>	usethis	Automate Package and Project Setup	2.0.1	🌐	✕
<input type="checkbox"/>	utf8	Unicode Text Processing	1.2.2	🌐	✕

Install Packages

Install from: [? Configuring Repositories](#)

Repository (CRAN) ▼

Packages (separate multiple with space or comma):

Install to Library:

F:/R-4.1.1/library [Default] ▼

☒ Install dependencies

Install

Cancel

Data files

read.table()

```
>  
> mydata=read.table("demographics.csv",sep=',',header=TRUE,stringsAsFactors = FALSE)  
>
```

filename
sep
header
stringAsFactors

read.csv()

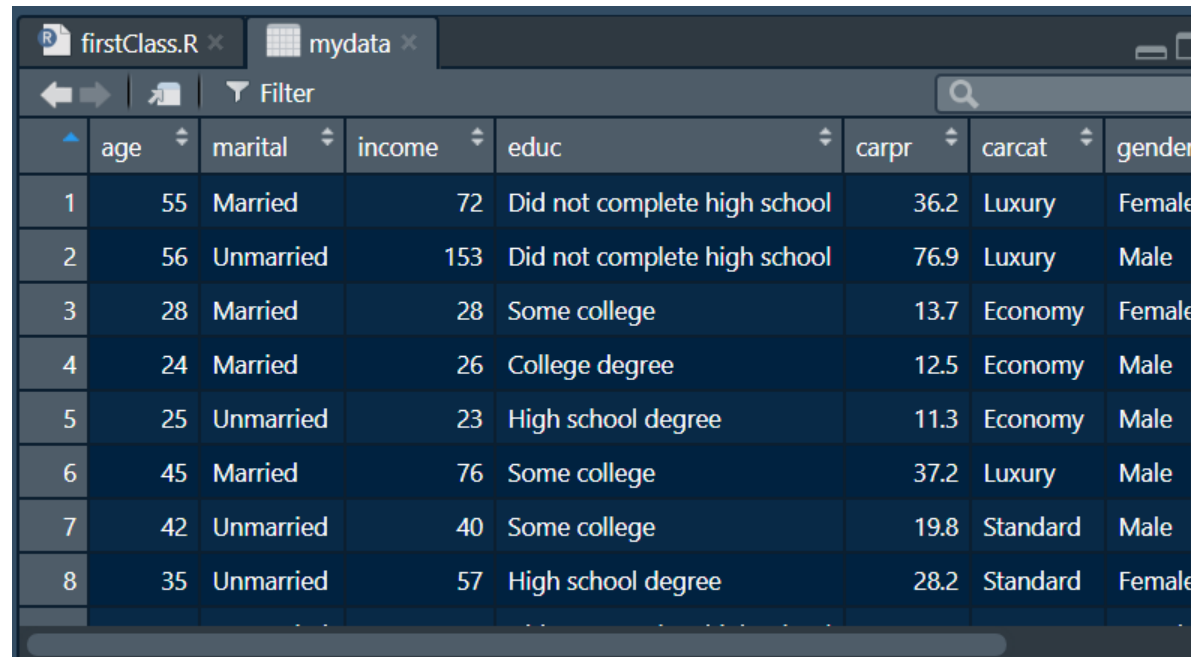
Lee un archivos

```
mydata<-read.csv("demographics.csv")
```

Display data

view()

```
>  
> view(mydata)  
> |
```



	age	marital	income	educ	carpr	carcat	gender
1	55	Married	72	Did not complete high school	36.2	Luxury	Female
2	56	Unmarried	153	Did not complete high school	76.9	Luxury	Male
3	28	Married	28	Some college	13.7	Economy	Female
4	24	Married	26	College degree	12.5	Economy	Male
5	25	Unmarried	23	High school degree	11.3	Economy	Male
6	45	Married	76	Some college	37.2	Luxury	Male
7	42	Unmarried	40	Some college	19.8	Standard	Male
8	35	Unmarried	57	High school degree	28.2	Standard	Female

str()

```
>
> str(mydata)
'data.frame':  510 obs. of  8 variables:
 $ age      : int  55 56 28 24 25 45 42 35 46 34 ...
 $ marital: chr   "Married" "Unmarried" "Married" "Married" ...
 $ income  : int  72 153 28 26 23 76 40 57 24 89 ...
 $ educ    : chr   "Did not complete high school" "Did not complete
high school" "Some college" "College degree" ...
 $ carpr   : num   36.2 76.9 13.7 12.5 11.3 37.2 19.8 28.2 12.2 46.1
 ...
 $ carcat  : chr   "Luxury" "Luxury" "Economy" "Economy" ...
 $ gender  : chr   "Female" "Male" "Female" "Male" ...
 $ retired: chr   "No" "No" "No" "No" ...
> |
```

names()

```
> names(mydata)
[1] "age"      "marital" "income"  "educ"    "carpr"
[6] "carcat"   "gender"  "retired"
> |
```

stringAsFactors=TRUE

```
> str(mydata)
'data.frame':  510 obs. of  8 variables:
 $ age      : int  55 56 28 24 25 45 42 35 46 34 ...
 $ marital: Factor w/ 2 levels "Married","Unmarried": 1 2 1 1 2
1 2 2 2 1 ...
 $ income  : int  72 153 28 26 23 76 40 57 24 89 ...
 $ educ    : Factor w/ 5 levels "College degree",...: 2 2 5 1 3 5
5 3 2 5 ...
 $ carpr   : num  36.2 76.9 13.7 12.5 11.3 37.2 19.8 28.2 12.2 4
6.1 ...
 $ carcat  : Factor w/ 3 levels "Economy","Luxury",...: 2 2 1 1 1
2 3 3 1 2 ...
 $ gender  : Factor w/ 2 levels "Female","Male": 1 2 1 2 2 2 2 1
1 2 ...
 $ retired: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1
...
> |
```

`head()`

Muestra los primeros 6 renglones

`tail()`

Muestra los últimos 6 renglones

`dim()`

Muestra la dimensión del data frame

`nrow()`

Numero de renglones

`ncol()`

Numero de columnas

`str()`

Estructura del data frame

`names()` /
`colnames`

Nombres de cada columna

`rownames`

Nombres de los renglones

summary()

```
> summary(mydata)
```

age	marital	income
Min. :18.00	Married :254	Min. : 9.00
1st Qu.:32.00	Unmarried:256	1st Qu.: 28.00
Median :41.00		Median : 45.00
Mean :42.06		Mean : 78.59
3rd Qu.:51.00		3rd Qu.: 86.00
Max. :73.00		Max. :1116.00

educ	carpr
College degree :113	Min. : 4.40
Did not complete high school:125	1st Qu.:13.82
High school degree :132	Median :22.20
Post-undergraduate degree : 27	Mean :30.84
Some college :113	3rd Qu.:42.42
	Max. :98.60

carcat	gender	retired
Economy :147	Female:250	No :485
Luxury :190	Male :260	Yes: 25
Standard:173		

summary()

```
> summary(mydata)

      age      marital      income
Min.   :18.00  Length:510  Min.    :  9.00
1st Qu.:32.00  Class  :character 1st Qu.: 28.00
Median :41.00  Mode   :character  Median : 45.00
Mean   :42.06                                     Mean  : 78.59
3rd Qu.:51.00                                     3rd Qu.: 86.00
Max.   :73.00                                     Max.   :1116.00

      educ      carpr      carcat
Length:510    Min.    : 4.40  Length:510
Class  :character 1st Qu.:13.82  Class  :character
Mode   :character Median :22.20  Mode   :character
                                     Mean  :30.84
                                     3rd Qu.:42.42
                                     Max.   :98.60

      gender      retired
Length:510    Length:510
Class  :character  Class  :character
Mode   :character  Mode   :character
```

$$\bar{x} = \frac{\sum_{i=1}^N x_i}{N}$$

4, 2, 8, 1, 15
 → 1, 2, 4, 8, 15
 MEDIAN

mean(x)

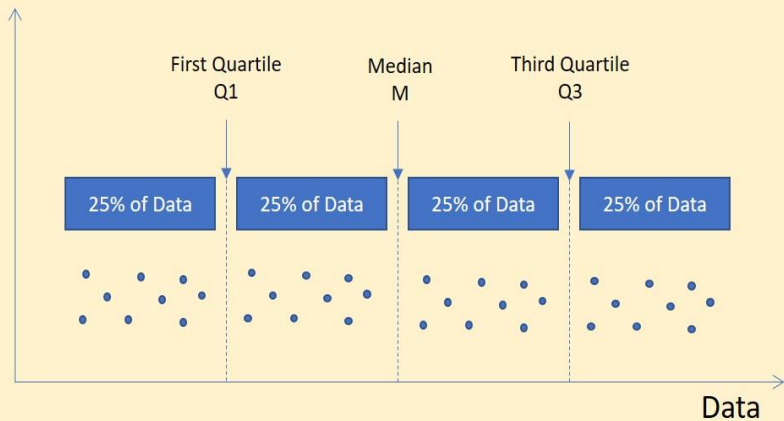
median(x)

1st quartile

3rd quartile

min(x)

max(x)



```
> mydata2=as.data.frame(starwars,stringAsFactors=TRUE)
> mydata2=mydata2[-c(12:14)]
> str(mydata2)
'data.frame':  87 obs. of  11 variables:
 $ name      : chr  "Luke Skywalker" "C-3PO" "R2-D2" "Darth Vader" ...
 $ height    : int  172 167 96 202 150 178 165 97 183 182 ...
 $ mass      : num  77 75 32 136 49 120 75 32 84 77 ...
 $ hair_color: chr  "blond" NA NA "none" ...
 $ skin_color: chr  "fair" "gold" "white, blue" "white" ...
 $ eye_color : chr  "blue" "yellow" "red" "yellow" ...
 $ birth_year: num  19 112 33 41.9 19 52 47 NA 24 57 ...
 $ sex       : chr  "male" "none" "none" "male" ...
 $ gender    : chr  "masculine" "masculine" "masculine" "masculine" ...
 $ homeworld : chr  "Tatooine" "Tatooine" "Naboo" "Tatooine" ...
 $ species   : chr  "Human" "Droid" "Droid" "Human" ...
> |
```

Slicing and selecting data

`dat[1, 3]`

Accesar a un elemento

`dat[["y"]]`

Accesar a una columna

`dat$y`

Accesar a una columna

`head()`

Muestra los primeros 6 renglones

`tail()`

Muestra los últimos 6 renglones

`dim()`

Muestra la dimensión del data frame

`nrow()`

Numero de renglones

`ncol()`

Numero de columnas

`str()`

Estructura del data frame

`names() /
colnames`

Nombres de cada columna

Basic statistics

Frequency distribution table

Frequency distribution table

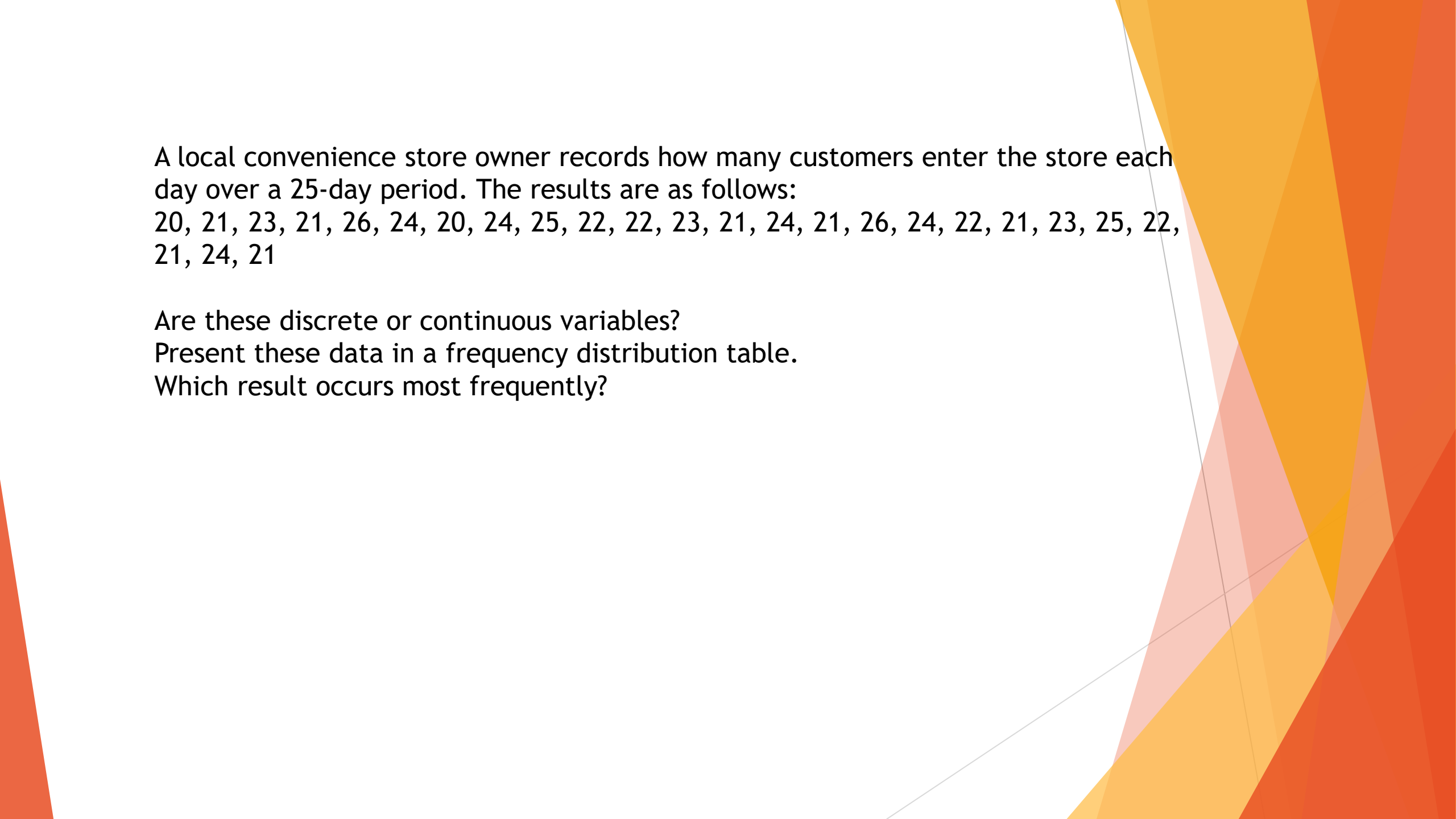
Frequency Tables	
Value	frequency
3	2
4	3
5	6
7	1
8	5

Handwritten data for frequency table:

- Value 3: frequency 2 (data: 3, 5, 8, 4)
- Value 4: frequency 3 (data: 8, 3, 8, 5, 5)
- Value 5: frequency 6 (data: 8, 7, 4, 4, 5)
- Value 7: frequency 1 (data: 8, 5, 5)

frequency distribution table		
A data table that lists a set of scores and their frequency.		
score	tally	frequency (f)
1		4
2	 	9
3	 	6
4	 	7
5		3
6		2

© Jenny Esther 2014



A local convenience store owner records how many customers enter the store each day over a 25-day period. The results are as follows:

20, 21, 23, 21, 26, 24, 20, 24, 25, 22, 22, 23, 21, 24, 21, 26, 24, 22, 21, 23, 25, 22, 21, 24, 21

Are these discrete or continuous variables?

Present these data in a frequency distribution table.

Which result occurs most frequently?

Frequency distribution

The frequency distribution of a data variable is a summary of the data occurrence in a collection of non-overlapping categories.

Frequency distribution
table

Barplot

`table()`

Create a table of the counts at each combination of factor levels.

```
library(MASS)
```

```
> str(painters)
'data.frame':  54 obs. of  5 variables:
 $ Composition: int  10 15 8 12 0 15 8 15 4 17 ...
 $ Drawing    : int   8 16 13 16 15 16 17 16 12 18 ...
 $ Colour     : int  16 4 16 9 8 4 4 7 10 12 ...
 $ Expression : int   3 14 7 8 0 14 8 6 4 18 ...
 $ School     : Factor w/ 8 levels "A","B","C","D",...: 1 1 1 1 1 1
1 1 1 1 ...
> |
```

```
school <- painters$School
schooltable <- table(school)
```

```
> schooltable
school
 A  B  C  D  E  F  G  H
10  6  6 10  7  4  7  4
> |
```

Categorical data

Calcula tabla de frecuencia del campo composicion

```
cbind(schooltable)
```

De que escuela hay más
pintores?

De que escuela hay
menos pintores?

```
names(schooltable)
```

```
> cbind(schooltable)
schooltable
A          10
B           6
C           6
D          10
E           7
F           4
G           7
H           4
> |
```

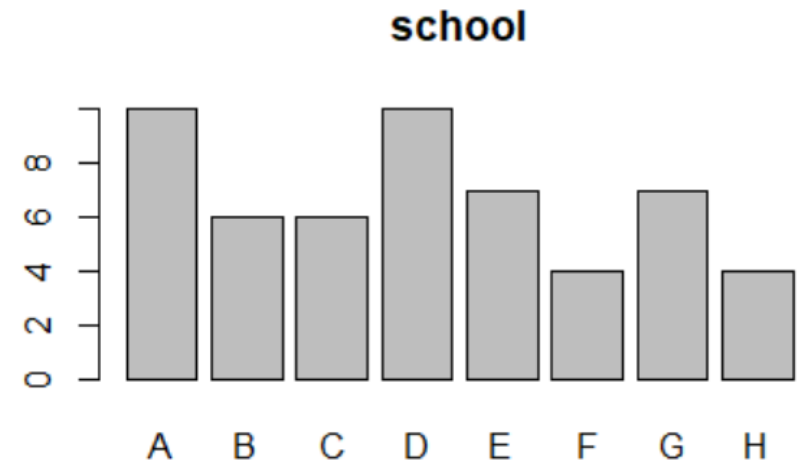
Categorical data

```
barplot(schooltable)
```

consists of vertical parallel bars that shows the frequency distribution graphically.

```
barplot(schooltable, main="school")
```

```
>  
> barplot(schooltable, main="school")  
> |
```



max(schooltable)

```
>  
> max(schooltable)  
[1] 10  
> |
```

which.max(schooltable)

which.min(schooltable)

```
> which.max(schooltable)  
A  
1  
> which.min(schooltable)  
F  
6  
> |
```

Return the first
maximum in the data

```
maxST=max(schooltable)
```

```
> maxST=max(schooltable)
> maxST
[1] 10
```

```
Res=(schooltable==maxST)
```

```
> Res
school
      A      B      C      D      E      F      G      H
TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
> |
```

Operation
element-wise

Look for a value in the frequency table, return a logical vector

```
x=schooltable[Res]
```

```
> x=schooltable[Res]
> x
school
  A  D
10 10
> |
```

Select the registers
where Res is true

```
names(x)
```

```
>
> names(x)
[1] "A" "D"
> |
```

Return all the column's names with
value equal to the max

Basic statistics

histogram

Quantitative Data / continuous data

frequency distribution of a data variable is a summary of the data occurrence in a collection of non-overlapping categories.

- Select data
- Find range
- Defining a sequence of equal distance for the break points (intervals)
- Divide the data using the intervals
- Compute the frequency in each interval

Range

The range of a data set is the difference of its largest and smallest data values.
It is how much the data spreads.



```
mydata      510 obs. of 8 variables
 $ age      : int  55 56 28 24 25 45 42 35 46 34 ...
 $ marital  : Factor w/ 2 levels "Married", "Unmarried":
 $ income   : int  72 153 28 26 23 76 40 57 24 89 ...
 $ educ     : Factor w/ 5 levels "College degree",...: 2...
 $ carpr    : num  36.2 76.9 13.7 12.5 11.3 37.2 19.8 2...
 $ carcat   : Factor w/ 3 levels "Economy", "Luxury",...: ...
 $ gender   : Factor w/ 2 levels "Female", "Male": 1 2 1...
 $ retired  : Factor w/ 2 levels "No", "Yes": 1 1 1 1 1 ...
```

```
> salary=mydata$income
> range(salary)
[1]      9 1116
> |
```

Create the intervals

```
breaks=seq(0,1150,by=50)
```

```
> breaks=seq(0,1150,by=50)
> length(breaks)
[1] 24
> breaks
[1] 0 50 100 150 200 250 300 350 400 450 500 550
[13] 600 650 700 750 800 850 900 950 1000 1050 1100 1150
> |
```

Divide the data using the intervals

```
salary_cut=cut(salary, breaks, right = FALSE)
```

```
> salary_cut=cut(salary,breaks,right = FALSE)
> str(salary_cut)
Factor w/ 23 levels "[0,50)","[50,100)","...: 2 4 1 1 1 2 1 2 1 2 ...
> head(salary_cut)
[1] [50,100) [150,200) [0,50) [0,50) [0,50) [50,100)
23 Levels: [0,50) [50,100) [100,150) [150,200) ... [1.1e+03,1.15e+03)
```

()
[)
(]

Convert numeric to factor

```
> levels(salary_cut)
[1] "[0,50)" "[50,100)" "[100,150)"
[4] "[150,200)" "[200,250)" "[250,300)"
[7] "[300,350)" "[350,400)" "[400,450)"
[10] "[450,500)" "[500,550)" "[550,600)"
[13] "[600,650)" "[650,700)" "[700,750)"
[16] "[750,800)" "[800,850)" "[850,900)"
[19] "[900,950)" "[950,1e+03)" "[1e+03,1.05e+03)"
[22] "[1.05e+03,1.1e+03)" "[1.1e+03,1.15e+03)"
>
```

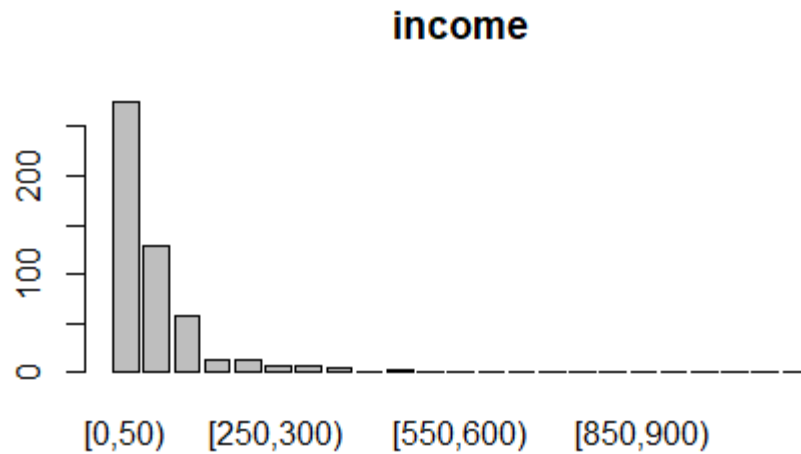
Numeric as a factor

```
temp=cbind(mydata$income,salary_cut)
```

```
> head(temp)
      salary_cut
[1,]    72      2
[2,]   153      4
[3,]    28      1
[4,]    26      1
[5,]    23      1
[6,]    76      2
>
```

Graph the barplot

```
tableSalary=table(salary_cut)  
barplot(tableSalary, main="income")
```



```
salary=mydata$income
```

```
range(salary)  
breaks=seq(0,1150,by=50)
```

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

```
salary_cut=cut(salary,breaks,right = FALSE)
```

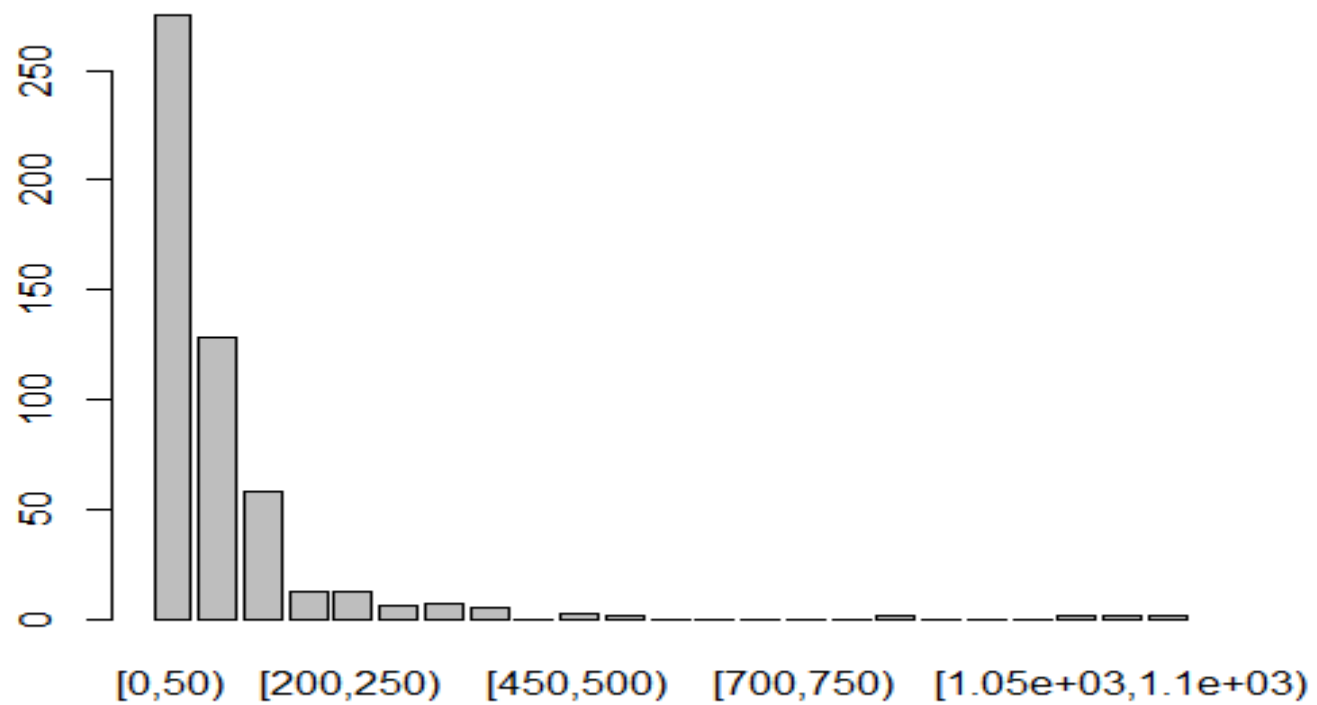
```
tableSalary=table(salary_cut)  
barplot(tableSalary, main="income")
```

- Select data
- Find range
- Defining a sequence of equal distance break points
- Divide the data using the intervals
- Compute the frequency in each interval

[min, max]

[min, max)

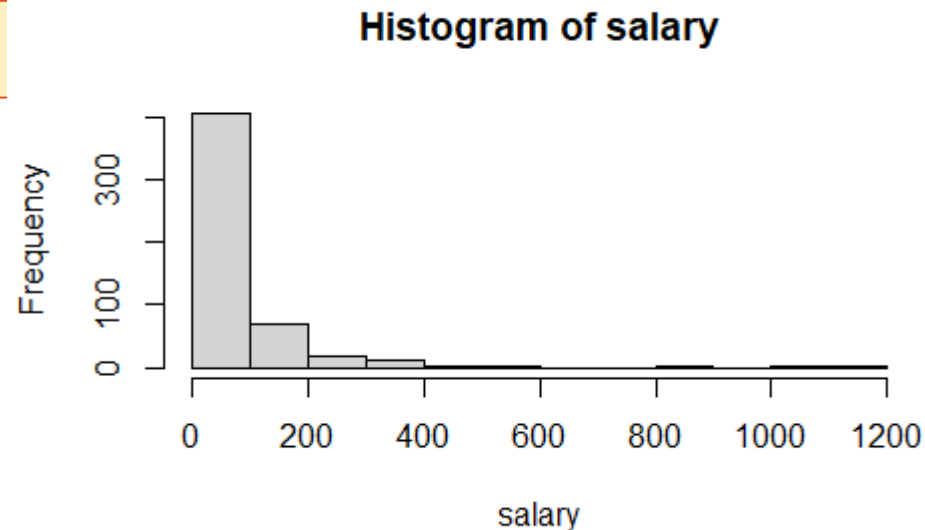
income



histogram

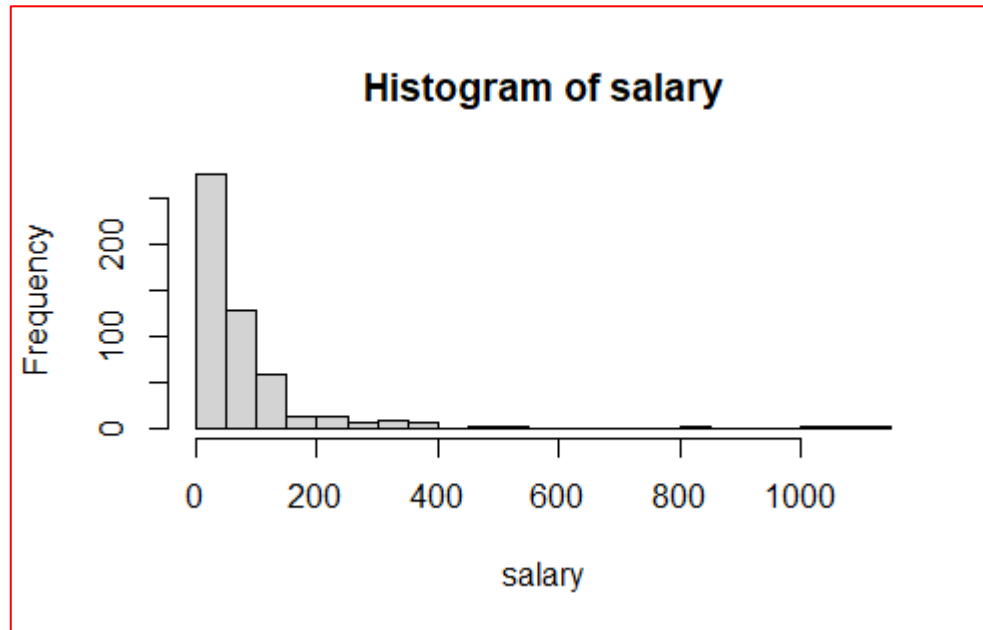
histogram consists of parallel vertical bars that graphically shows the frequency distribution of a quantitative variable. The area of each bar is proportional to the frequency of items found in each class.

```
hist(salary)
```



histogram

```
h = hist(salary, breaks=breaks, right=FALSE)
```



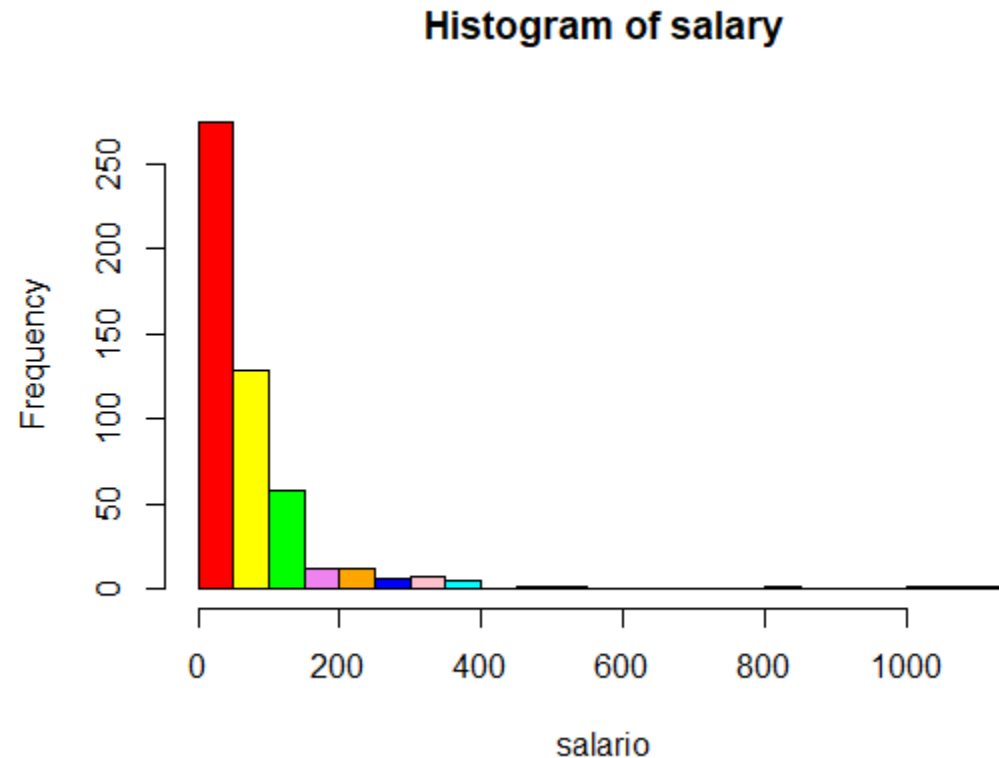
```
h = hist(salary, breaks=breaks, right=FALSE, plot=FALSE)
```

histogram

```
h = hist(salary, breaks=breaks, right=FALSE, plot=FALSE)
```

```
> h = hist(salary, breaks=breaks, right=FALSE, plot=FALSE)
> str(h)
List of 6
 $ breaks  : num [1:24] 0 50 100 150 200 250 300 350 400 450 ...
 $ counts  : int [1:23] 275 128 58 12 12 6 7 5 0 2 ...
 $ density : num [1:23] 0.010784 0.00502 0.002275 0.000471 0.000471
 ...
 $ mids     : num [1:23] 25 75 125 175 225 275 325 375 425 475 ...
 $ xname    : chr "salary"
 $ equidist: logi TRUE
 - attr(*, "class")= chr "histogram"
>
```

```
hist(salary, breaks=breaks, right=FALSE, col = colors,xlab="salario")
```



```
> colors=c("red","blue","green","yellow")  
> hist(salary, breaks=breaks, right=FALSE, col = colors,xlab="salario")  
> |
```