Introduction to language R

Daniela Zugna

R

- It provides a set of tools for statistical analysis of data
- It is a language used to describe statistical models, also extremely complex
- It allows for graphical representation
- It is a *object-oriented* language
- It is free and *open source*

The programming language R starts from *objects*, such as vector, dataset, table, graphic, model and so on

R console

R version 3.4.2 (2017-09-28) – "Short Summer"
Copyright (C) 2017 The R Foundation for Statistical Computing
R è un software libero ed è rilasciato SENZA ALCUNA GARANZIA.
Siamo ben lieti se potrai redistribuirlo, ma sotto certe condizioni.
Scrivi 'license()' o 'licence()' per dettagli su come distribuirlo.
R è un progetto di collaborazione con molti contributi esterni.
Scrivi 'contributors()' per maggiori informazioni e 'citation()'
per sapere come citare R o i pacchetti di R nelle pubblicazioni.
Scrivi 'demo()' per una dimostrazione, 'help()' per la guida in linea, o
'help.start()' per l'help navigabile con browser HTML.
Scrivi 'q()' per uscire da R.
[Caricato workspace precedentemente salvato]

- All commands have to be written after prompt $>$

Packages

library() lists installed packages

install.packages("pkg") connects to CRAN mirror to download a package

library(pkg) loads package for a session

update.packages() updates your packages

Help

?q

help(q)

- These commands ask to R to open a new window containing a guide on typed command
- The window describes accurately the command and report some examples on its use at the end of the window
- The + indicates that the command is not complete: press *Esc* button

example(plot) shows the examples of command plot

```
> example(plot)
plot> require(stats) # for lowess, rpois, rnorm
plot> plot(cars)
```

Calculator

```
> 2+2
[1] 4
> pnorm(1.96)
[1] 0.9750021
> pchisq(3.84,1)
[1] 0.9499565
```

Vectors and variables

```
> x<-4
> x
[1] 4
```

- The command assigns the value 4 to variable x
- R interpret the scalar as a vector of length equal to 1 and [1] indicates that value 4 represents the content of the first vector value

```
> y<-c(2,7,4,1)
> y
[1] 2 7 4 1
```

- *c* connects elements of vector

Vectors and variables

```
> ls()
[1] "x" "y"
```

- It provides the list of all objects loaded in the workspace

Matrix and algebraic operations

```
> x<-4
> y<-c(2,7,4,1)
> x*y
[1]  8 28 16  4
> y*y
[1]  4 49 16  1
> y^2
[1]  4 49 16  1
```

Matrix and algebraic operations

```
> y<-c(2,7,4,1)
> z<-y %*% t(y)
> z
     [,1] [,2] [,3] [,4]
[1,]    4   14    8    2
[2,]   14   49   28    7
[3,]    8   28   16    4
[4,]    2    7    4    1
```

- t is the transposed vector
- z is a matrix of 4 rows and 4 columns

Matrix and algebraic operations

```
> a<-matrix(1:30,5,6)
> a
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    6   11   16   21   26
[2,]    2    7   12   17   22   27
[3,]    3    8   13   18   23   28
[4,]    4    9   14   19   24   29
[5,]    5   10   15   20   25   30
```

- *a* is a matrix of 5 rows and 6 columns containing numbers from 1 to 30

Matrix and algebraic operations

```
> matrix(1:30,5,6,byrow=T)  #filling per row
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    2    3    4    5    6
[2,]    7    8    9   10   11   12
[3,]   13   14   15   16   17   18
[4,]   19   20   21   22   23   24
[5,]   25   26   27   28   29   30
```

- # allows to include comments

Matrix and algebraic operations

```
> matrix(c(1,2,3,4),2,4)
     [,1] [,2] [,3] [,4]
[1,]    1    3    1    3
[2,]    2    4    2    4
> matrix(,2,3)
     [,1] [,2] [,3]
[1,]   NA   NA   NA
[2,]   NA   NA   NA
```

- NA (not available) indicates missing values

Matrix and algebraic operations

```
> x<-c(1,2,3,4)
> y<-c(2,4,6,8)
> v1<-x+y
> v1
[1]  3  6  9 12
> v2<-x-y
> v2
[1] -1 -2 -3 -4
```

Matrix and algebraic operations

```
> v3<-x*y
> v3
[1]  2  8 18 32
> v4<-x/y
> v4
[1] 0.5 0.5 0.5 0.5
```

Matrix and algebraic operations

```
> cbind(v1,v2)
     v1 v2
[1,]  3 -1
[2,]  6 -2
[3,]  9 -3
[4,] 12 -4
> rbind(v1,v2)
   [,1] [,2] [,3] [,4]
v1    3    6    9   12
v2   -1   -2   -3   -4
```

- *cbind* merge two vectors (or matrices) on the columns
- *rbind* merge two vectors (or matrices) on the rows

Vector's elements

```
> x<-1:4
> x
[1] 1 2 3 4
> seq(-3,6,2)
[1] -3 -1  1  3  5
> seq(-3,1,length=10)
 [1] -3.0000000 -2.5555556 -2.1111111
 [4] -1.6666667 -1.2222222 -0.7777778
 [7] -0.3333333  0.1111111  0.5555556
[10]  1.0000000
```

- If the length=10, 10 equidistant elements will be created

Vector's elements

```
> x<--3:8
> x
 [1] -3 -2 -1  0  1  2  3  4  5  6  7  8
> A<-matrix(x,2,6)
> A
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]   -3   -1    1    3    5    7
[2,]   -2    0    2    4    6    8
```

Vector's elements

```
> x
 [1] -3 -2 -1  0  1  2  3  4  5  6  7  8
> x[3]
[1] -1
```

- [3] is the third element of the vector x

Vector's elements

```
> which(x<2)
[1] 1 2 3 4 5
```

- *which* provides the position of elements of x which are < 2

```
> which((x>=1) & (x<=5))
[1] 5 6 7 8 9
```

- *&* is the logical operator *and*

```
> which((x>=1) | (x<=5))
 [1]  1  2  3  4  5  6  7  8  9 10 11 12
```

- | is the logical operator *or*

Vector's elements

```
> x
 [1] -3 -2 -1  0  1  2  3  4  5  6  7  8
> z<-which((x>=1) & (x<=5))
> x[z]
[1] 1 2 3 4 5
> (x>=1) & (x<=5)
 [1] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE
 [8]  TRUE  TRUE FALSE FALSE FALSE
> which((x>=1) & (x<=5))
[1] 5 6 7 8 9
```

Matrix's elements

```
> A
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]   -3   -1    1    3    5    7
[2,]   -2    0    2    4    6    8
> A[2,3]
[1] 2
```

- $[2, 3]$ is the second row and the third column of matrix A

Matrix's elements

```
> A[2,]
[1] -2  0  2  4  6  8
> A[,3]
[1] 1 2
```

- $[2,]$ is the second row of matrix A
- $[,3]$ is the third column of matrix A

Matrix's names

```
> rownames(A)
NULL
> colnames(A)
NULL
> rownames(A)<-c("a","b")
> colnames(A)<-c("a","b","c","d","e","f")
> A
   a  b c d e f
a -3 -1 1 3 5 7
b -2  0 2 4 6 8
```

Workspace

```
> ls()
[1] "a"  "A"  "v1" "v2" "v3" "v4" "x"  "y"
[9] "z"
```

- It provides the objects saved in the workspace
- save.image: It saves the content of the workspace in a file named .RData memorised in the working directory
- getwd: It shows the working directory
- setwd: It allows to change working directory

Workspace

```
> setwd("/Users/lab/Documents/R_2020")
> save(x,A,file="prova.rda")
```

- Objects x and A are saved in prova.rda file in working directory Users lab Documents R_2020

```
> rm(v1,v2,v3,v4)
```

- Objects v1,v2,v3,v4 are removed from workspace

```
> rm(list=ls())
> ls()
character(0)
```

- All objects are removed from workspace

Workspace

```
> load("/Users/lab/Documents//R_2020/prova.rda")
> ls()
[1] "A" "x"
```

- Objects A and x are loaded in working directory

Functions

```
> somma<-function(a,b) {a+b}
> somma(2,3)
[1] 5
```

- *somma* is the name of the *function*
- *function* specifies the arguments in round brackets

```
> prod.somma<-function(a,b,c){a*b+c}
> prod.somma(3,4,2)
[1] 14
```

Functions

```
> body(prod.somma)
{
    a * b + c
}
```

ifelse

```
> somma<-function(a,b){
+ if((a>0) & (b>0))
+ a+b
+ else
+ -1
+ }
> somma(2,3)
[1] 5
> somma(2,-4)
[1] -1
```

ifelse

```
> somma<-function(a,b){
+ if((a>0) | (b>0))
+ a+b
+ else
+ -1
+ }
> somma(2,3)
[1] 5
> somma(2,-4)
[1] -2
> somma(-2,-4)
[1] -1
```

Cycle for

```
> for (i in 1:9) print(1:i)
[1] 1
[1] 1 2
[1] 1 2 3
[1] 1 2 3 4
[1] 1 2 3 4 5
[1] 1 2 3 4 5 6
[1] 1 2 3 4 5 6 7
[1] 1 2 3 4 5 6 7 8
[1] 1 2 3 4 5 6 7 8 9
```

Cycle for

```
> a<-0
> for (i in 1:10000) a<-a+i
> a
[1] 50005000
> sum(1:10000)
[1] 50005000
```

Object's types

Everything in R is an object and every object belongs to a class

```
> x<-1:3
> str(x)
 int [1:3] 1 2 3
> x<-numeric(3)
> x
[1] 0 0 0
> str(x)
 num [1:3] 0 0 0
```

- *str* defines the object's type: x is a vector of integer numbers
- *numeric* creates a numeric vector of length 3 (of real numbers)

Object's types

```
> str<-c("Silvia","Laura","Alberto")
> str
[1] "Silvia"  "Laura"   "Alberto"
> str(str)
 chr [1:3] "Silvia" "Laura" "Alberto"
```

- Object *str* is a vector of strings

Object's types

```
> log<-c(TRUE,FALSE,TRUE)
> log
[1]  TRUE FALSE  TRUE
> str(log)
 logi [1:3] TRUE FALSE TRUE
```

- Object *log* is a logical vector cointaining TRUE or FALSE

Object's types

```
> A<-matrix(1,4,2)
> lista<-list(nomi=str,logic=log,matrix=A)
> lista
$nomi
[1] "Silvia"  "Laura"   "Alberto"

$logic
[1]  TRUE FALSE  TRUE

$matrix
     [,1] [,2]
[1,]    1    1
[2,]    1    1
[3,]    1    1
[4,]    1    1
```

Object's types

```
> str(lista)
List of 3
 $ nomi  : chr [1:3] "Silvia" "Laura" "Alberto"
 $ logic : logi [1:3] TRUE FALSE TRUE
 $ matrix: num [1:4, 1:2] 1 1 1 1 1 1 1 1
```

- *list* is a vector whose elements can be objects of different types
- *llista* is an object containing 3 elements named nomi, logic, matrix

Object's types

```
> lista$nomi
[1] "Silvia"  "Laura"   "Alberto"
> lista[[1]]
[1] "Silvia"  "Laura"   "Alberto"
> lista[[1]][3]
[1] "Alberto"
```

Data types

- qualitative (nominal or ordinal)
- quantitative (discrete or continuous)

```
> sesso<-c("M","M","M","F", "F", "F","F")
> str(sesso)
 chr [1:7] "M" "M" "M" "F" "F" "F" "F"
> eta<-c("giovane","giovane","adulto","adulto",
+         "anziano","giovane","anziano")
> str(eta)
 chr [1:7] "giovane" "giovane" "adulto" ...
```

Data types

```
> sesso2<-factor(sesso)
> str(sesso2)
 Factor w/ 2 levels "F","M": 2 2 2 1 1 1 1
> sesso2
[1] M M M F F F F
Levels: F M
```

- *sesso2* is a qualitative vector with levels representing the categories of the variable
- F=1, M=2: in absence of an order R considers the alphabetical order

Data types

```
> eta2<-factor(eta)
> str(eta2)
 Factor w/ 3 levels "adulto","anziano",..: 3 3 1 1 2 3 2
> eta2
[1] giovane giovane adulto  adulto  anziano
[6] giovane anziano
Levels: adulto anziano giovane
> ordered(eta2,levels=c("giovane","adulto","anziano"))
[1] giovane giovane adulto  adulto  anziano
[6] giovane anziano
Levels: giovane < adulto < anziano
```

Data types

```
> eta2<-factor(eta,levels=c("giovane","adulto","anziano"),
ordered=T)
> eta2
[1] giovane giovane adulto  adulto  anziano
[6] giovane anziano
Levels: giovane < adulto < anziano
> eta2<-ordered(eta,levels=c("giovane","adulto","anziano"))
> eta2
[1] giovane giovane adulto  adulto  anziano
[6] giovane anziano
Levels: giovane < adulto < anziano
```

Data types

```
> eta2
[1] giovane giovane adulto  adulto  anziano
[6] giovane anziano
Levels: giovane < adulto < anziano
> as.numeric(eta2)
[1] 1 1 2 2 3 1 3
> unclass(eta2)
[1] 1 1 2 2 3 1 3
attr(,"levels")
[1] "giovane" "adulto"  "anziano"
```

See practical on Introduction to R (practical_intro) with corresponding solutions (solution_intro) and R script (main_intro).

Dataframe

Variable	Code
id	Identity
bweight	Birth weight of baby (g)
lowbw	Indicator for birth weight less than 2500 g
gestwks	Gestation period (weeks)
preterm	Indicator for gestation period less than 37 weeks
matage	Maternal age
hyp	Indicator for maternal hypertension
sex	Sex of baby: 1:Male, 2:Female

Dataframe

```
> library(foreign)
> data<-read.table("/Users/lab/Documents/R_2020/data/
+  data.births.csv",header = TRUE, sep = ",", row.names = 1)
> head(data)
  id bweight lowbw gestwks preterm matage hyp
1  1    2974     0   38.52       0     34   0
2  2    3270     0      NA      NA     30   0
3  3    2620     0   38.15       0     35   0
4  4    3751     0   39.80       0     31   0
5  5    3200     0   38.89       0     33   1
6  6    3673     0   40.97       0     33   0
  sex
1   2
2   1
3   2
4   1
5   1
6   2
> View(data)
```

Dataframe

```
> class(data)
[1] "data.frame"
> str(data)
'data.frame': 500 obs. of  8 variables:
 $ id      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ bweight: int  2974 3270 2620 3751 3200 3673 3628 3773 3960 34
 $ lowbw   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ gestwks: num  38.5 NA 38.2 39.8 38.9 ...
 $ preterm: int  0 NA 0 0 0 0 0 0 0 0 ...
 $ matage : int  34 30 35 31 33 33 29 37 36 39 ...
 $ hyp     : int  0 0 0 0 1 0 0 0 0 0 ...
 $ sex     : int  2 1 2 1 1 2 2 1 2 1 ...
```

Dataframe

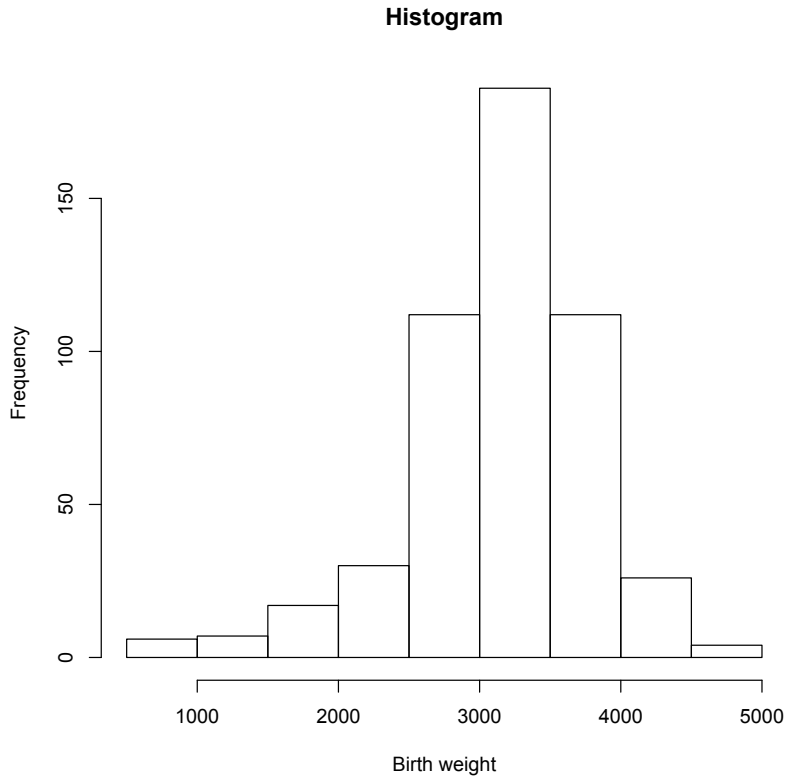
```
> head(data)
  id bweight lowbw gestwks preterm matage hyp
1  1    2974     0   38.52       0      34   0
2  2    3270     0      NA      NA      30   0
3  3    2620     0   38.15       0      35   0
4  4    3751     0   39.80       0      31   0
  sex
1   2
2   1
3   2
4   1
> data[1:2,]
  id bweight lowbw gestwks preterm matage hyp
1  1    2974     0   38.52       0      34   0
2  2    3270     0      NA      NA      30   0
  sex
1   2
2   1
> data$bweight[1:2]
[1] 2974 3270
```

Continuous variable

```
> summary(data$bweight)
   Min. 1st Qu.  Median    Mean 3rd Qu.
    628    2862    3188    3137    3551
   Max.
   4553
> table(cut(data$bweight,breaks=c(0,2500,3000,3500,4000,5000),
+           right=F))

    [0,2.5e+03) [2.5e+03,3e+03)
             60             112
[3e+03,3.5e+03) [3.5e+03,4e+03)
            186             112
  [4e+03,5e+03)
             30
```

Histogram

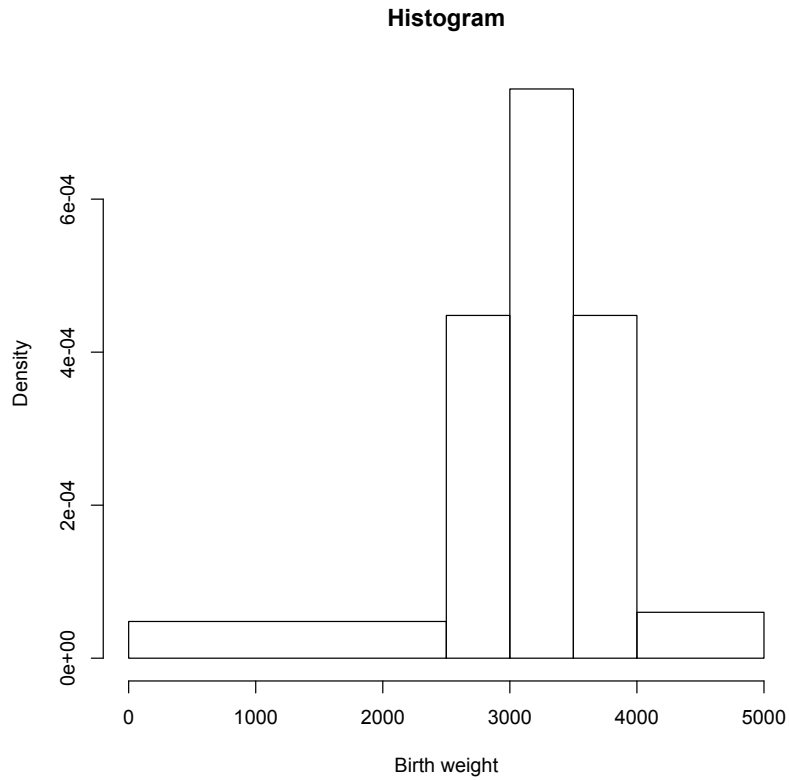
```
> hist(data$bweight,main="Histogram",xlab="Birth weight")
```



#histogram of birth weight: base x height=frequency

Histogram

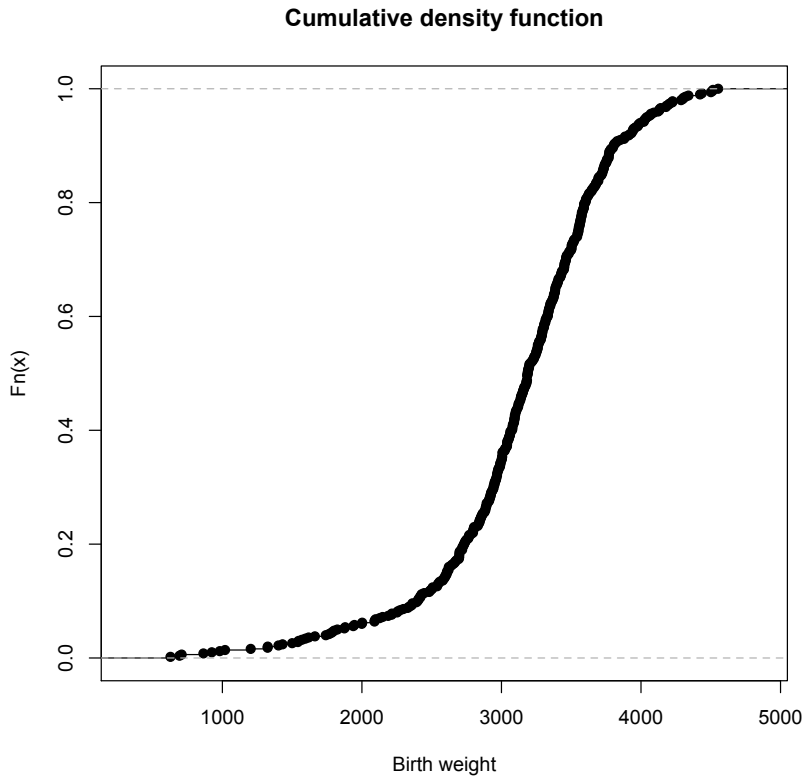
```
> hist(data$bweight,c(0,2500,3000,3500,4000,5000),
+       main="Histogram",xlab="Birth weight") #histogram of birth
```



Cumulative density function

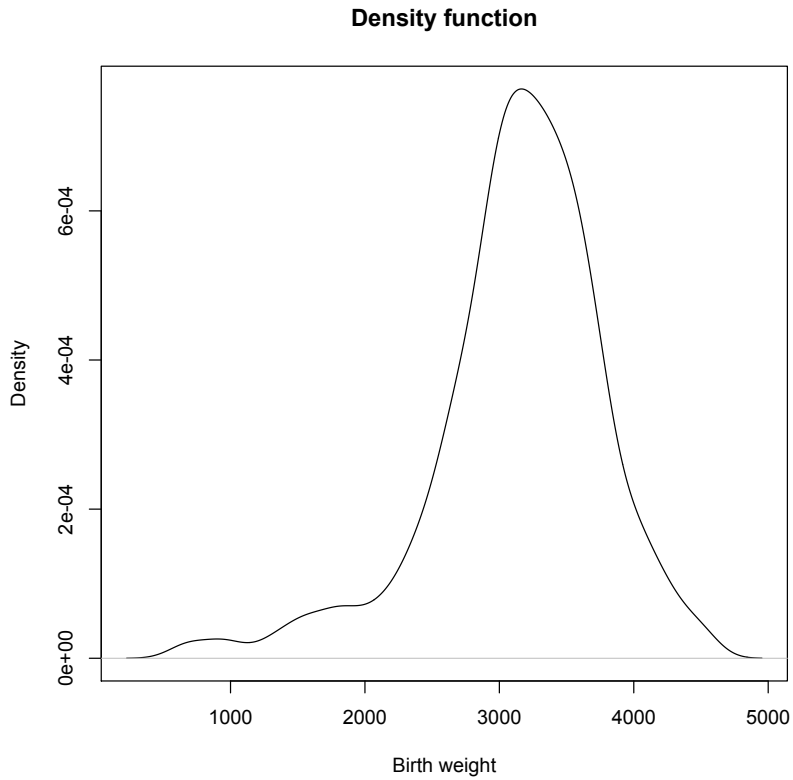
Cumulative density function: $Pr(X < x)$

```
> plot(ecdf(data$bweight),main="Cumulative density function" ,
+   xlab="Birth weight")
```



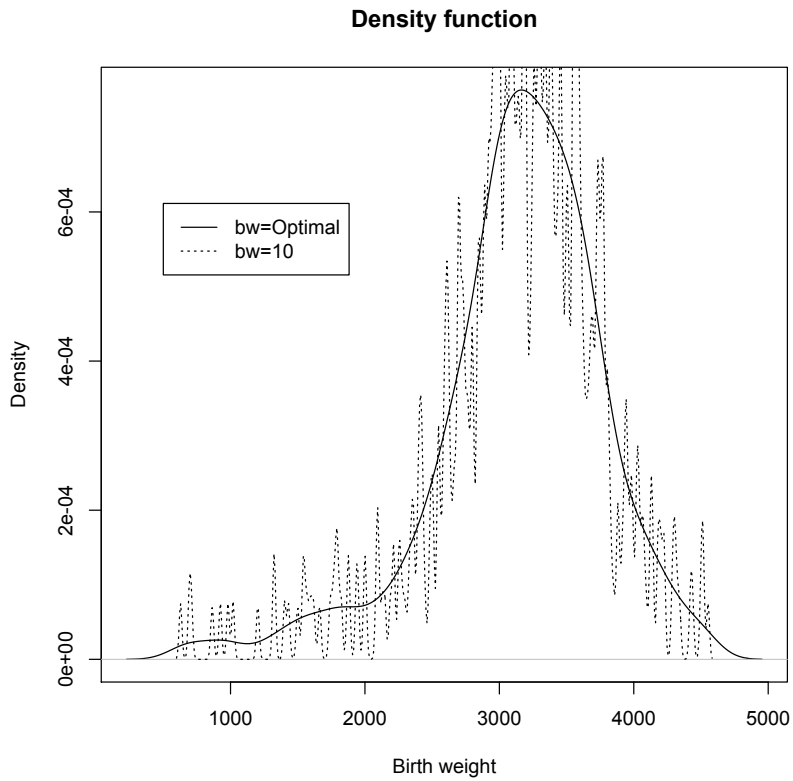
Density function

```
> plot(density(data$bweight),main="Density function",
+   xlab="Birth weight")
```



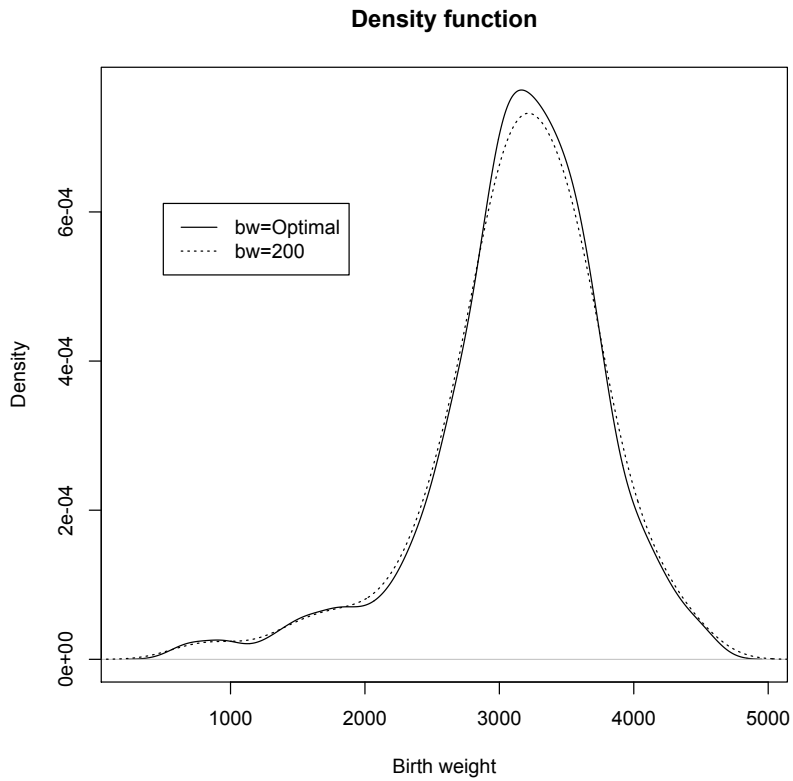
Density function

```
> plot(density(data$bweight),main="Density function" ,
+  xlab="Birth weight")
> lines(density(data$bweight,bw=10),lty=3)
> legend(500,6.114e-04,c("bw=Optimal","bw=10"),lty=c(1,3))
```



Density function

```
> plot(density(data$bweight),main="Density function" ,
+  xlab="Birth weight")
> lines(density(data$bweight,bw=200),lty=3)
> legend(500,6.114e-04,c("bw=Optimal","bw=200"),lty=c(1,3))
```



Density function

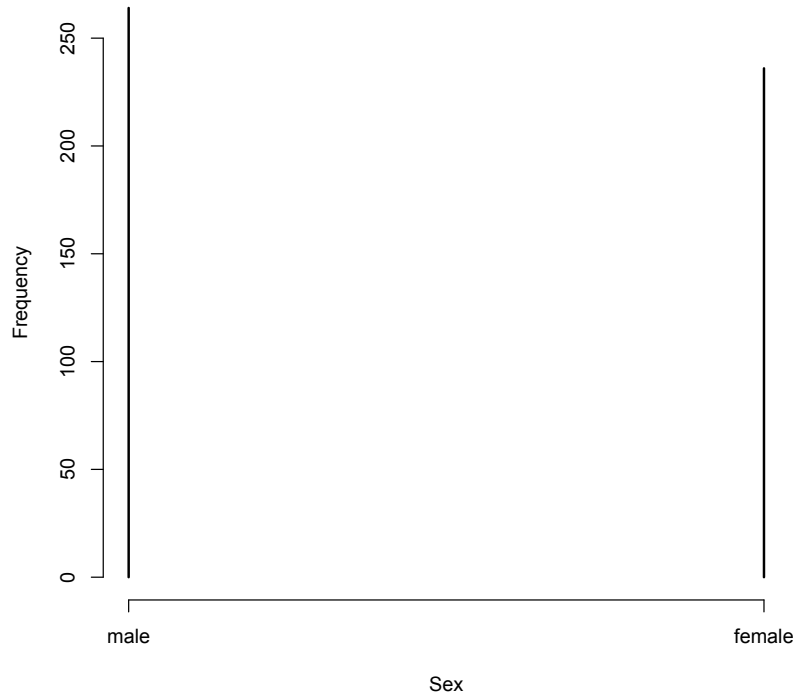
```
> density(data$bweight)

Call:
density.default(x = data$bweight)

Data: data$bweight (500 obs.); Bandwidth 'bw' = 133.6

       x                y
 Min.   : 227.3   Min.   :9.120e-08
 1st Qu.:1408.9   1st Qu.:2.501e-05
 Median :2590.5   Median :7.318e-05
 Mean   :2590.5   Mean   :2.114e-04
 3rd Qu.:3772.1   3rd Qu.:3.448e-04
 Max.   :4953.7   Max.   :7.635e-04
```

Categorical variables

```
> summary(data$sex)
   Min. 1st Qu.  Median    Mean 3rd Qu.
  1.000   1.000   1.000   1.472   2.000
   Max.
  2.000
> data$sex <- factor(data$sex,levels = c(1,2),
+  labels = c("male", "female"))
> table(data$sex)  #frequency of sex

  male female
   264    236
```

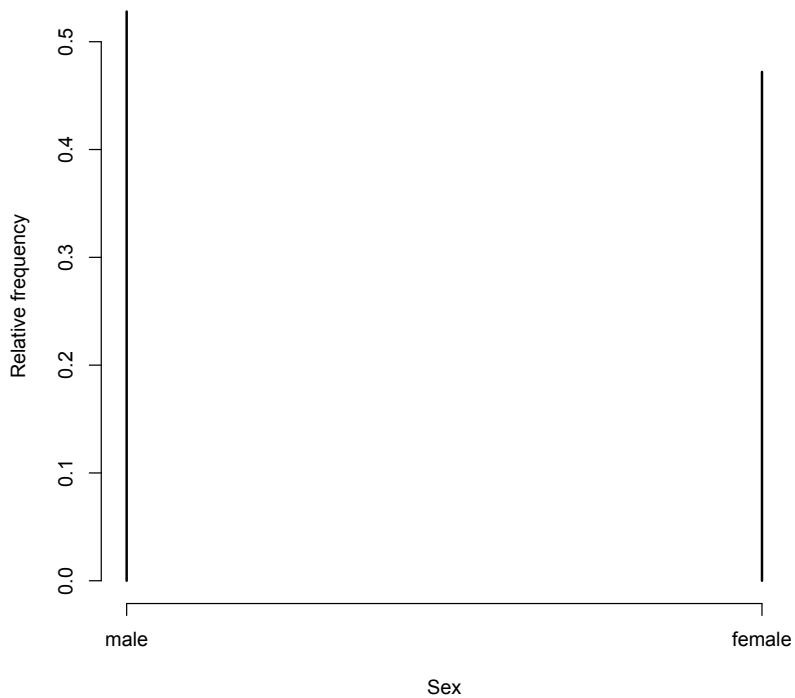
Categorical variables

```
> plot(table(data$sex),xlab="Sex",ylab="Frequency")
#bar chart of frequency
```



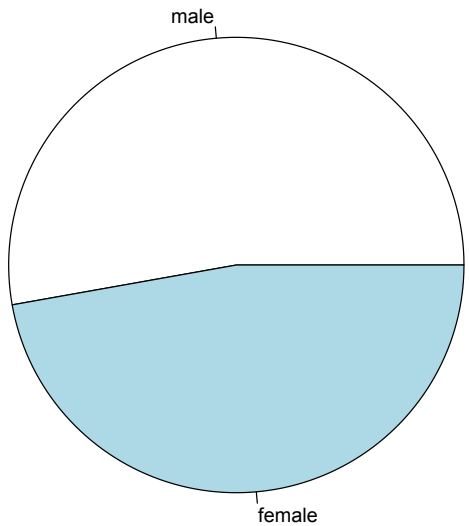
Bar chart

```
> plot(table(data$sex)/sum(table(data$sex)),xlab="Sex",
+  ylab="Relative frequency")
> #bar chart of relative frequency
```



Pie chart

```
> pie(table(data$sex))
```



Mean, median, quantiles

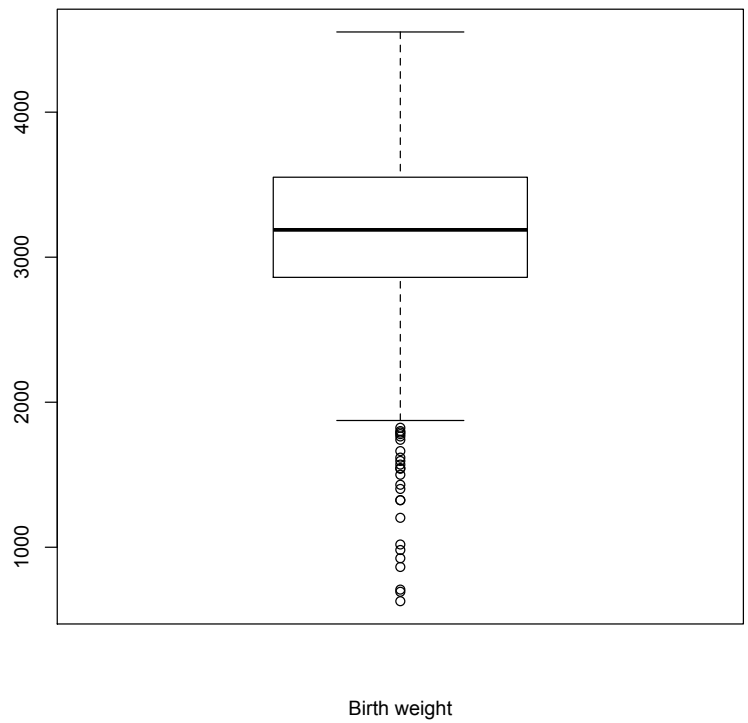
```
> summary(data$bweight)  #distribution of birth weight
   Min. 1st Qu.  Median    Mean 3rd Qu.
    628    2862    3188    3137    3551
   Max.
   4553
> mean(data$bweight)     #mean of birth weight
[1] 3136.884
> median(data$bweight)   #median of birth weight
[1] 3188.5
```

Quantiles and range

```
> quantile(data$bweight) #quantiles of birth weight
     0%     25%     50%     75%    100%
 628.00 2862.00 3188.50 3551.25 4553.00
> min(data$bweight)      #minimum of birth weight
[1] 628
> max(data$bweight)      #maximum of birth weight
[1] 4553
> range(data$bweight)    #range of birth weight
[1]  628 4553
```

Box plot

```
> boxplot(data$bweight,xlab="Birth weight")
#box-plot of birth weight
```



Summarizing...

```
> summary(data)
       id              bweight
 Min.   :  1.0   Min.   : 628
 1st Qu.:125.8   1st Qu.:2862
 Median :250.5   Median :3188
 Mean   :250.5   Mean   :3137
 3rd Qu.:375.2   3rd Qu.:3551
 Max.   :500.0   Max.   :4553

     lowbw           gestwks
 Min.   :0.00   Min.   :24.69
 1st Qu.:0.00   1st Qu.:37.94
 Median :0.00   Median :39.12
 Mean   :0.12   Mean   :38.72
 3rd Qu.:0.00   3rd Qu.:40.09
 Max.   :1.00   Max.   :43.16
                NA's   :10
    preterm            matage
 Min.   :0.0000   Min.   :23.00
 1st Qu.:0.0000   1st Qu.:31.00
```

Summarizing...

Index	Nominal qual.	Ordinal qual.	Quantitative
Mode	Yes	Yes	Yes
Mean	No	No	Yes
Median	No	Yes	Yes
Quantile	No	Yes	Yes
Boxplot	No	No	Yes
Range	No	No	Yes

Variability indices

```
> var(data$bweight)   #variance of birth weight
[1] 406344.4
> cv<-function(x){
+   sqrt(var(x)/abs(mean(x)))
+ }
> cv(data$bweight)    #variation coefficient of birth weight
[1] 11.38146
```

- Variation's coefficient is used to compare the variability of different phenomenons

Test for equality of two proportions

```
> n1<-8
> n2<-12
> x1<-rbinom(8,1,0.7)
> x2<-rbinom(12,1,0.5)
> p.test<-function(x,y,n,m){
+ p1<-sum(x)/n
+ p2<-sum(y)/m
+ p<-(sum(x)+sum(y))/(n+m)
+ se<- sqrt((n+m)*p*(1-p)/(n*m))
+ t.oss<-(p1-p2)/se
+ 1-pnorm(t.oss)
+ }
> p.test(x1,x2,n1,n2)
[1] 0.2219694
```

Test z for one sample

Let $\boldsymbol{X} = (X_1, \ldots, X_n)$ be a sample generated by a casual variable X: $E(X) = \mu$ e $Var(X) = \sigma^2$ known.

$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$$

For limit central theorem (approximately):

$$\frac{\overline{X} - \mu}{\sigma/\sqrt{n}} \sim N(0, 1)$$

$$H_0 : \mu = \mu_0 \qquad H_1 : \mu \neq \mu_0$$

$$H_0 : T = \frac{\overline{X} - \mu_0}{\frac{\sigma}{\sqrt{n}}} \sim N(0, 1)$$

Test z for one sample

$$H_0 : \mu = 74.5 \qquad H_1 : \mu \neq 74.5$$

```
> x<- c(75,76,73,75,74,73,76,76,73,79)
> z.test<-function(x,sigma,mu0){
+ n<-length(x)
+ xbar<-mean(x)
+ se<- sigma/sqrt(n)
+ t.oss<-(xbar-mu0)/se
+ 2*pnorm(-t.oss)
+ }
> z.test(x,1.5,74.5)
[1] 0.2918405
```

Test t for one sample

Let $\boldsymbol{X} = (X_1, \ldots, X_n)$ be a sample generated by a casual variable X: $X \sim N(\mu, \sigma^2)$, σ^2 unknown.

$$H_0 : \mu = \mu_0 \qquad H_1 : \mu \neq \mu_0$$

$$H_0 : T = \frac{\overline{X} - \mu_0}{\frac{S}{\sqrt{n}}} \sim t_{n-1}$$

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \qquad S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

Test t for one sample

$$H_0 : \mu = 74.5 \qquad H_1 : \mu \neq 74.5$$

```
> t.test(x,mu=74.5)

One Sample t-test

data:  x
t = 0.83853, df = 9, p-value = 0.4234
alternative hypothesis: true mean is not equal to 74.5
95 percent confidence interval:
 73.65111 76.34889
sample estimates:
mean of x
       75
```

t-test for no-paired samples with equal variance

Let $\boldsymbol{X} = (X_1, \ldots, X_n)$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$ be two samples from two independent casual variables $X \sim N(\mu_x, \sigma^2)$ e $Y \sim N(\mu_y, \sigma^2)$

$$H_0 : \mu_y - \mu_x = 0 \qquad H_1 : \mu_y - \mu_x \neq 0$$

$$H_0 : T = \frac{\overline{Y} - \overline{X}}{\sqrt{S^2(\frac{1}{n} + \frac{1}{m})}} \sim t_{n+m-2}$$

$$S_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2, \qquad S_Y^2 = \frac{1}{m-1}\sum_{i=1}^{m}(Y_i - \overline{Y})^2$$

$$S^2 = \frac{(n-1)S_X^2 + (m-1)S_Y^2}{n+m-2}$$

t-test for no-paired samples with equal variance

$$H_0 : \mu = 74.5 \qquad H_1 : \mu \neq 74.5$$

```
> x1<-c(15, 10, 13, 7, 9, 8, 21, 9, 14, 8)
> x2<-c(15, 14, 12, 8, 14, 7, 16, 10, 15, 12)
> t.test(x2,x1,var.equal=TRUE)

Two Sample t-test

data:  x2 and x1
t = 0.53311, df = 18, p-value = 0.6005
alternative hypothesis: true difference in means is not equal to
95 percent confidence interval:
 -2.646765  4.446765
sample estimates:
mean of x mean of y
     12.3      11.4
```

t-test for no-paired samples with equal variance

- Study aim: to evaluate if pupils' reading skills improve after some supplementary courses
- Treatment: Whether student participated in activities (treated) or not (control)
- Response: Score on Degree of Reading Power test

$$H_0 : \mu_1 = \mu_2 \qquad H_1 : \mu_1 \neq \mu_2$$

```
> data<-read.table("/Users/lab/Documents/R_2020/data/pupil.dat",
                   sep="",header=T)
> names(data)
[1] "Treatment" "Response"
```

t-test for no-paired samples with equal variance

```
> t.test(data$Response[data$Treatment=="Control"],
            data$Response[data$Treatment=="Treated"])

Welch Two Sample t-test

data:  data$Response[data$Treatment == "Control"]
and data$Response[data$Treatment == "Treated"]
t = -2.3109, df = 37.855, p-value = 0.02638
alternative hypothesis: true difference in means is not
equal to 0
95 percent confidence interval:
 -18.67588  -1.23302
sample estimates:
mean of x mean of y
 41.52174  51.47619
```

t-test for paired samples

$\boldsymbol{X} = (X_1, \ldots, X_n)$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$ two samples from two dependent casual variables $X \sim N(\mu_x, \sigma_x^2)$ and $Y \sim N(\mu_y, \sigma_y^2)$.

$$H_0 : \mu_y - \mu_x = 0 \qquad H_1 : \mu_y - \mu_x \neq 0$$

$$H_0 : T = \frac{\overline{Z} - 0}{\frac{S}{\sqrt{n}}} \sim t_{n-1}$$

$$Z = Y - X$$

$$\overline{Z} = \frac{1}{n} \sum_{i=1}^{n} Z_i, \qquad S^2 = \frac{1}{n-1} \sum_{i=1}^{n} (Z_i - \overline{Z})^2$$

t-test for paired samples

```
 > x1<-c(4.72, 3.42, 2.29, 3.77, 3.17, 4.63, 5.26, 4.70,
              4.01, 1.73)
> x2<-c(5.94, 3.58, 6.87, 6.41, 4.66, 5.65, 6.45, 5.43,
             4.66, 6.11)
> t.test(x2,x1,paired=TRUE)

Paired t-test

data:  x2 and x1
t = 3.6814, df = 9, p-value = 0.005065
alternative hypothesis: true difference in means is not
equal to 0
95 percent confidence interval:
 0.6962347 2.9157653
sample estimates:
mean of the differences
                  1.806
```

t-test for paired samples

- Study aim: compare the employment rates of women in 1972 and 1968
- City: City United States, 1972: Labor Force Participation rate of women in 1972, 1968: Labor Force Participation rate of women in 1968

```
> data<-read.table("/Users/lab/Documents/R_2020/data/women.dat",
                   sep="",header=T)
> names(data)
[1] "City"  "X1972" "X1968"
```

t-test for paired samples

```
> t.test(data$X1972,data$X1968,paired=T)

Paired t-test

data:  data$X1972 and data$X1968
t = 2.4577, df = 18, p-value = 0.02435
alternative hypothesis: true difference in means is not
equal to 0
95 percent confidence interval:
 0.004889895 0.062478527
sample estimates:
mean of the differences
              0.03368421
```

Cross-tabulation

```
> data<-read.table("/Users/lab/Documents/R_2020/data/
      data.births.csv",header = TRUE, sep = ",", row.names = 1)
> table(data$lowbw)  #tabulate the case-control variable

  0   1
440  60
> table(data$lowbw,data$sex)
  #cross-tabulation of low birth weight and sex

     1   2
  0 237 203
  1  27  33
```

Cross-tabulation

```
> mytable<-table(data$lowbw,data$sex)
   #save the cross-tabulation in table named mytable
> margin.table(mytable,1) #row total

  0   1
440  60
> margin.table(mytable,2) #column total

  1   2
264 236
```

Cross-tabulation

```
> prop.table(mytable) # percentages

       1     2
  0 0.474 0.406
  1 0.054 0.066
> prop.table(mytable,1) # row percentages

            1         2
  0 0.5386364 0.4613636
  1 0.4500000 0.5500000
```

Cross-tabulation

```
> prop.table(mytable, 2) # column percentages

            1         2
  0 0.8977273 0.8601695
  1 0.1022727 0.1398305
> summary(mytable) #chi-square test to evaluate the association
                   #between being low birth weight and sex
Number of cases in table: 500
Number of factors: 2
Test for independence of all factors:
Chisq = 1.6645, df = 1, p-value = 0.197
```

χ^2 test

```
> chi2<-chisq.test(mytable, correct = TRUE)
> chi2$expected

        1      2
  0 232.32 207.68
  1  31.68  28.32
> chi2$p.value
[1] 0.2491915
```

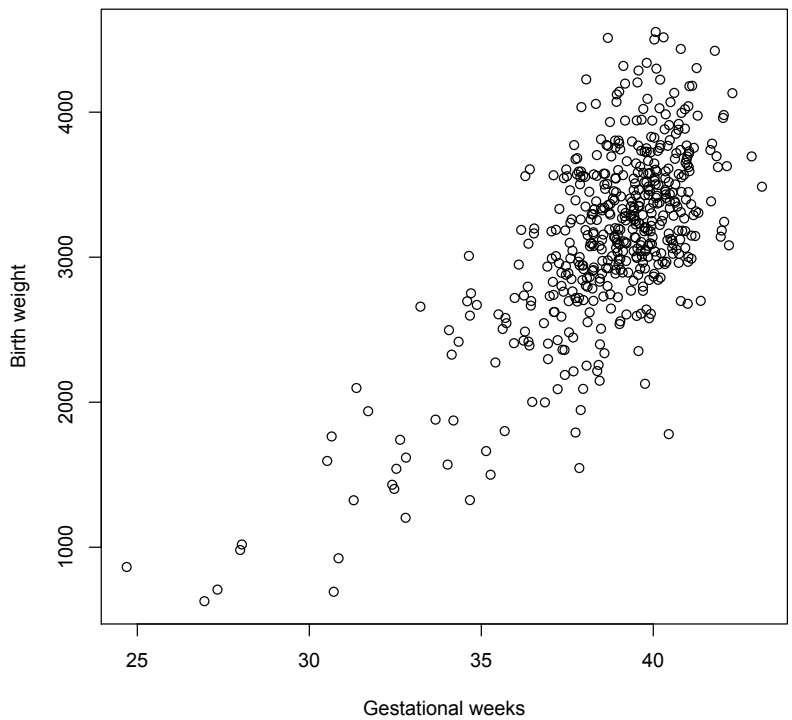
Tabulation of three variables

```
> mytable2<-table(data$lowbw,data$sex,data$hyp)
   #tabulation of three variables
> ftable(mytable2)
        0   1

0 1   206  31
  2   182  21
1 1    15  12
  2    25   8
> dim(mytable2) #table dimensions
[1] 2 2 2
```

See practical on Descriptive analysis in R (practical_descriptive) with corresponding solutions (solution_descriptive) and R script (main_descriptive).

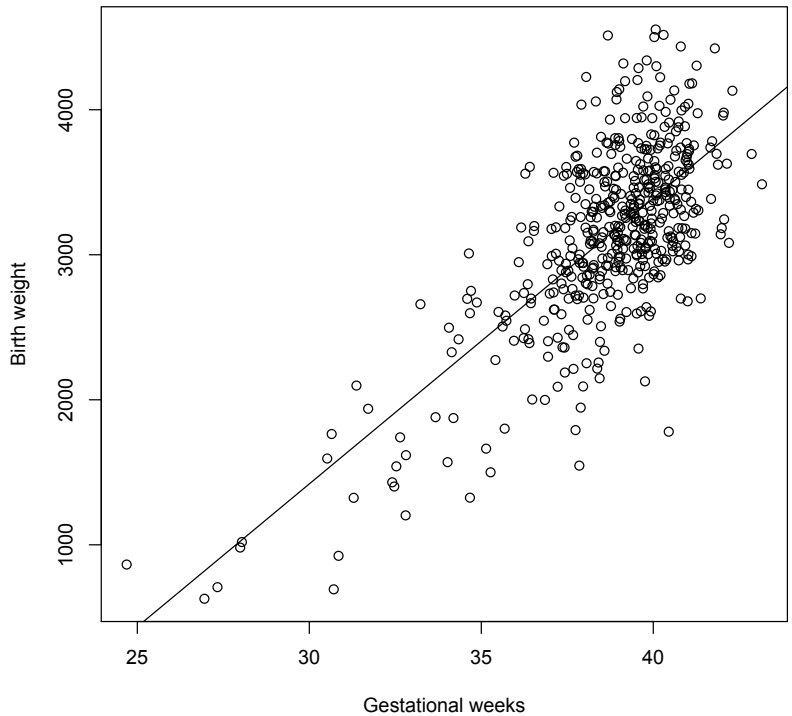
Linear regression

```
> plot(data$gestwks,data$bweight,xlab="Gestational weeks",
ylab="Birth weight")
```



Scatter plot with linear fit

```
> plot(data$gestwks,data$bweight,xlab="Gestational weeks",
ylab="Birth weight")
> abline(lm(data$bweight~data$gestwks))
```



Linear regression model

$$y = \beta_0 + \beta_1 x + \epsilon \qquad \epsilon \sim N(0, \sigma^2)$$

- Y is the dependent variable, X is the independent variable
- Y is normally distributed
- $Var(Y) = Var(\epsilon) = \sigma^2$
- β_0 is the intercept: value of Y when X=0
- β_1 is the slope: the increase of Y when X increases of one unit

```
> linear.model<-lm(bweight~gestwks,data=data)
```

Linear regression model

```
> summary(linear.model)

Call:
lm(formula = bweight ~ gestwks, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1698.40  -280.14    -3.64   287.61  1382.24

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4489.140    340.899  -13.17   <2e-16 ***
gestwks       196.973      8.788   22.41   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 449.7 on 488 degrees of freedom
  (10 observations deleted due to missingness)
Multiple R-squared:  0.5073,Adjusted R-squared:  0.5062
F-statistic: 502.4 on 1 and 488 DF,  p-value: < 2.2e-16
```

Linear regression model

```
> linear.model$coefficients
   #estimated coefficients from linear regression
(Intercept)     gestwks
 -4489.1398    196.9726
> confint(linear.model)
   #estimated confidence intervals from linear regression
                  2.5 %     97.5 \%
(Intercept) -5158.9503 -3819.3293
gestwks       179.7054   214.2399
```

Linear regression model

- The intercept (β_0) is -4489 (95% CI: -51590;-3819.3): when gestational weeks are 0 the estimated average infant birth weight is -4489 g (of difficult interpretation when gestational weeks variable is not centered at its mean) (p$<$0.001)
- The slope (β_1) is 197 (95% CI: 179.7;214.2): for a unit increase in gestational weeks the infant birth weight increases on average of 197 g (p¡0.001). There is evidence of an effect of gestational weeks on infant birth weight
- Multiple R-Squared=0.51: 51% of the total variability of infant birth weight is explained by gestational weeks
- F test compares the intercept-only model to the model that we specify (with gestational weeks): p$<$0.001 suggests the presence of an effect of gestational weeks on infant birth weight

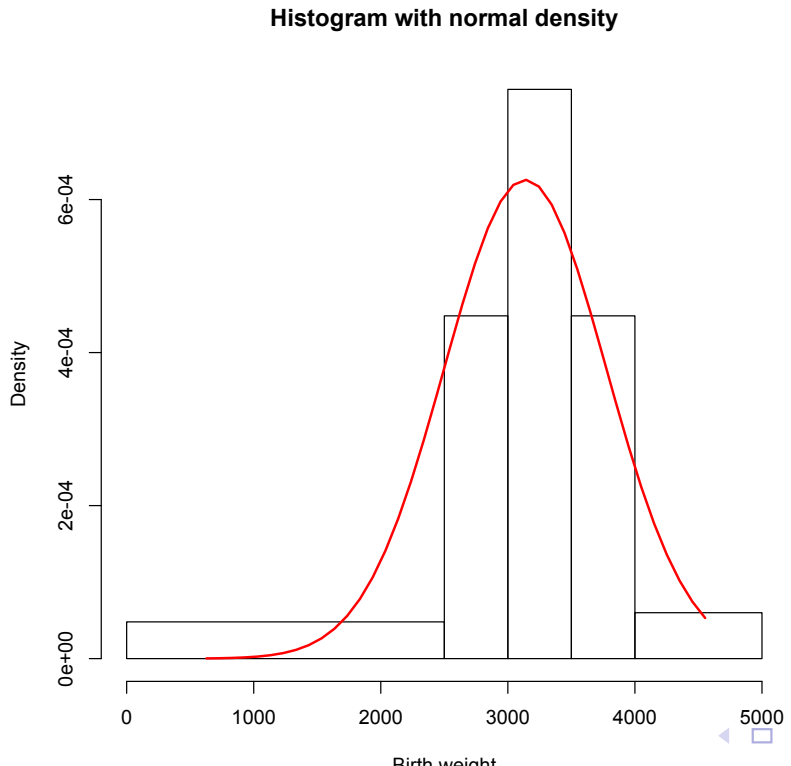
$$\widehat{y} = -4489.1 + 196.9x$$

Prediction

```
> predict(linear.model,data.frame(gestwks=c(37,38,39,40)))
       1        2        3        4
2798.847 2995.820 3192.793 3389.765
```

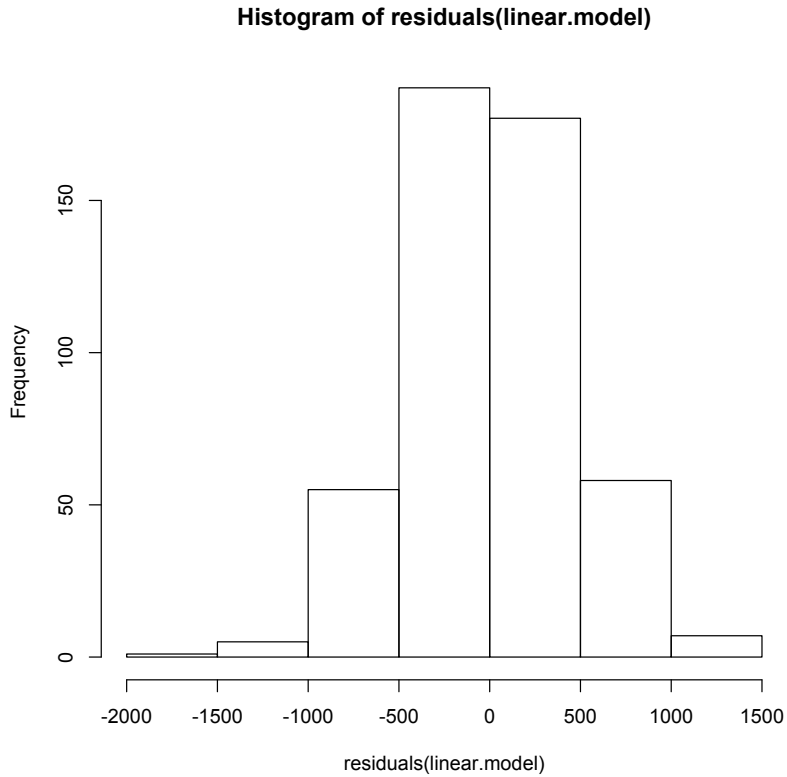
Normality assumption

```
h<-hist(data$bweight,c(0,2500,3000,3500,4000,5000),
    main="Histogram with normal density",xlab="Birth weight")
m<-mean(data$bweight)
std<-sqrt(var(data$bweight))
x<-seq(min(data$bweight), max(data$bweight), length = 40)
y<-dnorm(x, mean=m, sd=std)
lines(x,y,col="red", lwd=2)
```



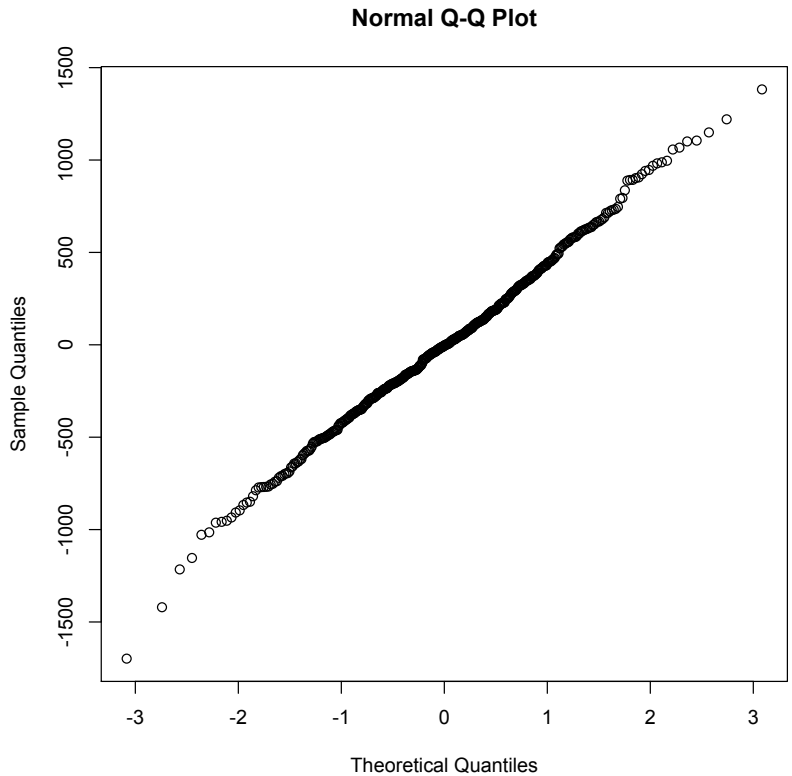
Residuals

```
> hist(residuals(linear.model))
```



qqplot

```
> qqnorm(residuals(linear.model))
```



Test for normality

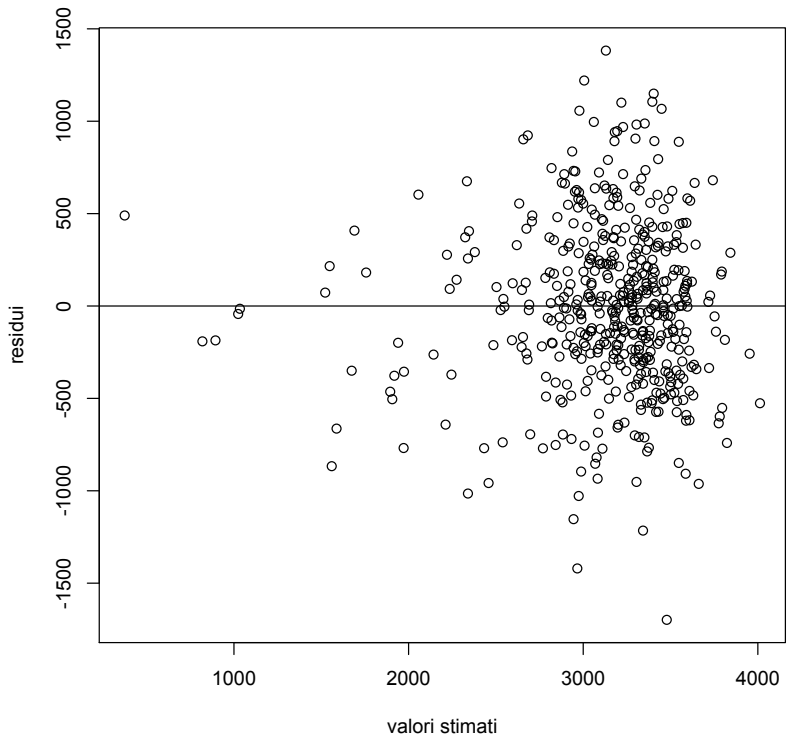
```
> shapiro.test(residuals(linear.model))

Shapiro-Wilk normality test

data:  residuals(linear.model)
W = 0.99672, p-value = 0.4241
```

- Test of normality: the null hypothesis is the normality
- p=0.42, there is no evidence against the normality in residuals' distribution

Deviance residuals

- Residuals have to be distributed casually (without any trend)

```
> plot(fitted(linear.model),residuals(linear.model),
      xlab="valori stimati",ylab="residui")
> abline(h=0)
```



Multivariable linear regression

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \epsilon$$

- Y is the dependent variable, X_1, X_2, X_3 are the independent variables
- β_0 is the intercept: value of Y when X=0
- β_1 measures the increase of Y when X_1 increases of one unit, maintaining constant X_2 and X_3
- β_2 measures the increase of Y when X_2 increases of one unit, maintaining constant X_1 and X_3
- β_3 measures the increase of Y when X_3 increases of one unit, maintaining constant X_1 and X_2

Multivariable linear regression

```
> linear.model1<-lm(bweight~gestwks+matage+hyp,data=data)
> summary(linear.model1)

Call:
lm(formula = bweight ~ gestwks + matage + hyp, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1710.42  -282.71    -8.73   282.78  1362.63

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -4259.3811   390.6432 -10.904   <2e-16 ***
gestwks       192.2524     8.9655  21.444   <2e-16 ***
matage         -0.7659     5.2060  -0.147   0.8831
hyp          -144.2234    58.9971  -2.445   0.0149 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 447.9 on 486 degrees of freedom
```

Multivariable linear regression

```
> linear.model1$coefficients
     #estimated coefficients from linear regression
 (Intercept)      gestwks       matage          hyp
-4259.381056   192.252362    -0.765879  -144.223427
> confint(linear.model1)
     #estimated confidence intervals from linear regression
                   2.5 %       97.5 %
(Intercept) -5026.93912 -3491.822991
gestwks       174.63642   209.868302
matage        -10.99496     9.463198
hyp          -260.14422   -28.302634
```

Multivariable linear regression

- The intercept is -4259.4 (95% CI:-5.026;-3491.8): value of birth weight when gestational weeks are 0, in women without hypertension and aged 0 at the beginning of pregnancy
- Comparing women with equal maternal age and hypertension status, the low birth weight increases on average of 192.2 g (95% CI:174.6;209.9) for an unit increase of gestational weeks: there is evidence of an effect of gestational weeks
- Comparing women with equal hypertension status and gestational weeks, the low birth weight decreases on average of -0.77 g (95% CI::-11.0;9.5) for an unit increase of gestational weeks: there is no evidence of an effect of maternal age
- Comparing women with equal maternal age and gestational weeks, the low birth weight increases on average of 144.2 g (95% CI:-260,1;-28.3) in women with hypertension compared to women without hypertension: there is evidence of an effect of hypertension

Multivariable linear regression

```
> predict(linear.model1,data.frame(gestwks=c(37,38,39,40),
        matage=c(30,30,30,30),hyp=c(1,1,1,1)))
       1        2        3        4
2686.757 2879.009 3071.261 3263.514
> predict(linear.model1,data.frame(gestwks=rep(c(37,38,39,40),2)
       ,matage=rep(30,8),hyp=c(rep(1,4),rep(0,4))))
       1        2        3        4        5        6        7
2686.757 2879.009 3071.261 3263.514 2830.980 3023.232 3215.485 3
```

See practical on Linear regression in R (practical_linear regression) with corresponding R script (main_linear).

Logistic regression model

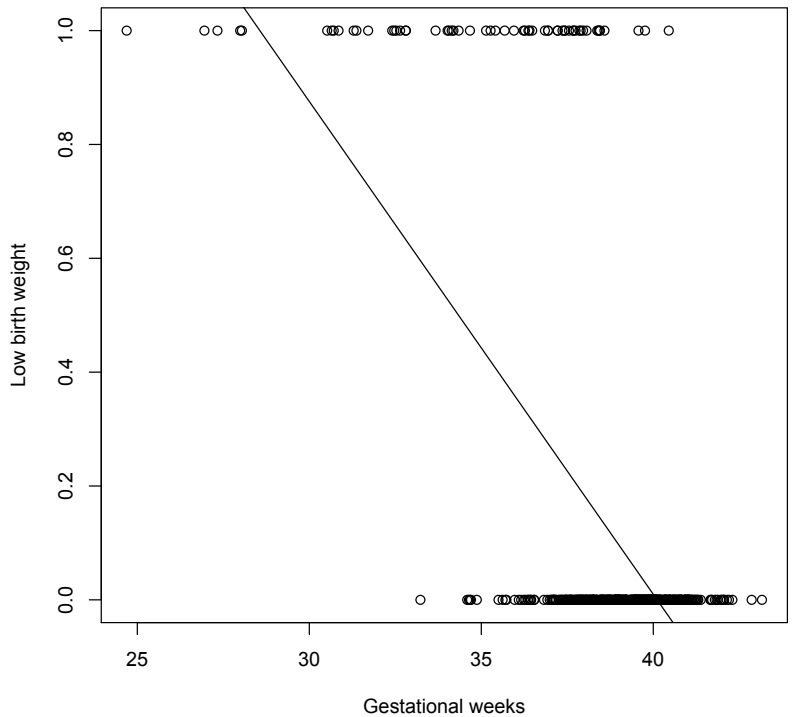
It measures association between a binary outcome variable and continuous or binary predictors

```
> data<-read.table("/Users/lab/Documents/R_2020/data/
  data.births.csv",header = TRUE,  sep = ",", row.names = 1)
> data[1:5,c(2,4)]
  bweight gestwks
1    2974   38.52
2    3270      NA
3    2620   38.15
4    3751   39.80
5    3200   38.89
```

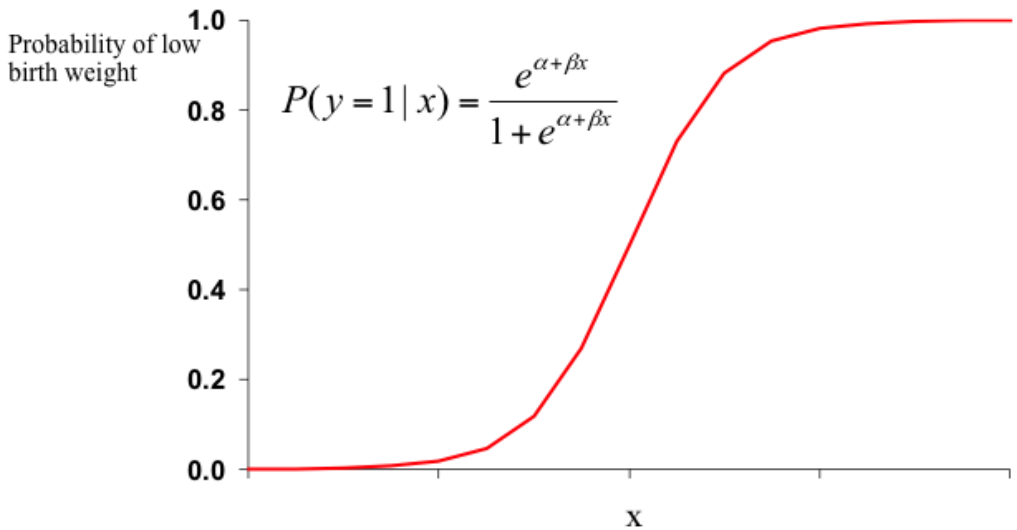
Scatter plot with linear fit

It measures association between a binary outcome variable and continuous or binary predictors

```
> plot(data$gestwks,data$lowbw,xlab="Gestational weeks",
    ylab="Low birth weight")
> abline(lm(data$lowbw~data$gestwks))
```



Logistic function



Logistic model

- $\frac{Pr(y=1|x)}{1-Pr(y=1|x)}$: odds of disease
- $log(\frac{Pr(y=1|x)}{1-Pr(y=1|x)})$: logit of $Pr(y = 1|x)$=log(odds)

$$Pr(y = 1|x) = \frac{e^{\alpha+\beta x}}{1 + e^{\alpha+\beta x}}$$

$$\frac{Pr(y = 1|x)}{1 - Pr(y = 1|x)} = e^{\alpha+\beta x}$$

$$log\left(\frac{Pr(y = 1|x)}{1 - Pr(y = 1|x)}\right) = \alpha + \beta x$$

Logistic model

$$log(odds(x+1)) = \alpha + \beta(x+1)$$

$$log(odds(x)) = \alpha + \beta x$$

$$log(odds(x+1)) - log(odds(x)) = log\left(\frac{odds(x+1)}{odds(x)}\right) = log(OR)$$

$$log(OR) = \alpha + \beta(x+1) - \alpha - \beta x = \beta$$

$$OR = exp(\beta)$$

Logistic regression model

- If x is continuous:
  α: log(odds) at x=0
  β: change in log(odds) for an unit increase of x
  $OR = exp(\beta)$: odds ratio for a unit increase of x

- If x is binary: α: log(odds) in unexposed subjects
  $\alpha + \beta$: log(odds) in exposed subjects
  β: change in log(odds) when subject is exposed
  $OR = exp(\beta)$: odds ratio of exposure

Logistic regression model

```
> logistic.model<-glm(lowbw~gestwks,data=data,family=binomial)
> summary(logistic.model)

Call:
glm(formula = lowbw ~ gestwks, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0873  -0.3623  -0.2223  -0.1369   2.9753

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  31.8477     4.0574   7.849 4.18e-15 ***
gestwks      -0.8965     0.1084  -8.272  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 360.38  on 489  degrees of freedom
```

Logistic regression model

```
> logistic.model$coefficients
     #coefficients estimated by logistic model
(Intercept)     gestwks
 31.8476573  -0.8964603
> confint(logistic.model)
   #confidence intervals estimated by logistic model
Waiting for profiling to be done...
                 2.5 %     97.5 %
(Intercept) 24.469787 40.4340771
gestwks     -1.126091 -0.6996476
```

Logistic regression model

```
> exp(-0.8965)   #crude OR
[1] 0.4079951
> exp(coef(logistic.model))
 (Intercept)      gestwks
6.780502e+13 4.080114e-01
> exp(confint(logistic.model))
Waiting for profiling to be done...
                    2.5 %       97.5 %
(Intercept) 4.237341e+10 3.633261e+17
gestwks     3.242986e-01 4.967603e-01
```

Logistic regression model

- Gestational weeks is continuous:
  $\alpha = 31.84$: log(odds) at 0 gestational weeks
  $\beta = -0.8965$: change in log(odds) for an unit increase of gestational weeks
  $OR = 0.41$: odds ratio for a unit increase of gestational weeks
  There is evidence that for each unit increase of gestational weeks, mothers are less likely to have low birth weight babies (OR=0.41, 95% CI: 0.32;0.50)

Logistic regression model

```
> logistic2.model<-glm(lowbw~hyp,data=data,family=binomial)
> summary(logistic2.model)

Call:
glm(formula = lowbw ~ hyp, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8067  -0.4430  -0.4430  -0.4430   2.1773

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.2721     0.1661 -13.682  < 2e-16 ***
hyp           1.3166     0.3111   4.232 2.32e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 366.92  on 499  degrees of freedom
```

Logistic regression model

```
> logistic2.model$coefficients
    #coefficients estimated by logistic model
(Intercept)          hyp
  -2.272126     1.316614
> confint(logistic2.model)
   #confidence intervals estimated by logistic model
Waiting for profiling to be done...
                  2.5 %     97.5 %
(Intercept) -2.6128761 -1.960165
hyp          0.6938372  1.919027
```

Logistic regression model

```
> exp(1.316614)   #crude OR
[1] 3.730768
> exp(coef(logistic2.model))
(Intercept)         hyp
  0.1030928   3.7307692
> exp(confint(logistic2.model))
Waiting for profiling to be done...
                  2.5 %    97.5 %
(Intercept) 0.07332335 0.1408352
hyp         2.00138051 6.8143241
```

Logistic regression model

- Hypertension is binary:
  $\alpha = -2.72$: log(odds) in no hypertensive women
  $\alpha + \beta = -0.95$: log(odds) in hypertensive women
  $\beta = 1.32$: change in log(odds) when woman is hypertensive
  $OR = exp(\beta)$=3.73: odds ratio of exposure
  There is evidence that mothers with hypertension are about 3.73 times more likely to have low birth weight babies than mothers without hypertension (OR=3.73, 95% CI: 2.00;6.81)

Multivariable logistic model

- The measures introduced are a crude estimate of the effect of the risk factor on the occurrence of the disease, as other factors related to the factor of risk associated with the occurrence of the disease, could contribute at its value (confounding factors)

$$log(odds) = \alpha + \beta_1 x_1 + \beta_2 x_2$$

```
> logistic.adj.model<-glm(lowbw~hyp+matage,data=data,
                          family=binomial)
```

Multivariable logistic model

```
> logistic.adj.model$coefficients
 (Intercept)           hyp       matage
-1.949468288  1.310004028 -0.009469024
> confint(logistic.adj.model)
Waiting for profiling to be done...
                    2.5 %     97.5 %
(Intercept) -4.37396505 0.39709198
hyp          0.68534017 1.91434871
matage      -0.07861054 0.06032476
```

Multivariable logistic model

- When adjusting the regression model for maternal age, mothers with hypertension are about 3.71 times more likely to have low birth weight babies than mothers without hypertension, there is evidence of an effect (OR=3.71, 95% CI: 1.98;6.78)

```
> exp(logistic.adj.model$coefficients)
(Intercept)         hyp      matage
  0.1423497   3.7061886   0.9905757
```

Prediction

- The probability of being low birth weight when maternal age is 34 years old is: 9.3% when mother is not hypertensive and 27.6% when mother is hypertensive

```
> newdata<-data.frame(matage=mean(data$matage),hyp=c(0,1))
> newdata
  matage hyp
1 34.028   0
2 34.028   1
> prediction<-predict(logistic.adj.model,newdata,
              type="response")
> prediction
         1          2
0.09349571 0.27654271
```

See practical on Logistic regression in R (practical_logistic regression) with corresponding R script (main_logistic).