UniLaSalle
Terre & Sciences

F.FABRI
Practical Tests (R Software)
Promotion 2023 - ISAFE

Programming with R: datasheet 4
Inferential statistics in R

I) Parametric tests

The mean

The *t.test* function allows you to manage all the tests of means. However, you will have to choose the right parameters depending on the test you want to perform. Below the arguments to use to perform the appropriate test for the situation.

Arguments	Descriptions
x	A numeric vector to perform a mean test *(x = seq (11))*
y	An optional numeric vector which you can use to perform the test of homogeneity of means (comparison between *x* and *y*). To do a test of conformity, do not assign any value. By default, it will be empty.
alternative	To perform a bilateral or unilateral test. By default, it will be bilateral. *alternative="two.sided"* ou *alternative="t"* *** bilateral test *alternative="greater"* ou *alternative="g"* *** unilateral test *alternative="less"* ou *alternative="l"* *** test unilateral
mu	A number indicating the true value of mean, by default it is equal to 0 (*mu* = *2*).
paired	This parameter specifies whether the data is paired or unpaired. By default, it is *FALSE*. In the case where the data are paired, it is necessary to specify (*paired=TRUE*).
var.equal	This parameter is only useful for homogeneity testing. If we can consider the variances as equal then we must put (*var.equal=TRUE*). By default it is FALSE.
conf.level	This is the argument is used to determine the confidence interval (percentage), by default it is 95%.
formula	To compare two groups of the same series. We can apply the following command: *formula=serie~factor* Where *serie* is a numeric variable giving the data values and *factor* a factor with two levels giving the corresponding groups.

Finally, you should know that we can find:

- **the test statistic as follows**: *t.test(seq(10))$statistic*
  The value we want is named "statistic"*. To extract it, we can use the dollar sign notation, or double square brackets: t.test(seq(10))[['statistic']]*
- **the confidence interval of the mean as follows:**

t.test(seq(10))$conf.int *** The population must be Gaussian or *n>30*

Examples: take the time to understand the results given!
Let's test at the risk of 5% if the height of a maple tree is 20 m with the following series:

18 ; 16 ; 23 ; 25 ; 20 ; 22 ; 19 ; 23 ; 17 ; 24

```
t.test(c(18,16,23,25,20,22,19,23,17,24),mu=20,conf.level=.95)
```

Conformity testing indicating mean = 1, bilateral test, confidence level 90%:

```
t.test(runif(100,0,2),mu=1,conf.level=.9)
```

Test of homogeneity of means considering variance as equal, unilateral test, conf. level is 95%.

```
t.test(runif(100,0,2),rnorm(150),var.equal=TRUE,alternative="g")
```

Homogeneity test of two groups of the same series according to the factor F classes with variances assumed to be equal, bilateral test at 95%

```
df=data.frame("S"=seq(100),"F"=c(rep("A",50),rep("B",50)))
t.test(formula=df$S~df$F,var.equal=TRUE)
```

Aplications :

🕐 After a treatment on a variety of animals, a sample of animals is taken and weighed. We obtain the weights in kg: 83; 81; 84; 80; 85. The average weight for this variety of animals is 87.6 kg. It is assumed that the weight of this variety of animals is normally distributed. Does the average weight of the treated animals differ significantly from the standard at the 5% risk?

Test statistic T =-5.39

🕐 We observed the amount of potassium (t / ha) existing in the soil of a beech forest and in the soil of a coppice, each time in six different places chosen at random and independently. Based on the following results, estimate the difference in means existing between the two media and test whether this difference is significant. On suppose que les populations parentes sont normales et de variances égales. We will take a 5% risk.

Beech grove	63	84	82	61	81	74
Coppice	64	60	45	59	48	59

Test statistic T =3.58

🕐 Tree heights were measured upright by a trigonometric method. To check whether this method does not consistently give too high or too low results, 12 of the trees were felled and measured on the ground. It is assumed that the populations are Gaussian. The results (in m) are as follows:

Standing trees	20,4	25,4	25,6	25,6	26,6	28,6	28,7	29	29,8	30,5	30,9	31,1
Felled trees	21,7	26,3	26,8	28,1	26,2	27,3	29,5	32	30,9	32,3	32,3	31,7

What can we conclude statistically at the 5% risk?

Statistique de test T=3.23 (fait en TD MOI3)

A) <u>The variance</u>

Only the homogeneity tests can be performed by a command on R (*var.test()*). The arguments to be used for this function can be found in the table below:

Arguments	Descriptions
x	A first numeric vector on which we want to perform a test of homogeneity of variance (*x=seq(11)*). The *x*= it is not mandatory.
y	A second numeric vector on which we want to perform a test of homogeneity of variance (*y=seq(101,111)*). The *y*= it is not mandatory.
alternative	To specify if you want to perform a bilateral or unilateral test. By default, it will be bilateral. - *alternative="two.sided"* ou *alternative="t"* *** bilateral test - *alternative="greater"* ou *alternative="g"* *** unilateral test - *alternative="less"* ou *alternative="l"* *** test unilateral
conf.level	This is the argument is used to determine the confidence interval (percentage), by default it is 95%.
formula	To compare two groups of the same series. We can apply the following command: *formula=serie~factor* Where *serie* is a numeric variable giving the data values and *factor* a factor with two levels giving the corresponding groups.

Examples :

Homogeneity of variance test, bilateral test, confidence level 90%:
var.test(rnorm(100,0,2),runif(100,0,1))

Homogeneity of variance test to compare two groups of the same series divided into two-state factor (populations considered Gaussian), bilateral test at 90%
df=data.frame("Norm"=rnorm(100),"F"=c(rep(c("A","B"),50)))
var.test(formula=df$Norm~df$F,conf.level=0.9)

Aplications :

- The same variety of oats was sown at different densities in two fields and yields from eleven small plots were observed on both sides. Based on the following results (in g / m2), check whether the variances can be considered equal. The populations are normal.

Density 1	538	491	508	438	382	409	433	491	420	547	478
Density 2	450	423	439	463	356	431	467	335	326	431	393

Test statistic F=1.148

B) Proportion test

The command used for this test is: *prop.test*() The arguments that can be used for this function are explained in the table below :

Arguments	Descriptions
x	For proportion compliance tests, *x* is a vector of counts of successes. For the homogeneity of proportions tests, x is a 2 * 2 matrix. The first column vector represents the number of successes for the first group and the second group. The second column vector represents the number of failures of the first group then of the second group.
n	To be used only for compliance testing, it represents the total number of tests.
p	To be used only for compliance testing, it represents the probability to be tested. The default is p = 0.5.
alternative	To specify if you want to perform a bilateral or unilateral test. By default, it will be bilateral. - *alternative="two.sided"* ou *alternative="t"* *** bilateral test

	• *alternative="greater"* ou *alternative="g"* *** unilateral test • *alternative="less"* ou *alternative="l"* *** test unilateral
conf.level	This is the argument is used to determine the confidence interval (percentage), by default it is 95%.
correct	Set the value FALSE. By default, this parameter is TRUE. It brings a slight correction to the test seen in progress.

Examples:

Proportion conformity test *p=0.5*, bilateral test at 95%:

```
prop.test(length(which(rnorm(1000)>=0)),1000,correct=FALSE)
```

Homogeneity test of proportions between two groups, bilateral test at 90%:

	Succès	Echecs
Groupe 1	50	100
Groupe 2	50	150

```
M=matrix(c(50,50,100,150),2,2)
prop.test(M,conf.level=0.9,correct=FALSE)
```

Aplications :

🕐 We observe a species of migratory birds passing a pass and note their sex. There are 81 females and 75 males in a sample. We assume that the proportion of women migrating on this day is p = 0.41. Can we say, at the risk of 5%, that the proportion of females is 0.41 or admit that it is higher?

Test statistic U=2.774

🕐 We are interested in the onset of a certain disease. In a group of 200 goats, we are experimenting with the effects of a vaccine. Another group of 200 goats served as control. The following results were obtained after a certain period:

	Vaccinated group	A group of witnesses
Onset of the disease	20	40
No onset of disease	180	160

Are the observed differences significant (□= 5 %)?

Test statistic U=-2.8

Note :

On peut retrouver l'**intervalle de confiance d'une proportion** comme suit :

```
prop.test(length(which(rnorm(1000)>=0)),1000,correct=FALSE)$conf.int
```

II) Non-parametric tests

A) Shapiro-Wilk test

To test the normality of the data. It is possible to use a simple test in R. This is the Shapiro-Wilk test. The function is *shapiro.test()*. There is only one parameter to use: the series to test. There is a technical constraint. The size of the series must be between 3 and 5000 data.

Examples :

```
shapiro.test(rnorm(1000))
shapiro.test(rbinom(1000,1000,.5))
shapiro.test(runif(1000))
```

Do the results obtained seem consistent to you?

B) Conformity tests with respect to a theoretical chi-square law

To perform a chi-square test of conformity with respect to a theoretical law, the function is *chisq.test()*. The usable arguments are explained below:

Arguments	Descriptions
x	*x* is the vector of observed frequency
p	*p* is the vector of theoretical probabilities. By default, the uniform law is tested.

Examples:

Conformity test with respect to a uniform law on *{1,…, 6}* for a dice.

Side	1	2	3	4	5	6
Observed frequency	12	11	14	15	30	5

chisq.test(c(12,11,14,15,30,5))

Conformity test with respect to fish low with lambda parameter =1.48

```
fish=function(i,lambda=1)
{	return(exp(-lambda)*lambda^i/factorial(i))}
proba=fish(0:3,1.48) ; chisq.test(c(37,46,39,19,9),p=c(proba,1-sum(proba)))
```

Aplications :

⌚ The Mendelian theory of heredity suggests that by crossing 2 types of plants, we must obtain products of type A, B, C and D in the proportions 9/16, 3/16, 3/16 and 1 / 16. After the result of experiments, 154 products of type A, 44 of type B, 63 of type C and 21 of type D are observed. What can we think, in this case, of the Mendelian theory, (□=5 %)?

⌚ A test on a variety of wheat is carried out on 500 plots. The yield of this variety is noted between 0 and 5 for all the plots. The results are given below

Yield score	0	1	2	3	4	5
Observed frequency	20	89	157	154	73	7

Test the binomial law *B(5,0.4768)*. We can use the command *dbinom(0:5,5,0.4768)* to obtain the probabilities of the binomial distribution of parameter 5 and 0.4768.

C) Chi-Square Independence Tests

To perform a chi-square test of conformity with respect to a theoretical law, the function is *chisq.test()*. The usable parameters are explained below:

Arguments	Descriptions
x	Can *x* correspond to the contingency table (matrix format) or else *x* corresponds to a vector (*y* must be entered). R will take care of making the contingency table between *x* and *y*.
y	Do not forget if we entered for *x* as a vector.

Example:

We are interested in *M* disease in cattle. Following the use of treatments, A or B, we have grouped below the condition of the cattle having been treated with treatments A or B

	Healing (H)	Amelioration (A)	Steady state (S)	Totals
A	280	210	110	600
B	220	90	90	400
Totals	500	300	200	1.000

Can we say that treatments A and B are different?

First version: matrix writing

```
chisq.test(matrix(c(280,220,210,90,110,90),2,3))
```

Second version: vector writing (x and y are factors)

```
x=c(rep("G",500),rep("A",300),rep("E",200))
y=c(rep("A",280),rep("B",220),rep("A",210),rep("B",90),rep("A",110),rep("B",90))
chisq.test(x,y)
```

III) Confidence interval applications

🕐 Apply the following code in R

```
n=100
x=rnorm(n,0,1)
bi=c()
bs=c()
alpha=0.05
for (beta in seq(0.005,0.05,0.005))
{
        bi=c(bi,(n-1)*var(x)/qchisq(1-beta,n-1))
        bs=c(bs,(n-1)*var(x)/qchisq(alpha-beta,n-1))
}
I=bs-bi
I
for (i in 1:10)
{
        beta=0.005+(i-1)*0.005
        print(c(bi[i],bs[i],I[i],alpha-beta,1-alpha,beta))
}
k=which(I==min(I))
beta=0.005+(k-1)*0.005
print(c(min(I),alpha-beta,1-alpha,beta))
```

What do you think?

🕐 Apply the following code in R

```
tmp=0
for (i in 1:100)
{
        a=rnorm(100)
        if ((t.test(a)$conf.int[1]<=0)&(t.test(a)$conf.int[2]>=0))
                tmp=tmp+1
}
tmp
```

What do you think?