Introduction to R

4th lecture

Alessandro FERMI – Giovanna VENUTI
Outline of the lecture

In this lecture we will introduce

- Remind of probability distribution in R
- Basic built-in tools for hypothesis testing
- Statistical models in R
- Linear models for multiple regression
- One- and two-way analysis of variance
Probability distributions in R

In R all most important probability distributions are implemented. A list of them can be found in the manuals.

Functions are provided to evaluate density and distribution functions and to compute any quantile $P(X < l) > q$.

Prefix the name of the probability distribution by
- ‘d’ for the density,
- ‘p’ for the CDF,
- ‘q’ for the quantile function
- ‘r’ for simulation (random deviates).

The first argument is x for dxxx, q for pxxx, p for qxxx and n for rxxx
Example

• extract a sample from Student’s t-distribution with degree of freedom equal to 10 (for instance)

• use the function qqnorm() to compare this sample with the normal distribution

• extract a sample from the Fischer distribution (degree of freedom $df1 = 5$, $df2 = 7$) and compare this sample with the normal distribution

Remark. To generate a random sample from a uniform distribution
You may also use the ‘runif()’ function.
Moreover to generate a random vector of integers the function ‘sample()’ is available.
Hypothesis testing in R

In R many «classical» tests for hypothesis testing are implemented!
Let us continue the example with the data frame ‘faithful’.

Example

```r
> F_long3 <- ecdf(long3)
> x <- seq(0.0, by=0.01, to=5.5)
> lines(x, pnorm(x, mean=mean(long3), sd=sqrt(var(long3))), lty=3)
```

We can carry out a Shapiro-Wilk test for checking the normality

```r
> shapiro.test(long3)
```

Shapiro-Wilk normality test

data:  long3
W = 0.9793, p-value = 0.01052

The null hypothesis is accepted!
Furthermore, we can also carry out a Kolmogorov-Smirnov test on the shape of the distribution density.

**Example**

```r
> ks.test(long3, "pnorm", mean = mean(long3), sd = sqrt(var(long3)), alternative="two.sided")
```

One-sample Kolmogorov-Smirnov test

data: long3  
D = 0.0661, p-value = 0.4284  
alternative hypothesis: two-sided

The null hypothesis is accepted.
Hypothesis testing in R

Now if we want to carry out, for instance, a t-test to test whether our sample mean is equal to some theoretical value, we may consider the function

\[ \text{t.test}(x, y = \text{NULL}, \text{alternative} = \text{c("two.sided", "less", "greater")}, \mu = 0, \text{paired} = \text{FALSE}, \text{var.equal} = \text{FALSE}, \text{conf.level} = 0.95, \ldots) \]

**Example.**

> t.test(eruptions, mu=3.6, var.equal=FALSE)

One Sample t-test

data: eruptions
\text{t} = -1.6215, \text{df} = 271, \text{p-value} = 0.1061
alternative hypothesis: true mean is not equal to 3.6
95 percent confidence interval:
3.351534 3.624032
sample estimates:
mean of x
3.487783
Hypothesis testing in R

We want to carry out a t-test to test whether two sample means are equal.

t.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), mu = 0, paired = FALSE, var.equal = FALSE, conf.level = 0.95, ...)

Example

```r
> zwd_daniS <- read.table('C:/Users/Alessandro/Documents/R_Work/DAN1SIGMAALL.TRP', header=TRUE)
> zwd_daniQ <- read.table('C:/Users/Alessandro/Documents/R_Work/DANIQIFALL.TRP', header=TRUE)
> t.test(zwd_daniS$CORR_U, zwd_daniQ$CORR_U, var.equal=FALSE)
```
t.test(zwd_daniS$CORR_U, zwd_daniQ$CORR_U, var.equal=FALSE)

Welch Two Sample t-test

data:  zwd_daniS$CORR_U and zwd_daniQ$CORR_U
t = 0.1294, df = 340, p-value = 0.8971
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.006713596  0.007659327
sample estimates:
mean of x  mean of y
0.1338109  0.1333380

> qt(0.975, df=340)
[1] 1.966966

The null hypothesis is accepted!
As seen above, we have considered the optional argument “var.equal=FALSE” and carried out a Welch test, which is an adaptation of Student’s t test.

We can apply a F-test to check for equality in the variances of the two samples, provided that the two samples are from normal distributions. This will also enable us to directly apply a Student’s t-test.

The general syntax is

\[
\text{var.test}(x, y, \text{ratio} = 1, \text{alternative} = c("two.sided", "less", "greater"), \text{conf.level} = 0.95, \ldots)
\]

**Example. 1)** Perform a F-test between the samples `zwd_daniS$CORR_U` and `zwd_daniQ$CORR_U`

2) Carry out a Student’s t-test assuming equal variances if the F-test holds true.
Hypothesis testing in R

The output of the F-test is

```r
> var.test(zwd_daniS$CORR_U, zwd_daniQ$CORR_U)

F test to compare two variances

data:  zwd_daniS$CORR_U and zwd_daniQ$CORR_U
F = 0.9983, num df = 170, denom df = 170, p-value = 0.991
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
  0.7383171 1.3497626
sample estimates:
ratio of variances
  0.9982749
```

The null hypothesis is true!
Hypothesis testing in R

The output of a Student’s t-test is

```r
> t.test(zwd_daniS$CORR_U, zwd_daniQ$CORR_U, var.equal=TRUE)

Two Sample t-test

data:  zwd_daniS$CORR_U and zwd_daniQ$CORR_U
t = 0.1294, df = 340, p-value = 0.8971
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  -0.006713596  0.007659327
sample estimates:
mean of x  mean of y
0.1338109  0.1333380
```

The null hypothesis is true!
R provides many features to make fitting and analysis of statistical models simple and efficient.

Recall that a general linear statistical model is given in matrix notation by

\[ y = Ax + b + e \]

where \( e \sim N(0, \sigma^2) \) (independent, homoscedastic errors).

The operator \( \sim \) is used in R to define a model formula. The general syntax is

\[ \text{obs} \sim \text{op}_1 \text{term}_1 \text{op}_2 \text{term}_2 \ldots \text{op}_k \text{term}_k \]

where
- \text{obs} is the vector (or matrix) defining the observations or response variables
op_i is an operator (+ or -) implying the inclusion or exclusion of regression variables.

term_i is either

- vector or matrix or 1
- factor
- a formula expression consisting of factors, vectors or matrices connected by formula operators.

Examples.

\[ y \sim x \]
\[ y \sim 1 + x \] : simple linear regression of y on x; the first has an implicit intercept term, the second an explicit one.

\[ \log(y) \sim x1 + x2 \] : multiple regression of the transformed obs on two independent variables x1 and x2.
y ~ A : single classification analysis of variance model on y, with classes
determined by the factor A

y ~ A*B*C – A:B:C
y ~ (A+B+C)^2 : three factor experiments with a model containing main
effects and two factor interactions only. Both formulae
specify the same model.

y ~ A*B + Error(C) : an experiment with two treatment factors A and B and
error strata determined by the factor C
Statistical models in R

Example.

```r
> x <- 1:20
> w <- 1 + sqrt(x)/2
> dummy <- data.frame(x=x, y= x + rnorm(x)*w)
> fit <- lm(y ~ x, data=dummy)
> anova(fit)

> anova(fit)
Analysis of Variance Table

Response: y

                     Df Sum Sq Mean Sq F value  Pr(>F)
 x                    1 740.13  740.13  80.589 4.574e-08 ***
 Residuals          18 165.31    9.18
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```
The general syntax of the function `lm()` is

```r
> fitted.model <- lm(formula, data=data.frame)
```

Its output is an object of class 'lm'. Information from `fitted.model` can be extracted by several generic functions, among which we mention:

- Anova
- Plot
- Print
- Summary
- Residuals
- Deviance
- Etc...

Many more information can be found in the manual.
Another important function used to fit linear models is the `aov()` function.

```r
> fitted.model <- aov(formula, data=data.frame)
```

This function allows an analysis of models and an error term can be also added.

```r
> fitted.model <- aov(response ~ mean.formula + Error(strata.formula), data=data.frame)
```

Many extraction functions defined for `lm()` can be used for `aov()` as well.
Example. We want to verify if the following vector is dependent of the factors A and B, or if they are independent.

```r
> income = c(15,18,22,23,24, 22,25,15,15,14, 18,22,15,19,21,
+ 23,15,14,17,18, 23,15,26,18,14, 12,15,11,10,8, 26,12,23,15,18,
+ 19,17,15,20,10, 15,14,18,19,20, 14,18,10,12,23, 14,22,19,17,11,
+ 21,23,11,18,14)
> A <- gl(12,5)
> B <- gl(5,1,60)
> fit <- aov(income ~ A + B)
> anova(fit)
```
Example. Analysis of Variance Table

Response: income

<table>
<thead>
<tr>
<th></th>
<th>Df</th>
<th>Sum Sq</th>
<th>Mean Sq</th>
<th>F value</th>
<th>Pr(&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>11</td>
<td>308.45</td>
<td>28.041</td>
<td>1.4998</td>
<td>0.1660</td>
</tr>
<tr>
<td>B</td>
<td>4</td>
<td>44.17</td>
<td>11.042</td>
<td>0.5906</td>
<td>0.6712</td>
</tr>
<tr>
<td>Residuals</td>
<td>44</td>
<td>822.63</td>
<td>18.696</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example. Suppose to have the following table

<table>
<thead>
<tr>
<th>Subject</th>
<th>time1</th>
<th>time2</th>
<th>time3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>45</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>56</td>
<td>18</td>
<td>29</td>
</tr>
<tr>
<td>3</td>
<td>59</td>
<td>10</td>
<td>24</td>
</tr>
<tr>
<td>4</td>
<td>49</td>
<td>15</td>
<td>25</td>
</tr>
</tbody>
</table>

A parameter has been measured in four subjects in three different epochs. Does time influence the measurements?
Let us construct the data frame in R.

```r
subj <- rep(1:4, each=3)
time <- rep(c("time1", "time2", "time3"), 4)
weights <- c(45, 20, 30, 56, 18, 29, 59, 10, 24, 49, 15, 25)
mydata <- data.frame(factor(subj), factor(time), weights)
names(mydata) <- c("subj", "time", "weights")
mydata
myanova <- aov(weights ~ time + Error(subj/time), data=mydata)
summary(myanova)
```

Error: subj
Df Sum Sq Mean Sq F value Pr(>F)
Residuals 3 34.667 11.556

Error: subj:time
Df Sum Sq Mean Sq F value Pr(>F)
time 2 2795.17 1397.58 49.086 0.0001911 ***
Residuals 6 170.83 28.47
Statistical models in R

Error: subj
Df Sum Sq Mean Sq  F value  Pr(>F)
Residuals  3 34.667 11.556

Error: subj:time
Df Sum Sq Mean Sq  F value  Pr(>F)
time  2 2795.17 1397.58  49.086 0.0001911 ***
Residuals  6  170.83  28.47

---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

The symbol «***» means that the differences among the three groups are statistically significant.
THANK YOU FOR YOUR ATTENTION!