Initiation to R software Session VI

Pierre Michel

Master AMSE 1st year, 2019

Statistics in R

Introduction: statistics in R

Functions available for statistical analysis:

- Package `stats`: functions for classical statistical analysis (hypotheses tests, linear models, distributions, statistical summary, time series, multivariate analyses). It is loaded when R starts. USe `library(help = "stats")`
- Specific packages: other packages available for statistical methods, distributed with R or downloadable: `class` (for classification), `boot` (for bootstrap), `survival` (for survival analysis)...

Introduction: statistical modelling

Functions from `stats` for **statistical modelling**:

function_name	description
SSlogis	Logistic model
aov	Analysis of variance
glm	Generalized linear models
lm	Linear models
manova	Mutivariate analysis of variance
ksmooth	Kernel smoother regression
step	Model selection with Stepwise algorithm

Introduction: statistical tests

Functions from `stats` for **statistical tests**:

function_name	description
bartlett.test	Bartlett's variances homogeneity test
binom.test	Exact binomial test
chisq.test	Chi2 test
cor.test	Correlation test
t.test	Student's means comparison t-Test
var.test	Fisher's variances comparison F-test
wilcox.test	Wilcoxon's rank test
ks.test	Kolmogorov-Smirnov's test

Introduction: estimation and data analysis

Functions for **estimation**:

function_name	description
density	Kernel density estimation
ecdf	Empirical cumulative distribution function

Functions for **data analysis**:

function_name	description
hclust	Hierarchical clustering
kmeans	Partitioning clustering
princomp	Principal components analysis
dist	Computes distance matrices

Formulas

Formulas: syntax

Many statistical functions use formulas: `function(formula, data, ...)`

- `formula` of type `response ~ predictors`
- `data` data table containing the variables in `formula`

`response` is the target variable, `predictors` is the set of predictor variables, separated by arithmetic symbols.

Formulas: examples

- `y~a`: predictor `a`
- `y~a+b`: predictors `a` and `b`
- `y~M`: as many models as predictors in `M` (matrix)
- `y~x-1`: model without intercept
- `y~log(b)`: predictor `log(b)`
- `y~a+I(b+c)`: predictors `a` and `b+c`
- `y~a:b`: interaction of `a` and `b`
- `y~a*b:a+b+a:b`: main effects and interaction of `a` and `b`
- `y~(a+b)^2`: main effects `a` and `b` and second-order interaction of `a+b+a:b`
- `y~a*b-a:b`: `a+b`

Formulas: examples with formulas

Example 1: Simple linear regression of y on x (x and y quantitative).

```
x=sample(1:10,200,replace=TRUE)
y=3+7*x+rnorm(200,0,100)
linreg=lm(y~x)
```

Example 2: Multivariate linear regression of fertility on education and infant mortality (all variables quantitative).

```
data(swiss)
mlinreg = lm(Fertility~Education+Infant.Mortality,swiss)
```

Example 3: 1-factor analysis of variance: analyze the effect of a bug spray (6 different types) on the number of insects (count), based on the observation of 12 cultures sprayed succesively with the bug sprays. The response is qualitative.

```
data(InsectSprays)
anova = aov(sqrt(count) ~ spray, data = InsectSprays)
```

Formulas: examples without formulas

Example 4: χ^2-test to test the relation between qualitative variables X and Y, based on a sample in the form of a contengency table.

```
O = matrix(c(442,514,38,6),nrow=2,byrow=TRUE)
colnames(O) = c("male","female");
rownames(O) = c("sighted","blind")
X2 = chisq.test(O,correct=FALSE)
```

Example 5: Student's t-test to test the equality between two means, based on a sample of each subpopulation $X = (X_1, .., X_{n_1})$ and $Y = (Y_1, .., Y_{n_2})$

```
x = rnorm(100,1,1); y = rexp(200,1)
st = t.test(x,y)

x = rnorm(100,1,1); z = rep(c(T,F),50)
st2 = t.test(x~z)
```

Outputs

Outputs: Linear regression (`lm()`)

```
linreg
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Coefficients:
## (Intercept)            x
##      29.610        4.592
```

Outputs: Multivariate linear regression (`lm()`)

```
mlinreg
```

```
##
## Call:
## lm(formula = Fertility ~ Education + Infant.Mortality, d
##
## Coefficients:
##      (Intercept)         Education  Infant.Mortality
##          48.8213           -0.8167             1.5187
```

Outputs: Analysis of variance (aov())

```
anova
```

```
## Call:
##    aov(formula = sqrt(count) ~ spray, data = InsectSpray
##
## Terms:
##                       spray Residuals
## Sum of Squares  88.43787  26.05798
## Deg. of Freedom         5         66
##
## Residual standard error: 0.6283453
## Estimated effects may be unbalanced
```

Outputs: χ^2-test (`chisq.test()`)

```
X2
```

```
##
##  Pearson's Chi-squared test
##
## data:  O
## X-squared = 27.139, df = 1, p-value = 1.894e-07
```

Outputs: Student's *t*-test (`t.test()`)

```
st
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -0.16849, df = 197.7, p-value = 0.8664
## alternative hypothesis: true difference in means is not
## 95 percent confidence interval:
##  -0.2606861  0.2196456
## sample estimates:
## mean of x mean of y
## 0.9942497 1.0147699
```

Outputs: Student's *t*-test (`t.test()`)

```
st2
```

```
##
##  Welch Two Sample t-test
##
## data:  x by z
## t = -0.77103, df = 92.686, p-value = 0.4427
## alternative hypothesis: true difference in means is not
## 95 percent confidence interval:
##  -0.5958588  0.2625704
## sample estimates:
## mean in group FALSE  mean in group TRUE
##           0.8617696           1.0284138
```

Outputs

In R, statistical functions return an object of class inherited from the name of the function (e.g `aov()` return an object of class `aov`, `lm()` returns an object of class `lm`, tests return objects of class `htest`).

The returned object contains the results of the analysis. This is generally a list, whose displayed results depend on its class (e.g `aov()` and `lm()` do not provide the same lists).

The elements in the list-object can be printed using `attributes()` and `names()`.

Outputs: Linear regression and ANOVA

```
names(linreg)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "ra
##  [5] "fitted.values" "assign"        "qr"            "di
##  [9] "xlevels"       "call"          "terms"         "mo
```

```
names(anova)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "ra
##  [5] "fitted.values" "assign"        "qr"            "di
##  [9] "contrasts"     "xlevels"       "call"          "te
## [13] "model"
```

Outputs: χ^2-test

```
attributes(X2)
```

```
## $names
## [1] "statistic" "parameter" "p.value"   "method"    "dat
## [7] "expected"  "residuals" "stdres"
##
## $class
## [1] "htest"
```

Outputs: Simple/Multivariate linear regression

```r
coefficients(mlinreg) # or mlinreg$coefficients
```

```
##       (Intercept)         Education Infant.Mortality
##        48.8212736        -0.8166573         1.5187190
```

```r
residuals(linreg) # or linreg$residuals
```

```
##             1             2             3             4
## -197.9339498  -95.3936445  -43.2006639  -61.2268520   43
##             6             7             8             9
##  -90.4280540  -35.8655301   19.5964264 -140.8449593  -71
##            11            12            13            14
##  118.3705958   33.5042972  -84.9596497  -65.3849881  -23
##            16            17            18            19
##   65.1376230 -115.2425095 -133.4402101  116.0808205 -103
##            21            22            23            24
##  146.3313526  -20.4186140 -177.7767247  160.9829140   55
##            26            27            28            29
## -155.6741978  134.3449689  -19.6699709  127.5716138  112
```

Outputs: χ^2-test

```
X2$expected # theoretical counts
```

```
##              male female
## sighted 458.88 497.12
## blind    21.12  22.88
```

```
X2$residuals # residuals (theoretical - observed)
```

```
##                male     female
## sighted -0.7879939  0.7570801
## blind    3.6730385 -3.5289413
```

```
sum(X2$residuals^2) # Khi2 statistic value
```

```
## [1] 27.13874
```

Generic functions

Generic functions

Some functions are also used to extract the desired results: the **generic functions**.

They work specifically, according to the object class.

Generic functions have a **single syntax** for all cases.

Generic functions

- `print()` returns a short summary of the analysis.
- `summary()` returns a detailed summary of the analysis.
- `df.residuals()` returns the number of degrees of freedom of residual.
- `coef()` returns the estimated coefficients (sometimes with standard errors).
- `residuals()` returns the residuals.
- `fitted()` returns values predicted by the model.
- `logLik()` computes the log-likelihood and the number of parameters of a model.
- `AIC()` computes Akaike's information criterion.
- `anova()` table of analysis of variance.
- `plot()` returns a plot adapted to the analysis.

Generic functions: summary()

summary() prints a detailed summary of the analysis, specific to the object class.

```
apropos("^summary")
```

```
##  [1] "summary"                    "Summary"
##  [3] "summary.aov"                "summary.connection"
##  [5] "summary.data.frame"         "Summary.data.frame"
##  [7] "summary.Date"               "Summary.Date"
##  [9] "summary.default"            "Summary.difftime"
## [11] "summary.factor"             "Summary.factor"
## [13] "summary.glm"                "summary.lm"
## [15] "summary.manova"             "summary.matrix"
## [17] "Summary.numeric_version"    "Summary.ordered"
## [19] "summary.POSIXct"            "Summary.POSIXct"
## [21] "summary.POSIXlt"            "Summary.POSIXlt"
## [23] "summary.proc_time"          "summary.srcfile"
## [25] "summary.srcref"             "summary.stepfun"
## [27] "summary.table"              "summary.warnings"
```

Generic functions: summary()

```
summary(linreg)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -332.41  -69.66    2.23   66.15  287.10
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   29.610     16.083   1.841   0.0671 .
## x              4.592      2.448   1.876   0.0621 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
##
## Residual standard error: 99.43 on 198 degrees of freedom
```

Generic functions: summary()

```
summary(anova)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## spray         5  88.44  17.688    44.8 <2e-16 ***
## Residuals    66  26.06   0.395
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Generic functions: `plot()`

`plot()` returns graphics adapted to the current analysis.

```
apropos("^plot")
```

```
##  [1] "plot"                    "plot.default"            "plot.d
##  [4] "plot.ecdf"               "plot.function"           "plot.h
##  [7] "plot.spec.coherency"     "plot.spec.phase"         "plot.s
## [10] "plot.ts"                 "plot.window"             "plot.x
```

Generic functions: plot()

```
par(mfrow = c(2,2))
plot(anova)
```