

Introduction à R

Régression Logistique

Ricco Rakotomalala

http://eric.univ-lyon2.fr/~ricco/cours/cours_programmation_R.html

(1) R est un langage de programmation. L'objet de base est un vecteur de données.

C'est un « vrai » langage c.-à-d. types de données, branchements conditionnels, boucles, organisation du code en procédures et fonctions, découpage en modules.

Mode de d'exécution : transmettre à R le fichier script « .r »

(2) R est un logiciel de statistique et de data mining, pilotée en ligne de commande. Il est extensible (quasiment) à l'infini via le système des packages.

Les instructions servent à manipuler les objets R c.-à-d. les ensembles de données, les vecteurs, les modèles, etc.

Mode de d'exécution : introduire commandes dans le terminal, manipulation interactive

→ C'est le mode que nous exploiterons dans ce tutoriel.

<http://www.r-project.org/>

The screenshot shows the R Project website for Statistical Computing. The page features a navigation menu on the left with links for 'About R', 'Download, Packages', 'R Project Foundation', 'Documentation', and 'Misc'. The main content area displays several statistical plots: a PCA plot with 5 variables (Fertility, Examination, Education, Catholic, Agriculture), a Clustering dendrogram with 4 groups, and two histograms for Factor 1 (41%) and Factor 3 (19%). A 'Getting Started' section contains a list of bullet points, with a red arrow pointing to the 'download R' link. Below this is a 'News' section with updates on R versions and events. At the bottom, it states the server is hosted by the Institute for Statistics and Mathematics at WU Wien.

Getting Started:

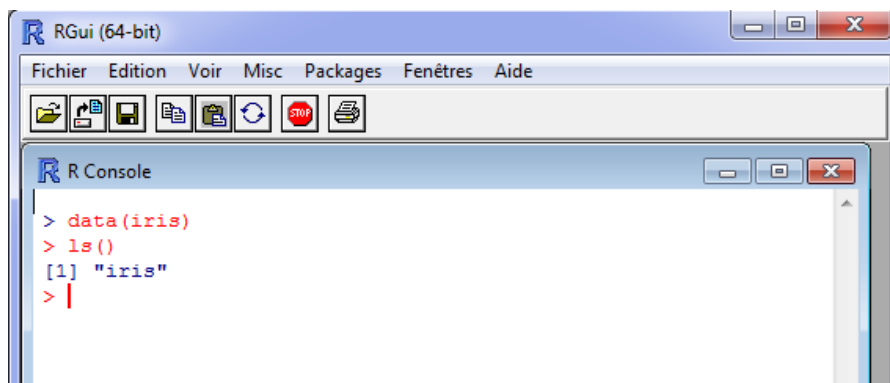
- R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please visit your preferred [CRAN mirror](#).
- If you have questions about R like how to download and install software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

News :

- **R version 2.14.1** (December Snowflakes) has been released on 2011-12-22.
- [The R Journal Vol.3/2](#) is available.
- [useR! 2012](#), will take place at Vanderbilt University, Nashville Tennessee, USA, June 12-15, 2012.
- **R version 2.13.2** has been released on 2011-09-30.

This server is hosted by the [Institute for Statistics and Mathematics](#) of the [WU Wien](#).

R peut fonctionner sous **Windows**, **Mac OS X**, **Linux**



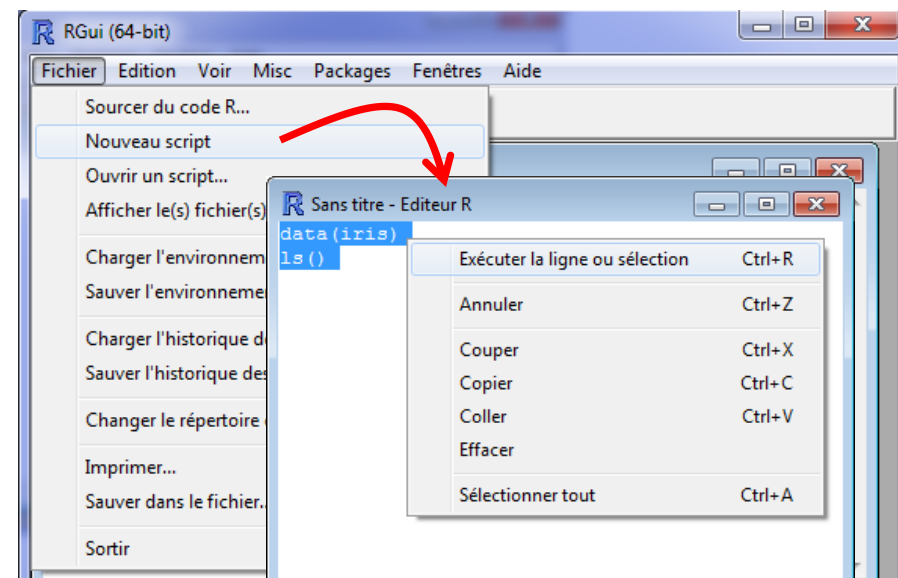
```
> data(iris)
> ls()
[1] "iris"
> |
```

Mode « terminal »

+ interactivité, visualisation immédiate des résultats

+ avec « ↑ », on retrouve les anciennes commandes

- pas de sauvegarde des commandes si fermeture de R (si en fait, avec *FICHIER / SAUVER L'HISTORIQUE DES COMMANDES*)



Mode « script »

+ interactivité, visualisation immédiate des résultats (CTRL + R)

+ maintien d'une liste « propre » des commandes utiles uniquement

+ possibilité d'E/S (chargement ou sauvegarde d'un fichier script « .r »)

→ mode conseillé pour nous

Si on veut programmer (mode programmation), mieux vaut passer par un éditeur externe (ex. TINN-R, R-STUDIO, ECLISPE + StatET,...)



demo_reglog.txt - Bloc-notes

| age | taux | angine | coeur |
|-----|-------|--------|----------|
| 50 | 126.0 | 1 | presence |
| 49 | 126.0 | 0 | presence |
| 46 | 144.0 | 0 | presence |
| 49 | 139.0 | 0 | presence |
| 62 | 154.0 | 1 | presence |
| 35 | 156.0 | 1 | presence |
| 67 | 160.0 | 0 | absence |
| 65 | 140.0 | 0 | absence |
| 47 | 143.0 | 0 | absence |
| 58 | 165.0 | 0 | absence |
| 57 | 115.0 | 1 | absence |
| 59 | 145.0 | 0 | absence |
| 44 | 175.0 | 0 | absence |
| 41 | 153.0 | 0 | absence |
| 54 | 152.0 | 0 | absence |
| 52 | 169.0 | 0 | absence |
| 57 | 168.0 | 1 | absence |
| 50 | 158.0 | 0 | absence |
| 44 | 170.0 | 0 | absence |
| 49 | 171.0 | 0 | absence |

Nom du data.frame

Séparateur de colonnes

1^{ère} ligne = nom des variables

Nom du fichier

Point décimal

```
R Console
> heart <- read.table(file="demo_reglog.txt", sep="\t", dec=".", header=T)
> class(heart)
[1] "data.frame"
> summary(heart)
      age      taux      angine      coeur
Min.   :35.00  Min.   :115.0  Min.   :0.00  absence :14
1st Qu.:46.75  1st Qu.:142.2  1st Qu.:0.00  presence: 6
Median :50.00  Median :153.5  Median :0.00
Mean   :51.75  Mean   :151.4  Mean   :0.25
3rd Qu.:57.25  3rd Qu.:165.8  3rd Qu.:0.25
Max.   :67.00  Max.   :175.0  Max.   :1.00
> |
```

« âge », « taux » et « angine » sont considérées comme quantitatives.

« cœur » est une variable qualitative.

Fichier texte, séparateur tabulation.

data.frame = matrice de données = liste de vecteurs de même longueur.

Vecteur = variable.

Les variables sont typées. Les plus utilisées sont « numeric / integer » (variables quantitatives) et « factor » (variables qualitatives)

Remarque : on peut accéder aux variables d'un data.frame avec l'opérateur \$

```
R Console
> class(heart$age)
[1] "integer"
> class(heart$taux)
[1] "numeric"
> class(heart$angine)
[1] "integer"
> class(heart$coeur)
[1] "factor"
> mean(heart$age)
[1] 51.75
> |
```

Package ?

- Un package est une bibliothèque externe
- Sous Windows → fichiers binaires pré-compilés
- Extension .zip
- Il est toujours documenté : fichier HTML (aide sous R) et PDF

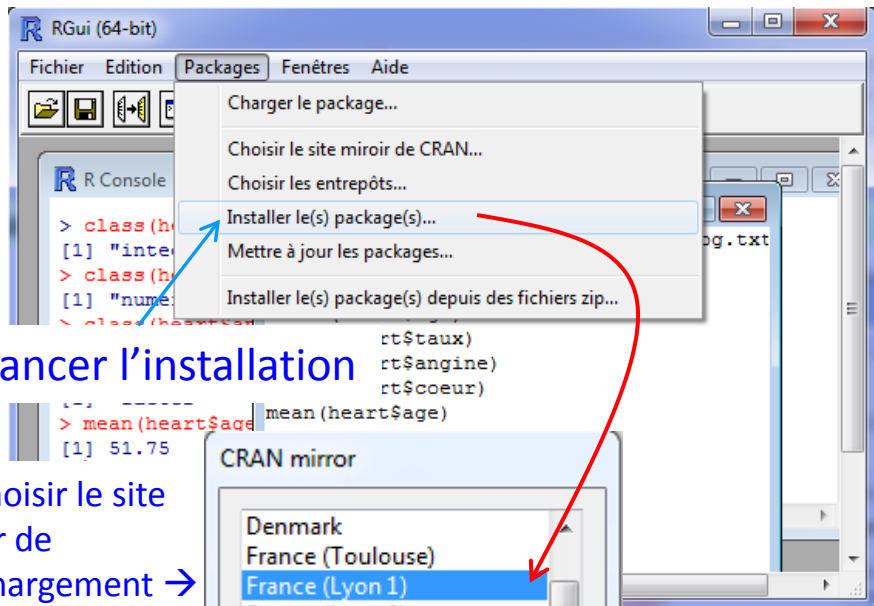
Quel intérêt ?

- Un package contient des collections de fonctions utilisables sous R
- Souvent centrés sur un sujet particulier (ex. *rpart* pour les arbres de décision, etc.)
- Gestion affinée des packages : nous pouvons les installer, désinstaller, charger, décharger et mettre à jour à notre guise

Ce système permet d'augmenter considérablement la puissance de R !!!

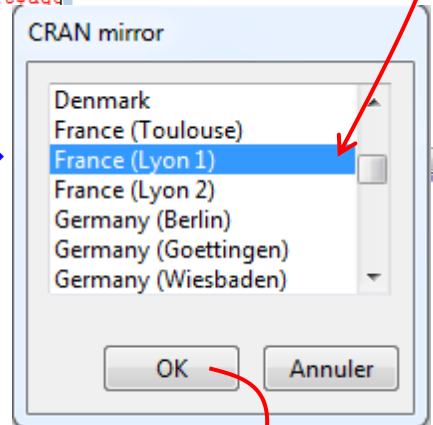
Ex. installer et charger le package « xlsx » permettant de lire directement les fichiers Excel (*.xls et *.xlsx)

Installation (une fois) et chargement d'un package (à chaque utilisation)

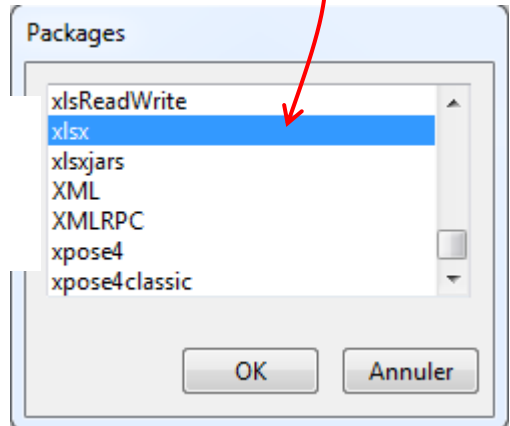


(1) Lancer l'installation

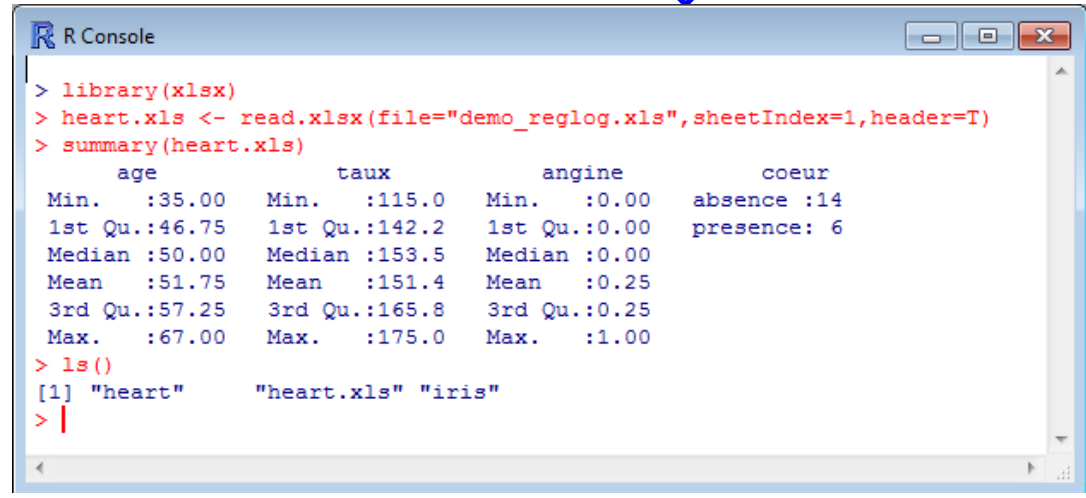
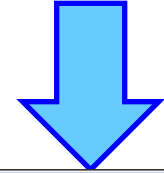
(2) Choisir le site miroir de téléchargement →



(3) Choisir le package à installer →



- > Chargement du package [library]
- > Lecture du fichier Excel (1^{ère} feuille, nom de variable sur 1^{ère} ligne) [read.xlsx]
- > Statistiques descriptives [summary]
- > ls() liste le contenu de la mémoire



```
R Console
> modele <- glm(coeur ~ age+taux+angine,data=heart,family=binomial)
> class(modele)
[1] "glm" "lm"
> print(modele)

Call:  glm(formula = coeur ~ age + taux + angine, family = binomial,
          data = heart)

Coefficients:
(Intercept)          age          taux          angine
 14.49379      -0.12563     -0.06356      1.77901

Degrees of Freedom: 19 Total (i.e. Null);  16 Residual
Null Deviance:      24.43
Residual Deviance: 16.62      AIC: 24.62
> print(summary(modele))

Call:
glm(formula = coeur ~ age + taux + angine, family = binomial,
    data = heart)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9773  -0.5437  -0.3876   0.5093   1.7577

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) 14.49379     7.95464   1.822  0.0684 .
age         -0.12563     0.09380  -1.339  0.1805
taux        -0.06356     0.04045  -1.572  0.1161
angine       1.77901     1.50449   1.182  0.2370
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 24.435  on 19  degrees of freedom
Residual deviance: 16.618  on 16  degrees of freedom
AIC: 24.618

Number of Fisher Scoring iterations: 5

> |
```

`glm()` pour la régression linéaire généralisée
 $\text{Logit}(\text{cœur}) = a_0 + a_1 \cdot \text{âge} + a_2 \cdot \text{taux} + a_3 \cdot \text{angine}$
`data` = heart (data.frame des données)
`family` = binomial (régression logistique)
→ « `modele` » (objet régression généralisée)

Description succincte du modèle
(coefficients, ddl, déviance du modèle trivial,
déviance du modèle, AIC – critère Akaike)

Description détaillée (coefficients, écarts-type
des coefficients, test de significativité, ...)

Manipulation de l'objet « modele »

Ex. Test du rapport de vraisemblance pour évaluer la significativité globale du modèle

Tous les objets R sont typés (`class` permet de le connaître), certains ont des champs (`attributes` fournit la liste).

```
R Console
> ls()
[1] "heart"      "heart.xls"  "iris"       "modele"
> class(modele)
[1] "glm" "lm"
> attributes(modele)
$names
 [1] "coefficients"      "residuals"      "fitted.values"
 [4] "effects"           "R"               "rank"
 [7] "qr"                "family"          "linear.predictors"
[10] "deviance"          "aic"             "null.deviance"
[13] "iter"              "weights"         "prior.weights"
[16] "df.residual"       "df.null"         "y"
[19] "converged"         "boundary"        "model"
[22] "call"              "formula"         "terms"
[25] "data"              "offset"          "control"
[28] "method"           "contrasts"       "xlevels"

$class
[1] "glm" "lm"

> chi2 <- modele$null.deviance - modele$deviance
> print(chi2)
[1] 7.816884
> ddl <- modele$df.null - modele$df.residual
> print(ddl)
[1] 3
> pvalue <- pchisq(chi2,ddl,lower.tail=F)
> print(pvalue)
[1] 0.04995172
> ls()
[1] "chi2"      "ddl"      "heart"    "heart.xls" "iris"
[6] "modele"    "pvalue"
> |
```

Champs associés à l'objet « modele ». Certains sont des scalaires (*vecteur de taille 1*), d'autres des vecteurs, d'autres encore des matrices, des listes, des data.frame,...

Rapport de vraisemblance = Déviance du modèle trivial (réduit à la constante) – déviance du modèle étudié
DDL = écart entre les deux DDL
P-VALUE = loi du KHI-2 à DDL degrés de liberté
Le modèle est globalement significatif à 5%

Nouveaux objets présents en mémoire après les calculs (chi2, ddl et pvalue, *tous des scalaires ici...*)

Prediction avec la regression logistique – Matrice de confusion, taux d'erreur

Modèle utilisé pour la prédiction

Données à prédire (ça peut être un autre data.frame, ex. échantillon test)

Prediction = Proba d'être positif (présence de cœur)

Prediction avec `predict(...)`

```
R Console
> pred.proba <- predict(modele, newdata=heart, type="response")
> print(pred.proba)
      1      2      3      4      5      6      7      8      9     10
0.87894733 0.58154537 0.39220275 0.37820752 0.21335852 0.87655486 0.01640958 0.07103688 0.37750865 0.03624840
     11     12     13     14     15     16     17     18     19     20
0.85841939 0.10575388 0.10366373 0.40566043 0.12437705 0.05836647 0.17271990 0.13818549 0.13712678 0.07370700
> pred.moda <- factor(ifelse(pred.proba > 0.5, "presence", "absence"))
> print(pred.moda)
      1      2      3      4      5      6      7
presence presence absence absence absence presence absence
     13     14     15     16     17     18     19
absence absence absence absence absence absence absence
Levels: absence presence
> ls()
[1] "chi2"      "ddl"      "heart"    "heart.xls" "iris"    "modele"  "pred.moda" "pred.proba"
[9] "pvalue"
> |
```

Transformer les probas en affectation (`ifelse`), variable qualitative (`factor`)

`pred.proba` et `pred.moda` sont deux nouveaux vecteurs accessibles dans la mémoire de R

Matrice de confusion

```
R Console
> mc <- table(heart$coeur, pred.moda)
> class(mc)
[1] "table"
> print(mc)
      pred.moda
      absence presence
absence      13      1
presence      3      3
> err <- (mc[2,1]+mc[1,2])/sum(mc)
> print(err)
[1] 0.2
> ls()
[1] "chi2"      "ddl"      "err"      "heart"
[5] "heart.xls" "iris"     "mc"       "modele"
[9] "pred.moda" "pred.proba" "pvalue"
> |
```

`table(...)` construit un tableau croisé entre la cible observée (`coeur`) et la prédiction du modèle (`pred.moda`)

La table `mc` se comporte comme une matrice à 2 dimensions, on en déduit le taux d'erreur

Nouvelle liste des objets disponibles en mémoire

Sélection de variables – Backward – Optimisation du critère AIC avec **stepAIC**

Librairie pour la sélection pas-à-pas.

Modèle de départ (avec toutes les variables)

Plage de recherche

Direction de recherche

```
R Console
> library(MASS)
> modele.back <- stepAIC(modele,scope=list(lower="coeur ~ 1", upper="coeur ~ age+taux+angine"),direction="backward")
Start:  AIC=24.62
coeur ~ age + taux + angine

      Df Deviance  AIC
- angine  1  18.151 24.151
<none>    16.618 24.618
- age     1  19.094 25.094
- taux    1  19.702 25.702

Step:  AIC=24.15
coeur ~ age + taux

      Df Deviance  AIC
<none>    18.151 24.151
- age     1  20.682 24.682
- taux    1  22.945 26.945
> print(modele.back)

Call:  glm(formula = coeur ~ age + taux, family = binomial, data = heart)

Coefficients:
(Intercept)      age      taux
 16.25444    -0.12011   -0.07438

Degrees of Freedom: 19 Total (i.e. Null);  17 Residual
Null Deviance:      24.43
Residual Deviance: 18.15      AIC: 24.15
> |
```

AIC modèle complet = 24.62
1^{ère} meilleure variable à retirer : angine, AIC = 24.151 (OK)
2^{ème} variable à retirer : âge, mais AIC = 24.682 (PAS OK)
Arrêt de la procédure

Print du modèle simplifié

Modèle de départ
(cœur = a0)

Direction de
recherche

```

R Console
> modele.trivial <- glm(coeur ~ 1,data=heart,family=binomial)
> modele.forward <- stepAIC(modele.trivial,scope=list(lower="coeur ~ 1", upper="coeur ~ age+taux+engine"),direction="forward")
Start:  AIC=26.43
coeur ~ 1

      Df Deviance   AIC
+ taux  1  20.682 24.682
+ engine 1  21.742 25.742
<none>   24.435 26.435
+ age    1  22.945 26.945

Step:  AIC=24.68
coeur ~ taux

      Df Deviance   AIC
+ age  1  18.151 24.151
<none>  20.682 24.682
+ engine 1  19.094 25.094

Step:  AIC=24.15
coeur ~ taux + age

      Df Deviance   AIC
<none>  18.151 24.151
+ engine 1  16.618 24.618
> print(modele.forward)

Call:  glm(formula = coeur ~ taux + age, family = binomial, data = heart)

Coefficients:
(Intercept)      taux      age
  16.25444    -0.07438   -0.12011

Degrees of Freedom: 19 Total (i.e. Null);  17 Residual
Null Deviance:      24.43
Residual Deviance: 18.15      AIC: 24.15
> |
    
```

AIC modèle trivial= 26.43

1^{ère} meilleure variable à ajouter : taux, AIC = 24.682 (OK)

2^{ème} variable à ajouter : âge, AIC = 24.151 (OK)

3^{ème} variable à ajouter : engine, mais AIC = 24.618 (PAS OK)

Arrêt de la procédure

Print du modèle après sélection

De la documentation à profusion (n'achetez jamais des livres sur R)

Site du cours

http://eric.univ-lyon2.fr/~ricco/cours/cours_programmation_R.html

Programmation R

<http://www.duclert.org/>

Quick-R

<http://www.statmethods.net/>

POLLS (Kdnuggets)

Data Mining / Analytics Tools Used - <http://www.kdnuggets.com/polls/2011/tools-analytics-data-mining.html>

(Mai 2011, R en 2nde position)

What languages you used for data mining / data analysis?

<http://www.kdnuggets.com/polls/2011/languages-for-data-mining-analytics.html>

(Août 2011, langage R en 1^{ère} position)

Article New York Times (Janvier 2009)

“Data Analysts Captivated by R’s Power” - http://www.nytimes.com/2009/01/07/technology/business-computing/07program.html?_r=1