Exploration de données ouvertes avec R

François Pelletier

Exploration de données ouvertes avec R

- François Pelletier
- francois@francoispelletier.org
- LinuQ

- 31 octobre 2015

Contenu de la présentation

- Présentation du logiciel statistique R
- Opérateurs, objets et types
- Exemple
- Ressources

Présentation du logiciel statistique R

- Logiciel libre
- Langage de programmation S
- Multi-paradigme: objet, impératif et fonctionnel

Environnement

- Ensemble de fonctions de base
- Extension avec des paquets
- Extenstion écrites souvent en C, C++ et FORTRAN

```
library("MASS")
```

Développer en R

- RStudio
- Emacs + ESS
- Eclipse + StatET

Opérateurs de base

```
# Assignation
(x <- c(1,2,3))
```

```
[1] 1 2 3
```

```
# Extraction
(y <- data.frame(a=x[1],b=x[2]))
```

```
  a b
1 1 2
```

```
# Objets
(z <- y$a)
```

```
[1] 1
```

Les objets en R

Les objets dans R ont une classe, un type et une dimension

```
monVecteur <- c(1,2,3,4)
class(monVecteur)
```

```
[1] "numeric"
```

```
typeof(monVecteur)
```

```
[1] "double"
```

```r
dim(monVecteur)
```

```
NULL
```

```r
maMatrice <- matrix(nrow = 2,ncol = 2, data = monVecteur)
class(maMatrice)
```

```
[1] "matrix" "array"
```

```r
typeof(maMatrice)
```

```
[1] "double"
```

```r
dim(maMatrice)
```

```
[1] 2 2
```

Les types en R

Nous avons vu le vecteur et la matrice.

Il y a aussi les facteurs, qui permettent d'utiliser des modalités qualitatives:

```r
factor(c("oui","non","non","oui","nsp","oui","non"))
```

```
[1] oui non non oui nsp oui non
Levels: non nsp oui
```

les tableaux, une extension multidimensionnelle des matrices:

```r
array(1:8,dim=c(2,2,2))
```

```
, , 1

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

```
, , 2

     [,1] [,2]
[1,]    5    7
[2,]    6    8
```

les listes, qui sont des collections d'objets de types différents:

```
list("1",TRUE,c(1,2,3),function(x) x^2)
```

```
[[1]]
[1] "1"

[[2]]
[1] TRUE

[[3]]
[1] 1 2 3

[[4]]
function (x)
x^2
```

et les cadres de données, semblables à des tables SQL:

```
data.frame(numero_membre=c(1,2,3,4),
           nom_membre=c("François","Jean","Marie","Stéphanie"),
           age_membre=c(26,53,41,32),
           stringsAsFactors = TRUE)
```

```
  numero_membre nom_membre age_membre
1             1   François         26
2             2       Jean         53
3             3      Marie         41
4             4  Stéphanie         32
```

Structures de contrôle

```r
# option
if(TRUE) "vrai" else "faux"
```

```
[1] "vrai"
```

```r
# sélection
1:5<3
```

```
[1]  TRUE  TRUE FALSE FALSE FALSE
```

```r
(1:10)[-3:-5]
```

```
[1]  1  2  6  7  8  9 10
```

Boucles

Explicites (à éviter)

```r
for (i in 1:2)
  print(i)
```

```
[1] 1
[1] 2
```

Implicites

```r
y <- sapply(3:4,print)
```

```
[1] 3
[1] 4
```

Les fonctions

Déclaration:

```r
maFonction <- function(x, ...)
{
  if (x)
    sum(...)
  else
    0
}
maFonction(FALSE,1,2,3)
```

```
[1] 0
```

```r
maFonction(TRUE,1,2,3)
```

```
[1] 6
```

Quelques statistiques

```r
set.seed(123)
mesDonnees <- rnorm(10,mean = 5,sd = 3)
range(mesDonnees)
```

```
[1]  1.204816 10.145195
```

```r
summary(mesDonnees)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.205   3.405   4.760   5.224   6.134  10.145
```

```r
quantile(mesDonnees,c(seq(.25,.75,.25)))
```

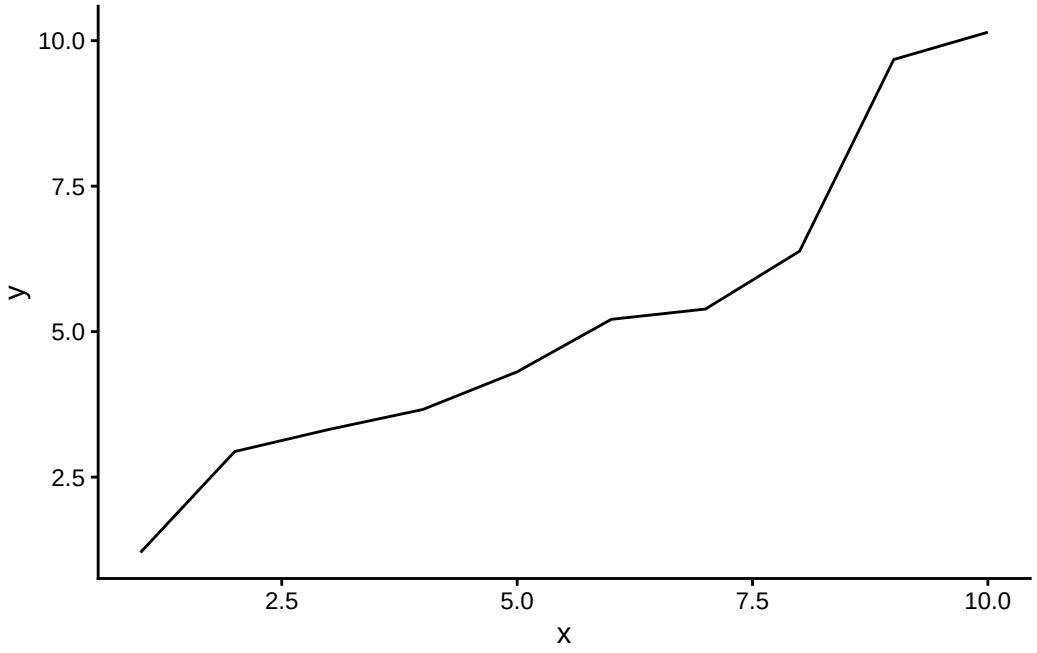
```
     25%      50%      75%
3.404683 4.760496 6.134027
```

Un premier graphique

```r
library("ggplot2")
monData <-
  data.frame(x=seq_along(mesDonnees),
                        y=mesDonnees[order(mesDonnees)])

monGraph <-
  ggplot(data=monData,aes(x=x,y=y)) +
  geom_line() +
  theme_classic()
```

```r
monGraph
```



Un premier modèle

```r
(monModele <- lm(y~x,data=monData))
```

```
Call:
lm(formula = y ~ x, data = monData)
```

```
Coefficients:
(Intercept)            x
     0.2566       0.9031
```

Sommaire du modèle

```
summary(monModele)
```

```
Call:
lm(formula = y ~ x, data = monData)

Residuals:
     Min       1Q   Median       3Q      Max
-1.19072 -0.46365 -0.08055  0.73103  1.29127

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.25662    0.61074   0.420    0.685
x            0.90314    0.09843   9.175 1.61e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.894 on 8 degrees of freedom
Multiple R-squared:  0.9132,    Adjusted R-squared:  0.9024
F-statistic: 84.19 on 1 and 8 DF,  p-value: 1.608e-05
```

Analyse de variance du modèle

```
anova(monModele)
```

```
Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq F value    Pr(>F)
x          1 67.292  67.292   84.19 1.608e-05 ***
Residuals  8  6.394   0.799
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ressources

- R Project
- Revolution R
- CRAN R Other Docs (incluant livres en français)
- Wikibooks R Programming
- R Bloggers
- Awesome R