Intitation à RStudio

Contents

Introduction **1**
Les Chunks 1

Probability and statistics with R **2**
Vectors 2

Utiliser des données qui se trouvent dans un Package **6**

Utiliser les pipes **7**

Des graphiques plus sophistiqués **7**

Recode variables with dplyr **10**

Statistiques descriptives **11**

Stem and leaf plot **13**

Statistiques de base **14**

Charger des données sur un lien **15**

Exercice Yogurt **15**

Introduction

- Nous allons étudier la programmation avec **R**.
- Nous sommes dans un *Notebook*.
- Nous pouvons écrire des fomules de mathématiques en *LATEX*.
- Par exemple, je peux écrire une équation comme :

$$x^2 - 12x + 14 = 0.$$

- Des formules un peu plus sophistiquées

$$\int_{-\infty}^{+\infty} e^{-x^2} dx = \sqrt{\pi}.$$

Mettre un lien hypertexte Le wikipedia du Cnam.

Les Chunks

- Pour exécuter des lignes de code en **R**, vous pouvez insérer ces lignes à partir du menu.

```r
1+1
```

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

Probability and statistics with R

Vectors

```r
x <- 5
x
```

```
## [1] 5
```

```r
y <- c(7, 3, 5)
y
```

```
## [1] 7 3 5
```

```r
z <- c(2, 4, 6, 8)
length(z)
```

```
## [1] 4
```

```r
length(x)
```

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

```r
length(y)
```

```
## [1] 3
```

```r
x + y
```

```
## [1] 12  8 10
```

```r
y + z # Opération non souhaitée car pas de même longueur
```

```
## Warning in y + z: longer object length is not a multiple of shorter object
## length
```

```
## [1]  9  7 11 15
```

```r
# Supposons que z soient des prix en euros. Pour les convertir en dollars, il suffit de faire
0.87*z
```

```
## [1] 1.74 3.48 5.22 6.96
```

LogVec <- (x < z) # logical vector LogVec # 5 < 2, 5 < 4, 5 < 6, 5 < 8 [1] FALSE FALSE TRUE TRUE typeof(LogVec)

```r
LogVec <- (x < z)
LogVec
```

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

```r
typeof(LogVec)
```

```
## [1] "logical"
```

```r
typeof(x)
```

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

```r
z
```

```
## [1] 2 4 6 8
```

```r
z[2]
```

```
## [1] 4
```

```r
typeof(z)
```

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

```r
z<-as.integer(z)
typeof(z)
```

```
## [1] "integer"
```

```r
LETTERS ## vecteur contenant les lettres de l'alphabet
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S"
## [20] "T" "U" "V" "W" "X" "Y" "Z"
```

```r
LETTERS[10] ## 10ème lettre
```

```
## [1] "J"
```

```r
LETTERS[c(1, 2, 3, 4)] ## Les quatre premières lettres
```

```
## [1] "A" "B" "C" "D"
```

```r
LETTERS[1:20] ## Les vingt premières lettres
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S"
## [20] "T"
```

```r
1:20 ## crée une liste (ou un vecteur) des nombres de 1 à 20
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

```r
1980:2021     ## Créez un vecteur d'années de 1980 à 2021
```

```
##  [1] 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994
## [16] 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009
## [31] 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021
```

```r
seq(1980,2021,4)              ##  Créez un vecteur des années bisextiles depuis  1980
```

```
##  [1] 1980 1984 1988 1992 1996 2000 2004 2008 2012 2016 2020
```

```r
z
```

```
## [1] 2 4 6 8
```

```r
z[1] ## Garder le premier élément
```

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

```r
z[-1] ## Enlever le premier élément
```

```
## [1] 4 6 8
```

```r
z[c(1,2)]      ## Garder les deux premiers éléments
```

```
## [1] 2 4
```

```r
z[-c(1,2)] ## Enlever les deux premiers éléments
```

```
## [1] 6 8
```

```r
z[-(1:2)] ## Idem
```

```
## [1] 6 8
```

```r
z[z>5] ## Garder les éléments supérieurs à 5
```

```
## [1] 6 8
```

```r
data<-c(rep(c("Nord","Oui"),0.3*30000),
rep(c("Nord","Non"),0.1*30000),
rep(c("Sud","Oui"),0.2*30000),
rep(c("Sud","Non"),0.4*30000))

data<-matrix(data,nrow=30000,ncol=2,byrow=TRUE)
data<-as.data.frame(data)
typeof(data)
```

```
## [1] "list"
```

```r
class(data)
```

```
## [1] "data.frame"
```

```r
is.data.frame(data)
```

```
## [1] TRUE
```

```r
class(x)
```

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

```r
typeof(x)
```

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

```r
names(data)<-c("Région","Réponse")
head(data)
```

```
##   Région Réponse
## 1   Nord     Oui
## 2   Nord     Oui
## 3   Nord     Oui
## 4   Nord     Oui
## 5   Nord     Oui
## 6   Nord     Oui
```

```r
te<-table(data) ## Tableau d'effectifs
te
```

```
##       Réponse
## Région   Non   Oui
##   Nord  3000  9000
##   Sud  12000  6000
```

```r
  ## Tableau de proportions
prop.table(te) ## Calcule le tableau en proportions à partir du tableau en effectifs
```

```
##       Réponse
```

```
## Région Non Oui
##   Nord 0.1 0.3
##   Sud  0.4 0.2
```

```
addmargins(prop.table(te)) ## Rajoute les marges, sommes en ligne et en colonne
```

```
##       Réponse
## Région Non Oui Sum
##   Nord 0.1 0.3 0.4
##   Sud  0.4 0.2 0.6
##   Sum  0.5 0.5 1.0
```

```
addmargins(prop.table(te,1),2) ## Calcule les prop. en ligne (en conditionnant par les lignes) et rajou
```

```
##       Réponse
## Région       Non       Oui       Sum
##   Nord 0.2500000 0.7500000 1.0000000
##   Sud  0.6666667 0.3333333 1.0000000
```

```
addmargins(prop.table(te,2),1) ## Calcule les prop. en colonne (conditionnelles) et rajoute les marges
```

```
##       Réponse
## Région Non Oui
##   Nord 0.2 0.6
##   Sud  0.8 0.4
##   Sum  1.0 1.0
```

```
Grades <- c("A", "D", "C", "D", "C", "C", "C", "C", "F", "B") # Crée la variable
Grades # Affiche la variable
```

```
##  [1] "A" "D" "C" "D" "C" "C" "C" "C" "F" "B"
```

```
table(Grades) # Crée une table de fréquence
```

```
## Grades
## A B C D F
## 1 1 5 2 1
```

```
xtabs(~Grades) # Idem avec la fonction xtabs
```

```
## Grades
## A B C D F
## 1 1 5 2 1
```

```
table(Grades)/length(Grades) # Crée la table en prop.
```

```
## Grades
##   A   B   C   D   F
## 0.1 0.1 0.5 0.2 0.1
```

```
prop.table(table(Grades)) # Idem
```

```
## Grades
##   A   B   C   D   F
## 0.1 0.1 0.5 0.2 0.1
```

```
prop.table(xtabs(~Grades)) # Idem
```

```
## Grades
##   A   B   C   D   F
```

```
## 0.1 0.1 0.5 0.2 0.1
```

Utiliser des données qui se trouvent dans un Package

```
library(MASS) # Active le Package MASS
# data(package="MASS") # Montre tous les fichiers de données du Package MASS
head(quine) # Montre les 6 premières ligne de du fichier quine
```

```
##   Eth Sex Age Lrn Days
## 1   A   M  F0  SL    2
## 2   A   M  F0  SL   11
## 3   A   M  F0  SL   14
## 4   A   M  F0  AL    5
## 5   A   M  F0  AL    5
## 6   A   M  F0  AL   13
```

```
table(quine$Age)
```

```
##
## F0 F1 F2 F3
## 27 46 40 33
```

```
xtabs(~Age,data=quine) # Effectifs par Age
```

```
## Age
## F0 F1 F2 F3
## 27 46 40 33
```

```
names(quine) # Donne le nom des variables
```

```
## [1] "Eth"  "Sex"  "Age"  "Lrn"  "Days"
```

```
xtabs(~Age+Sex,data=quine) # On croise la variable Age et Sex
```

```
##     Sex
## Age   F  M
##   F0 10 17
##   F1 32 14
##   F2 19 21
##   F3 19 14
```

```
addmargins(xtabs(~Age+Sex,data=quine))
```

```
##      Sex
## Age    F   M Sum
##   F0   10  17  27
##   F1   32  14  46
##   F2   19  21  40
##   F3   19  14  33
##   Sum  80  66 146
```

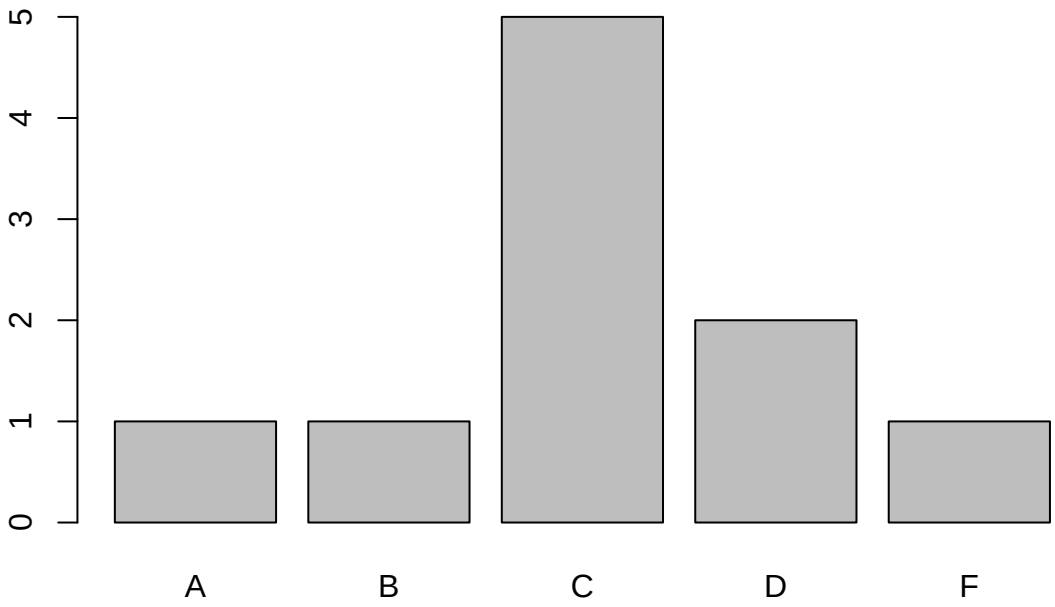
```
#table(quine$Age)  accessing Age using dollar prefixing
```

Utiliser les pipes

```
library(magrittr)
library(MASS) # Charger une librairie pour les pipes
xtabs(~Age+Sex,data=quine) %>%
addmargins()
```

```
##      Sex
## Age    F   M Sum
##   F0   10  17  27
##   F1   32  14  46
##   F2   19  21  40
##   F3   19  14  33
##   Sum  80  66 146
```

```
xtabs(~Grades,data=quine)%>%
barplot()
```

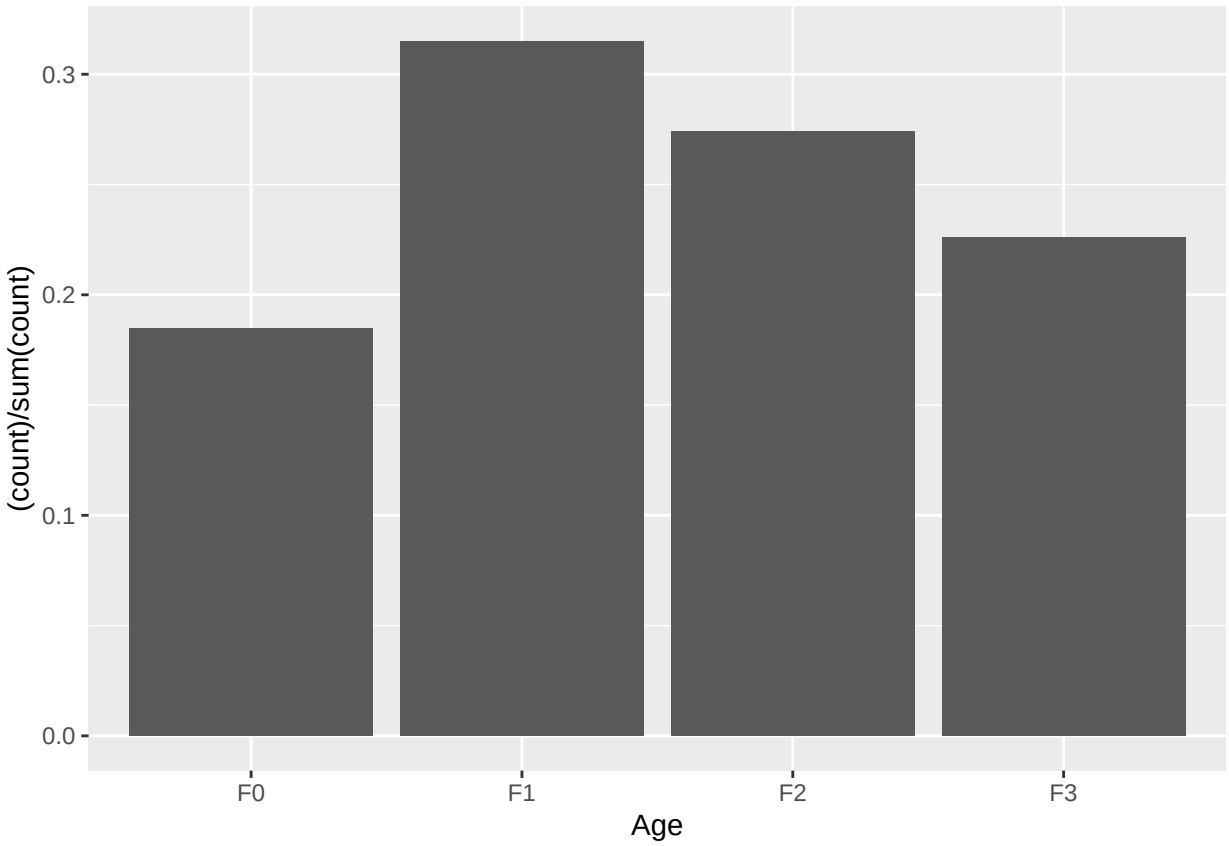


Des graphiques plus sophistiqués

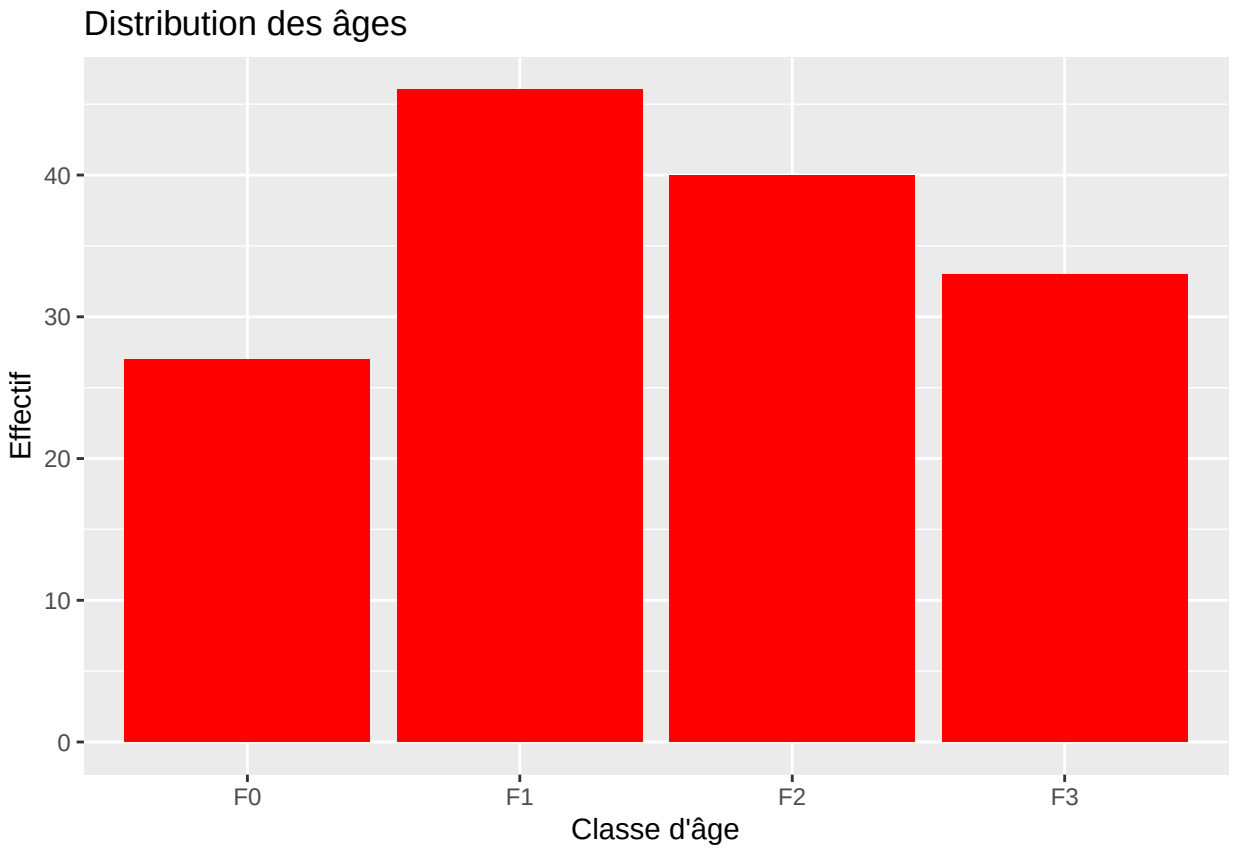
```
library(ggplot2) # On charge cette librairie de dataviz
library(ggthemes) # Librairie de thèmes supplémentaires
is.data.frame(quine)
```

```
## [1] TRUE
```

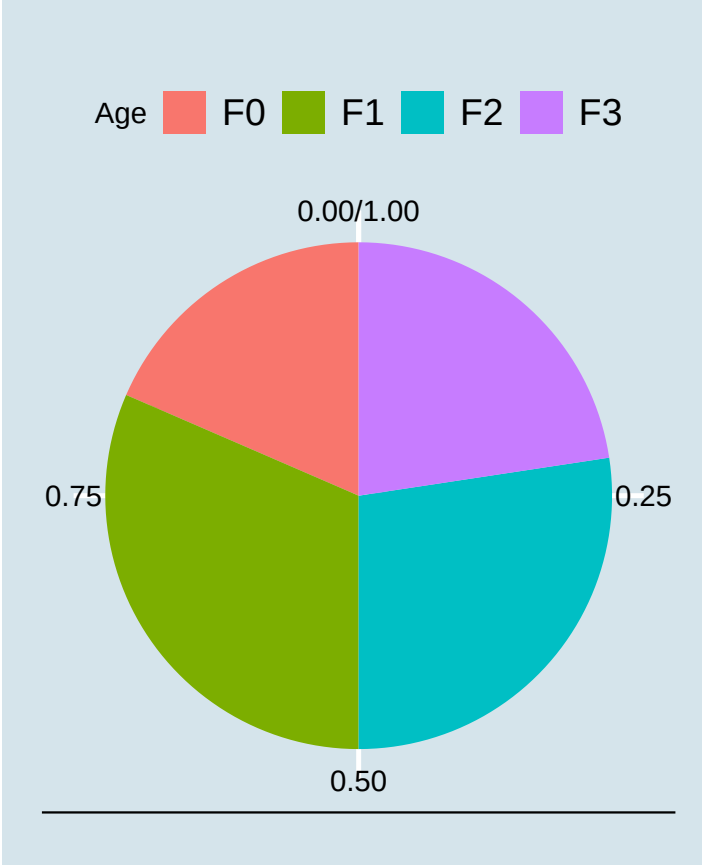
```
ggplot(data=quine, aes(x=Age))+
geom_bar(aes(y = (..count..)/sum(..count..)))
```




```
bp<-ggplot(data=quine, aes(x=Age))+
geom_bar(aes(y = (..count..)),fill = "red")+ggtitle("Distribution des âges") +ylab("Effectif")+xlab("Cl
bp
```




```
ggplot(quine,aes(x= "", fill=Age)) +geom_bar(aes(y = (..count..)/sum(..count..)))+ggtitle(" ") +theme_e
  coord_polar("y", start=0)+labs(x=NULL, y=NULL)
```



Recode variables with dplyr

```
library(dplyr)
```

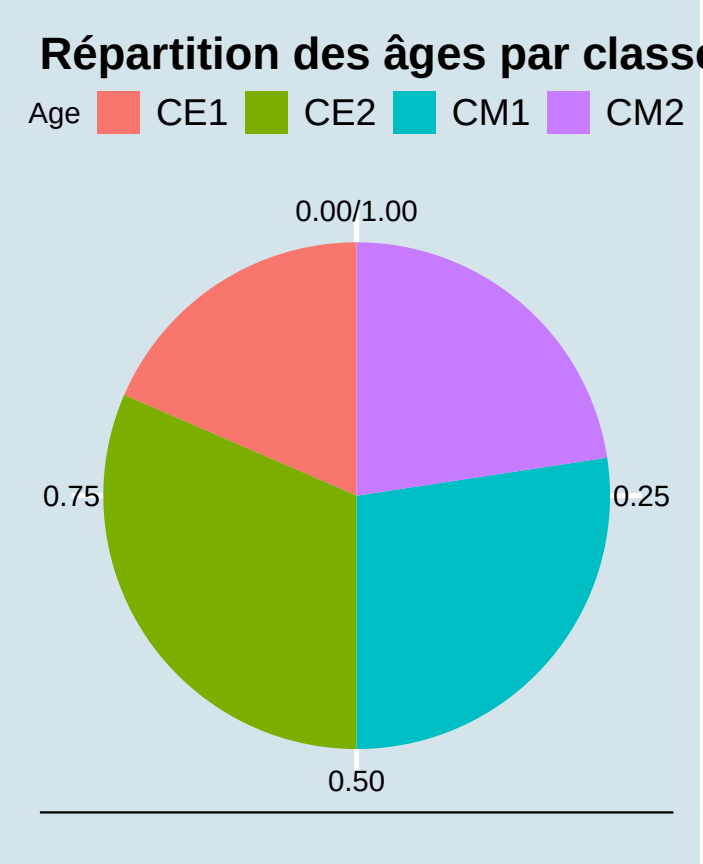
```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
quine%>%mutate(Age=recode(Age,"F0"="CE1","F1"="CE2","F2"="CM1","F3"="CM2"))->df
ggplot(df,aes(x= "", fill=Age)) +geom_bar(aes(y = (..count..)/sum(..count..)))+ggtitle("Répartition des
  coord_polar("y", start=0)+labs(x=NULL, y=NULL)
```



Statistiques descriptives

```
library(knitr)
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(qwraps2)
stargazer(df,type="text")
```

========== Statistic N Mean St. Dev. Min Pctl(25) Pctl(75) Max ———————————————————- Days 146 16.459 16.253 0 5 22.8 81 ———————————————————-

```
summary(df)%>%
kable()
```

Eth	Sex	Age	Lrn	Days
A:69	F:80	CE1:27	AL:83	Min. : 0.00
N:77	M:66	CE2:46	SL:63	1st Qu.: 5.00
NA	NA	CM1:40	NA	Median :11.00
NA	NA	CM2:33	NA	Mean :16.46

Eth	Sex	Age	Lrn	Days
NA	NA	NA	NA	3rd Qu.:22.75
NA	NA	NA	NA	Max. :81.00

```
summary_table(df)
```

	df (N = 146)
Eth	
A	69 (47)
N	77 (53)
Sex	
F	80 (55)
M	66 (45)
Age	
CE1	27 (18)
CE2	46 (32)
CM1	40 (27)
CM2	33 (23)
Lrn	
AL	83 (57)
SL	63 (43)
Days	
minimum	0
median (IQR)	11.00 (5.00, 22.75)
mean (sd)	16.46 ± 16.25
maximum	81

```
library(dummies)
```

```
## dummies-1.5.6 provided by Decision Patterns
```

```
dummy(quine$Eth)%>%
data.frame()->dfEth
```

```
## Warning in model.matrix.default(~x - 1, model.frame(~x - 1), contrasts = FALSE):
## non-list contrasts argument ignored
```

```
dummy(quine$Sex)%>%
data.frame()->dfSex
```

```
## Warning in model.matrix.default(~x - 1, model.frame(~x - 1), contrasts = FALSE):
## non-list contrasts argument ignored
```

```
dummy(quine$Age)%>%
data.frame()->dfAge
```

```
## Warning in model.matrix.default(~x - 1, model.frame(~x - 1), contrasts = FALSE):
## non-list contrasts argument ignored
```

```
dummy(quine$Lrn)%>%
data.frame()->dfLrn
```

```
## Warning in model.matrix.default(~x - 1, model.frame(~x - 1), contrasts = FALSE):
## non-list contrasts argument ignored
```

```
cbind(dfEth,dfSex,dfAge,dfLrn,quine$Days) ->df
names(df)<-c("Eth A","Eth N","Female","Male","F0","F1","F2","F3","AL","SL","Days")
```

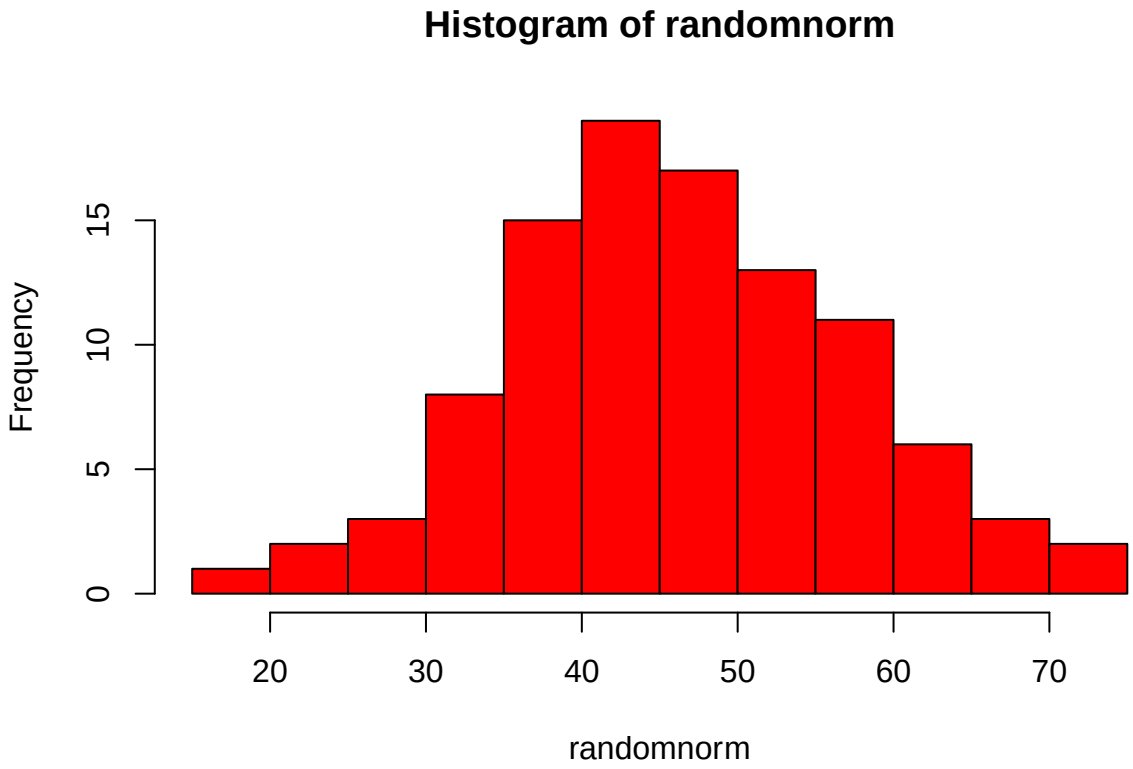
```
knitr::opts_chunk$set(echo = FALSE)
stargazer(df,type="latex")
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Thu, Apr 08, 2021 - 09:27:04

Table 2:

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
Eth A	146	0.473	0.501	0	0	1	1
Eth N	146	0.527	0.501	0	0	1	1
Female	146	0.548	0.499	0	0	1	1
Male	146	0.452	0.499	0	0	1	1
F0	146	0.185	0.390	0	0	0	1
F1	146	0.315	0.466	0	0	1	1
F2	146	0.274	0.448	0	0	1	1
F3	146	0.226	0.420	0	0	0	1
AL	146	0.568	0.497	0	0	1	1
SL	146	0.432	0.497	0	0	1	1
Days	146	16.459	16.253	0	5	22.8	81

Stem and leaf plot

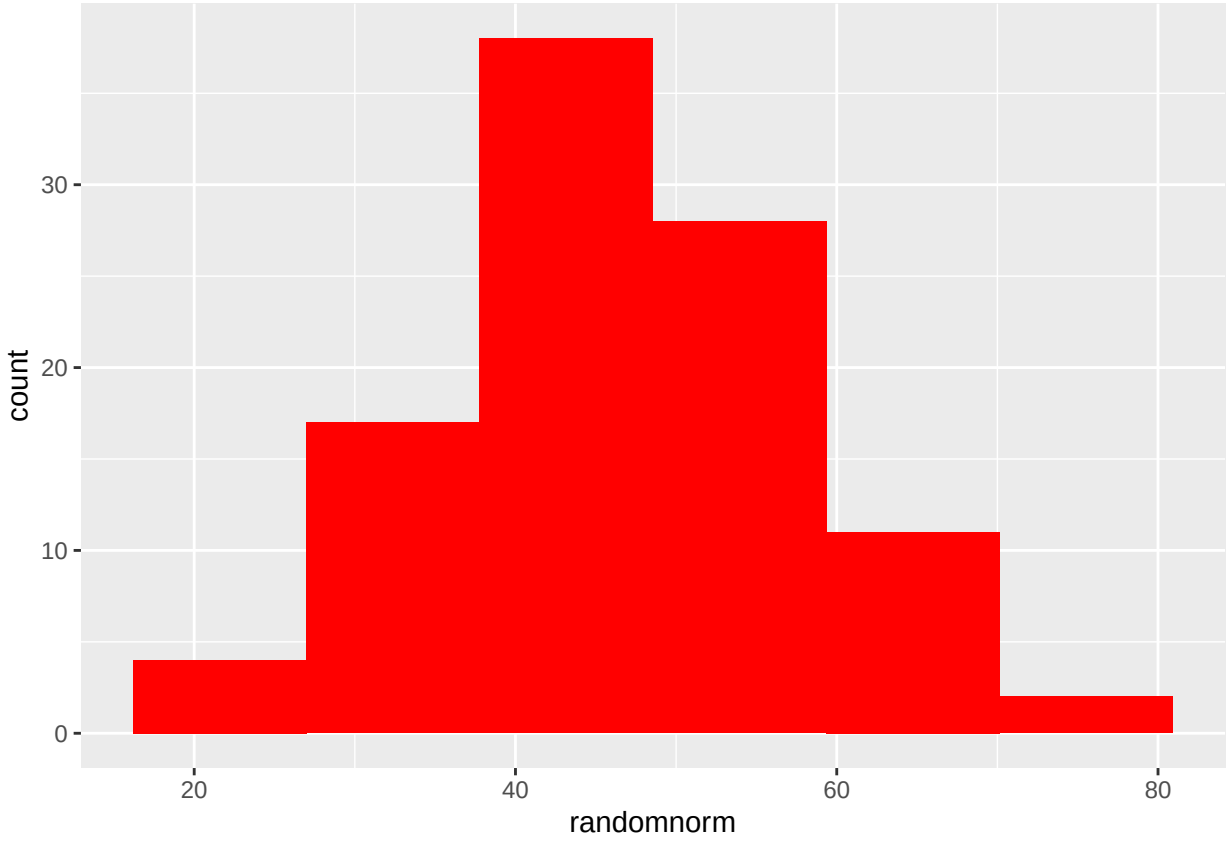



```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##   1 | 7
```

```
##   2 | 156
##   3 | 00012223336677777888999
##   4 | 000111112222223344455666677788899
##   5 | 00000011222333455666777899
##   6 | 001133466
##   7 | 011
```

```
## [1] 17.28997
```

```
## [1] 71.248
```



Statistiques de base

```
## [1] 46.08487
```

```
## [1] 10.95379
```

```
## [1] 119.9855
```

```
## [1] 45.74108
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.29   39.07   45.74   46.08   53.30   71.25
```

```
##    randomnorm
##  Min.   :17.29
##  1st Qu.:39.07
##  Median :45.74
##  Mean   :46.08
##  3rd Qu.:53.30
```

```
##  Max.   :71.25
```

Charger des données sur un lien

```
##    age  fat sex
## 1   23  9.5   M
## 2   23 27.9   F
## 3   27  7.8   M
## 4   27 17.8   M
## 5   39 31.4   F
## 6   41 25.9   F
## 7   45 27.4   M
## 8   49 25.2   F
## 9   50 31.1   F
## 10  53 34.7   F
## 11  53 42.0   F
## 12  54 29.1   F
## 13  56 32.5   F
## 14  57 30.3   F
## 15  58 33.0   F
## 16  58 33.8   F
## 17  60 41.1   F
## 18  61 34.5   F
```

Exercice Yogurt

- Les données sont disponibles dans le package Ecdat
- Les données sont traitées dans l'article MISRA, Sanjog. Generalized reverse discrete choice models. Quantitative Marketing and Economics, 2005, vol. 3, no 2, p. 175-200.