

```
# Initiation à R
# Lecture de quelques fichiers

# le repertoire de travail # Attention au sens des /
setwd("C:/R_ITA/")
#setwd("D:/Mes Documents/R_work/formation_R/ACTA/")

# lister les fichiers dans data
dir("data/")

# lire des données et les enregistrer dans un data.frame
statenz <- read.table("data/stat_enzyme.csv", sep=";", dec=".", skip=3, header=TRUE,
stringsAsFactors=TRUE)
#statenz <- read.delim("data/stat_enzyme.csv", sep=";")
#statenz <- read.csv2("data/stat_enzyme.csv", dec=".")

# afficher les premières lignes du tableau
head(statenz)

# la structure de data
str(statenz)

#####
```

```

# Initiation à R
# TP4. Lecture de quelques fichiers exemples dans des formats divers.
# fichier XLS.
# conseil : utiliser le fichier XLS uniquement pour la saisie (données brutes) et éviter
les calculs, les analyses dedans.
# afin d'assurer la traçabilité données brutes---->résultats analyses
#####
# Option1 (on vous encourage à cela, au moins pour débiter). Convertir les onglets du
fichier XLS en csv (avec enregistrer sous...)
# => voir le TP4_lecturefichier.r

# Option2 directement à partir d'un fichier xls
# => besoin d'installer une library supplémentaire
#install.packages("xlsx")
library(xlsx)

setwd("C:/R_ITA/")
#setwd("D:/data/R/R_ITA/")

# lecture de l'onglet 1 - stat_enzyme
#data1= read.xlsx("data/stat_enzyme.xls",
#colNames = TRUE, sheet = 1, type = "data.frame", from=4, rowNames = NA, colClasses = NA,
#checkNames = TRUE,
#dateTime = "numeric", naStrings = "-", stringsAsFactors = FALSE)
# read data and save them in a data.frame
statenz_1= read.xlsx("data/stat_enzyme.xlsx", sheetName="stat_enzyme", startRow=4,
header=TRUE, stringsAsFactors = FALSE )
#statenz_1= read.xlsx("data/stat_enzyme.xlsx", sheetIndex=1, startRow=4, header=TRUE,
stringsAsFactors = FALSE )

# show the first lines of the table
head(statenz_1)

# rmq1 : startRow=4 <-> skip=3
# rmq2. pbl de chiffre significatif affiché (moins en passant directement en xls, et je
ne voie pas le réglage

# # lecture de l'onglet 2 - cultivar
statenz_2= read.xlsx("data/stat_enzyme.xlsx", sheetName="cultivar", startRow=1,
header=TRUE, stringsAsFactors = FALSE)
# remplace les valeurs manquante par NA
statenz_2[statenz_2=="-"]=NA
statenz_2$TT_x = as.numeric(statenz_2$TT_x)

# si pertinent, jointure sur les deux onglets/tables:
# rmq1. Essayer de garder précisément les mêmes noms de colonne lors de la conception
des fichiers de saisie (dont majuscule/minuscule/accent)
statenz<-merge(statenz_1, statenz_2, by.x=c("cultivar"),by.y=c("cultivar"))

head(statenz)
str(statenz)

# un premier graphique...
plot(statenz$TT_x, statenz$LOX)

# end of file

```

```
1: # nuage de points
2: df1 <- statenz[statenz$cultivar=="v1", c("PO", "jours")]
3: plot(PO~jours, df1, pch=19, col="red")
4:
5: # diagramme en bâtons
6: x <- c(1,5,8,-2,7)
7: barplot(x)
8:
9: # histogramme
10: setwd("D:/R_ITA/")
11: weather<- read.delim("data/meteo.dat",dec=',')
12: hist(weather$stmin, xlab="Tmin", main=" ")
13:
14: # boxplot
15: boxplot(statenz$GST)
16:
17: # camembert
18: pie(table(statenz$cultivar), col=1:5, clockwise=TRUE, cex=2)
19:
20: # graph multiples
21: par(mfrow=c(2,2))
22: x<- seq(0,2,by=0.1)*pi
23: y<- sin(x)
24: z<- cos(x)
25: plot(x,y)
26: plot(x,y,type="l")
27: plot(x,y,type="l")
28:   points(x,z)
29: plot(x,y,type="l",xlab="Input", ylab="Output", xlim=c(-1,7))
30:   lines(x,z,lwd=3,lty=4)
31:
32: # nuage de points avec nom des individus
33: par(mfrow=c(1,1), cex=0.5)
34: plot(weather$stmin, weather$stmax, type="n")
35: text(weather$stmin, weather$stmax, labels=weather$day)
36:
37:
```

```
1: # comparaison de 2 moyennes
2: t.test(LOX~inocule, statenz, var.equal=TRUE)
3:
4: # comparaison de 2 variances
5: var.test(LOX~inocule, statenz)
6:
7: # corrélation
8: cor(statenz$PO, statenz$jours)
9: cor.test(~PO+jours,statenz)
10:
11: # tableau croisé : 1 variable quali * 1 variable quanti
12: with(statenz, tapply(GST,cultivar, mean))
13:
14: # tableau croisé : 2 variables quali * 1 variable quanti
15: with(statenz, tapply(GST,list(cultivar, inocule), mean))
16:
17: # tableau croisé : 2 variables quali, et test du chi2
18: temp<- data.frame(
19:   V1=sample(letters[1:5],300, replace=T),
20:   V2=sample(LETTERS[1:5],300, replace=T))
21: table(temp$V1, temp$V2)
22: chisq.test(table(temp))
23:
24: # test de normalité
25: W <- statenz$GST
26: shapiro.test(W)
27:
28: # régression linéaire simple
29: setwd("C:/R_ITA/")
30: weather <- read.table("data/meteo.dat", h = TRUE, dec = ',')
31: head(weather)
32: Fit<- lm(weather$Tmax~weather$radiation)
33: print(Fit)
34: summary(Fit)
35: COEF <- Fit$coefficients
36: plot(weather$radiation,weather$Tmax, xlab="radiation", ylab="Tmax")
37: abline(a=COEF[1], b=COEF[2],lty=2)
38:
39: # analyse de la variance
40: toto <- statenz[statenz$cultivar=="v1" &
41:   statenz$inocule=="i",]
42: toto$series <- as.factor(toto$series)
43: res_lm <- lm(PO~jours + series, toto)
44: anova(res_lm)
45: summary(res_lm)$sigma # ETR
46:
47:
48:
49:
```

```
1: setwd("D:/R_ITA/")
2: # sauvegarde d 'un graphique au format png
3: png("output/nom_fichier.png")
4:   barplot(1:5)
5: dev.off()
6:
7: # sauvegarde de fichier
8: data(iris)
9: write.table(iris, file="iris.txt", row.names=F, sep="\t")
10:
11: # sauvegarde de résultats
12: sink("output/synthese.txt")
13:   anova(res_lm)
14:   summary(weather)
15: sink()
16:
```

```
1:
2: # boucle
3: X<- matrix(seq(1,60,by=1), nrow=10, ncol=6)
4: X
5: Y<- rep(NA, 10)
6:
7: for (i in 1:10) {
8:   Y[i]<- sum(X[i,]) }
9: Y
10:
11: apply(X, 1, sum)
12: apply(X, 2, sum)
13:
14: # condition
15: cond <- TRUE
16: if (cond) print("OK") else print("pas OK")
17:
18: # boucle et condition
19: setwd("D:/R_ITA/")
20: weather <- read.table("data/meteo.dat", h = TRUE, dec = ',')
21:
22: TMIN <- weather$tmin
23: for (i in 1:length(TMIN)) {
24:   if (TMIN[i] < 24) TMIN[i] <- 0
25: } # NON !!! pas efficace
26:
27: TMIN[TMIN < 24] <- 0
28:
29: # fonction
30: MeanTemp<- function(Tmin, Tmax) {
31:   Tmean<-(Tmin+Tmax)/2
32:   return(Tmean)
33: }
34: MeanTemp(weather$tmin, weather$tmax)
35:
36:
```

```
1: # variable quanti
2: x<- statenz$GST
3: x[c(10,20,30)]<- NA
4: sum(is.na(x))
5: x[!is.na(x)]
6: x[is.na(x)]<- mean(x, na.rm=TRUE)
7:
8: # variable quali
9: x<- as.character(statenz$cultivar)
10: x[c(10,20,30)]<- NA
11: sum(is.na(x))
12: x[is.na(x)]<- "inconnu"
13:
14:
15:
16:
```