

```

install.packages("readr") #Installer le package "readr"

library(readr) #package permet d'importer les fichiers csv
Tabagisme <- read.csv2("//calebasse/niare/Bureau/Exemples bases de données/Tabagisme.csv",header= TRUE,
sep=";",dec=",")
View(Tabagisme) #Voir la base de données "Tabagisme"

names(Tabagisme) #names( ): permet d'afficher toutes les variables

dim(Tabagisme) #dim( ): permet d'afficher le nombre d'individus et de variables

getwd()# connaître de repertoire de travail

install.packages("questionr")# Installer le package "questionr"

library(questionr)# permet de renommer les variables
rename.variable(df, old, new)# renommer une variable,df est le nom de la base, old c'est l'ancien nom et new le nouvo
rename.variable(Tabagisme,"fum_reg","fumeur_regulier")

Tabagisme= rename.variable(Tabagisme,"fum_reg","fumeur_regulier")

# Accéder à un sous ensemble de variables ou d'individus: [. , .]
#Une partie des individus
Tabagisme[c(1:50, 88, 95,115), ]

#Une partie des variables
Tabagisme[,c(2:4,8:10,12)]

#Une partie des variables et des individus
Tabagisme[c(1:25, 30:60,72,120), c(2:4,8:10,12)]

#La fonction subset( ) Elle permet d'extraire un sous-ensemble en posant une condition sur une variable
Femmes=subset (Tabagisme, sexe=="Femme")
Femmes

Femmes1=subset(Tabagisme, sexe=="Femme", c(age, epargne, exp_tabac))
Femmes1

#ou
Femmes1=subset(Tabagisme,sexe=="Femme", c(2,8,12))
Femmes1

#Extraction avec deux conditions:" être femme " et n'avoir aucune données manquante sur les variables "fum_reg" et
"nbcig"

F=subset(Tabagisme, sexe=="Femme" & complete.cases (fumeur_regulier, nbcig))
F

#Extraction avec trois conditions: " être homme ", " avoir une épargne >=1000 euros" et n'avoir aucune données
manquante sur les variables "fum_reg" et "nbcig"

Hommes=subset(Tabagisme, sexe=="Homme" & epargne>=1000 & complete.cases (fumeur_regulier, nbcig), c(age,
sexe,fumeur_regulier,nbcig, epargne))
Hommes

str(Tabagisme) #Afficher les variables et leur type

#Changer le type des variables inappropriées

Tabagisme$etude=as.factor(Tabagisme$etude)
Tabagisme$csp=as.factor(Tabagisme$csp)
Tabagisme$statut_tab=as.factor(Tabagisme$statut_tab)

summary(Tabagisme)#Afficher les statistiques descriptives

csp1=subset(Tabagisme,csp=="1")# Extraire la csp1
csp1
mean(csp1$epargne)#Calcul de la moyenne de l'épargne de la csp1

Taba<-na.omit(Tabagisme)#Supprimer les valeurs manquantes

#Graphiques
#Graphique en moustache
boxplot(Tabagisme$age,ylab="Âges",main="Répartition de l'âge des 250 personnes")
boxplot(Tabagisme$revenu2014~Tabagisme$sexe,ylab="Revenu en 2014",main="Le revenu en 2014 des 250 personnes selon le
sexe",col = "cyan",range=0)
# range=0:Permet de prendre en compte les valeurs isolées dans le graphique

#Graphique nuage de points

```

```

plot(Tabagisme$epargne,ylab="epargne",xlab="individus",main="Epargne des 250 personnes")
plot(Tabagisme$age,Tabagisme$revenu2010,xlab="Âges",ylab="Revenus en 2010",col="blue",main="Le revenu en 2010 en
fonction de l'âge")
pairs(Tabagisme[,c(2,5,6,12)])# Nuage de points regroupés
library(lattice)
xyplot(revenu2010~age,data=Tabagisme,main="Le revenu en 2010 en fonction de l'âge")

#Graphique ou diagramme en barre
barplot(table(Tabagisme$csp),ylab="Effectifs",xlab="CSP",main="La catégorie socio-pro",col = "cornflowerblue")
barplot(table(Tabagisme$sexe),ylab="Effectifs",xlab="Sexe",col = "orange",space =0.6)

###Tris-à-plat
library(questionr) #le package permettant de faire les tris-à-plats et les tableaux croisés
sexe=table(Tabagisme$sexe) #les effectifs
sexe

freq(sexe)# les pourcentages
#val%=la répartition en pourcentages, données manquantes exclues

freq(sexe, cum = TRUE, total=TRUE,sort = "inc",digits = 1,exclude = NA)

###Tableaux croisés
statut=table(Tabagisme$sexe, Tabagisme$statut_tab) #les effectifs, tableau de contingence
statut
copie(statut)# exporter ou coller sur une feuille excel

lprop(statut, digits = 1, percent = TRUE)#lprop( ): permet le calcul des pourcentages en ligne
#percent=TRUE: permet d'afficher le symbole %

cprop(statut,digits = 0, percent = TRUE) #Cprop( ): permet le calcul des pourcentages en colonne

library(questionr)##le package permettant de faire les tris-à-plats et les tableaux croisés
library(R2HTML) #le package permettant d'exporter les tableaux croisés
#exporter le tableau du tri croisé % en colonne
table.export=cprop(statut,digits = 1, percent = TRUE)
copie(table.export) #après exécution, ouvrir une feuille excel et faire un contrôle v (coller)

### TEST DU KHI2 (Vérifier la relation entre 2 variables qualitatives)

statut= matrix(c(67,45,39,41,35,23),nrow=2,ncol = 3)
statut
chisq.test(statut)

#Ou simplement
chisq.test(statut)

#Vérifier la Relation de corrélation entre 2 variables quantitatives

cor(Tabagisme$nbcbig,Tabagisme$epargne,use="complete")#Affiche la valeur du coef de corrélation
cor.test(Tabagisme$nbcbig,Tabagisme$epargne)#Calcul du coef de corrélation avec intervalle de conf
#OU

TAB=na.omit(Tabagisme[,c(11,12)])#Créer une base "TAB" en supprimant les valeurs manquantes
cor(TAB$nbcbig,TAB$epargne)#Affiche la valeur du coef de corrélation
cor.test(TAB$nbcbig,TAB$epargne)#Calcul du coef de corrélation avec intervalle de conf

#Vérifier la relation entre une variable quali et d'une var quanti(ANOVA ou analyse de la variance)

Analyse=lm(epargne~csp,data=Tabagisme)
anova(Analyse)
summary(Analyse)#cellule de référence "csp=1"

summary(lm(epargne~ C(csp,base=2),data=Tabagisme))#Changer de cellule de référence "csp=2"

#Quelles sont les moyennes qui diffèrent significativement des autres?
#Comparaison deux à deux (méthode de Bonferroni)

pairwise.t.test(Tabagisme$epargne,Tabagisme$csp,p.adj="bonferroni")

#Effectuons un test de normalité
Femmes=subset (Tabagisme, sexe=="Femme")
Femmes

Femmes1=(Femmes[,12])#Création de la variable "indice_frontal" et affectation des indices correspondants
Femmes1
qqnorm(Femmes1,datax=TRUE)#Graphique 1 de la normalité
qqline(Femmes1,datax=TRUE)#Graphique 2 de la normalité
shapiro.test(Femmes1)#Test de normalité

#Effectuons un test de normalité

```

```
Hommes=subset (Tabagisme, sexe=="Homme")
Hommes

Hommes1=(Hommes[,12])#Création de la variable "indice_frontal" et affectation des indices correspondants
Hommes1
qqnorm(Hommes1,datax=TRUE)#Graphique 1 de la normalité
qqline(Hommes1,datax=TRUE)#Graphique 2 de la normalité
shapiro.test(Hommes1)#Test de normalité

#Conclusion: les p-value sont significatives, l'échantillon ne suit pas une loi normale.

t.test(Femmes1, Hommes1, paired=FALSE)#Test de student
```