

## Les fonctions du fichier data mining

```
#- - - - -#
# Lecture et mise en forme du fichier de données "visprem.dat" #
#- - - - -#

remplace_function(vec,modinit,modfin)
{
  vec1_rep(0,length(vec))
  for (i in 1:length(vec))
  {
    for (j in 1:length(modinit))
    {
      if (vec[i]==modinit[j]) vec1[i]=modfin[j];
    }
  }
  return(vec1)
}

qual2quant_function(vec)
{
  t=as.numeric(as.character(vec))
  attributes(t)$class="numeric"
  return(t)
}

#- - - - -#
# Analyse exploratoire #
#- - - - -#

isna_function(vec,caract="quantitatif")
{
  if (caract=="quantitatif")
  {
    sumna=sum(is.na(as.numeric(vec)))
    return(sumna)
  }
  if (caract=="qualitatif")
  {
    inc_substr(as.character(vec),2,4)=="inc"
    sumna=sum(inc)
    return(sumna)
  }
}

ana.uni.quanti_function(var,table=visprem,
  # - - - - arguments de la fonction boxplot - - - - #
  range = 1.5, width = NULL, varwidth = FALSE,
  notch = FALSE, outline = TRUE, border = par("fg"),
  col = NULL, pars = NULL, horizontal = FALSE,
  # - - - - arguments de la fonction hist - - - - #
  breaks2 = "Sturges",col2 = NULL, border2 = NULL,
  main2 = paste("Histogramme de la variable" , var),
  axes2=TRUE)
{
  vec_table[,var]
  vm_isna(vec)
  if (vm!= 0)
  {
    print("_____ ",quote =
FALSE)
    print(paste("Il y'a",vm,"valeur(s) manquante(s)"))
  }
  print("_____ ",quote = FALSE)
  print(paste("Distribution de la variable",var))
  print(summary(vec))
  print("_____ ",quote = FALSE)

  boxplot(vec,
    range = range, width = width, varwidth = varwidth,
    notch = notch, outline = outline, border = border,
    col = col, pars = pars, horizontal = horizontal)
  title(paste("Distribution de la variable",var))
  x11()
  hist(vec,breaks = breaks2,col = col2, border = border2,
```

Cette fonction sert à recoder les variables qualitatives. Les termes du vecteur *modinit* sont remplacés par leur homologue du vecteur *modfin*.

Cette fonction sert à modifier la classe de vecteurs quantitatifs, considéré comme qualitatif suite à l'importation de la table de données.

Cette fonction renvoie le nombre de valeurs manquantes d'un vecteur.

Cette fonction sert à produire les éléments nécessaires à l'analyse descriptive d'une variable quantitative.

L'utilisateur rentre le nom de la variable et le nom de la table dans laquelle elle se trouve. Il obtient en sortie « écran » les quartiles et 2 fenêtres graphiques : la boîte à moustache et l'histogramme.

```

    main=main2,axes=axes2)
}

ana.uni.quali_function(var,table=visprem,
                      na=TRUE,edges = 200, radius = 0.8,
                      density = NULL, angle = 45,main = NULL)
{
vec_table[,var]
dist_summary(vec)
vm_isna(vec,caract="qualitatif")
if (vm!= 0)
  {
  print("_____",quote = FALSE)
  print(paste("Il y'a",vm,"valeur(s) manquante(s)"))
  }
print("_____",quote = FALSE)
print(paste("Distribution de la variable",var))
print(dist)
print("_____",quote = FALSE)

if (na=="FALSE")
  {
  repart_dist[substr(attributes(dist)$names,2,4)!="inc"]
  }
if (na=="TRUE")
  {
  repart_dist
  }

prop_round(100*repart/sum(repart),2)
leg_rep(0,length(repart))
for (i in 1:length(repart))
  {
  leg[i]_paste(attributes(repart)$names[i]," (",prop[i],
%)",sep="")
  }
pie(repart, labels =leg, edges = edges, radius = radius,
    density = density, angle = angle, col
=rainbow(length(repart)),main = main)
title(paste("Distribution de la variable",var))
}

ana.bi.quanti_function(var1,var2,table=visprem,
                      var3="CARVP",mod="Coui",col="red",
                      type="p",xlab=var1, ylab=var2)
{
don_matrix(0,dim(table)[1],3)
vec1_table[,var1]
vec2_table[,var2]
vec3_as.character(table[,var3])

don_as.data.frame(cbind(table[,var1],table[,var2],as.character(table
[,var3])))
don[,1]_qual2quant(don[,1])
don[,2]_qual2quant(don[,2])

vm_sum(is.na(don[,1]*don[,2]))
if (vm!= 0)
  {
  print("_____",quote =
FALSE)
  print(paste(vm,"couple(s) d'observations est(sont)
incomplet(s)"))
  }

cor_cor(don[,1],don[,2],use="pairwise.complete.obs")
print("_____",quote = FALSE)
print(paste("Le coefficient de corrélation entre les
variables",var1,"et",var2,"est :",quote=FALSE))
print(cor)
print("_____",quote = FALSE)

don1_don[don[,3]!=mod,]
don2_don[don[,3]==mod,]
x11()
plot(don1[,1],don1[,2], xlim=range(don[,1]), ylim=range(don[,2]),

```

Cette fonction sert à produire les éléments nécessaires à l'analyse descriptive d'une variable qualitative.

L'utilisateur rentre le nom de la variable et le nom de la table dans laquelle elle se trouve. Il obtient en sortie « écran » la ventilation des individus observés sur les différentes modalités, et, dans une fenêtre, le camembert correspondant.

Par ailleurs, j'ai pensé qu'il pouvait être intéressant d'inclure ou non les valeurs manquantes dans la représentation graphique. Le paramètre *na* sert à cela.

Cette fonction sert à produire les éléments nécessaires à l'analyse de la liaison de 2 variables quantitatives (coefficients de corrélations et nuage de points).

Le nuage de point met en évidence les individus possédant une certaine caractéristique (modalité *mod* de la variable *var3*).

```

type=type,
  xlab=xlab, ylab=ylab)
points(don2[,1],don2[,2],col=col)
title(paste("Nuage de points des variables",var1,"et",var2))

}

ana.bi.quali_function(var1,var2,table=visprem)
{
vec1_table[,var1]
vec2_table[,var2]
nlev1_length(attributes(vec1)$levels)
nlev2_length(attributes(vec2)$levels)

conting_matrix(0,nlev1,nlev2)
cont_by(vec1,vec2,summary)
for (i in 1:nlev2)
  {
    conting[,i]_cont[[i]]
  }
nomi_attributes(vec1)$levels
nomj_attributes(vec2)$levels
dimnames(conting)_list(nomi,nomj)

print("_____",quote = FALSE)
print(paste("Table de contingence",var1,"x",var2,"(en effectif)",
quote = FALSE)
print(conting)

contingp_contingp_round(100*conting/sum(conting),2)
print("_____",quote = FALSE)
print(paste("Table de contingence",var1,"x",var2,"(en pourcentage du
total)",quote = FALSE)
print(contingp)

barplot(contingp, beside = TRUE,col = rainbow(nlev1),legend =
rownames(contingp), ylim = c(0, max(50,max(contingp)+5)))
title(paste("Table de contingence des
variables",var1,"et",var2,"\n(en pourcentage)"))

}

ana.bi.mixte_function(var,varcl,table=visprem,
  range = 1.5, width = NULL, varwidth = FALSE,
  notch= FALSE, outline= TRUE, border= par("fg"),
  col = NULL, pars = NULL, horizontal = FALSE)
{
vec1_table[,var]
vec2_table[,varcl]
nlev_length(attributes(vec2)$levels)

distcomp0_by(vec1,vec2,summary)
d1_distcomp0[[1]]
d2_distcomp0[[2]]
distcomp_as.matrix(rbind(d1,d2))
dimnames(distcomp)[[1]][1]_attributes(vec2)$levels[[1]]
dimnames(distcomp)[[1]][2]_attributes(vec2)$levels[[2]]

print("_____",quote = FALSE)
print(paste("Distribution de la variable",var,"selon les modalités
de la variable",varcl),quote = FALSE)
print(distcomp)
print("_____",quote = FALSE)

x11()
boxplot(vec1~vec2,
  range = range, width = width, varwidth = varwidth,
  notch = notch, outline = outline, border = border,
  col = col, pars = pars, horizontal = horizontal)
title(paste("Distribution de la variable",var,"\n\nselon les modalités
de la variable",varcl))
}

```

Cette fonction donne la table de contingence (en effectif et en pourcentage du total) et produit un diagramme en barre multiple.

Cette fonction sert à décrire la liaison entre une variable quantitative et une variable qualitative. Pour cela, elle fournit les quartiles pour chaque sous-échantillon, ainsi que les boites à moustaches multiples.