

A.C.P.

PACKAGES à installer : FactoMineR, factoextra,

Library à charger :

library(ggpubr) : pour enregistrer les sorties.

library(corrplot) : pour marquer les variables bien corrélées avec les différentes composantes principales.

```
install.packages(c("FactoMineR", "factoextra"))
```

Chargez-les dans R:

```
library("FactoMineR")
```

```
library("factoextra")
```

```
res=PCA(X, scale.unit = TRUE, ncp = q, graph = TRUE, quant.sup=, qual.sup=, ind.sup=)
```

X : tableau de données à analyser

scale.unit = T pour une A.C.P.N si F A.C.P.C

ncp = q (à donner) nombre de composantes principales

quant.sup = on donne le(les) numéro(s) de variable(s) quantitative supplémentaire.

De même pour qual.sup et ind.sup

VISUALISATION DES RESULTATS

```
eig.val=get_eigenvalue(res)
```

```
fviz_eig(res, addlabels = TRUE, ylim = c(0, 50))#scree plot
```

CAS DES VARIABLES

```
var=get_pca_var(res)
```

```
var$coord
```

```
var$cos2
```

```
fviz_pca_var(res, col.var = "cos2", gradient.cols = c("green", "blue", "red"), repel = TRUE)
```

#repel : Évite le chevauchement de texte

#col.var : les variables seront marquées par leur cos2 et gradient.cols marquera les variables à cos 2 faible en vert, moyen en bleu et élevé en rouge

```
library("corrplot")
```

```
corrplot(var$cos2)#visualiser les corrélations
```

CAS DES INDIVIDUS

```
ind=get_pca_ind(res)
```

```
ind
```

```
ind$coord # ou head(ind$coord)
```

```
ind$cos2
```

```
ind$contrib
```

```
fviz_pca_ind (res, col.ind = "cos2", gradient.cols = c("red", "blue", "green"), repel = TRUE)#par défaut le premier plan principal sinon on ajoute axes=c(r,s)
```

ou alors

```
fviz_pca_ind (res, pointsize = "cos2", pointshape = 21, fill = "#E7B800", repel = TRUE)
```

#modifier la taille des points en fonction du cos2 des individus correspondants.

Ou aussi

Pour modifier la taille et la couleur des points en fonction du cos2 :

```
fviz_pca_ind(res.pca, col.ind = "cos2", pointsize = "cos2", gradient.cols = c("#00AFBB",  
"#E7B800", "#FC4E07"), repel = TRUE)  
fviz_contrib(res, choice = "ind", axes = 1:2) # permet de repérer les individus ayant fortes  
contributions.
```

POUR FILTRER DES INDIVIDUS OU VARIABLES (DATA DE GRANDE TAILLE !!)

```
# Visualiser les variables avec cos2 >= 0.6  
fviz_pca_var(res, select.var = list(cos2 = 0.6))  
# Top 5 variables actives avec le cos2 le plus élevé  
fviz_pca_var(res, select.var = list(cos2 = 5))  
# Sélectionnez par noms  
name = list(name = c("", "", ""))  
fviz_pca_var(res, select.var = name)  
# Top 5 des individus/variables les plus contributifs  
fviz_pca_biplot(res.pca, select.ind = list(contrib = 5))
```

VISUALISATION DES CLASSES PAR kmeans

```
X = res$ind$coord  
Res.kmeans = kmeans(X[, 1:2], centers = k)  
Plotk = fviz_cluster(Res.kmeans, X[, 1:2])
```

ENREGISTRER DES SORTIES

GRAPHES :

```
library(ggpubr)  
ggexport(plotlist = list(scree.plot, ind.plot, var.plot, Plotk), nrow = 2, ncol = 2, filename =  
"PCA.pdf")
```

RESULTATS :

```
# Exporter vers un fichier TXT  
write.infile(res.pca, "pca.txt", sep = "\t")  
# Exporter vers un fichier CSV  
write.infile(res.pca, "pca.csv", sep = ";")
```