ACP

Yanice

20 novembre 2016

Contents

Introduction ACP

- On cherche dans l'espace des variables, une combinaison linéaire des variables de départ qui maximise la variance, avec orthogonalité
- On centre les données car le sous espace de projection qu'on cherche est celui qui passe par le centre de gravité. On pourra analyser les variations autour du centre de gravité.
- liens analysés : covariance ou corélations(ACP réduite).
- 1ere composante principale : axe qui maximise la variance.
- Trop d'information redondante.L'ACP réduit le nombre de variable.
- Indicateur sur les informations d'un graphique pour les individus : cos2 : bien ou mal présentés.
- Axe principal s'interprete en fonction des variables qui y sont corrélés. (cercle de correlation)
- Valeurs propres trop proches : pas de corrélations, orthogonale dans l'espace. Absence de corrélations, difficultés de réduire les axes, pas d'intret à former des axes principaux.
- Qualité des données -> boite a moustaches, valeurs abhérrentes.
- paramètre a faire varier pour observer plus de choses : pour 1000 habitants etc
- **inertie** : information restituée = variance, je choisis les axes qui maximisent l'inertie donc la variance.mesure générale de la variabilité.

- ACP : Trouver un nouveau système d'axe à l'aide de rotation pour avoir un point de vue ou al variance est maximale (diagonalisation)
- On veut que les projections des points sur les nouveaux axes soient le plus dispersés possible. #I. Visualisation des données - Centrage et reduction

I.1. Visualisation des données : plot

```
plot(mesures,col=couleur2[groupe],pch=15)
plot(objet)
```

I.1. Choix de Centrage et réduction ?

- Variances très différentes ou unités très différentes ==> Réduction pour ramener à la même échelle. (réduction : on divise par l'eccart type !!) On obtient la matrice de corrélation si pas de réduction et matrice de covariance si il ya réduction.

Cas de l'ex3 particulier :

```
apply(macon,2,sum)
    #Tous mes juges ont noté de 1 à 8
apply(macon,1,sum)
```

Remarque sur l'ACP: qu'elle soit normé ou pas ca ne change pas car on a la même somme totale mais on norme l'ACP car on interprete les résultats sur des ACP normées pour les cercles de corrélation.

Cas général

```
  colMeans(mesures)
  sapply(mesures,var)

  colMeans(objet)
  sapply(objet,var)

  boxplot(objet,horizontal=TRUE,las=1)
```

Les moyennes et variances sont très différentes d'une dépense à l'autre. Il faut donc **donner à chaque variable une même importance**. D'où le centrage et réduction des données.

```
library(ade4)
objet.cr=scalewt(objet,center=TRUE,scale=TRUE)#objet doit etre data.frame
#class(objet.cr)#matrix
#On rend notre objet en data.frame :
objetcr=as.data.frame(objet.cr)
```

```r
#Vérification que nos nouvelles données sont centrées réduites
  round(sapply(objet,mean),2)
  sapply(objetcr,var)
  sapply(objetcr,sd)

#Moyenne par groupe
  #Ne s'applique pas sur un data frame mais sur un vecteur!!! :
  tapply(objetcr[,2],groupe,mean)

#calcul la moyenne en fonction de groupe pour les n colonnes
  n=dim(objet)[2]
  sapply(1:n,function(x) tapply(objet[,x],groupe,mean))
```

Dans la représentation 3D des données centrées et réduite, le premier axe correspond au plus grand diamètre de l'ellipsoide (la longueu de la dragée), le 2nd à la largeur de la dragée et le 3eme à l'épaisseur de la dragée.

On cherche donc les axes qui nous donnent une représentation des données avec le moins de perte d'information par rapport au nuage de point initial. (Points les plus étalés dans le plan). On dit qu'on a conservé le maximum de l'inertie initiale du nuage de point.

II. ACP

```r
library(ade4)
acp=dudi.pca(objet,center=TRUE,scannf=FALSE,nf=2)
```

On conserve 3 axes pour mesures et 2 pour SR dans nos exemples.

II.1. Sorties de l'ACP

```r
  # Données du tableau initial après centrage et réduction
  head(acp$tab) #en 1/n pour la variance (scale en 1/n-1)

  #Poids
  acp$cw#poids des colonnes
  acp$lw#poids des lignes

  #valeurs propres dans le plus petit des deux espaces diagonalisé
  acp$eig#Nous renseigne sur la fraction de l'inertie totale prise en compte par chaque axe
  pve=100*acp$eig/sum(acp$eig)#Pourcentage d'inertie de chaque axe
  cumsum(pve)#Pourcentage cumulé

  #rang de la matrice diagonalisé : nombre de variables indépendantes
  acp$rank

  #Nombre de facteur conservé dans l'ACP
  acp$nf

  #AXES PRINCIPAUX (A: vecteurs propre de la diagonalisation dans R^n)
  acp$c1 #de norme1
```

```
  #Calcul de la norme :
  sapply(1:acp$nf,function(x) sum(acp$cw*acp$c1[,x]^2))

  #Coordonnés des individus (lignes, L=XQA) ou dans l'ACP Q=Identité de taille p
  acp$li #vecteur normés a la racine carré des valeurs propres correspondantes

  #Composantes principales (c'est K: vecteur propre de la diagonalisationd ans R^p)
  acp$l1#norme1
  sapply(1:acp$nf,function(x) sum(acp$lw*acp$l1[,x]^2))#pour vérifier que c'est de norme 1

  #Coordonnés des variable (C=t(X)DK) ou D=1/n
  acp$co#Les vecteurs sont normés à la racine carré des valeurs propres correspondantes
  #Doit etre égale aux valeurs propres
  sapply(1:acp$nf,function(x) sum(acp$cw*acp$co[,x]^2))==acp$eig

  #Liens entre c1 et c0
  acp$c1$CS1 * sqrt(acp$eig[1])#Pour l'axe 1
  t(t(acp$c1) * sqrt(acp$eig))

  #Liens entre l1 et li
  head(t(t(acp$l1) * sqrt(acp$eig)))#RS1 : pour l'axe 1 etc

  acp$call

  #moyenne des varaibles analysées (de départ)
  acp$cent

  #eccart type des var analysé sur racine de n
  acp$norm
  #Vérification
  var.n=function(x) sum((x-mean(x))^2)/length(x)
  sd.n= function(objet) sqrt(var.n(objet))
  apply(objet,2,sd.n)
  apply(objet,2,sd)*sqrt(dim(objet[1])-1)/sqrt(dim(objet)[1])

#reconstitution des données de départ
#Partant de acp$cent retrouver le tableau d'origine
recon=t(t(acp$tab)*((acp$norm*sqrt(length(mesures[,3]))/sqrt(length(mesures[,3]-1))))+acp$cent)
```

II.2. Choix des axes - Graph des valeurs propres

1. D'après le **critere de kaiser** : On conserve les valeurs propres au dessus de 1 dans une ACP normée.

2. D'après la **règle du point d'inflexion ou du coude** : On conserve les valeurs propres qui se situent avant un point d'inflexion sur l'histogramme des valeurs propres décroissant. Si il existe plusieurs points, on ne conserve que celles qui sont situées avant le 1er point.

On doit avoir **min(nbre individus,nbre de variable)** axes. Sinon, cela s'explique car la derniere ligne est une combinaison linéaire de toutes les autres variables. J'ai un rang en moins a ma matrice donc jai 8-1 donc 7 valeurs propres.

```
par(mfrow=c(1,2))
#Représentation 1
barplot(acpSR$eig,main="Représentation des valeurs propres",col="blue")
#Représentation 2
screeplot(acpSR,main="Avec Screeplot")
```

Pourcentage de l'inertie représentée par les axes :

```
  #Méthode 1 :
  summary(acp)
  #Méthode 2:
  (pve=100*acp$eig/sum(acp$eig))
  cumsum(pve)
```

Le premier axe factoriel extrait a% de l'inertie initiale et le deuxieme b% de l'inertie totale. Le premier plan factoriel représente donc a+b% de l'inertie initiale. (cela signifie qu'en prjettant les points dans le nouveau plan, on pert peu dinfo). Les valeurs propres nous renseignent sur la fraction de l'inertie totale prise en compte par chaque axe.

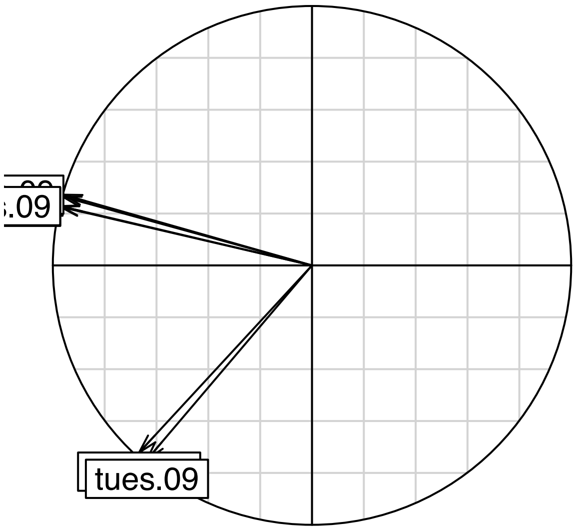
II.3. Projection des variables : Cercle des corrélations

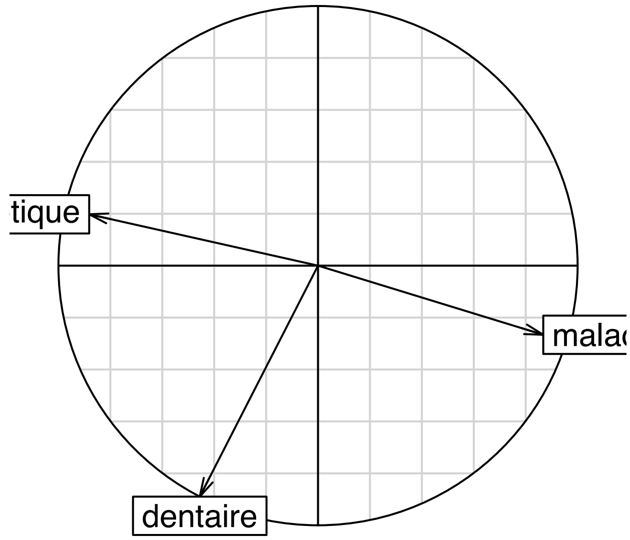
La projection des variables sur les composantes principales synthétise les covariances entre les variables initiales et les variables artificielles (i.e. les composantes principales).

Remarque : Dans le cas des variables réduites, la covariance s'interprète comme des corrélations.

Les projections des variables sont à interprétées comme des **directions**. Chaque carreau c'est 0.2, je projette le bout des fleches de maniere orthogonale sur les axes.

```
par(mfrow=c(1,2))
s.corcircle(acp$co,xax=1,yax=2)
s.corcircle(acpM$co)
```





- Dans l'ex1, On constate que les données sont structurées. Il y a coincidence parfaite entre ce qui se passe en 2009 et ce qui se passe en 2010.

Interprétation de l'axe horizontale

Il y a ici un **effet de taille**, i.e. que les variables sont toutes du même côté de l'axe. On pourra donc regrouper les départements qui cumulent toutes les variables et ceux qui en ont très peu. Elles font partie d'un même ensemble de variable : effet accidentodène. On pourra donc discerné les départements où j'ai beaucoup de problème de sécurité routière et d'autre ou j'en ai moins.

Interprétation de l'axe verticale

L'axe verticale représente 22% de la variabilité totale. Les variables tués sont fortement corrélées avec cet axe. Cet axe spécifie donc les départements où il y aura plus de tué que de blessés. La structure très forte entre les deux années indique qu'il n'ya rien eu de fait pour améliorer les problèmes de sécurité.

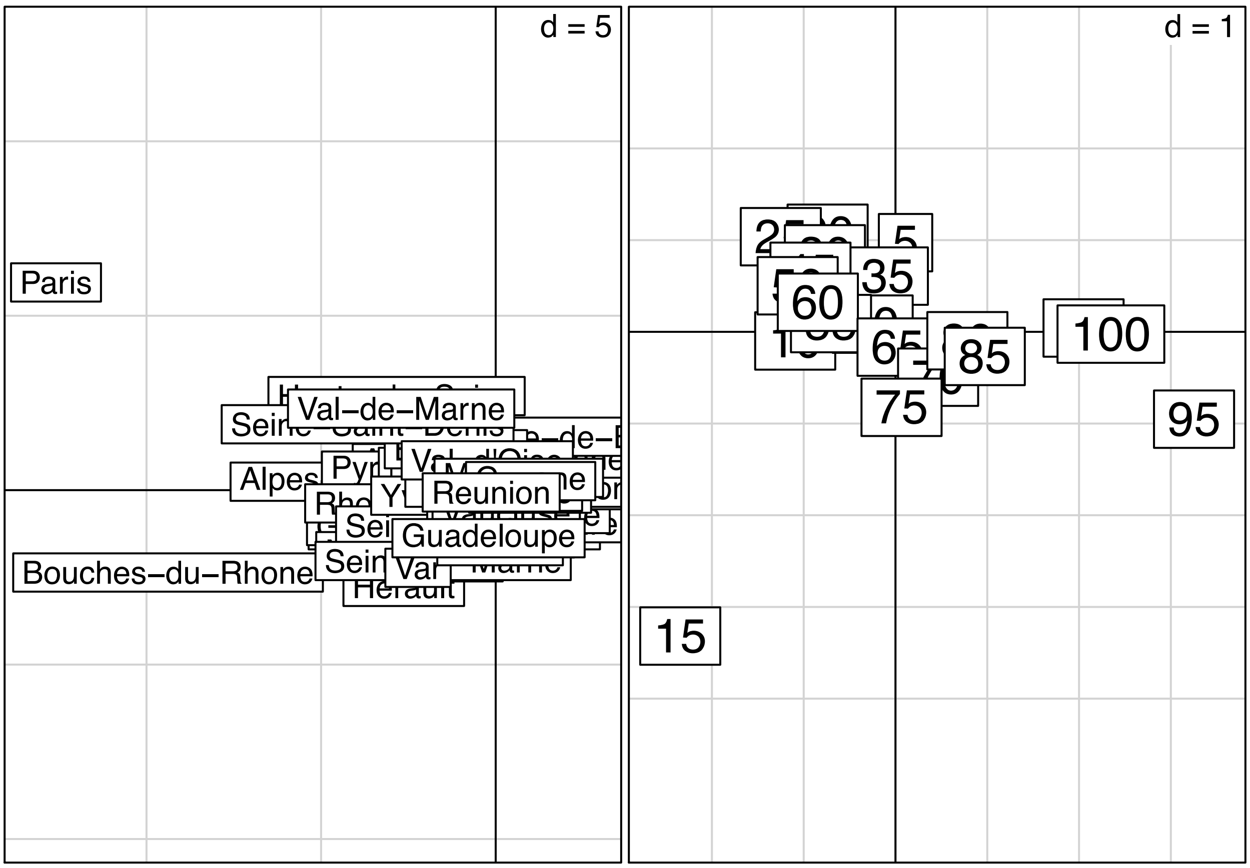
- Dans l'ex2 :l'axe horizontale : oppose les dépeneses liées a la maladie de ceux a l'optique et un petit peu au dentaire. Sur l'axe verticale : optique et maladie interviennent peu.
  Le dentaire est pratiquement à 0.9 , il intervient sur l'axe verticale

II.4. Représentation des individus

Les individus qui créent la variabilité sont ceux qui sont éloignés du centre de gravité. Les autres (ceux proches) ne sont aps interprétables

```
#s.label(acp$li) #attention ici c'est le numéro des lignes
#nom=
#s.label(acp$li,label=nom)

#s.label(acp$li,label=SR0910$numdep)
par(mfrow=c(1,2))
s.label(acp$li,label=SR0910$departement)
s.label(acpM$li,xax=1,yax=2,label=as.character((depsante$age)),clabel=1.5)
```

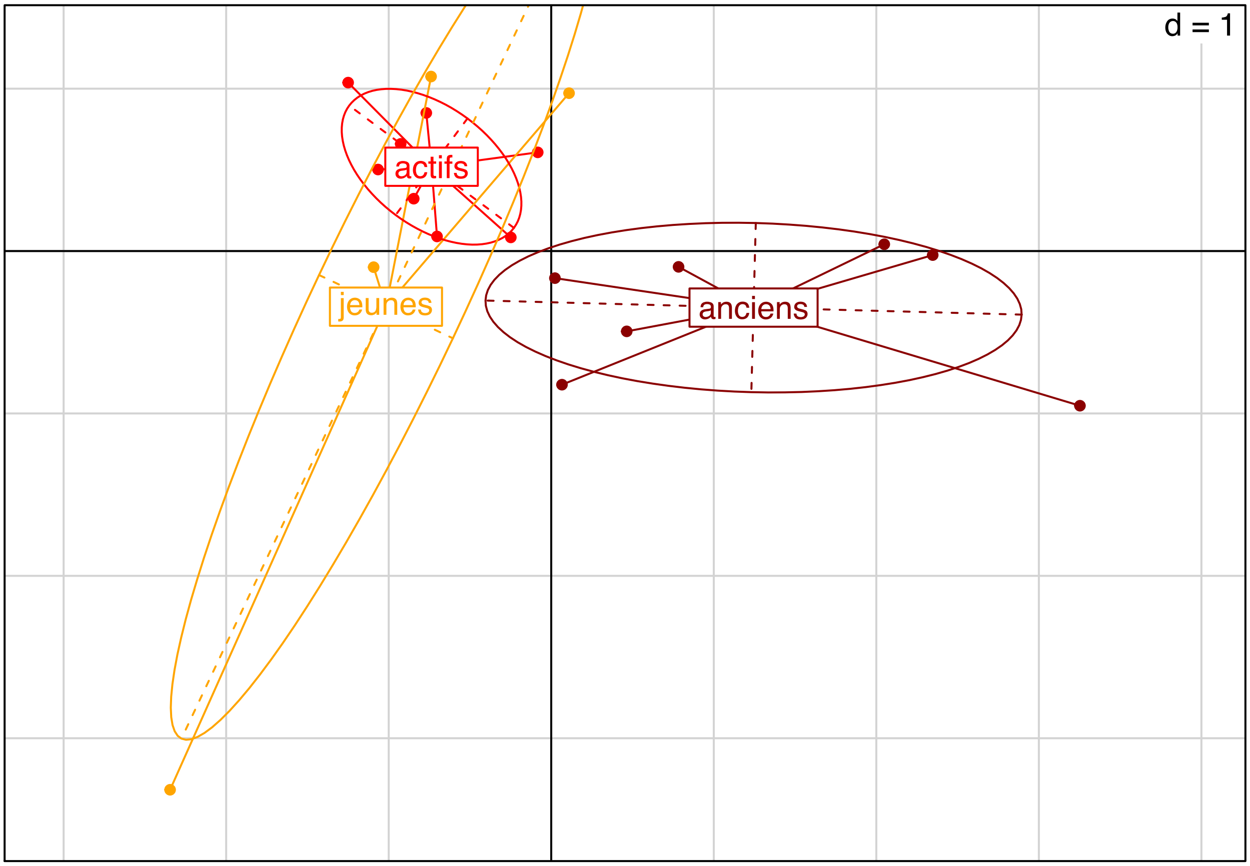


Interprétation

- Ex1 : On ne peut pas dire grand choses pour les départements proche du centre de gravité. Les individus Seine st denis, Var de marne et Haut de saine ont des profils semblables mais ne sont pas suffisamment loin du centre de gravité sur laxe 2 pour dire grand chose. Par contre, ceux qui créent la variabilité sont bouche du rhone et paris. Paris se distingue par un grand nombre d'accidents: petit corporel et blessé mais pas de tué Bouche du rhone : ou l'herault => il y a des tués sur les routes
- Ex2: L'axe 1 contient 58% de l'inertie totale et l'axe 2 30%. Donc l'individu 3 représente bien les 30% de variabilité. On a 88% de la variabilité expliquée par les deux axes. L'individu 3 crée la variabilité.

Rajout des groupes

```
#gcol : couleur assigné a chaque groupe, ici 3 groupes distinct
gcol=c("red1","red4","orange")
s.class(dfxy=acpM$li,fac=depsante$groupe,col=gcol,xax=1,yax=2)
```



Interprétation

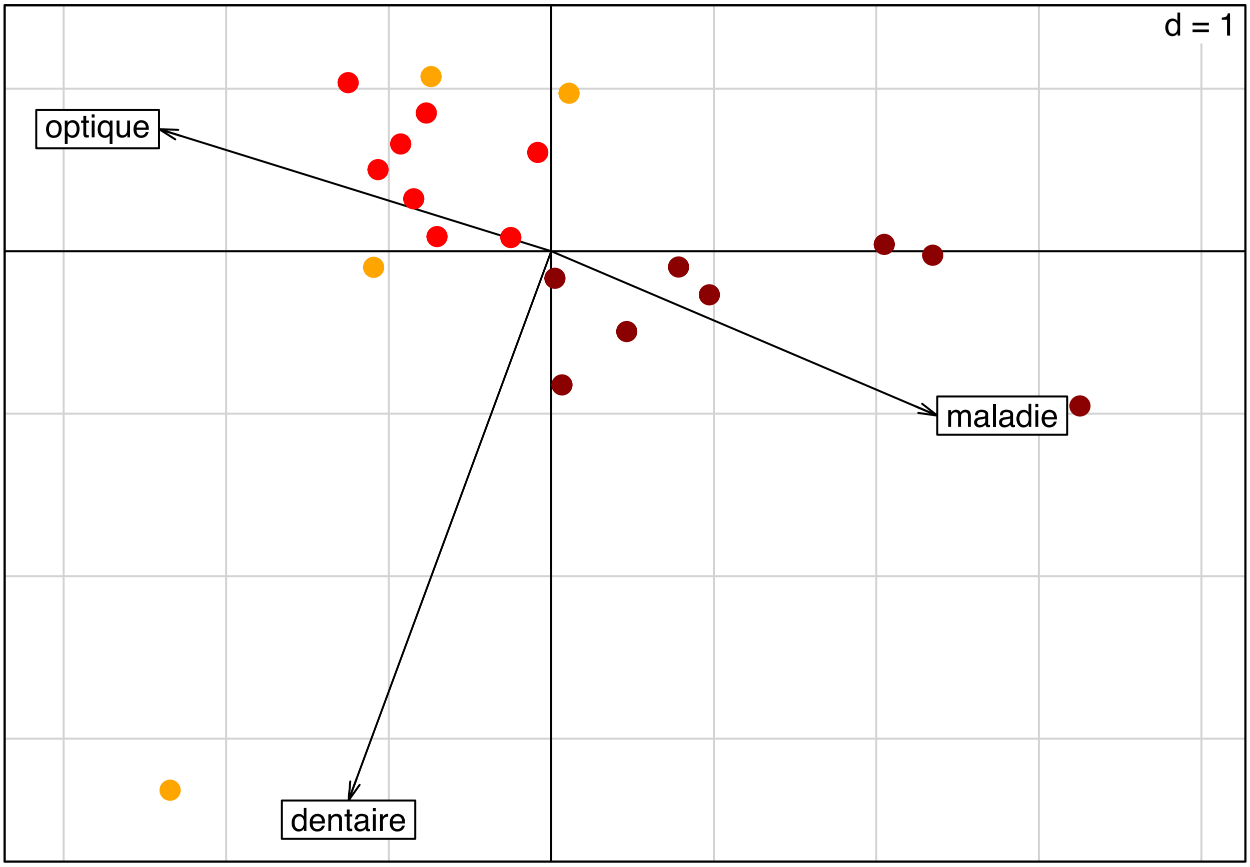
On observe unne grande variabilité sur l'axe verticale à cause des 15 ans qui perturbent avec leur pb dentaires, les actifs sont tres regroupés, ce n'est pas ceux qui ont le plus de dépenses, les anciens sur l'axe s'opposent aux jeunes et actifs.

II.5. Supperposition des deux graphiques

```r
#Superposition avec les groupes
scatter(acpM,clab.row=0,posieig="none")
```

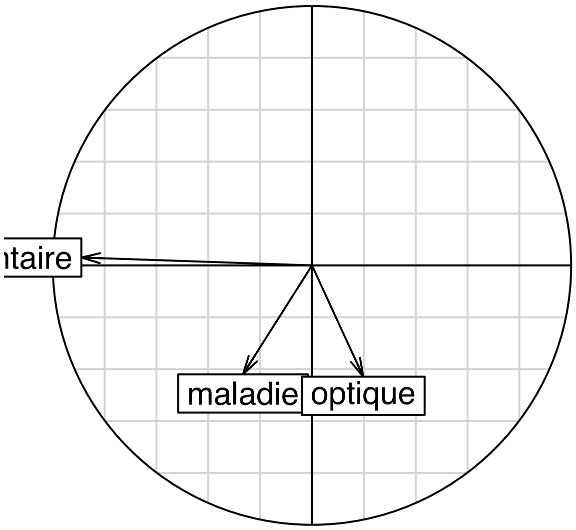
```
## NULL
```

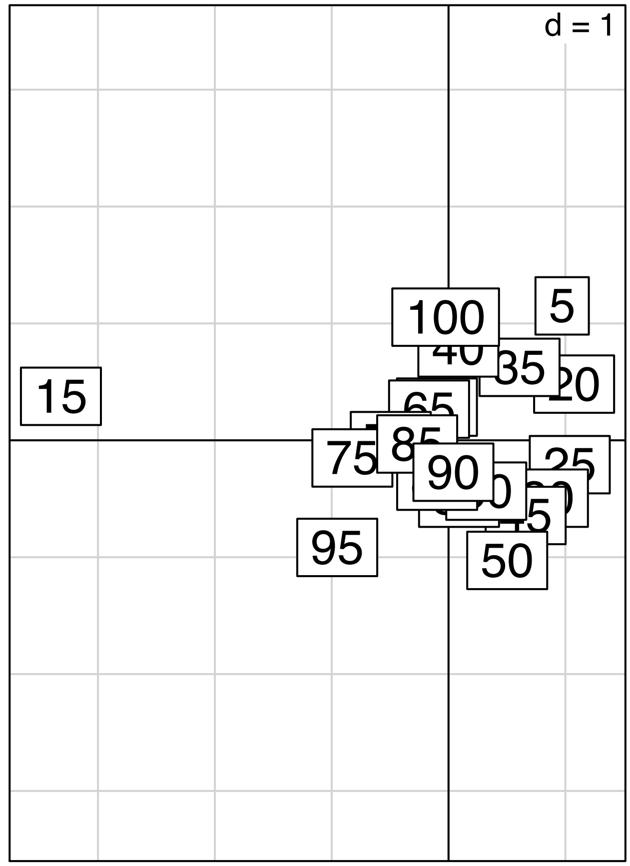
```r
s.class(acpM$li,fac=groupe,col=gcol,add.plot=TRUE,cstar=0,clabel=0,cellipse=0,cpoint=2)
```



III. Interprétation des autres axes(cas de l'ex 2)

```
par(mfrow=c(1,2))
      s.corcircle(acpM$co,xax=2,yax=3)
      s.label(acpM$li,xax=2,yax=3,label=as.character((depsante$age)),clabel=1.5)
```





L'inertie conservée sur laxe 3 étant tres faible, les correelation sont donc plus petites : pas bcp de variabilité Les groupes ayant des dépenses à la fois en maladie et en optique vont etre rarissime mais ils sont la quand même.
A 50 ans, on a beaucoup de dépenses d'optiques
A 95 ans non mais des dépenses liées à la maladie. On ne donne des explications que pour ceux qui se démarquent (loin du centre)

IV. Rajout d'un critère (ex : population sur l'ex1)

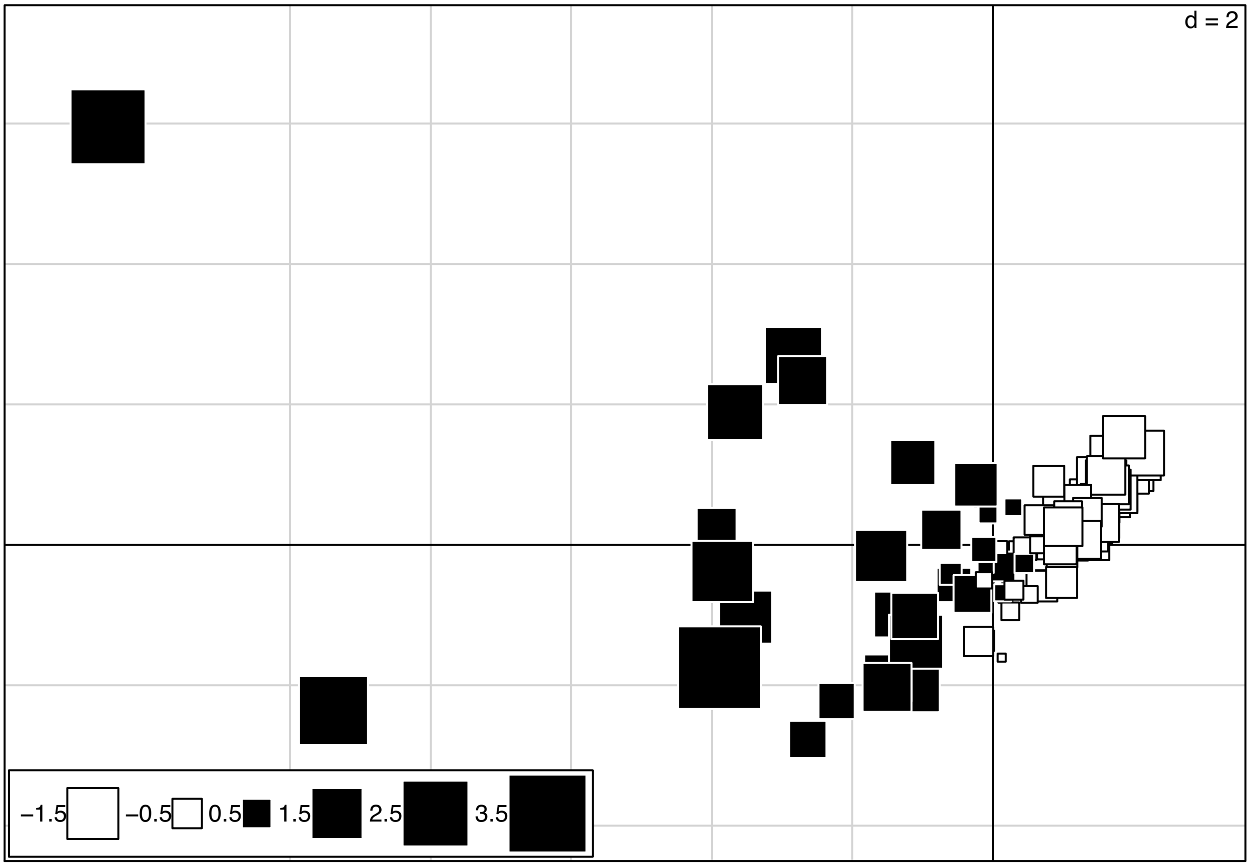
```
cor(SR0910$population,acp$li)
```

```
##            Axis1      Axis2
## [1,] -0.8686084 -0.1574972
```

Interprétation : La corrélation est très forte entre l'axe1 et la population. Plus les départements sont peuplés, plus on dénombre un grand nombre d'accidents. Il se pose donc la question de la pondération des individus en fonction de leur population pour une meilleur interprétation.

Remarque : Le signe négatif vient du faite que les flèches allaient vers la gauche.

```
#on centre et réduit les variables
popu=scale(SR0910$population,center=TRUE,scale=TRUE)
#On représente les données en fonction de la population
s.value(acp$li[,1:2],popu)
```



Conclusion

Ci dessus, j'illustre le probleme de la population sur les variables. Il faudrait diviser toutes les valeurs par la population pour rendre homogènes les plus et moins peuplés. (en blanc : en dessous de la moyenne, en noir: départements plus peuplés que la moyenne)

ACP sur R

L' analyse en composantes principales (ACP) , ou principal component analysis (PCA) en anglais, permet d'analyser et de visualiser un jeu de données contenant des individus décrits par plusieurs variables quantitatives.

C'est une méthode statistique qui permet d'explorer des données dites multivariées (données avec plusieurs variables). Chaque variable pourrait être considérée comme une dimension différente. Si vous avez plus de 3 variables dans votre jeu de données, il pourrait être très difficile de visualiser les données dans un "hyper-espace" multidimensionnelle.

L'ACP est utilisée pour extraire et visualiser les informations importantes contenues dans une table de données multivariées. L'ACP synthétise cette information en seulement quelques nouvelles variables appelées composantes principales. Ces nouvelles variables correspondent à une combinaison linéaire des variables originels. Le nombre de composantes principales est inférieur ou égal au nombre de variables d'origine.

L'information contenue dans un jeu de données correspond à la variance ou l'inertie totale qu'il contient. L'objectif de l'ACP est d'identifier les directions (i.e., axes principaux ou composantes principales) le long desquelles la variation des données est maximale.

En d'autres termes, l'ACP réduit les dimensions d'une donnée multivariée à deux ou trois composantes principales, qui peuvent être visualisées graphiquement, en perdant le moins possible d'information.

Notions de base

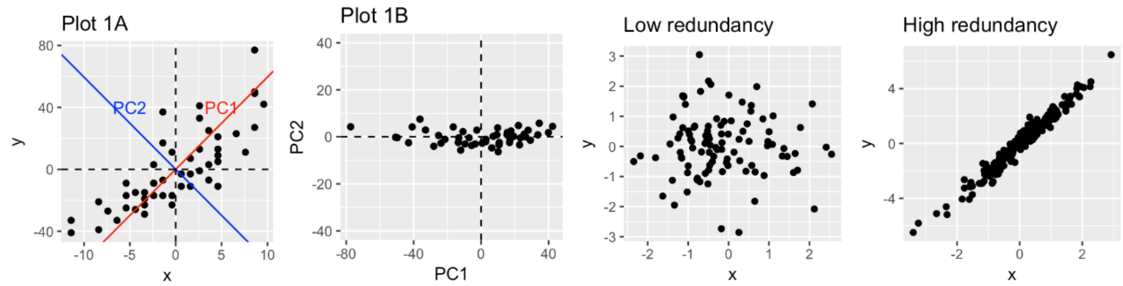
Comprendre les détails de l'ACP nécessite une connaissance de l'algèbre linéaire. Ici, nous n'expliquerons que les bases avec une représentation graphique simple des données.

Dans le Plot 1A ci-dessous, les données sont représentées dans le système de coordonnées X-Y. La réduction de la dimension est obtenue en identifiant les directions principales, appelées composantes principales, dans lesquelles les données varient.

L'ACP suppose que les directions avec les plus grandes variances sont les plus "importantes" (i.e., principales).

Dans la figure ci-dessous, l'axe PC1 est le premier axe principal le long duquel les échantillons présentent la plus grande variation. L'axe PC2 est la seconde direction la plus importante et orthogonal à l'axe PC1.

Les dimensions de notre jeu de données peuvent être réduites à une seule dimension en projetant chaque échantillon sur le premier axe principal (Plot 1B)



Techniquement parlant, la quantité de *variance expliquée* par chaque composante principale est mesurée par ce que l'on appelle *valeur propre*.

Notez que l'ACP est particulièrement utile lorsque les variables, dans le jeu de données, sont fortement corrélées. La corrélation indique qu'il existe une redondance dans les données. En raison de cette redondance, l'ACP peut être utilisée pour réduire les variables d'origine en un nombre plus petit de nouvelles variables (= **composantes principales**), ces dernières expliquant la plus grande partie de la variance contenue dans les variables d'origine.

En résumé, l'analyse en composantes principales permet:

- d'identifier des "profils cachés" dans un jeu de données,
- de réduire les dimensions des données en enlevant la redondance des données,
- d'identifier les variables corrélées

<u>Standardisation des données</u>

Dans l'analyse en composantes principales, les variables sont souvent normalisées. Ceci est particulièrement recommandé lorsque les variables sont mesurées dans différentes unités (par exemple: kilogrammes, kilomètres, centimètres, …); sinon, le résultat de l'ACP obtenue sera fortement affecté.

L'objectif est de rendre les variables comparables. Généralement, les variables sont normalisées de manière à ce qu'elles aient au final : <u>un écart type égal à 1</u> et <u>une moyenne égale à 0</u>.

Techniquement, l'approche consiste à transformer les données en soustrayant à chaque valeur une valeur de référence (la moyenne de la variable) et en la divisant par l'écart type. A l'issue de cette transformation les données obtenues sont dites données centrées-réduites. L'ACP appliquée à ces données transformées est appelée ACP normée.

Lors de la normalisation des variables, les données peuvent être transformées comme suit : **(xi−mean(x)) / sd(x)**

Où mean(x) est la moyenne des valeurs de x, et sd(x) est l'écart type (SD).

<u>Calcul</u>

On charge les Packages R comme suit :

```
library(ade4)

library(factoextra)
```

<u>Forme simplifier ACP :</u>

```
ACP ← dudi.pca( X,  center=TRUE, scale=TRUE, scannf=FALSE,   nf=2  )
```

Avec :

- **X** : jeu de données de type data frame. Les lignes sont des individus et les colonnes sont des variables numériques. (data frame = tableau de données)
- **center** / **scale** : Si TRUE, les données sont standardisées/normalisées avant l'analyse.
- **nf** : nombre de dimensions conservées dans les résultats finaux.
- **scannf** : Si TRUE un graphique est affiché.

<u>Visualisation et interprétation</u>

Les fonctions suivantes, de ***factoextra***, seront utilisées :

- get_eigenvalue(ACP) : Extraction des valeurs propres / variances des composantes principales
- fviz_eig(ACP) : Visualisation des valeurs propres (sreeplot)
- get_pca_ind(ACP), get_pca_var(ACP) : Extraction des résultats pour les individus et les variables, respectivement.
- fviz_pca_ind(ACP), fviz_pca_var(ACP) : visualisez les résultats des individus et des variables, respectivement.
- fviz_pca_biplot(ACP) : Création d'un biplot des individus et des variables.

Dans les sections suivantes, nous allons illustrer chacune de ces fonctions.

Les fonctions suivantes, de ***ade4***, seront utilisées :

- ACP$eig : Extraction des valeurs propres
- barplot(ACP$eig, main="histogramme des valeurs propres") : Visualisation des valeurs propres
- s.label(ACP$li, label=X$variable1, xax=1, yax=2, clabel = 1) : Représentation des individus
- scatter(ACP, posieig = "topright") : Représentation simultanée individus et variables.
- summary(ACP) : compte-rendu général

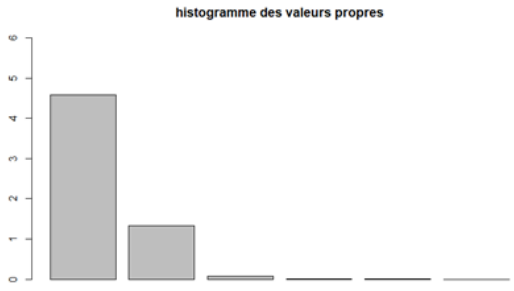
Valeurs propres / Variances

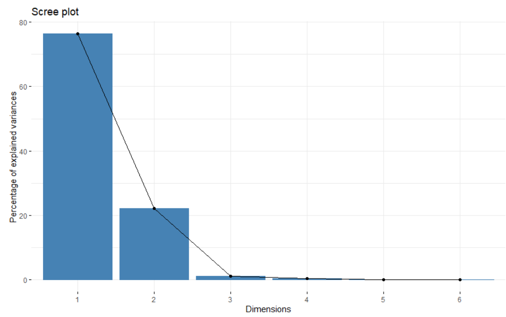
Comme décrit dans les sections précédentes, les valeurs propres (eigenvalues en anglais) mesurent la quantité de variance (i.e l'information) expliquée par chaque axe principal. Les valeurs propres sont grandes pour les premiers axes et petits pour les axes suivants. Autrement dit, les premiers axes correspondent aux directions portant la quantité maximale de variation contenue dans le jeu de données.

Afficher les valeurs propres : ACP$eig **ou** get_eigenvalue(ACP)

Représentation des valeurs propres :

Barplot : barplot(ACP$eig, main="histogramme des valeurs propres") **ou** Screeplot : fviz_eig(ACP)





Les valeurs propres peuvent être utilisées pour déterminer le nombre d'axes principaux à conserver après l'ACP :

<u>*Critère de Kaiser :*</u> on ne garde que les axes qui ont des valeurs propres supérieur à la valeur propre moyenne.

<u>*Critère du coude ou d'inflexion*</u> : sur l'histogramme des valeurs propres, on observe un décrochement (coude) suivi d'une décroissance régulière. On sélectionne les axes avant le décrochement.

<u>Un Critère de Kaiser :</u> ACP$eig>mean(ACP$eig)

OU

<u>Les valeurs propres nous renseignent sur la fraction d'inertie totale prise en compte par chaque axe :</u>

pve <- 100*ACP$eig/sum(ACP$eig)

cumsum(pve) (somme d'inertie cumulé des axes)

OU

summary(ACP) (contient inertie et inertie cumulé des axes)

<u>Extraire les résultats pour les variables :</u>

var ← get_pca_var(ACP)

var$coord : Coordonnées

var$contrib : Contributions aux axes

var$cos2 : Qualité de représentation

<u>Extraire les résultats pour les individus :</u>

ind ← get_pca_ind(ACP)

ind$coord : Coordonnées

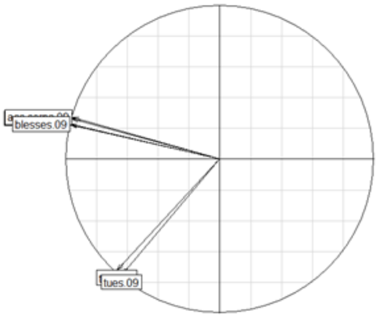
ind$contrib : Contributions aux axes

ind$cos2 : Qualité de représentation

<u>Cercle de corrélation</u>

La corrélation entre une variable et une composante principale (PC) est utilisée comme coordonnées de la variable sur la composante principale. La représentation des variables diffère de celle des observations : les observations sont représentées par leurs projections, mais les variables sont représentées par leurs corrélations.

Représentation : s.corcircle(ACP$co, xax=1, yax=2) ou fviz_pca_var(ACP, col.var = "black")



Le graphique suivant est également connu sous le nom de graphique de corrélation des variables. Il montre les relations entre toutes les variables. Il peut être interprété comme suit :

- Les variables positivement corrélées sont regroupées.
- Les variables négativement corrélées sont positionnées sur les côtés opposés de l'origine du graphique (quadrants opposés).
- La distance entre les variables et l'origine mesure la qualité de représentation des variables. Les variables qui sont loin de l'origine sont bien représentées par l'ACP.

<u>Avant l'interprétation</u>

<u>Variable</u>

<u>Qualité de représentation ou cos2 :</u> head(var$cos2)

- Un cos2 élevé indique une bonne représentation de la variable sur les axes principaux en considération. Dans ce cas, la variable est positionnée à proximité de la circonférence du cercle de corrélation.
- Un faible cos2 indique que la variable n'est pas parfaitement représentée par les axes principaux. Dans ce cas, la variable est proche du centre du cercle.
- <u>Représentation :</u> fviz_cos2(ACP, choice = "var", axes = 1) ou axes = 2 ou axes = 1 : 2

<u>Contributions des variables :</u> head(var$contrib)

Les contributions des variables dans la définition d'un axe principal donné, sont exprimées en pourcentage.

- Les variables corrélées avec PC1 (i.e. Dim.1) et PC2 (i.e. Dim.2) sont les plus importantes pour expliquer la variabilité dans le jeu de données.
- Les variables qui ne sont pas en corrélation avec un axe ou qui sont corrélées avec les derniers axes sont des variables à faible apport et peuvent être supprimées pour simplifier l'analyse globale.
- <u>Représentation :</u> fviz_contrib(ACP, choice = "var", axes = 1)

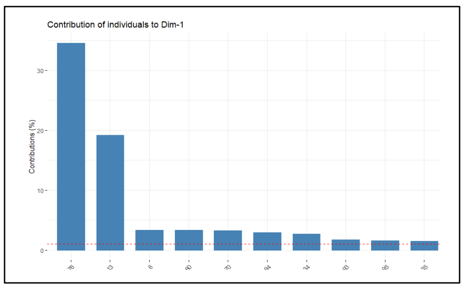
<u>Individus</u>

<u>Qualité de représentation ou cos2 :</u> head(ind$cos2)

- <u>Représentation :</u> fviz_cos2(ACP, choice = "ind", axes = 1, top = 10) ou axes = 2 ou axes = 1 : 2

<u>Contributions des individus :</u> head(ind$contrib)

- <u>Représentation :</u> fviz_contrib(ACP, choice = "ind", axes = 1, top = 10)



Interprétation

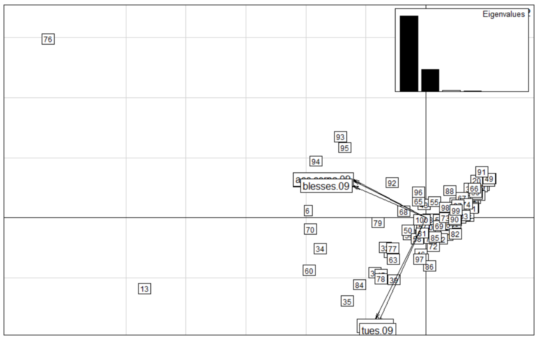
Biplot : représentation simultanée des individus et des variables

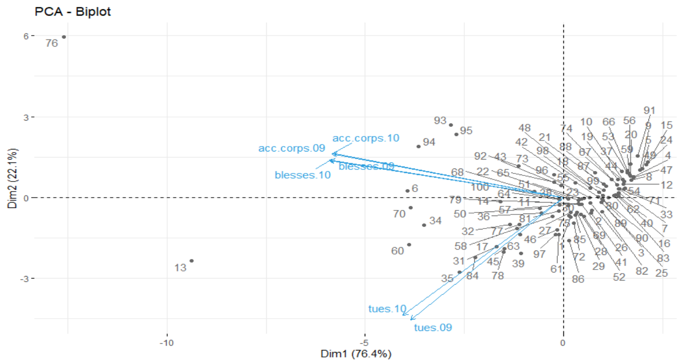
```
library(ade4)

scatter(ACP, posieig = "topright")
```

```
library(factoextra)

fviz_pca_biplot(ACP, repel = TRUE, col.var = "#2E9FDF", col.ind = "#696969")
```





Notez que le biplot n'est utile que s'il existe un faible nombre de variables et d'individus dans le jeu de données, sinon le graphique final serait illisible.

Notez également que les coordonnées des individus et des variables ne sont pas construites dans le même espace. Par conséquent, dans le biplot, vous devriez vous concentrer principalement sur la direction des variables mais pas sur leurs positions absolues sur le graphique.

Globalement, un biplot peut être interprété comme suit :

- un individu qui se trouve du <u>même côté d'une variable</u> donnée a une <u>valeur élevée</u> pour cette variable;
- un individu qui se trouve sur le <u>côté opposé d'une variable</u> donnée a une <u>faible valeur</u> pour cette variable.

Filtrer des résultats

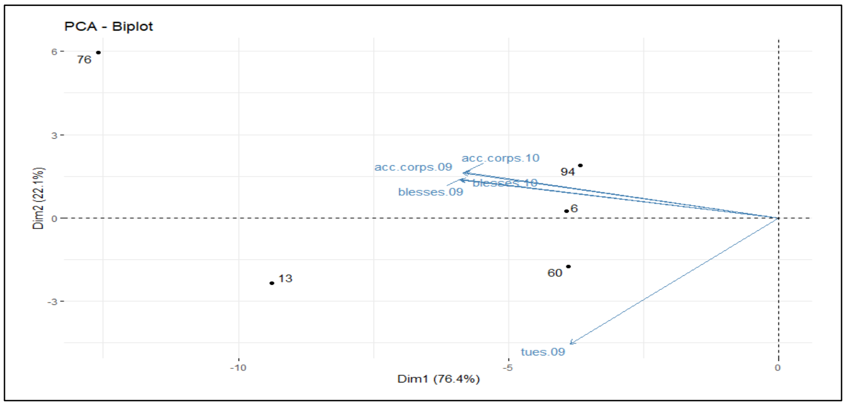
Si vous avez un nombre élevé d'individus / variables, il est possible de visualiser seulement certains d'entre eux en utilisant les arguments select.ind et select.var.

- Visualiser les variables avec cos2> = 0.6 : fviz_pca_var (ACP, select.var = list(cos2 = 0.6))
- Top 5 variables actives avec le cos2 le plus élevé : fviz_pca_var (ACP, select.var = list(cos2 = 5))
- Sélectionnez par noms : name <- list (name = c ("var1", "var2", "var3"))

  fviz_pca_var (ACP, select.var = name)

- Top 5 des individus/variables les plus contributifs :

  fviz_pca_biplot (ACP, select.ind = list (contrib = 5), select.var = list (contrib = 5), repel=TRUE)



ACP $_{\text{normée}}$ ou ACP centrée

Quel que soit le logiciel, les variables sont par défaut centrées (on retire la moyenne de la variable pour chaque observation). Généralement, elles sont aussi réduites (division par l'écart-type). Alors que le centrage est neutre pour l'analyse, la réduction ne l'est pas. Son avantage est d'assurer une comparaison entre variables mesurées dans des unités très différentes. Mais cette opération donne une importance identique à chaque variable. Selon la problématique de l'analyse, ce peut être un bien ou un mal. En effet, si toutes les variables sont mesurées dans la même unité, il peut être préférable de conserver leurs variances respectives. On parle alors d'ACP centrée.

Inertie

L'inertie est la somme pondérée des carrés des distances des individus au centre de gravité. Elle mesure la dispersion totale du nuage de points. L'inertie est donc aussi égale à la somme des variances des variables étudiées.

$I_{totale} = \sum_{i=0}^{n} var(Xi) \; = \sum_{j=0}^{J} vp_j$ avec X_i = variable i vp_j = valeur propre de l'axe i

Remarque : Dans le cas où les variables sont **centrées réduites**, la variance de chaque variable vaut 1. L'inertie totale est alors égale à n (nombre de variables). $\boxed{I_{totale} = \text{nombre de variable}}$

Critère de Kaiser

- Pour les variables centrées réduites : Le critère de Kaiser consiste à retenir les seuls axes dont la part d'inertie expliquée est supérieur à 1 i.e. supérieur à la part d'inertie moyenne.
- Pour les variables non-réduites : Le critère de Kaiser consiste à retenir les seuls axes dont la valeur propre est supérieure à la valeur propre moyenne.

$$vp_i > \frac{\sum_{k=1}^{n} vp_k}{n}$$

- Version « adoucie » :
  - on ne retient que les axes dont la valeur propre est supérieure à 0,7(valeur propre moyenne).

$$vp_i > 0{,}7 * \frac{\sum_{k=1}^{n} vp_k}{n}$$

  - Si variables centrées réduites, on retient les axes si : $vp_i > 0{,}7$

NB : Variance → Variable Valeur Propre → axe (ou composante ou dimension)

Cercle des corrélations : Que dire de l'angle entre les variables initiales ?

Dans R^n , le cosinus de l'angle entre 2 flèches correspond au coefficient de corrélation entre les 2 variables correspondantes. Si 2 flèches sont très proches, l'angle qui les sépare est proche de 0°, et cos(0)=1 , donc leur coefficient de corrélation est proche de 1 : ces 2 variables sont très corrélées. De même, si 2 flèches sont orthogonales (perpendiculaires), alors l'angle qui les sépare est de 90°. Le cosinus de cet angle valant 0, deux flèches orthogonales correspondent à des variables non corrélées (indépendantes).

ATTENTION : On ne peut déduire cela QUE si leurs 2 extrémités sont proches du cercle.

<u>L'effet taille</u> se rencontre assez fréquemment quand on réalise une ACP, il se manifeste par :

- Toutes les variables sont de **même signe sur le premier axe** factoriel donc elles sont toutes **corrélées positivement** entre elles.
- Dans ce cas, l'axe 1 constitue un **gradient** : il permet de classer les individus du plus "petit" au plus "grand", sur toutes les variables simultanément.

<u>Le choix de l'ACP</u>

Les données sont dispersées, on opte pour une ACP centrée réduite.

La réduction permet d'assurer une comparaison entre les variables en donnant une importance identique à chaque variable. On préfère alors une ACP normée. Dans la mesure où celles-ci sont centrées et réduites, leurs poids sont comparables.

<u>NB :</u> points en dehors de la boite à moustache représente les individus extrême.

TD 1103 ACP (1)

→ **NUAGE DE POINTS EN 3D** library (rgl) plot3d()

plot3d (dataframe, type = "s", col = vecteur_couleur)

[vecteur] créé auparavant

- colMeans(dataframe) : permet de calculer les moyennes des colonnes d'un dataframe
- si on veut faire apparaître l'ellipsoïde dans le NdP d'une ACP :
  plot3d (ellipse3d (cor(dataframe)), col = "grey", alpha = 0,5, add = TRUE)

> Centrage et réduction

Quand trop de différence dans le rôle des données : on opte pour un centrage et une réduction pour donner la même importance à chaque variable

→ **Effectuer centrage et réduction** : library (ade4) scalewt()

v1 ← scalewt (dataframe, center = TRUE, scale = TRUE)

↳ renvoie une matrice

alors on ajoute v2 ← as.data.frame(v1)

- fonction pour arrondir : round (dataframe, a)
  [précision de l'arrondi

- pour avoir un graph (2D ou 3D) pertinent quand valeur des axes limitées :
  On pose un vecteur lims ← c(min(vecteur / dataframe), max(vecteur / dataframe))
  et dans la fonction plot / plot3d (..., xlim = lims, ylim = lims, zlim = lims)

Si on réalise une ACP normée : les données sont centrées et réduites

- COURS : Forme générale du nuage : dragée (ou ellipsoïde)
  ↳ définie par ses 3 axes :
  1) longueur de la dragée : plus grand diamètre
  2) largeur de la dragée : diamètre moyen
  3) épaisseur de la dragée : plus petit diamètre

> ACP centrée-réduite dans ade4 : library(ade4) dudi.pca()

dudi.pca(dataframe, center = TRUE, scale = TRUE), $\underbrace{\text{scannf = FALSE, nf = 3}}_{\text{permet de conserver automatiquement 3 axes}}$)

Q ? : Select the number of axes ?

Objet renvoyé

- dataframe acp$tab : contient les données du tableau initial après centrage et réduction
- acp$cw : poids des colonnes : par défaut chaque variable a un poids de 1 → canonique
- acp$lw : poids des lignes : par défaut chaque individu a un poids de $\frac{1}{n}$ → uniforme
- acp$eig : valeurs propres (eigen values) de la plus petite des matrices à diagonaliser

↳ nous renseigne sur la fraction de l'inertie totale prise en charge par chaque axe

summary (acp) : Total inertia :

Eigenvalues :

Ax1 Ax2 A3

Projected inertia (%)

Cumulative projected inertia (%)

- acp$rank : donne le rang de la matrice diagonalisée : ici le nombre de composantes principales
- acp$nf : nombre de facteurs conservés dans l'analyse
- acp$c1 : coordonnées des variables (colonnes) : norme unité
- acp$l1 : coordonnées des individus (lignes) : norme unité
- acp$co : coordonnées des variables (colonnes) : normés à la $\sqrt{\text{valeur propre correspondante}}$
- acp$li : coordonnées des individus (lignes) : normés à la $\sqrt{\text{valeur propre correspondante}}$

↳ lien entre acp$c1 et acp$co

acp$c1

	CS1	CS2	CS3
V1	a	⋮	⋮
V2	b	⋮	⋮
V3	c	⋮	⋮

acp$co

	Comp 1	Comp 2	Comp3
V1	α	⋮	⋮
V2	β	⋮	⋮
V3	γ	⋮	⋮

$\alpha = a \times \sqrt{\text{valeur propre de axe 1}}$

Code R

acp$co $Comp1
== acp$c1 $CS1 *
sqrt(acp$eig[1])

Pour tous les axes

$\underbrace{\text{t(t(acp\$c1) * sqrt(acp\$eig)}}$

↗ donne acp$co

↳ lien entre acp$l1 et acp$li : idem

acp$l1 $RS1 * sqrt(acp$eig[1])

↳ donne 1ère colonne de acp$li

et t(t(acp$l1) * sqrt(acp$eig))

↳ donne acp$li

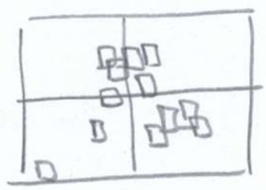
- acp$call : trace des calculs lors de l'appel de la fonction dudi.pca()
- acp$cent : donne les moyennes des variables analysées
- acp$norm : donne les écarts-types (sur $\sqrt{n}$) des variables analysées : $\frac{X-m}{\sqrt{n}}$

$(x\sqrt{n} \quad)+m$

> Représentation graphiques dans ade4

→ REPRESENTATION DES INDIVIDUS : slabel()

↳ sur les différents plans factoriels



slabel(acp$li, xax=a, yax=b, clabel = 1,5)

↳ (a,b) ↵

↳ plan factoriel sur lequel on veut projeter

⊕ rajouter en information supplémentaire une variable qualitative définissant des groupes d'individus : s.class()

s.class(dfxy = acp$li, fac = vecteur-groupes, col = vecteur-couleur, xax = a, yax = b)

→ REPRESENTATION DES VARIABLES : s.corcircle()

↳ cercle des corrélations

s.corcircle(acp$co, xax = 1, yax = 2) ? ?

→ REPRESENTATION SIMULTANEE INDIVIDUS/VARIABLES : scatter()

scatter(acp, posieig = { "bottomright" / "none")

> Changement de pondération

ACP classique : pondération associée aux individus est uniforme

acp$lw : tous les individus ont la même pondération (TD : 0,05)

poidsU ← acp$lw # pondération uniforme

poidsD ← depsante$effectif # on prend le vecteur qui contient les effectifs des individus (ici par âge moyen)

poidsD ← poidsD/sum(poidsD)

round(poidsD, 3)

↳ nouvelles pondérations