TP2

Taniel Remi

25 mars 2019

Analyse en composantes principales

QUESTION 1

On commence par charger les données stockées dans le fichier ocde.txt et on les stocke dans la variable ocde :

```
ocde <- read.table("ocde.txt", header=TRUE, sep = " ")
```

Analyse descriptive

On utilise la commande dim() pour obtenir les dimensions (nombre d'éléments et nombre de variables) de notre dataframe ocde:

```
dim(ocde)
```

```
## [1] 17 13
```

Notre dataframe contient donc 17 éléments pour 13 variables.

Pour analyser nos données, on se décide d'afficher les premières lignes de celle-ci, pour cela on utilise la commande head(), Voici donc les premieres lignes de notre dataframe :

```
head(ocde)
```

```
##   PAYS YEAR NATA CHOM APRI ASEC  PIB FBCF INFL RECC MINF PROT NRJ
## 1   AL 1975   97   41   73  460 6870  211   60  409  197   58 394
## 2   AU 1975  123   17  125  409 4990  267   78  391  205   56 307
## 3   BE 1975  121   42   36  399 6350  220   94  407  146   62 426
## 4   CA 1975  157   69   61  293 6990  242   83  374  155   65 878
## 5   DA 1975  142   49   98  315 7010  199   99  450  104   66 350
## 6   ES 1975  188   47  219  385 2870  232  139  231  121   50 173
```

On remarque qu'il n'y a qu'une seule variable qualitative, la variable 'PAYS', on décide donc de remplacer le numéro des lignes par la variable 'PAYS' afin de mieux identifier les points dans les prochaines analyses :

```
rownames(ocde) <- ocde$PAYS
ocde <- ocde[,-1]
```

Analyse descriptive univariée

Pour chaque variable, on décide d'afficher les valeurs minimales et maximales, la médianne, la moyenne ainsi que le 1er et 3e quartile, on utilise pour cela la commande summary() :

```
summary(ocde)
```

```
##       YEAR           NATA            CHOM            APRI      
##  Min.   :1975   Min.   : 97.0   Min.   :16.00   Min.   : 27.0  
##  1st Qu.:1975   1st Qu.:127.0   1st Qu.:23.00   1st Qu.: 64.0  
##  Median :1975   Median :142.0   Median :41.00   Median :102.0  
##  Mean   :1975   Mean   :147.8   Mean   :41.88   Mean   :116.8  
##  3rd Qu.:1975   3rd Qu.:157.0   3rd Qu.:49.00   3rd Qu.:149.0  
##  Max.   :1975   Max.   :216.0   Max.   :83.00   Max.   :282.0  
##       ASEC            PIB            FBCF            INFL      
##  Min.   :290.0   Min.   :1550   Min.   :136.0   Min.   : 60.0  
##  1st Qu.:336.0   1st Qu.:4070   1st Qu.:207.0   1st Qu.: 85.0  
##  Median :361.0   Median :5950   Median :220.0   Median : 96.0  
##  Mean   :364.5   Mean   :5368   Mean   :228.6   Mean   :108.3  
##  3rd Qu.:399.0   3rd Qu.:6990   3rd Qu.:242.0   3rd Qu.:138.0  
##  Max.   :460.0   Max.   :8460   Max.   :354.0   Max.   :169.0  
##       RECC            MINF            PROT            NRJ       
##  Min.   :230.0   Min.   : 83.0   Min.   :34.00   Min.   : 88.0  
##  1st Qu.:347.0   1st Qu.:105.0   1st Qu.:55.00   1st Qu.:297.0  
##  Median :395.0   Median :146.0   Median :59.00   Median :363.0  
##  Mean   :381.9   Mean   :155.7   Mean   :58.06   Mean   :401.5  
##  3rd Qu.:409.0   3rd Qu.:184.0   3rd Qu.:66.00   3rd Qu.:473.0  
##  Max.   :536.0   Max.   :379.0   Max.   :73.00   Max.   :878.0
```

On remarque que la variable 'YEAR' possède toujours la même valeur: 1975, nous pouvons donc la retirer notre dataframe ocde :

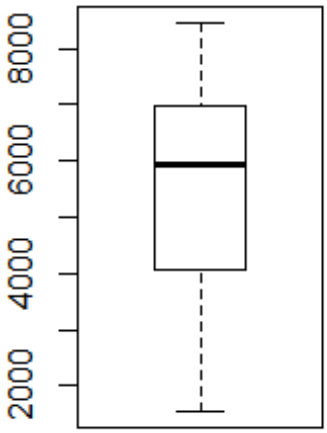
```
ocde <- ocde[,-1]
```

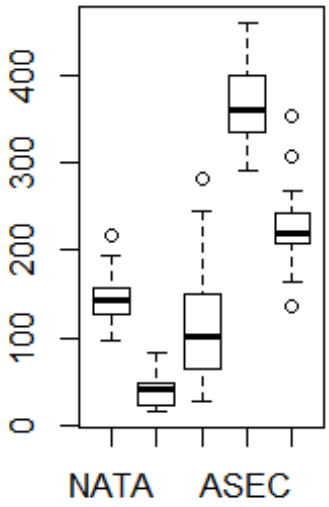
On remarque également que la variable 'PIB' possède des valeurs plus bien supérieurs aux restes des variables, cela risque donc de poser problème pour les prochains graphiques.

Analyse en boite à moustache

Pour chaque variable de notre dataframe ocde, on se décide d'afficher sa boite à moustache :

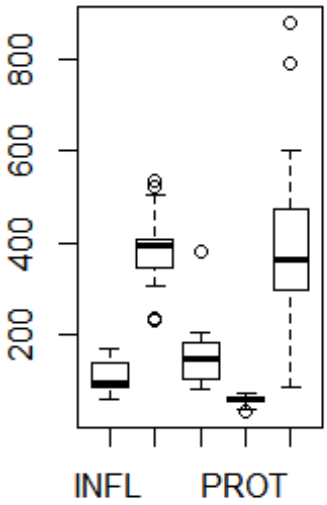
```
par(mfrow=c(1,2))
boxplot(ocde$PIB, xlab="PIB (par habitant)")
boxplot(ocde[,-c(5,7,8,9,10,11)], xlab="Autres (NATA, CHOM, APRI, ASEC,
FBCF)")
```





PIB (par habitant) utres (NATA, CHOM, APRI, ASEC,

```
boxplot(ocde[,-c(1,2,3,4,5,6)], xlab="Autres (INFL, RECC, MINF, PROT, NRJ)")
```

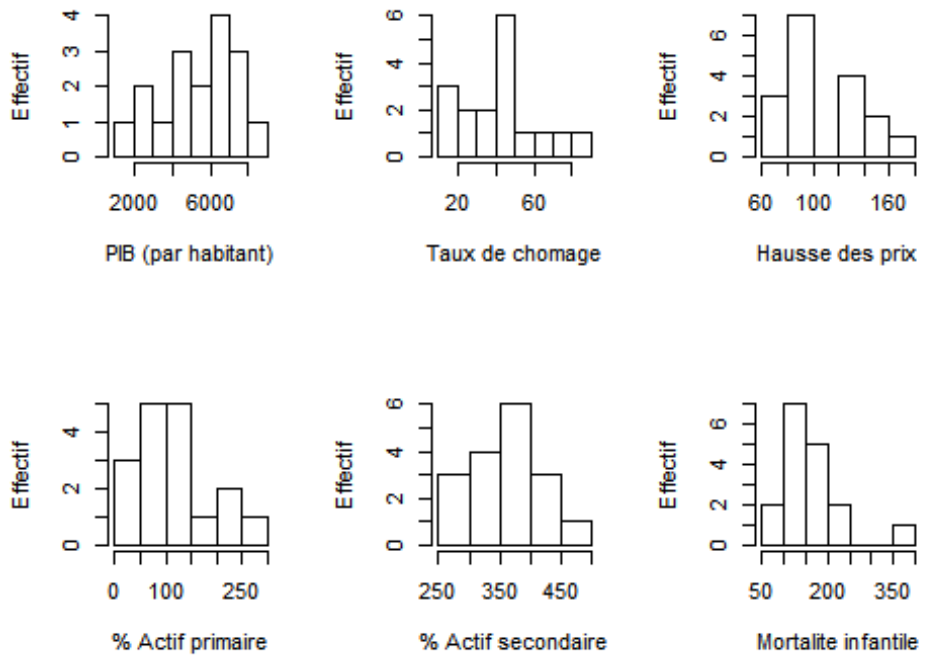


Autres (INFL, RECC, MINF, PROT,

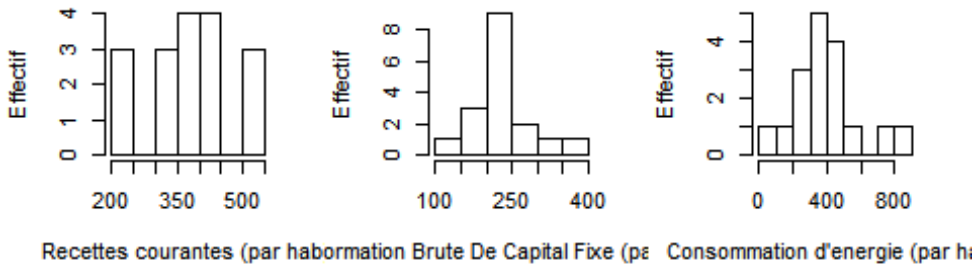
Analyse en histogramme

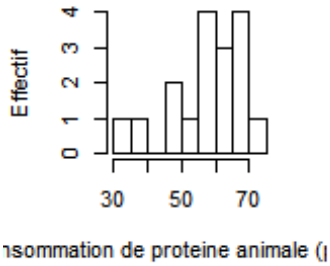
Pareil que précedemment mais cette fois-ci en histogramme :

```
par(mfrow=c(2,3))
hist(ocde$PIB, ylab="Effectif", xlab="PIB (par habitant)", main="")
hist(ocde$CHOM, ylab="Effectif", xlab="Taux de chomage", main="")
hist(ocde$INFL, ylab="Effectif", xlab="Hausse des prix", main="")
hist(ocde$APRI, ylab="Effectif", xlab="% Actif primaire", main="")
hist(ocde$ASEC, ylab="Effectif", xlab="% Actif secondaire", main="")
hist(ocde$MINF, ylab="Effectif", xlab="Mortalite infantile", main="")
```




```
hist(ocde$RECC, ylab="Effectif", xlab="Recettes courantes (par hab)",
main="")
hist(ocde$FBCF, ylab="Effectif", xlab="Formation Brute De Capital Fixe (par
hab)", main="")
hist(ocde$NRJ, ylab="Effectif", xlab="Consommation d'energie (par hab)",
main="")
hist(ocde$PROT, ylab="Effectif", xlab="Consommation de proteine animale (par
hab)", main="")
```





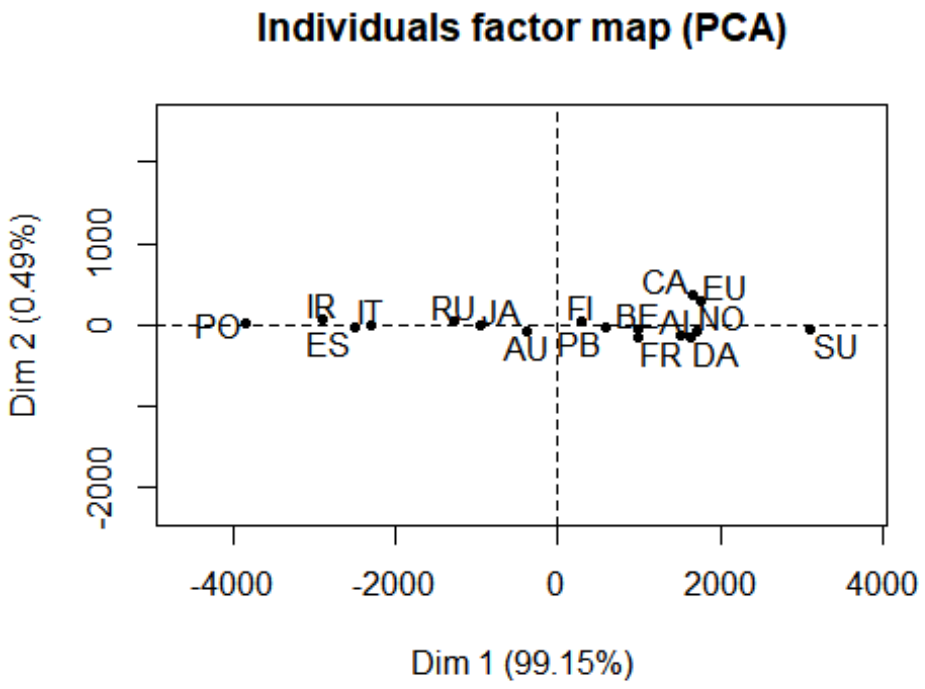
QUESTION 2

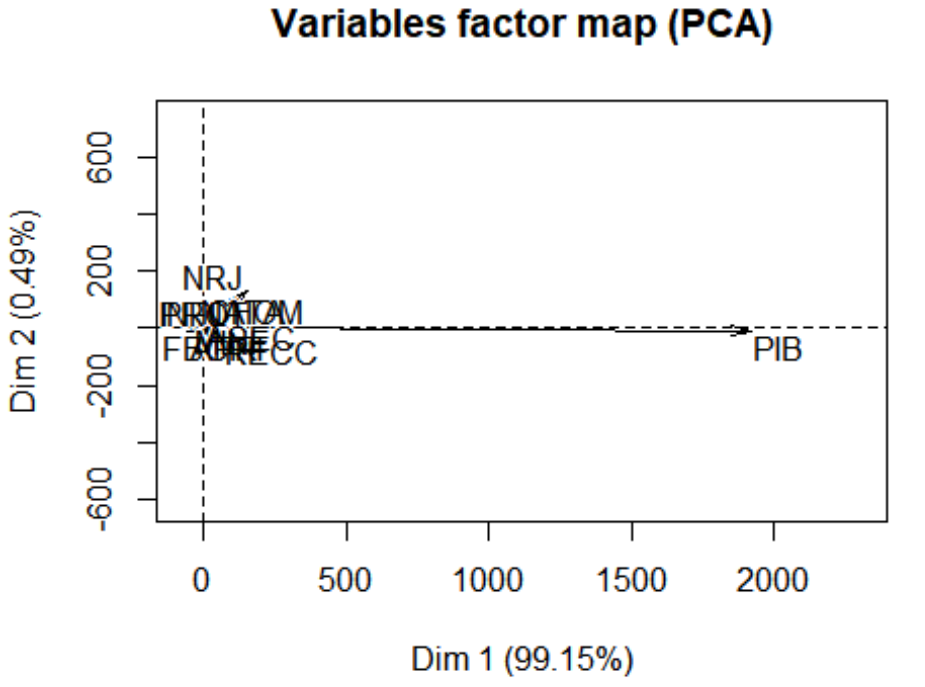
Nous devrons utiliser les packages 'FactoMineR', 'factoextra' et 'ade4', pour les utiliser, nous devons les importer avec les commandes suivantes :

```
library(FactoMineR)
library(factoextra)
library(ade4)
```

Nous allons dans un premier temps, faire une analyse en composante principales principales de la base de donnée sans réduire les donnees, pour cela on utilise la commande PCA() avec l'option 'scale.unit = FALSE' :

```
pca <- PCA(ocde, scale.unit=F)
```



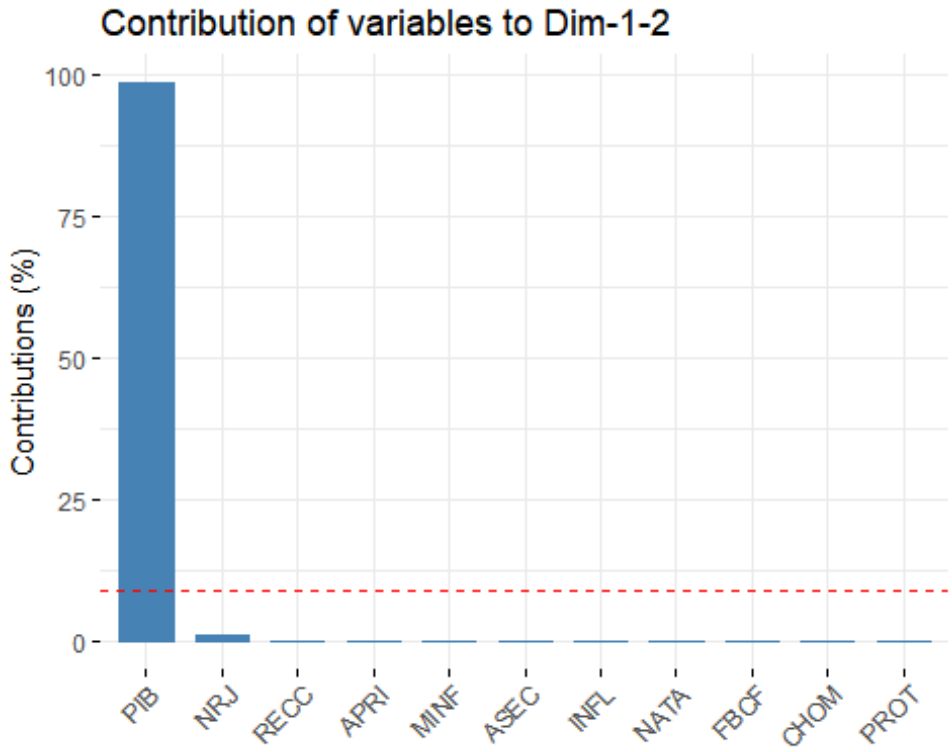


Sans réduire les données, il est difficile d'analyser les réultats etant donné que la variable 'PIB' "explose" le reste des variables, en effect, cette variable n'a pas le même ordre de grandeur que les autres variables que nous souhaitons analyser.

L'axe principal monopolise donc les données (99,15% de pourcentage d'inertie), en l'état, le graphe n'est pas exploitable, tout comme le reste de l'analyse en composantes principales…

Ceci est vérifé en affichant le graphique de contributions des variables par rapport aux 2 axes principaux :

```
fviz_contrib(pca, choice="var", axes=c(1,2))
```



Comme dit précedemment la variable 'PIB' monopolise la contribution avec quasiment 100% de contribution.

Question 4

On se propose donc de relancer une analyse en composantes principales, mais cette fois-ci en réduisant les données, on se décide également d'utiliser 3 variables qui calcule l'ACP d'une dataframe : - pca: calculée avec la fonction 'PCA()' du package 'FactoMineR' - pca_dudi: avec la fonction 'dudi.pca()' du package '' - pca_comp: avec la fonction 'prcomp()' du package 'ade4'

```
pca <- PCA(ocde, scale.unit=T, graph=F)
pca_dudi <- dudi.pca(ocde, scale=T, nf=11, scannf=F)
pca_comp <- prcomp(ocde, scale=T)
```

Fonctions d'analyse

Dans cette partie, je ne donnerai que les fonctions et leurs résultats, aucune analyse ne sera faite sur les résultats de ces fonctions, les analyses seront réalisées dans les questions suivantes.

Valeurs propres

On se décide d'afficher les valeurs propres, c'est a dire le pourcentage de variances (pourcentage d'inertie) expliqué pour chaque axe principal :

```
# PCA()
pca$eig[,1]
# prcomp()
pca_comp$sdev^2
# dudi.pca()
pca_dudi$eig
```

Pour afficher ces données avec quelque chose de plus graphique, on utilisera la fonction suivante (compatible avec les 3 fonctions calculant l'ACP) :

```
get_eigenvalue(pca_comp)
```

```
##        eigenvalue variance.percent cumulative.variance.percent
## Dim.1  5.14305497       46.7550452                    46.75505
## Dim.2  2.43153258       22.1048416                    68.85989
## Dim.3  1.40069807       12.7336189                    81.59351
## Dim.4  0.68931591        6.2665083                    87.86001
## Dim.5  0.42169549        3.8335954                    91.69361
## Dim.6  0.37359303        3.3963003                    95.08991
## Dim.7  0.22136534        2.0124121                    97.10232
## Dim.8  0.18253149        1.6593772                    98.76170
## Dim.9  0.07584782        0.6895256                    99.45122
## Dim.10 0.03740702        0.3400639                    99.79129
## Dim.11 0.02295827        0.2087116                   100.00000
```

Vecteurs propres

Pour obtenir les vecteurs propres, on utilise :

```
# PCA()
pca$svd$V[,2]
# prcomp()
pca_comp$rotation[,2]
# dudi.pca()
pca_dudi$c1[,2]
```

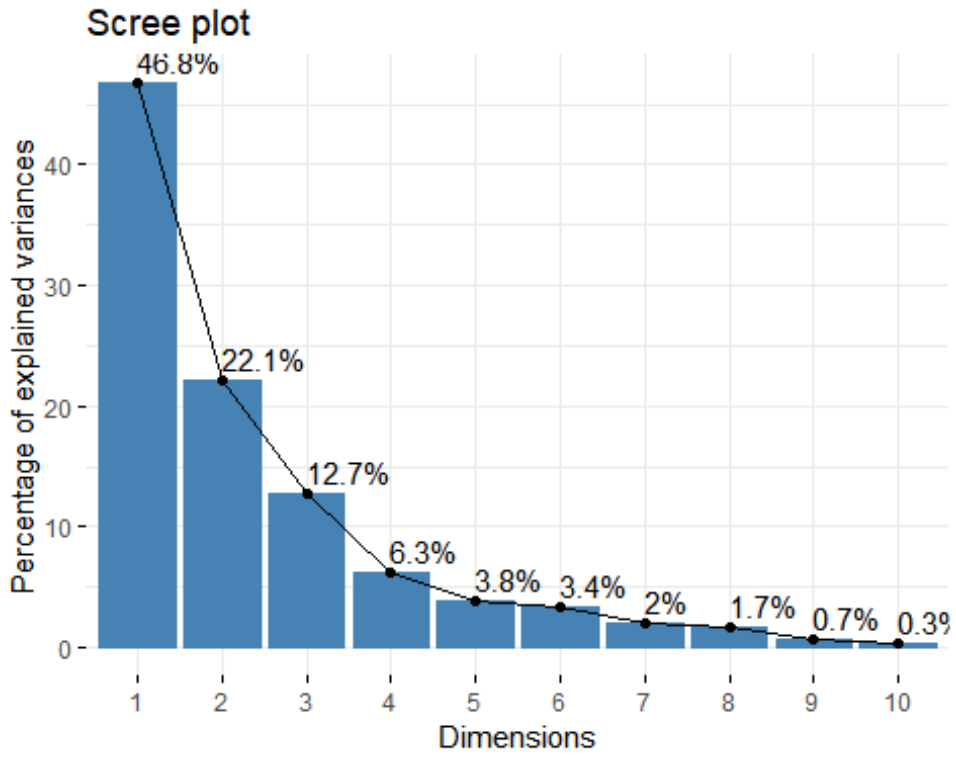
Part d'inertie exliquée par chaque axe

Pour obtenir la part d'inertie expliqué par chaque axe, on utilise :

```
# PCA()
pca$eig[,2]/sum(pca$eig[,2])
# prcomp()
pca_comp$sdev^2/sum(pca_comp$sdev^2)
# dudi.pca()
pca_dudi$eig/sum(pca_dudi$eig)
```

Pour afficher ces valeurs, on utilisera la fonction suivante (toujours compatible avec les 3 méthodes) :

```
fviz_eig(pca, addlabels=T)
```



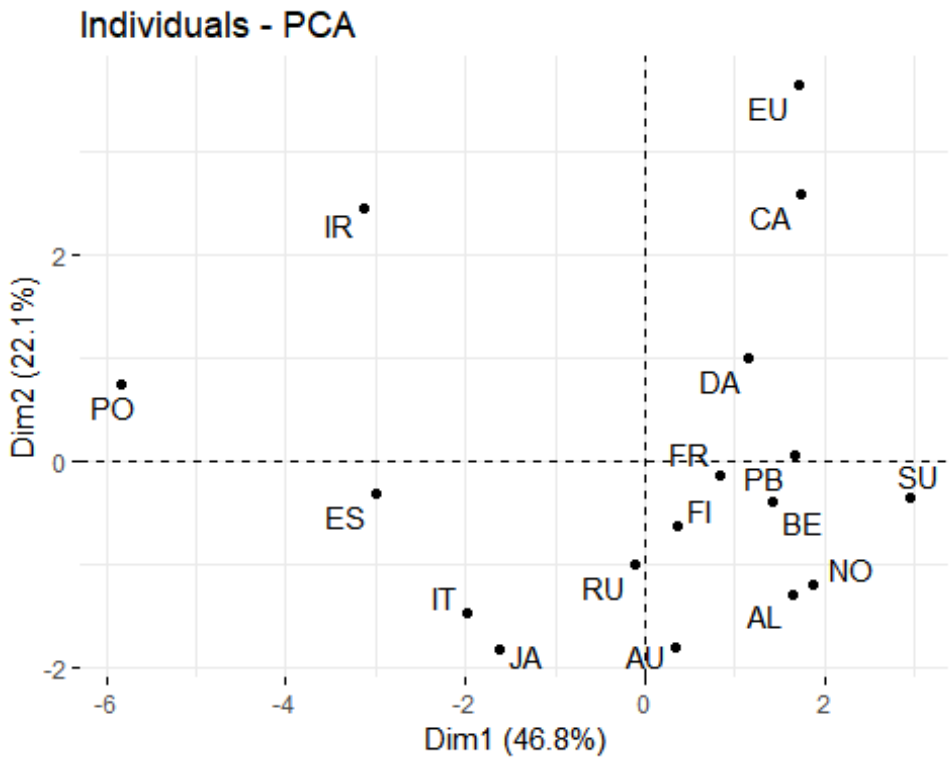
Coordonnées des individus dans la nouvelle base

Pour obtenir les coordonnées des individus dans la nouvelle base, on utilise :

```
# PCA()
pca$ind$coord
# prcomp()
pca_comp$x
# dudi.pca()
pca_dudi$li
```

Pour afficher ces coordonnées, on utilisera la fonction suivante :

```
fviz_pca_ind(pca, repel=T, axes=c(1,2))
```



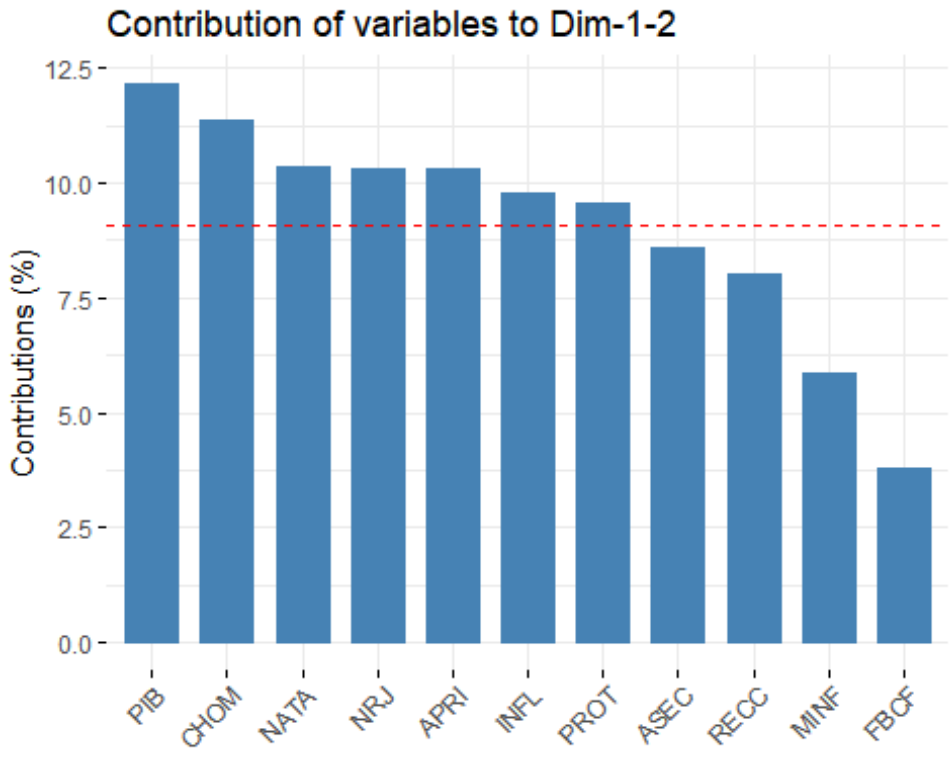
Contribution des variables sur les axes principaux

Pour obtenir les contributions des variables sur les axes principaux, on utilise :

```
# PCA()
pca$var$contrib
# prcomp()
100*pca_comp$rotation^2/apply(pca_comp$rotation^2, 2, sum)
# dudi.pca()
100*pca_dudi$co^2/apply(pca_dudi$co^2, 2, sum)
```

Pour afficher ces contributions sur les 2 axes principaux, on utilisera la fonction suivante :

```
fviz_contrib(pca, choice="var", axes=c(1,2))
```



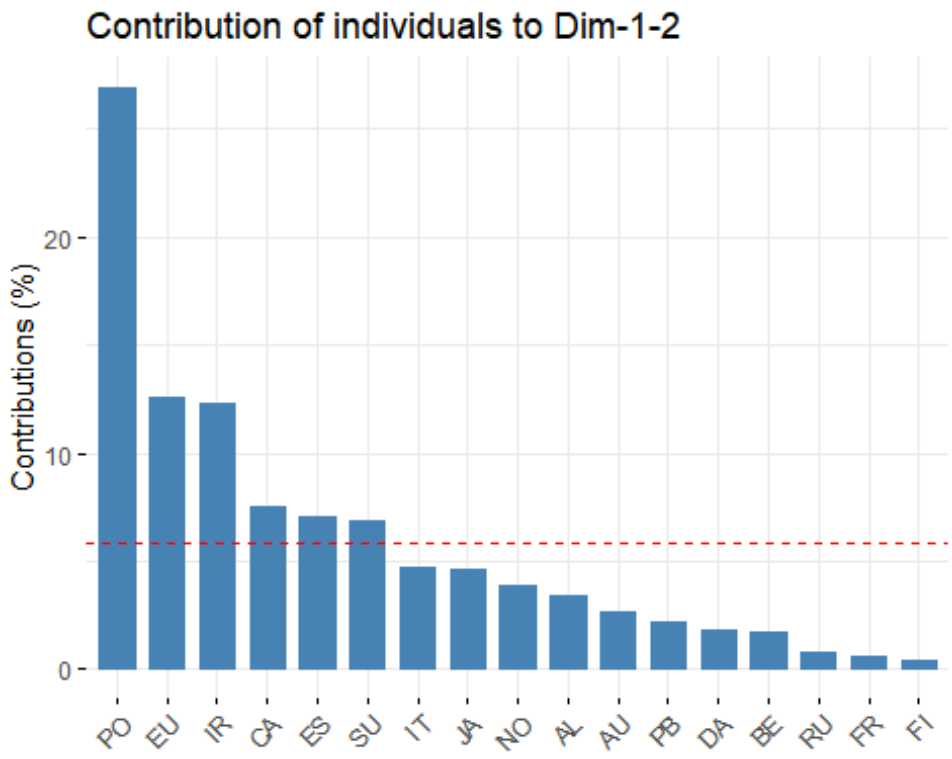
Contribution des individus sur les axes principaux

Pour obtenir les contributions des individus sur les axes principaux, on utilise :

```
# PCA()
pca$ind$contrib
# prcomp()
100*pca_comp$x^2/apply(pca_comp$x^2, 2, sum)
# dudi.pca()
100*pca_dudi$li^2/apply(pca_dudi$li^2, 2, sum)
```

Pour afficher ces contributions sur les 2 axes principaux, on utilisera la fonction suivante :

```
fviz_contrib(pca, choice="ind", axes=1:2)
```

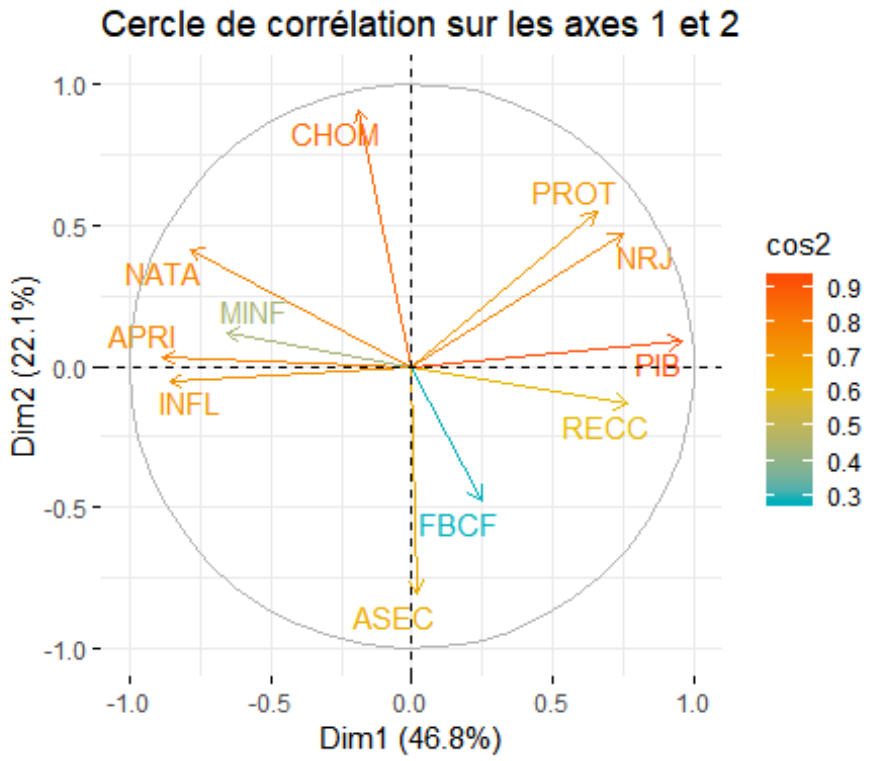


Cercle de corrélations

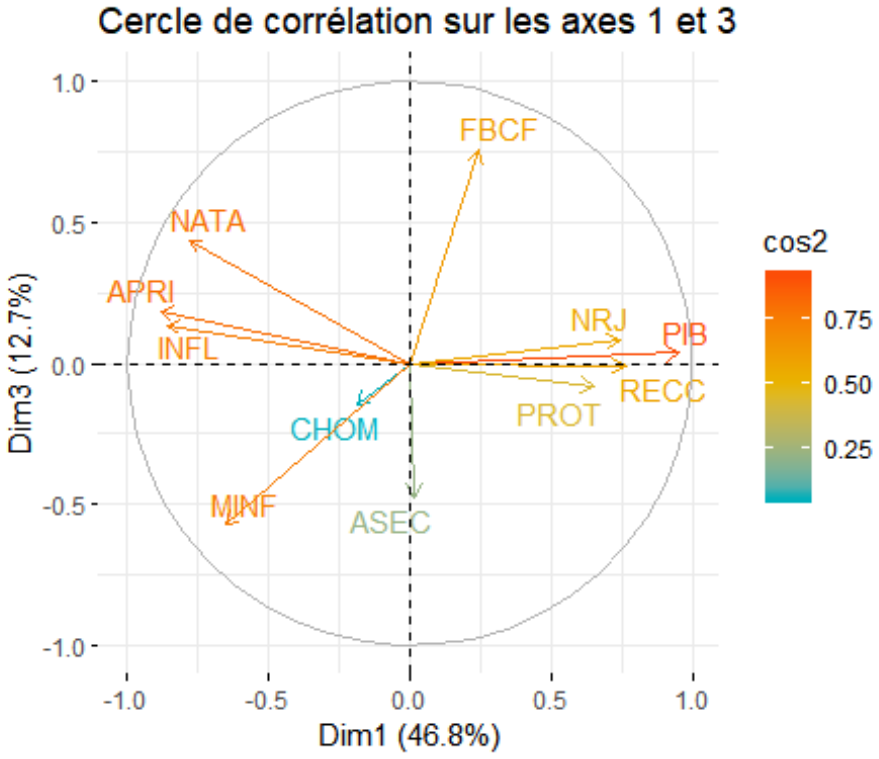
Graphique

Pour afficher le cercle de corrélations, on utilise la fonction suivante :

```
fviz_pca_var(pca, col.var="cos2", gradient.cols = c("#00AFBB", "#E7B800",
"#FC4E07"), axes=c(1,2), title="Cercle de corrélation sur les axes 1 et 2",
repel=T)
```




```
fviz_pca_var(pca, col.var="cos2", gradient.cols = c("#00AFBB", "#E7B800",
"#FC4E07"), axes=c(1,3), title="Cercle de corrélation sur les axes 1 et 3",
repel=T)
```



Analyse

Axe 1

Les variables 'PROT', 'NRJ', 'PIB' et 'RECC' sont posivitement corrélées entre elles et sont négativement corrélées aux variables 'APRI', 'INF', 'NATA' et 'MINF'.

On peut donc déduire que plus un pays est riche (fort PIB, forte recette courante par habitant), il est beaucoup moins vulnérable à une hausse des prix, la mortalité infantile et possède un faible pourcentage d'actifs dans le secteur primaire, ainsi qu'un fort taux de natalité.

Axe 2

Les variables 'ASEC' et 'FBCF' sont positivement corrélées et négativement corrélées avec la variable 'CHOM'.

Dans ce cas, on peut déduire que plus le taux de chômage dans un pays est élevés, moins il possède un fort pourcentage d'actifs dans le secteur secondaire et de formation brute de capital fixe par habitant.

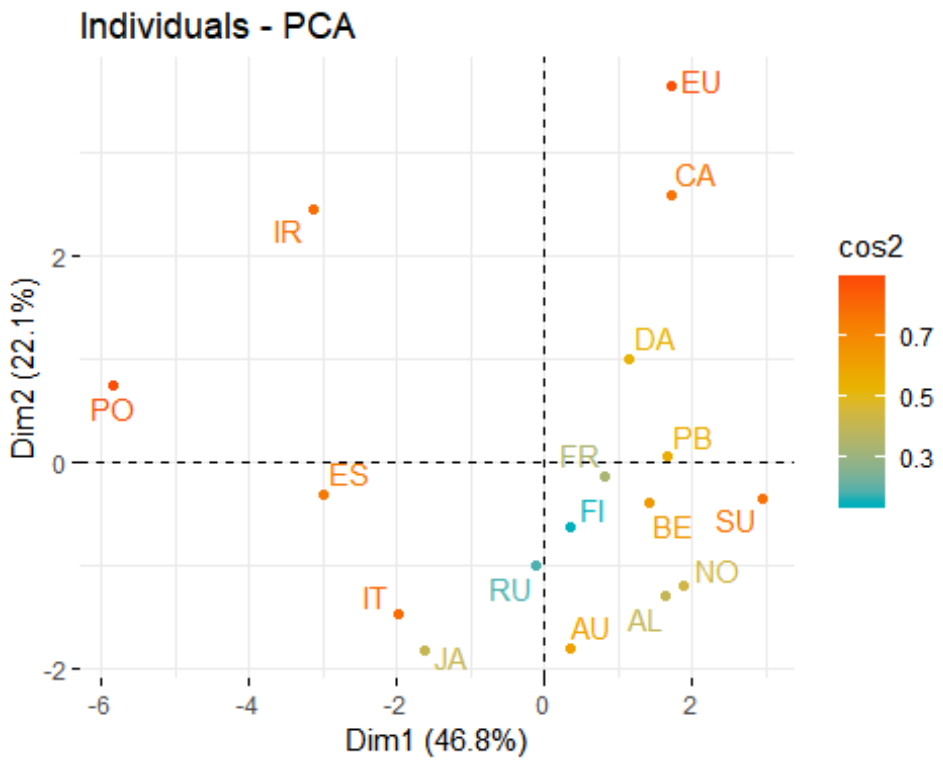
Axe 3

Nous pouvons analyser "que" 3 variables dans le cas de l'axe 3, la variable 'FBCF' est négativement corrélée avec les variables 'MINF' et 'ASEC',

Plus un pays a une mortalité infantile élevée, moins ses habitants produise du capital brute fixe. Néanmoins, l'axe 3 ne représentant que 12,7%, son analyse est beaucoup moins importante que l'axe 1 par exemple.

Graphique des individus dans la nouvelle base

```
fviz_pca_ind(pca, col.ind = "cos2", gradient.cols = c("#00AFBB", "#E7B800",
"#FC4E07"), repel=T)
```

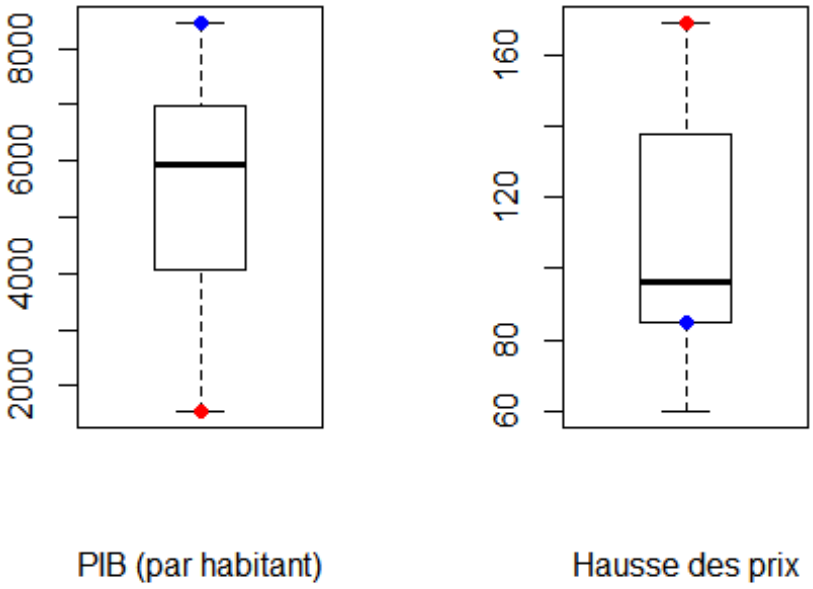


On décide garder seulement les individus les mieux representés (ayant un cos2 élevé), ici nous pouvons garder les individus EU, CA, SU, PO, E et IT.

Les individus les plus à droite possède un fort PIB, une forte consommation de de protéine animale (par habitant) par exemple, tandis que les pays les plus à gauche sur le graphe, possède un fort taux d'inflation et une forte mortalité infantile.

Nous pouvons vérifier cela avec des boxplot, par exemple décidons d'afficher la Pologne (en rouge) et la Suede (en bleu) sur le boxplot du taux d'inflation 'INFL' et celui du 'PIB', on obtient :

```
par(mfrow=c(1,2))
boxplot(ocde$PIB, xlab="PIB (par habitant)")
points(subset(ocde$PIB, rownames(ocde) == 'SU'), col='blue', pch=19)
points(subset(ocde$PIB, rownames(ocde) == 'PO'), col='red', pch=19)
boxplot(ocde$INFL, xlab="Hausse des prix")
points(subset(ocde$INFL, rownames(ocde) == 'SU'), col='blue', pch=19)
points(subset(ocde$INFL, rownames(ocde) == 'PO'), col='red', pch=19)
```



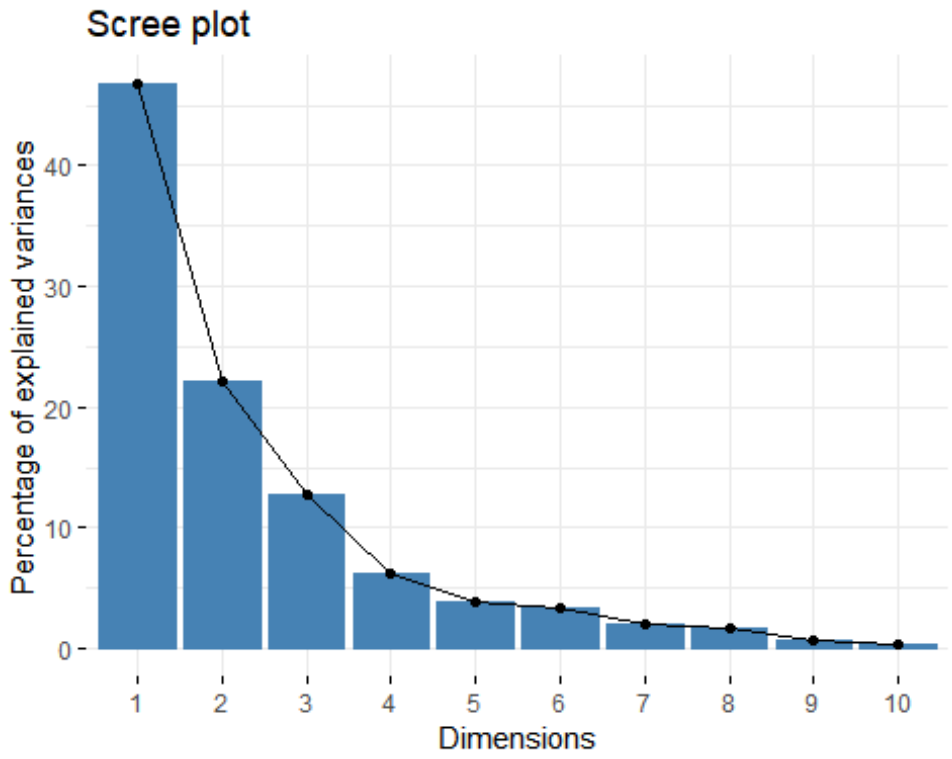
Question 5

Pour déterminer le nombre d'axes à retenir, nous pouvons utiliser 3 critères, qui sont les suivants : - Nombre d'axes à retenir pour obtenir un pourcentage cumulé d'inertie de 80% - Critère du "coude" - Régle de "Kaiser"

Critère du "coude"

Pour utiliser le critère du "coude", nous devons d'abord afficher le graphique representant la part de variance expliquée par chaque dimension grace à la fonction suivante :

```
fviz_eig(pca)
```



Avec le critere du “coude”, nous devons retenir les 3 premiers axes.

Seuil

Concernant le critere du seuil, nous devons sommer chaque pourcentage d’inertie expliqué des x premieres dimensions jusqu’a atteindre 80%, pour cela on utilise :

```
cumsum(pca$eig[,2])
```

```
##    comp 1    comp 2    comp 3    comp 4    comp 5    comp 6    comp 7
##  46.75505  68.85989  81.59351  87.86001  91.69361  95.08991  97.10232
##    comp 8    comp 9   comp 10   comp 11
##  98.76170  99.45122  99.79129 100.00000
```

Pour atteindre un seuil de 80% de pourcentage d’inertie, nous devons également retenir les 3 premiers axes.

Règle de “Kaiser”

Etant donné que notre ACP est normée, nous devons retenir les axes ayant une valeur propre supérieure a 1 :

```
sum(pca$eig[,1] > 1)
```

```
## [1] 3
```

Tout comme les 2 premiers critères, nous devons retenir 3 axes.

Conclusion

Les 3 critères ayant donné le même nombre d'axes a retenir, nous decidons de retenir 3 axes.

Question 6

Qualité de la representation

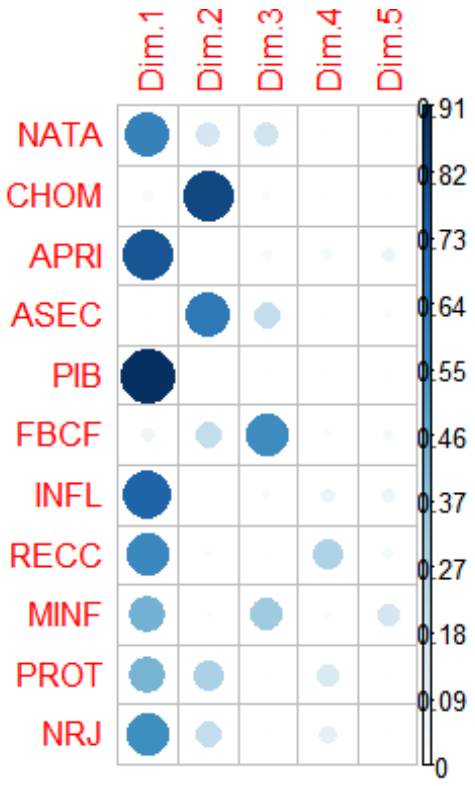
La qualité de representation des variables sur la carte de l'ACP s'appelle cos2, on peut visualiser ses valeurs avec la commande :

```
pca$var$cos2
```

```
##             Dim.1        Dim.2        Dim.3        Dim.4        Dim.5
## NATA 0.6134577683 0.1713265188 1.898596e-01 0.0017785830 0.0000743408
## CHOM 0.0359792418 0.8241929807 2.254383e-02 0.0017120568 0.0032057849
## APRI 0.7787812828 0.0009406057 3.432283e-02 0.0394733701 0.0701057103
## ASEC 0.0002759787 0.6490761682 2.243101e-01 0.0001131471 0.0218942202
## PIB  0.9133474926 0.0074819768 1.315149e-03 0.0035105438 0.0090879290
## FBCF 0.0607034545 0.2250158509 5.681884e-01 0.0214147543 0.0366913298
## INFL 0.7372094368 0.0026131417 1.853555e-02 0.0643831471 0.0646603418
## RECC 0.5880231366 0.0179986670 7.910268e-05 0.2757181070 0.0451753459
## MINF 0.4293519361 0.0141615097 3.279394e-01 0.0220896270 0.1697329330
## PROT 0.4256908676 0.2970080232 6.290350e-03 0.1527296707 0.0007678604
## NRJ  0.5602343721 0.2217171377 7.313760e-03 0.1063929036 0.0002996984
```

On utilise le package corrplot pour visualiser le cos2 de chaque variable sur chaque axe principal :

```
library(corrplot)
corrplot(pca$var$cos2, is.corr=F)
```



Un cos2 élevé (ici en bleu foncé) indique une bonne représentation de la variable sur l'axe principal en question, dans ce cas, sur le cercle de corrélations, la variable est proche de la circonférence du cercle.

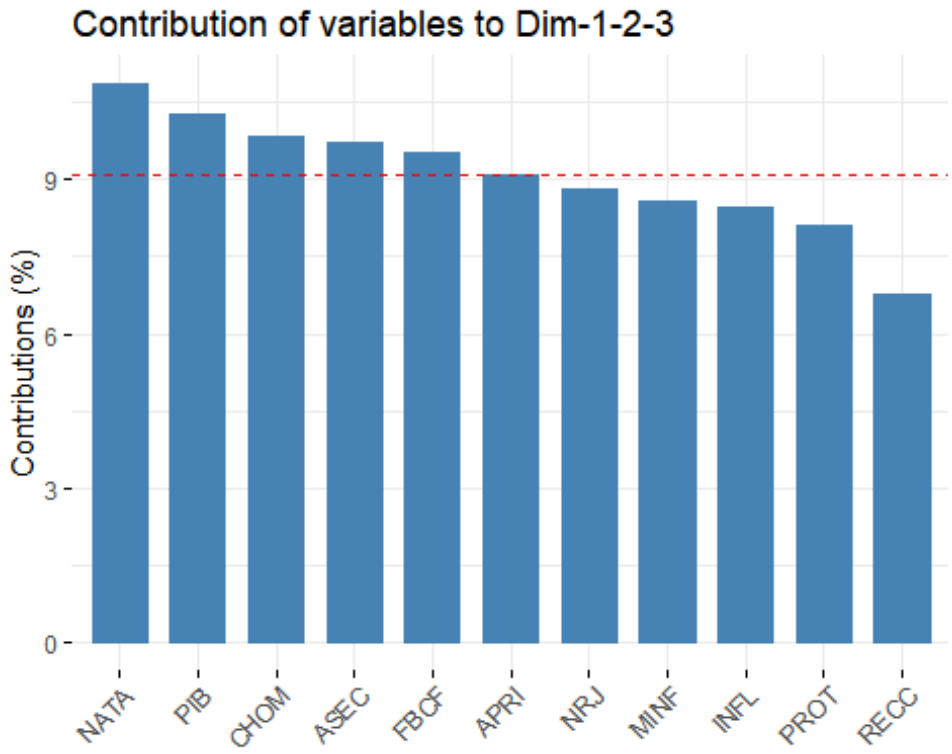
Inversement, un cos2 faible (ici en bleu clair) indique que la variable n'est pas parfaitement representé sur l'axe principal en question, dans ce cas, sur le cercle de corrélations, la variable est proche du centre du cercle.

Analyse des contributions

Nous allons maintenant étudier les contributions des variables et des individus aux 3 axes principaux retenus.

Contribution des variables aux 3 axes principaux

```
fviz_contrib(pca, choice="var", axes=1:3)
```

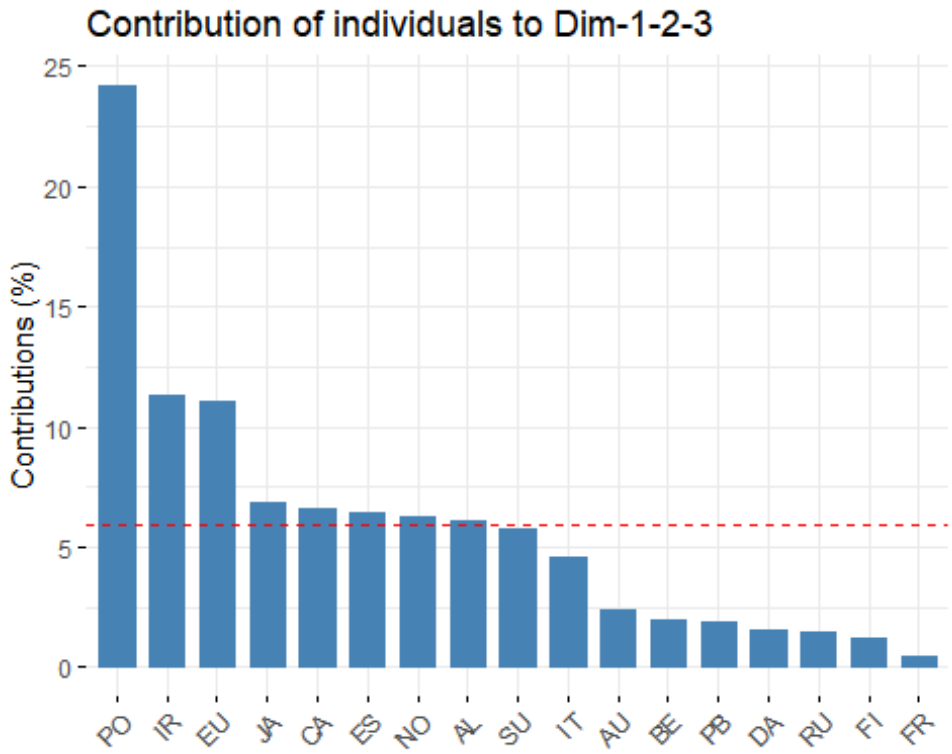


Les individus contribuant le plus aux 3 axes principaux sont celles au dessus de la ligne rouge en pointillé, dans notre cas, l'individu contribuant le plus aux 3 premiers axes est la Pologne avec quasiment 25% de contribution, le reste des individus sont : IR, EU, JA, CA, ES, NO, AL, SU.

Dans le cas de l'axe 1, ce sont les variables PIB, APRI, INFL, NRJ, RECC, PROT et MINF, tandis que pour l'axe 2 ce sont CHOM, ASEC, FBCF et PROT qui y contribuent le plus.

Contribution des individus aux 3 axes principaux

```
fviz_contrib(pca, choice="ind", axes=1:3)
```



Les individus contribuant le plus aux 3 axes principaux sont celles au dessus de la ligne rouge en pointillé, dans notre cas, l'individu contribuant le plus aux 3 premiers axes est la Pologne avec quasiment 25% de contribution, le reste des individus sont : IR, EU, JA, CA, ES, NO, AL, SU.