Devoir3

EL_Hadrami

23/12/2020

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

```
## Loading required package: ggplot2
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

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

```
## corrplot 0.84 loaded
```

Exercice 1

```
Z1 <- c(1:3,4,9)
Z2 <- c(5,10,rep(8,2),12)
n <- length(Z2)
mat <- matrix(c(Z1,Z2),nrow=2,ncol=5,byrow = TRUE,dimnames = list(c("Z1","Z2")))
meanZ1 <- mean(mat[1,])
meanZ2 <- mean(mat[2,])
miZ1 <- sd(mat[1,])
miZ2 <- sd(mat[2,])
Z1norm <- (Z1 - mean(Z1)) / sd(Z1)
Z2norm <- (Z2 - mean(Z2)) / sd(Z2)
matnorm <- matrix(c(Z1norm,Z2norm),nrow=2,ncol=5,byrow = TRUE,dimnames = list(c("Z1","Z2")))
# Matrice de correlation
matcorr <- (1/4) * (matnorm %*% t(matnorm))
# valeurs propres et vecteurs propres
eig <- eigen(matcorr)
valp1 <- eig$values[1]
vp1 <- eig$vectors[,1]
valp2 <- eig$values[2]
vp2 <- eig$vectors[,2]
# Cercle de correlation contenat les vecteurs X1 et X2
X1 <-  sqrt(valp1) * vp1
X2 <- sqrt(valp2) * vp2
```

1.ACP sur la main

```
df <- as.data.frame(mat)
res.pca <- PCA(df,scale.unit = TRUE ,graph = FALSE)
```

2. Interpretation

3.Utilisation des commandes

```
# Standardisation des données
s1 <- scale(x = Z1,center=TRUE,scale=TRUE)
```

```r
s2 <- scale(x = Z2,center=TRUE,scale=TRUE)
mats <- matrix(c(s1,s2),nrow = 2,ncol=5,byrow = TRUE)
```

fonction gsvd

```r
gsvd <- function(Z,r,c){
  #Z matrice numerique de dimension (n,p) et de rang k
  #r poids de la metrique des lignes N=diag(r)
  # c poids de la metrique des colonnes M=diag(c)
  #-----sortie---------------
  # d vecteur de taille k contenant les valeurs singulieres (racines carres des valeurs propres)
  # U matrice de dimension (n,k) des vecteurs propres de de ZMZ'N
  # V matrice de dimension (p,k) des vecteurs propres de de Z'NZM
  k <-qr(Z)$rank
  colnames<-colnames(Z)
  rownames<-rownames(Z)
  Z <-as.matrix(Z)
  Ztilde <-diag(sqrt(r)) %*% Z %*%diag(sqrt(c))
  e <-svd(Ztilde)
  U <-diag(1/sqrt(r))%*%e$u[,1:k]# Attention : ne s'ecrit comme cela que parceque N et M sont diagonale
  V <-diag(1/sqrt(c))%*%e$v[,1:k]
  d <- e$d[1:k]
  rownames(U) <- rownames
  rownames(V) <- colnames
  if(length(d)>1)
    colnames(U) <-colnames(V) <-paste("dim", 1:k, sep = "")
  return(list(U=U,V=V,d=d))
}
r <-rep(1/nrow(mats),nrow(mats)) #lignes ponderees par 1/n
c <-rep(1,ncol(mats)) #colonnes ponderees par 1
U <-gsvd(mats,r,c)$U
d <-gsvd(mats,r,c)$d
Psi <- U %*%diag(d)
#princomp(mat,cor=TRUE)
```

Exercice 2

```r
# load data
data_ski <- read.table("data/stations.txt",header = TRUE)
# extraction des variables quantitatives
data_ski_active <- as.matrix(data_ski[1:32,2:7])
rownames(data_ski_active) <- data_ski$Nom
summary(data_ski_active)
```

```
##     prixforf          altmin         altmax         pistes           kmfond
##  Min.   : 42.00   Min.   : 500   Min.   :1600   Min.   :  0.00   Min.   : 0.0
##  1st Qu.: 81.75   1st Qu.:1138   1st Qu.:2275   1st Qu.: 26.00   1st Qu.: 9.5
##  Median : 95.50   Median :1400   Median :2600   Median : 34.00   Median :22.0
##  Mean   :104.69   Mean   :1323   Mean   :2567   Mean   : 49.44   Mean   :27.5
##  3rd Qu.:140.00   3rd Qu.:1550   3rd Qu.:2838   3rd Qu.: 71.00   3rd Qu.:36.5
##  Max.   :160.00   Max.   :1850   Max.   :3450   Max.   :129.00   Max.   :80.0
##     remontee
##  Min.   :  4.00
##  1st Qu.: 17.00
##  Median : 23.00
```

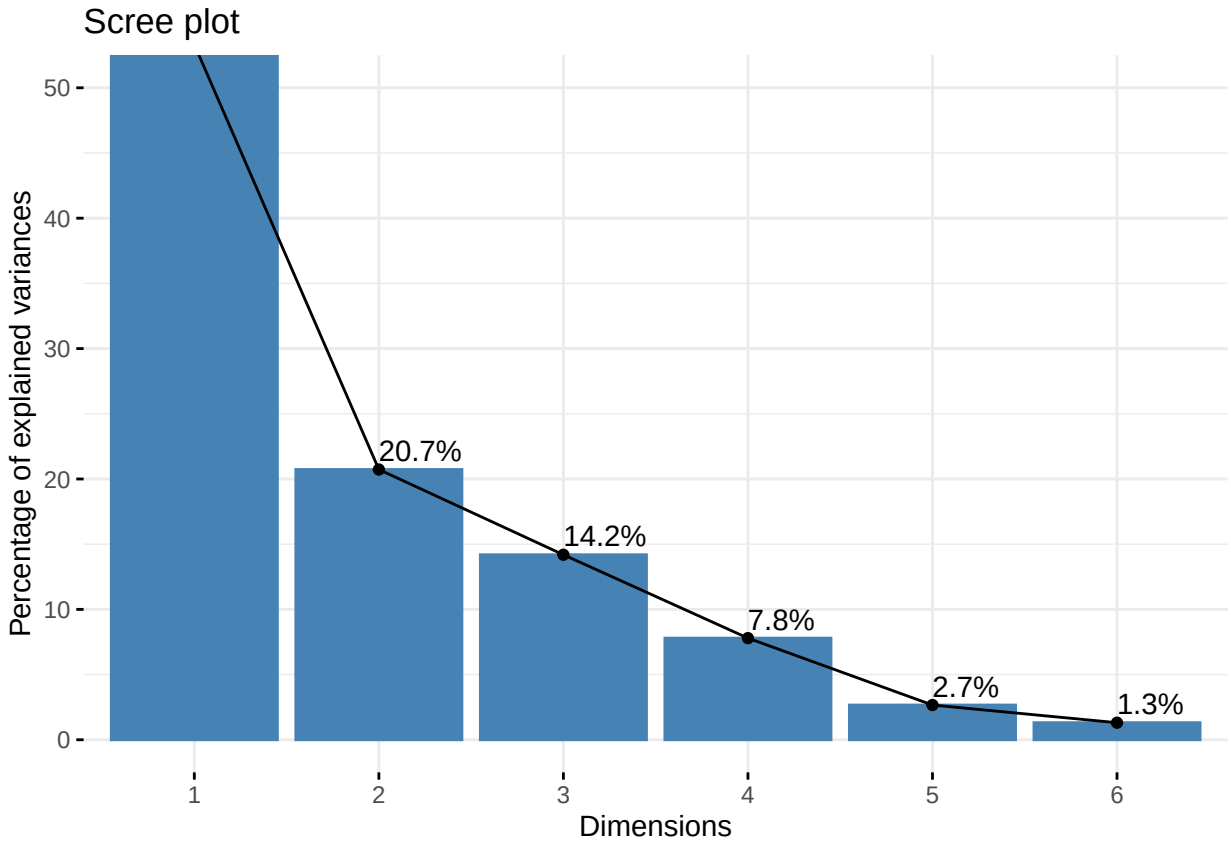
```
##  Mean   : 33.81
##  3rd Qu.: 45.75
##  Max.   :110.00
```

PCA

```
pcaski <- PCA(data_ski_active,scale.unit = T,graph = FALSE)
# Visualisation des valeurs propres
valp <- pcaski$eig
```

Graphe des valeurs propres

```
fviz_eig(pcaski, addlabels = TRUE, ylim = c(0, 50))
```



Les deux premières composantes principales expliquent 74% de la variation,donc les deux premiers axes peuvent etre acceptés.

Graphique des variables

```
var <- get_pca_var(pcaski)
```

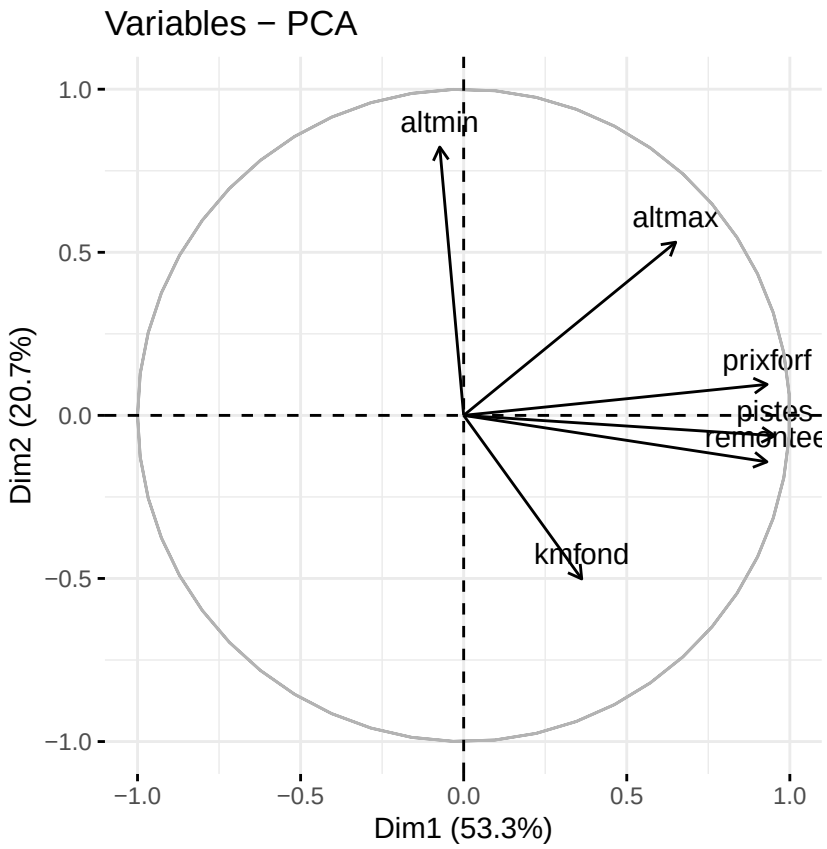
Coordonnées des variables

```
var$coord
```

```
##               Dim.1       Dim.2       Dim.3      Dim.4       Dim.5
## prixforf  0.93031706  0.09513297 -0.08572117  0.1251448 -0.31055554
## altmin   -0.07336694  0.82270492  0.48871130  0.2792394  0.02904032
## altmax    0.65006226  0.53099234 -0.03832048 -0.5398817  0.04967488
## pistes    0.95404437 -0.06226765 -0.11082956  0.1446174  0.05250905
## kmfond    0.36193326 -0.50154750  0.76829658 -0.1613207 -0.03250806
```

```
## remontee  0.92973674 -0.14239486 -0.03422684  0.1886935  0.23708189
```

```
fviz_pca_var(pcaski,axes = c(1,2))
```

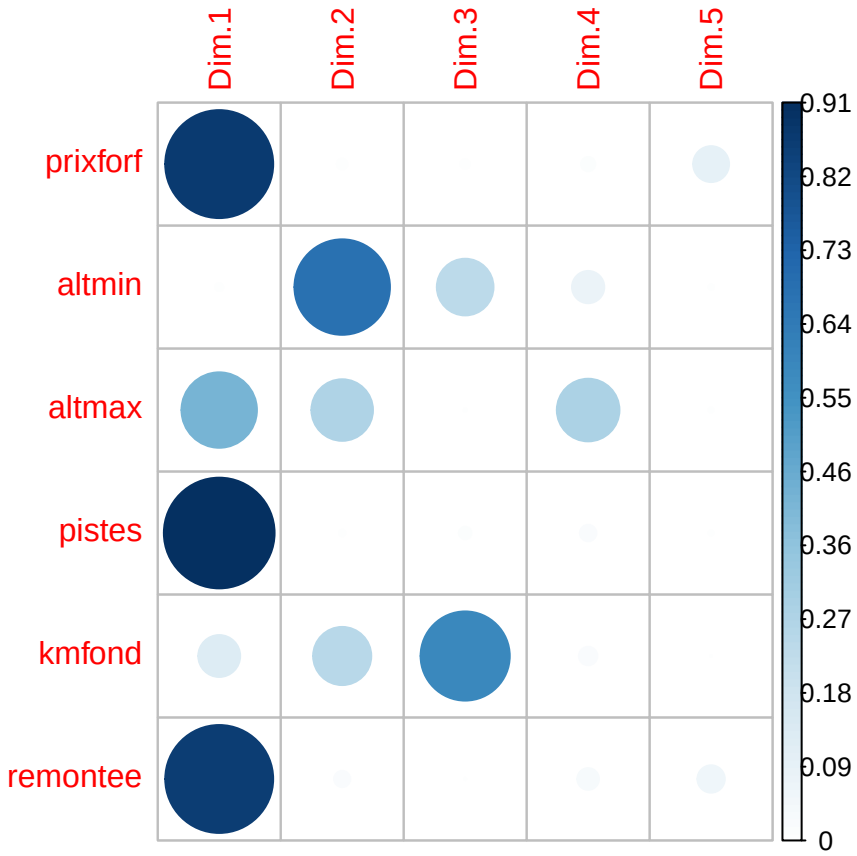


Interpretation

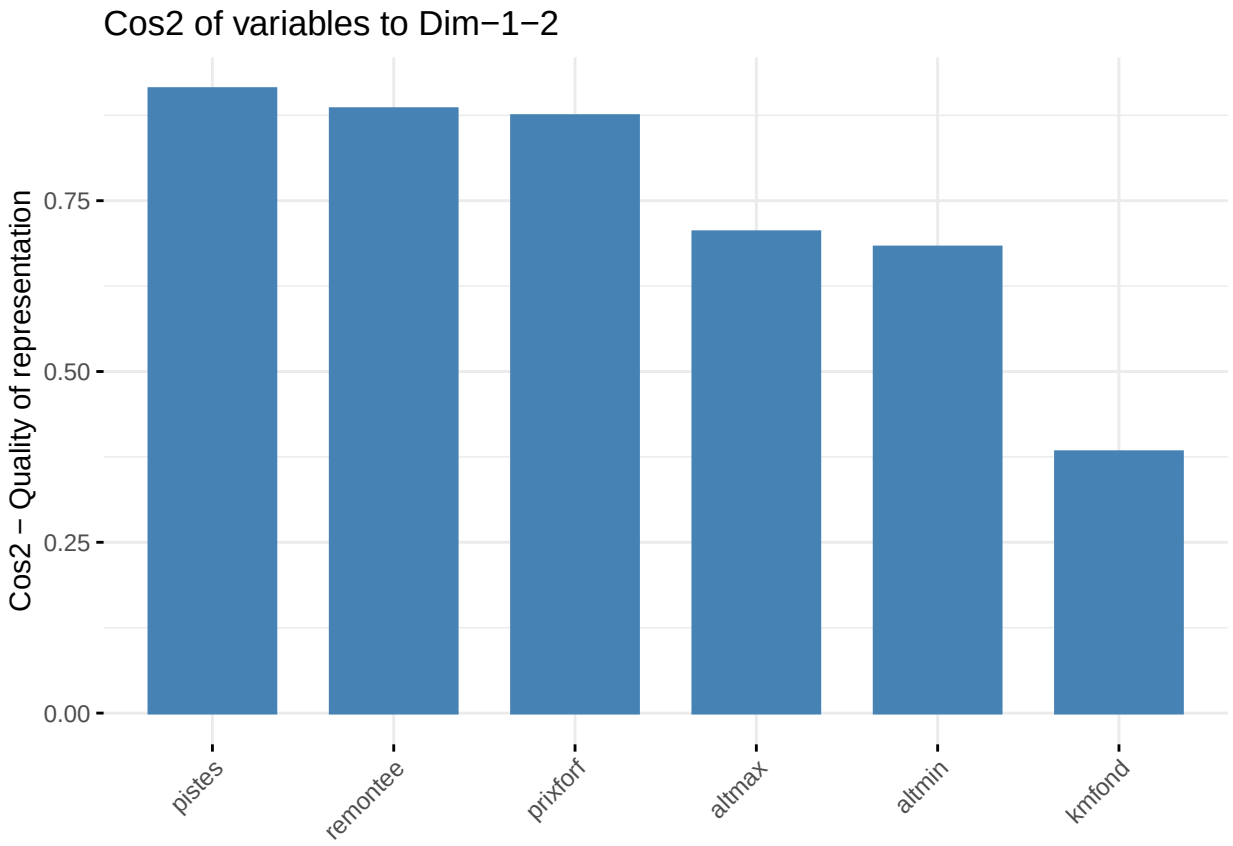
- Les variables positivement corrélées sont regroupées
- Les variables négativement corrélées sont positionnées sur les côtés opposés de l'origine du graphique (quadrants opposés).
- La distance entre les variables et l'origine mesure la qualité de représentation des variables, les variables qui sont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

Qualité de representation des variables

```
corrplot(var$cos2,is.corr = FALSE)
```



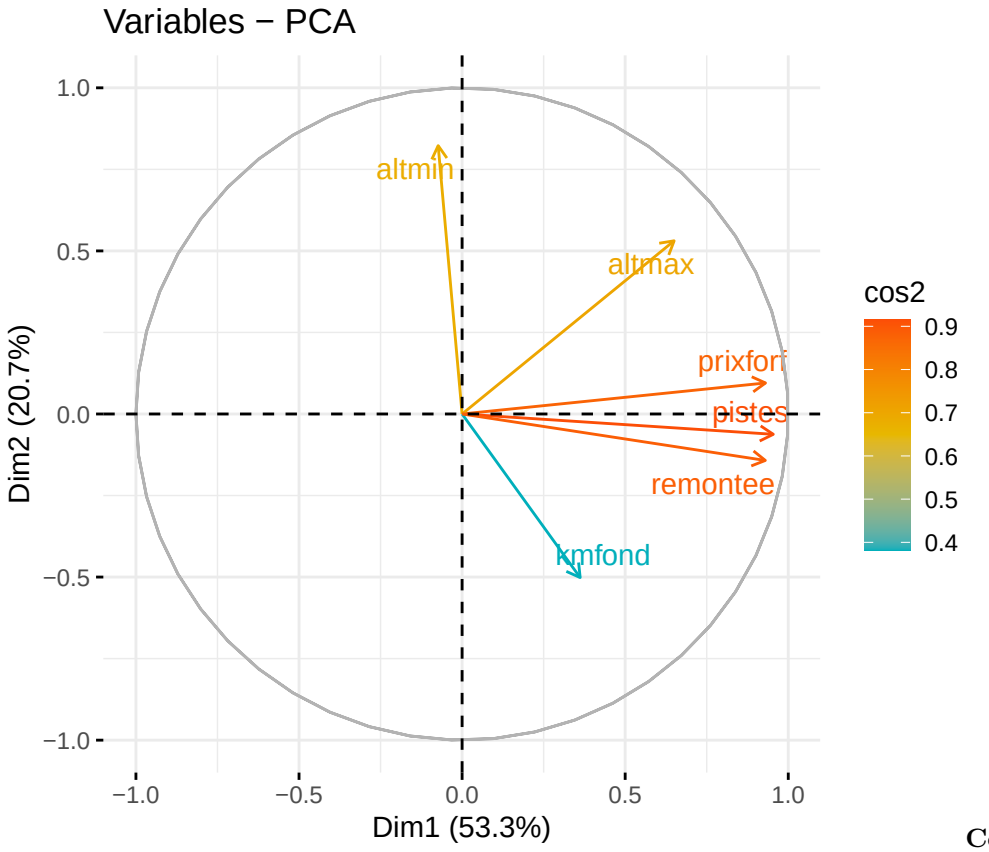

```
fviz_cos2(pcaski, choice = "var", axes = 1 :2)
```



Interpretations

Un cos2 élevé indique une bonne représentation de la variable sur les axes principaux en considération(comme on peut le voir dans le graphe ci dessus), dans ce cas la variable est positionnée à proximité de la circonférence du cercle de corrélation

```
fviz_pca_var(pcaski, col.var = "cos2",gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),repel = TRUE)
```

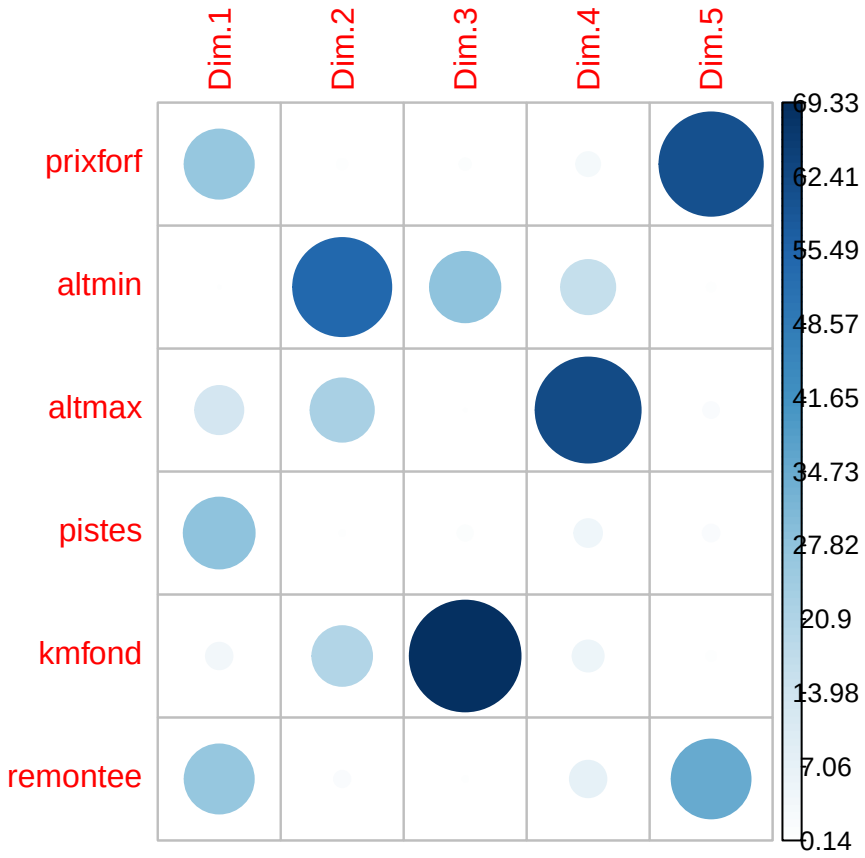


Contributions des variables

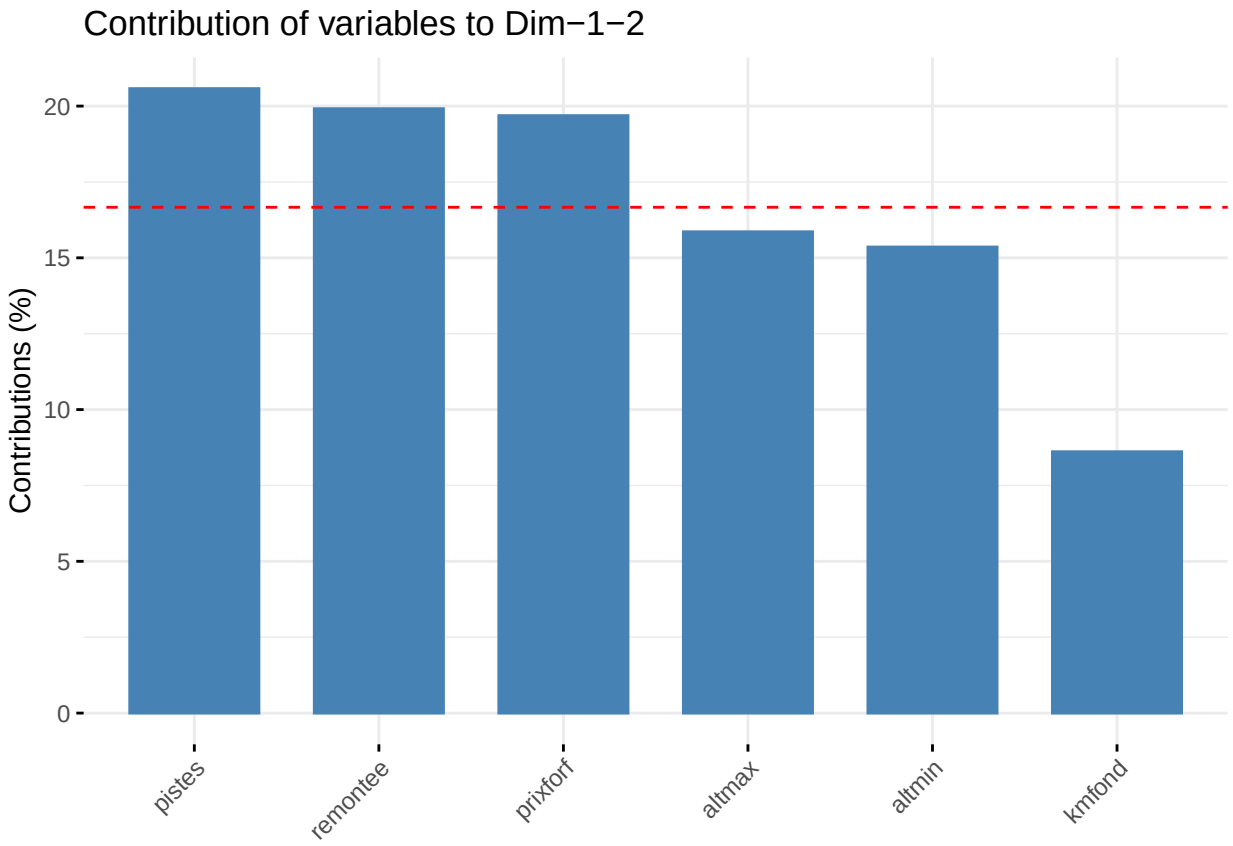
```
var$contrib
```

```
##               Dim.1      Dim.2      Dim.3     Dim.4      Dim.5
## prixforf 27.0545026  0.7277779  0.8630735  3.348903 60.3619164
## altmin    0.1682591 54.4283218 28.0528130 16.673650  0.5278217
## altmax   13.2095337 22.6732235  0.1724779 62.326754  1.5443938
## pistes   28.4521266  0.3117897  1.4427227  4.472173  1.7256503
## kmfond    4.0948176 20.2283699 69.3313170  5.564903  0.6614036
## remontee 27.0207604  1.6305172  0.1375958  7.613617 35.1788143
```

```
corrplot(var$contrib,is.corr = FALSE)
```




```
fviz_contrib(pcaski, choice = "var", axes = 1 :2, top = 6)
```

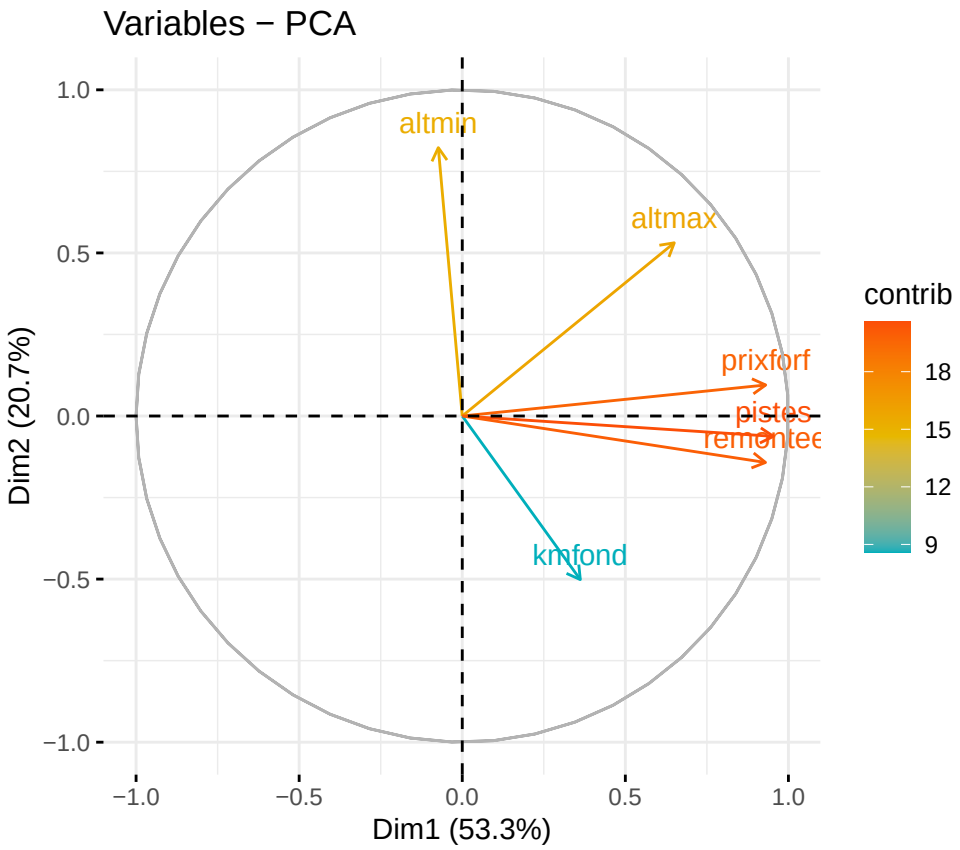


Interpretation

La ligne en pointillé rouge, sur le graphique ci-dessus, indique la contribution moyenne attendue.
Donc les variables les plus contributives sont **piste**,**remontee** et **prixfort**

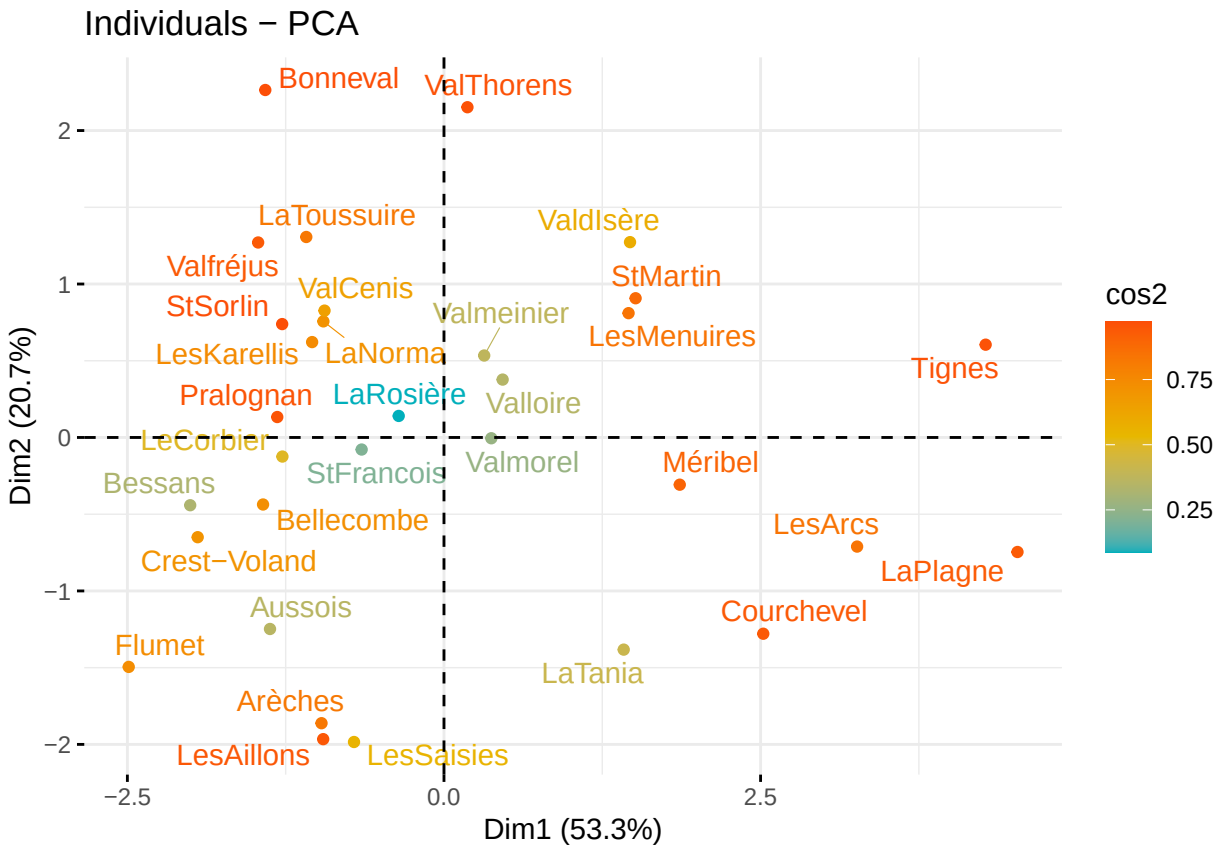
Diagramme circulaire des variables contributives

```
fviz_pca_var(pcaski, col.var = "contrib",gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),alha.var="c
```

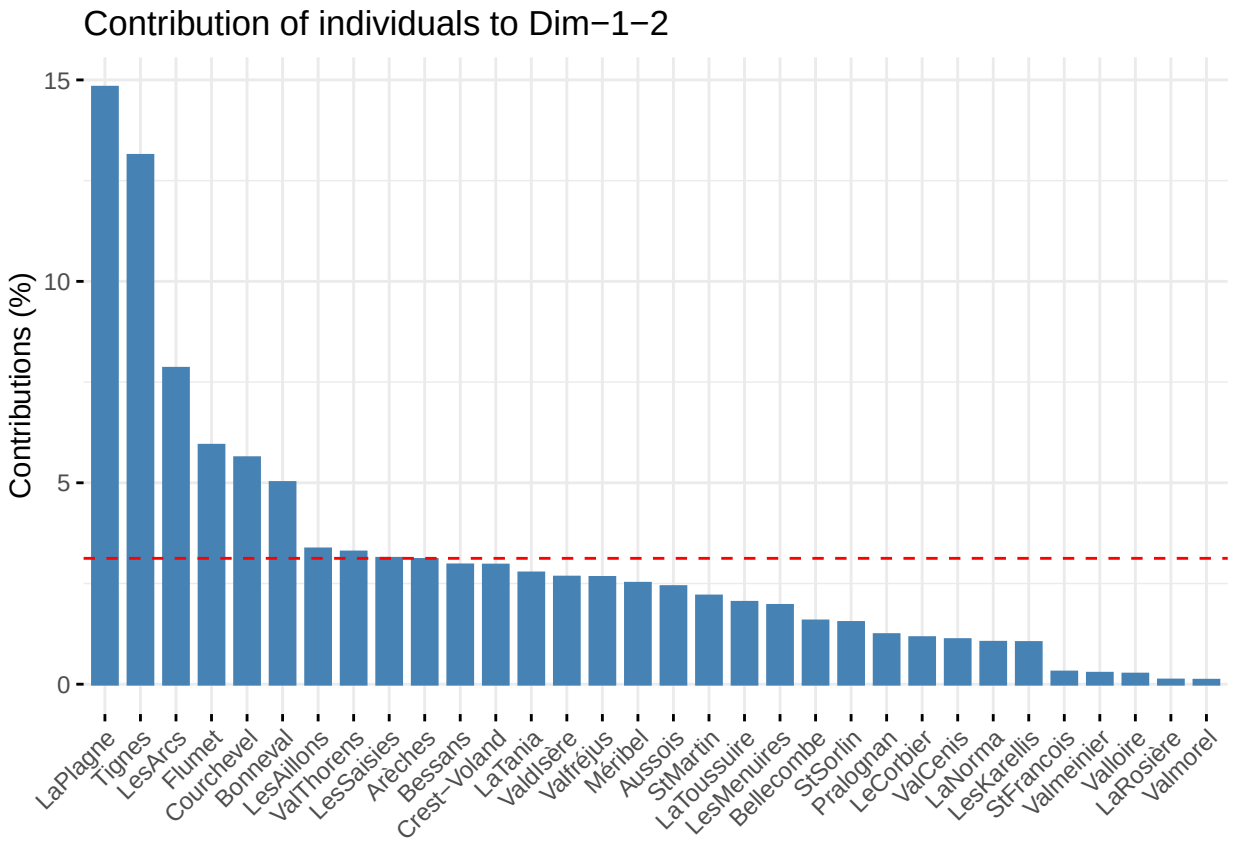


Graphiques des individus

```
ind <- get_pca_ind(pcaski)
fviz_pca_ind (pcaski, col.ind = "cos2",gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),repel = TRUE)
```

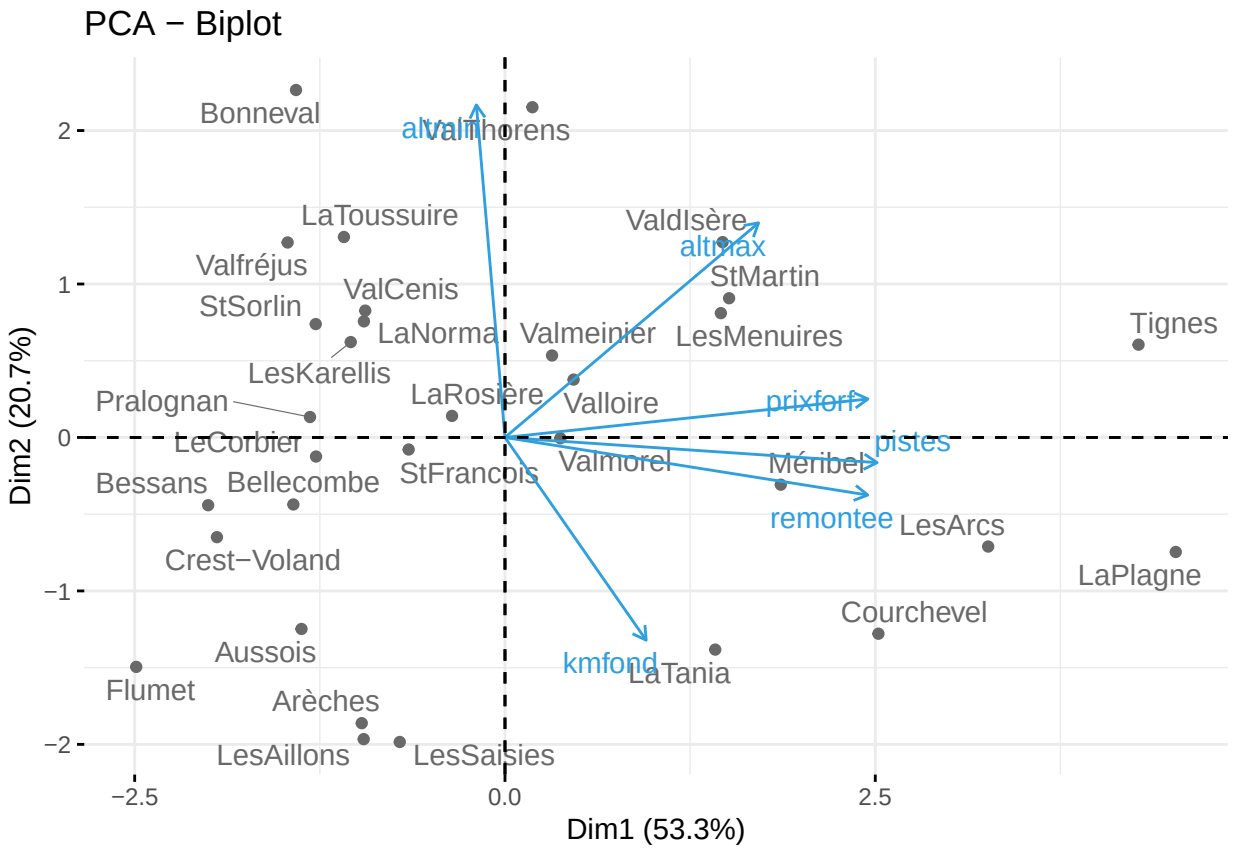



```
fviz_contrib(pcaski, choice = "ind", axes = 1 :2)
```



Biplot

```
fviz_pca_biplot(pcaski,
repel = TRUE,col.var = "#2E9FDF", # Couleur des variables
col.ind = "#696969") # Couleur des individues )
```


PCA − Biplot
Bonneval
altvalThorens
LaToussuire
ValdIsère
altmax
Valfréjus
StMartin
ValCenis
StSorlin
LaNorma
Valmeinier
LesMenuires
Tignes
LesKarellis
Pralognan
LaRosière
Valloire
prixforf
LeCorbier
pistes
Valmorel
Méribel
Bessans
Bellecombe
StFrancois
remontee
LesArcs
Crest−Voland
LaPlagne
Courchevel
Aussois
kmfond
LaTania
Flumet
Arèches
LesAillons
LesSaisies
Dim2 (20.7%)
Dim1 (53.3%)
2
1
0
−1
−2
−2.5
0.0
2.5