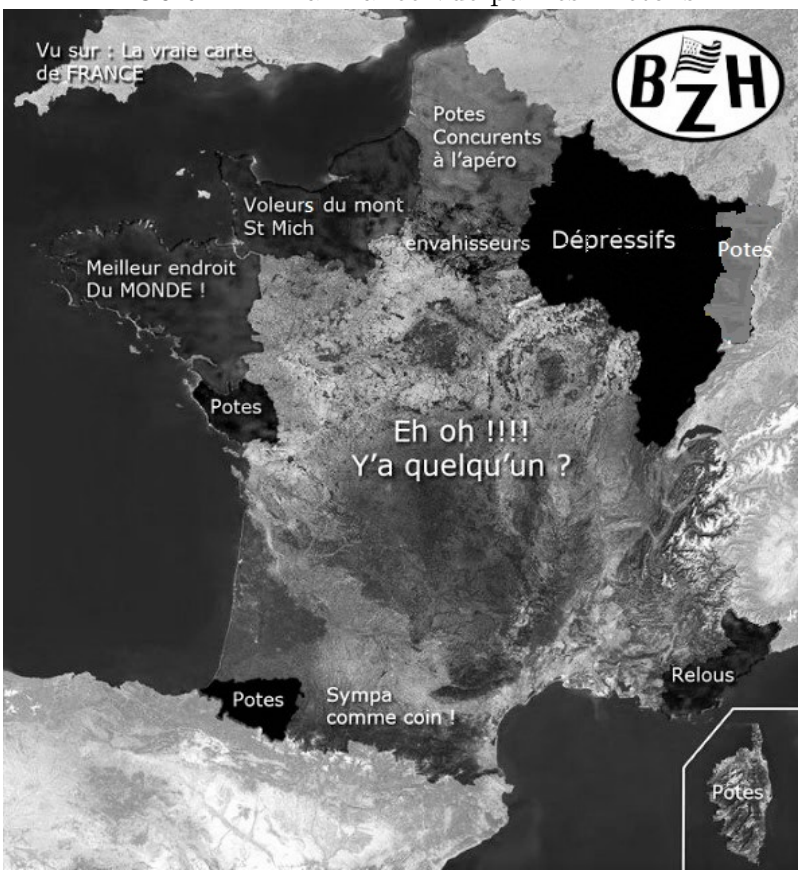
T. D. n° 1
Analyse des Composantes principales

Résumé

Ce document est le TD n° 1 du module Analyse exploratoire. Il reprend rapidement des éléments du cours et propose une mise en pratique interactive de l'ACP. Dans ce TD nous utiliserons une ACP centrée et réduite (appelée ACP normée). L'objectif est d'appliquer une ACP en utilisant le package *ade4* de R et d'interpréter les résultats.

FIGURE 1 – La France vue par les Bretons



source :`http://eoilugofrancais.blogspot.fr`

1 Le cidre

1.1 Chargement des données

Commencez par charger les données du fichier cidre.csv. Associez ces données à un dataframe (fonction *as.dataframe()*) et appliquez la fonction *summary()*. Il est

également possible de télécharger le fichier cidre.csv sur ce lien : `http://sebastien.ledien.free.fr/unofficial_factominer/livreR/index.html`.

```
       Type        S.Sucree         S.Acide          S.Amere
 Brut    :50   Min.   :3.444   Min.   :2.107   Min.   :2.143
 Demi-sec:30   1st Qu.:4.580   1st Qu.:3.625   1st Qu.:3.286
 Doux    :10   Median :5.250   Median :4.089   Median :3.964
               Mean   :5.169   Mean   :4.181   Mean   :4.274
               3rd Qu.:5.670   3rd Qu.:4.643   3rd Qu.:5.268
               Max.   :7.036   Max.   :6.536   Max.   :7.857

 S.Astringente
 Min.   :0.7143
 1st Qu.:1.4643
 Median :1.9821
 Mean   :2.0321
 3rd Qu.:2.4233
 Max.   :4.6786
```

```
> pearson.test(cidre$S.Sucree)

        Agostino-Pearson chi-square normality test

data:  cidre$S.Sucree
P = 13.422, p-value = 0.201

> pearson.test(cidre$S.Acide)

        Agostino-Pearson chi-square normality test

data:  cidre$S.Acide
P = 17.756, p-value = 0.05923

> pearson.test(cidre$S.Amere)

        Agostino-Pearson chi-square normality test

data:  cidre$S.Amere
P = 15.733, p-value = 0.1075

> pearson.test(cidre$S.Astringente)

        Agostino-Pearson chi-square normality test

data:  cidre$S.Astringente
P = 15.733, p-value = 0.1075
```

À vous !

a) Justifiez l'utilisation d'un test de Agostino-Pearson sur les données.

b) Quel autre test aurait pu être appliqué ?

1.2 Corrélation linéaire sur les variables

Une ACP se fait sur des variables quantitatives. Commencez par afficher la corrélation linéaire de Pearson sur les variables quantitatives à l'aide de la fonction *round()*. Ce coefficient permet de détecter la présence ou l'absence d'une relation linéaire entre deux caractères quantitatifs continus. En principe, le coefficient de Pearson n'est applicable que pour mesurer la relation entre deux variables X et Y ayant une distribution gaussienne et ne comportant pas de valeur exceptionnelles. Si ces conditions ne sont pas vérifiées (cas fréquent ...) l'emploi de ce coefficient peut aboutir à des conclusions erronées sur la présence ou l'absence d'une relation. On notera également que l'absence d'une relation linéaire ne signifie pas l'absence de toute relation entre les deux variables étudiées.

```
> round(cor(cidre[, c(2:5)]),2)
               S.Sucree S.Acide S.Amere S.Astringente
S.Sucree           1.00    0.10   -0.63          -0.49
S.Acide            0.10    1.00   -0.44          -0.16
S.Amere           -0.63   -0.44    1.00           0.83
S.Astringente     -0.49   -0.16    0.83           1.00
```

À vous !

a) Déterminez deux groupes d'attributs qui présentent des corrélations linéaires entre elles (r>0,5).

b) Justifiez l'utilisation d'une ACP.

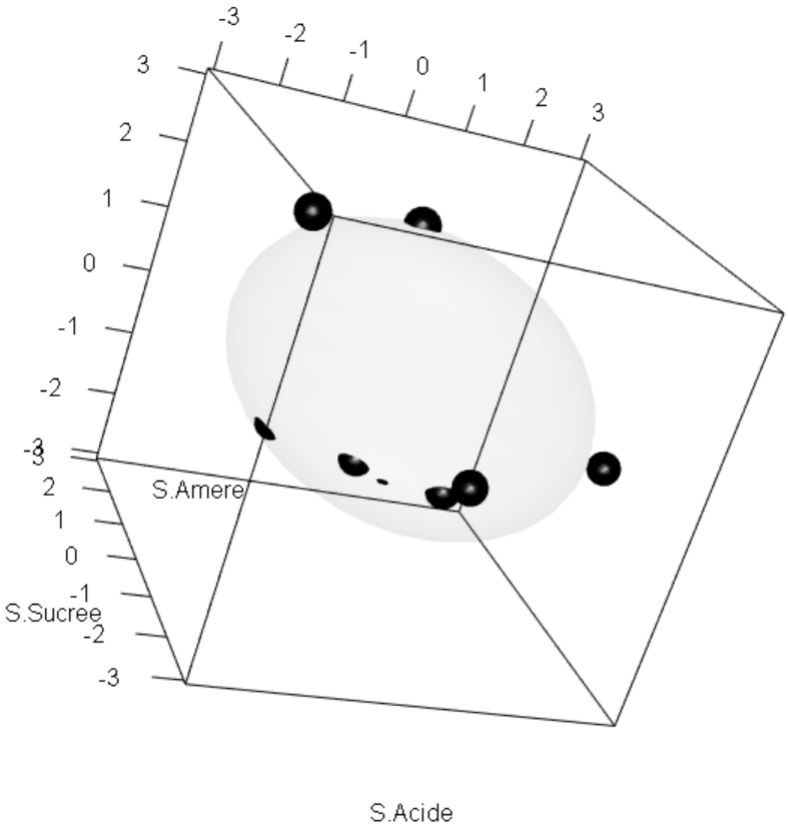
c) Expliquez les différences obtenues entre une ACP normée et non normée ?

1.3 Représentation 3D

Chargez le package *rgl*. Appelez également la fonction *attach()* sur votre dataframe. La fonction *attach()* permet de faire appel aux colonnes d'un *dataframe* en les nommant directement. Faites une représentation 3D des attributs S.Acide, S.Sucree et S.Amere :

```
> # Representation 3D : S.Acide, S.Sucree, S.Amere
> plot3d(S.Acide, S.Sucree,S.Amere,type="s",xlim=c(-3,3),ylim=c
   (-3,3),zlim=c(-3,3))
> plot3d(ellipse3d(cor(cbind(S.Acide,S.Sucree,S.Amere))),col="grey
   ",alpha=0.05,add=TRUE)
```

FIGURE 2 – Ellipse de corrélation linéaire : S.Acide, S.Sucree, S.Amere



La fonction *ellipse3d()* permet de représenter une ellipse de concentration. L'ellipse de concentration d'un sous–nuage de point est l'ellipse d'inertie telle qu'une distribution uniforme à l'intérieur de l'ellipse a une variance égale à celle du sous–nuage.

À vous !

a) Affichez l'ellipse de corrélation linéaire dans la représentation en 3D pour les attributs S.Astringente, S.Sucree et S.Amere.

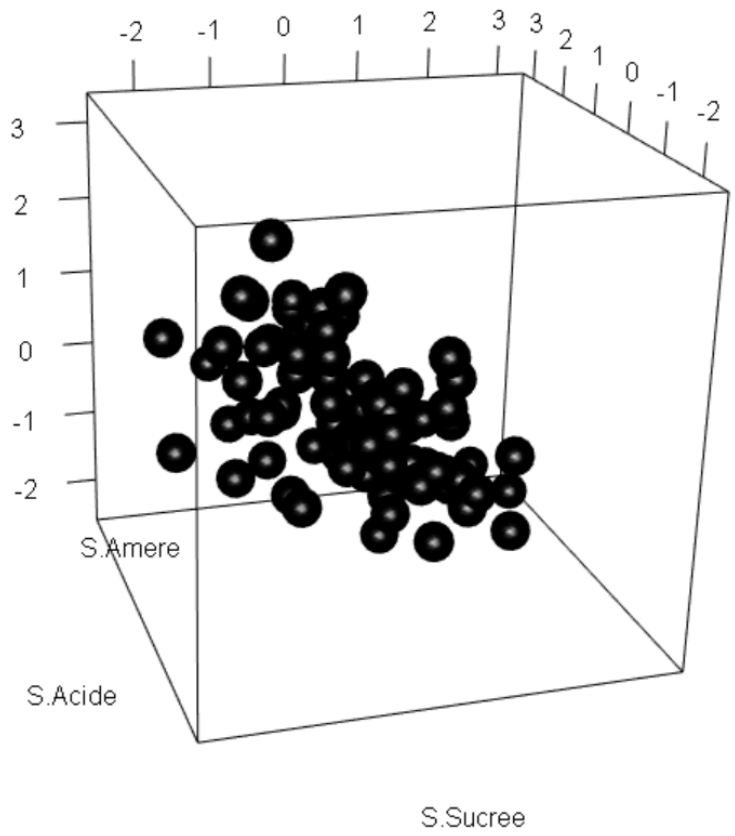
b) Expliquez les différences entre les ellipses obtenues dans les deux nuages.

1.4 Centrage

La fonction *scale()* permet de centrer les données puis divise les valeurs par l'écart-type. Appliquez cette fonction sur votre *dataframe*.

```
> cidre.cr <- scale(cidre[, c(2:5)])
> lims <- c(min(cidre.cr),max(cidre.cr))
> plot3d(cidre.cr, type = "s", xlim = lims, ylim = lims,zlim = lims
   )
```

FIGURE 3 – Plot3D après la fonction *scale()* : S.Acide, S.Sucree, S.Amere



À vous !

a) Quelles différences voyez-vous entre ce graphique et le plot 3D d'avant ?

1.5 ACP

Le package ade4 permet de réaliser une ACP. Il est téléchargable sur : `https://cran.r-project.org/web/packages/ade4/index.html`. Il est cependant possible de télécharger le package FactomineR qui permet également de faire une ACP. Utiliser la fonction *dudi.pca()* le package ade4 pour exécuter une ACP centrée réduite :

```
> install.packages("ade4")
> library(ade4)
> acp <- dudi.pca(cidre[, c(2:5)], center=TRUE, scale=TRUE, scannf
    = FALSE, nf = 3)
> names(acp)
 [1] "tab"   "cw"    "lw"    "eig"   "rank" "nf"    "c1"    "li"    "co"
      "l1"    "call" "cent" "norm"
```

À vous !

a) Que contient le *dataframe tab* ?
b) Comparez avec le tableau de données cidre.cr, expliquez la légère différence.
c) Quelle manipulation doit on réaliser pour retrouver exactement le tableau utilisé dans *dudi.pca()* ?

Le vecteur *cw* donne le poids des colonnes (*column weight)*, c'est-à-dire le poids des variables. Par défaut, chaque variable a un poids de 1.

```
> acp$cw
[1] 1 1 1 1
```

Le vecteur *lw* donne le poids des lignes (*line weight*), c'est-à-dire le poids des individus. Par défaut, chaque individu a un poids de $1/n$.

```
> head(acp$lw)
[1] 0.01111111 0.01111111 0.01111111 0.01111111 0.01111111
    0.01111111
> head(acp$lw)*nrow(cidre)
[1] 1 1 1 1 1 1
```

Les valeurs propres renseignent la part de l'inertie totale prise en compte par chaque axe.

```
> (pve <- 100*acp$eig/sum(acp$eig))
[1] 60.944152 23.661817 13.075336  2.318694
> cumsum(pve)
[1]  60.94415  84.60597  97.68131 100.00000
```

1.6 Informations associées à une ACP

Dans l'exemple, le premier axe factoriel extrait 61 % de l'inertie totale, le deuxieme axe factoriel 23 % de l'inertie totale. Le premier plan factoriel représente donc 84.6 % de l'inertie initiale. Ceci signifie que lorsqu'on projette le nuage de points initial dans R3 sur le plan défini par les deux premiers axes factoriels, il y a peu de perte d'informations.

À vous !

a) Quel pourcentage de l'inertie total avec 3 axes ?
b) Cherchez la signification du vecteur rank.
c) Cherchez la signification du vecteur nf.
d) Cherchez la signification du vecteur c1.
e) Cherchez la signification du vecteur l1.
f) Cherchez la signification du vecteur co.
g) Cherchez la signification de l'objet call.
h) Cherchez la signification du vecteur cent.
i) Cherchez la signification du vecteur norm.
j) Donnez le nombre de facteurs retenus.

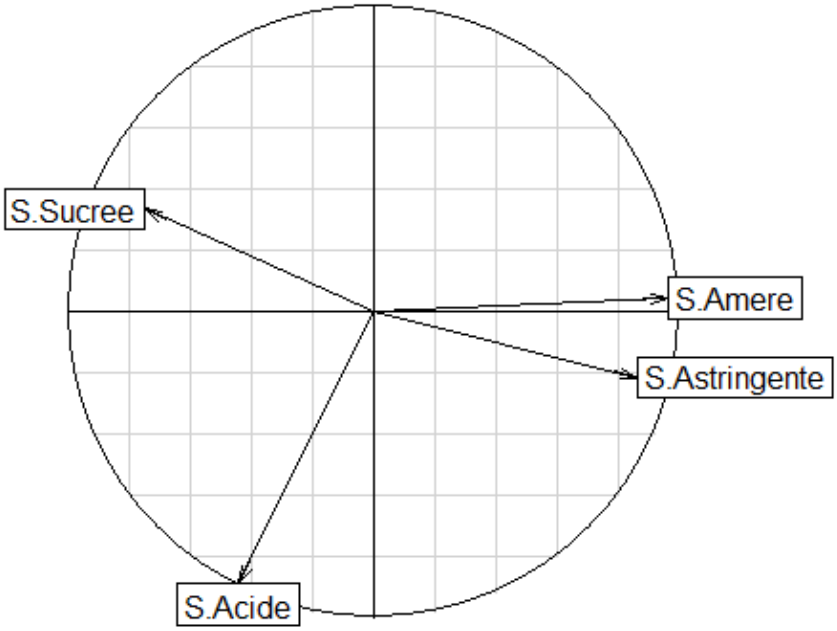
1.7 Analyse des variables

Observez les attributs de notre dataframe sur trois axes obtenus par l'ACP. La représentation des attributs se fait à travers un cercle de corrélation linéaire et on peut voir aisément la proximité des attributs dans le cercle.

```
> inertie <-inertia.dudi(acp, col.inertia=TRUE)
> # Coordonnees des attributs
> round(acp$co,2)
              Comp1 Comp2 Comp3
S.Sucree      -0.75  0.34  0.56
S.Acide       -0.44 -0.89  0.12
S.Amere        0.96  0.05  0.11
S.Astringente  0.86 -0.21  0.43
> # ctr en %
> inertie$col.abs/100
              Comp1 Comp2 Comp3
S.Sucree      23.24 12.05 60.28
S.Acide        8.01 82.92  2.65
S.Amere       38.17  0.24  2.34
S.Astringente 30.57  4.78 34.74
> # qlt en %
> inertie$col.re/100
               Comp1  Comp2 Comp3 con.tra
S.Sucree      -56.66  11.40 31.53      25
S.Acide       -19.54 -78.49  1.38      25
S.Amere        93.05   0.23  1.22      25
S.Astringente  74.53  -4.53 18.17      25

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

FIGURE 4 – Cercle des corrélations linéaires



À vous !

a) Comment reconnaît-on sur la figure qu'un attribut est bien représenté ?

b) Quel est l'attribut le moins bien représenté dans le cercle ? Justifiez votre réponse.

c) À l'aide de la figure précédente (figure 4), précisez l'attribut le plus corrélé positivement à l'astringente, le plus corrélé négativement à l'astringente et le moins corrélé à l'astringente.

d) Quels sont les attributs qui ont contribué à l'axe F1 ? Justifiez votre réponse.

e) Donnez une signification à cet axe.

f) Quels sont les attributs qui ont contribué à l'axe F2 ? Justifiez votre réponse.

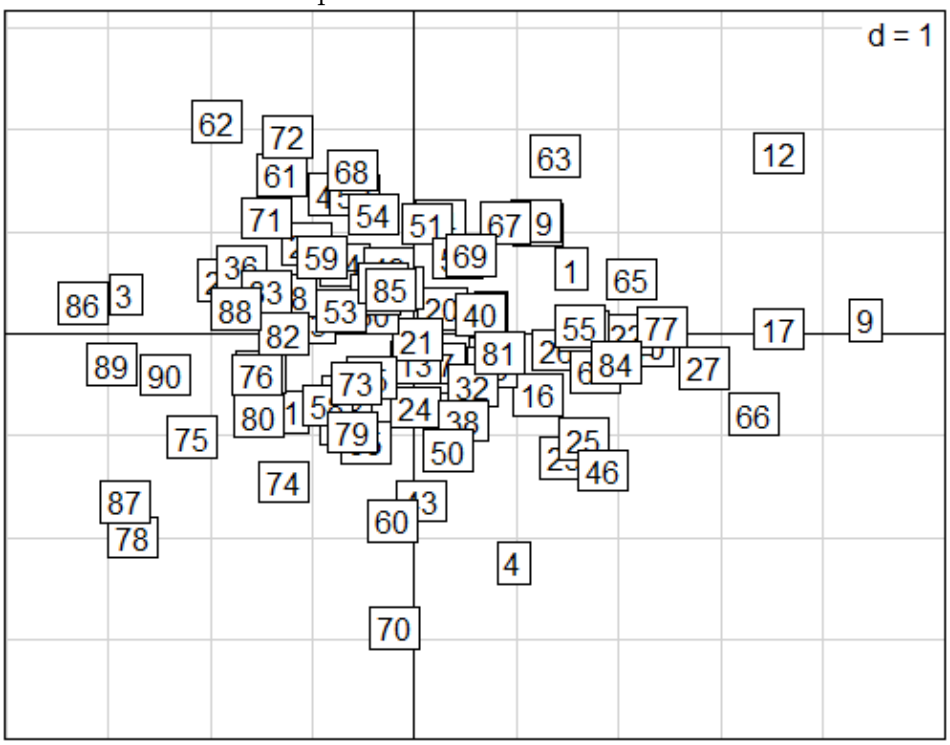
g) Donnez une signification à cet axe.

1.8 Conclusion

La fonction *s.label()* permet de représenter les individus sur les différents plans factoriels, par exemple sur le premier plan factoriel :

```
s.label(acp$li, xax = 1, yax = 2)
```

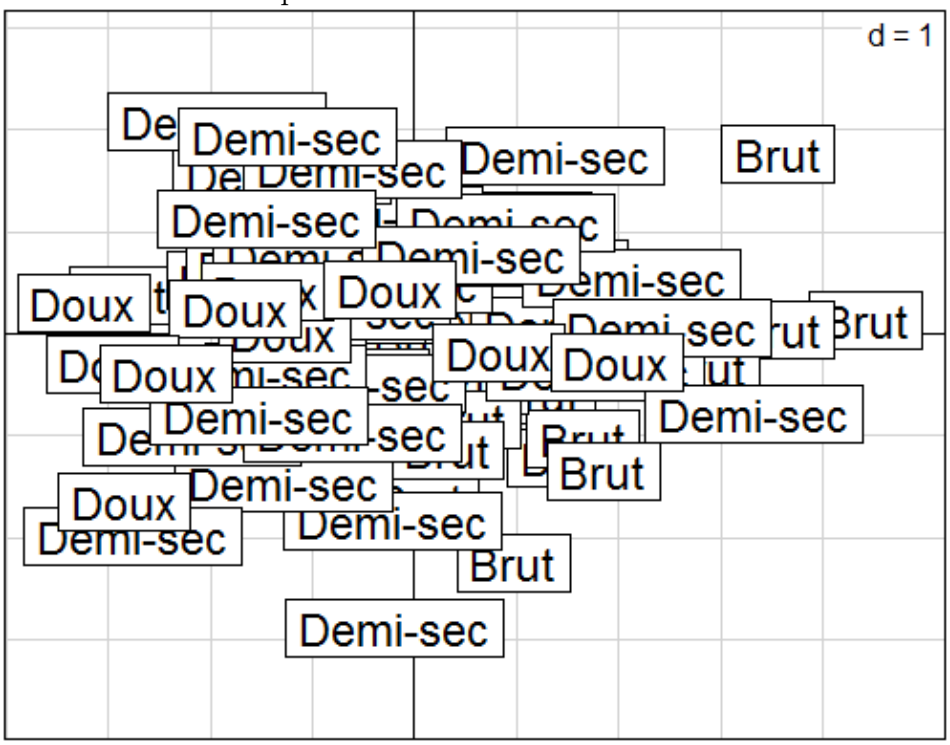
FIGURE 5 – Représentation des individus avec l'ACP.



Afin de bien interpréter les données, il est préférable d'utiliser comme étiquette d'un cidre son type.

```
s.label(acp$li, xax = 1, yax = 2, label=as.character(cidre$Type),
    clabel=1.5)
```

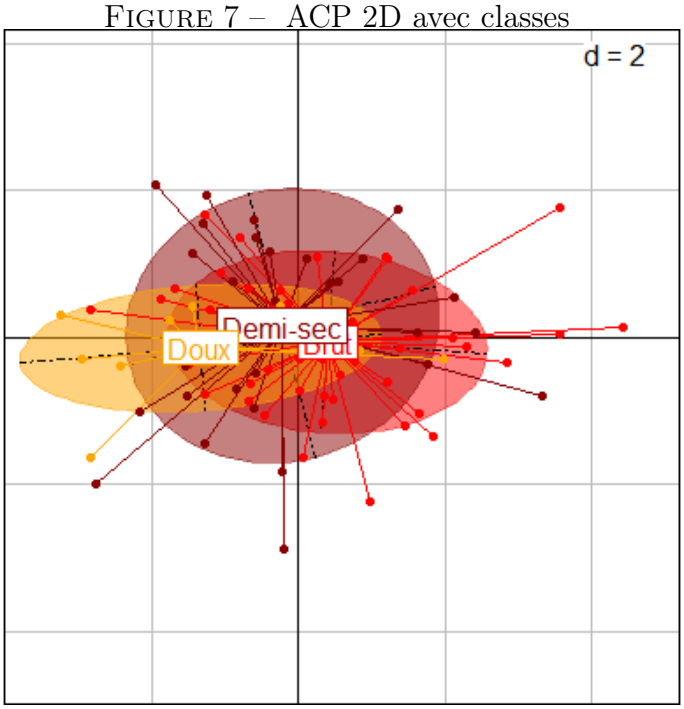
FIGURE 6 – Représentation labellisé des cidres avec l'ACP.



La fonction *s.class()* permet de porter en information supplémentaire une variable qualitative définissant des groupes d'individus, par exemple :

```
gcol <- c("red1", "red4","orange")
s.class(dfxy = acp$li, fac = cidre$Type, col = gcol, xax = 1, yax =
    2)
```

FIGURE 7 – ACP 2D avec classes



a) Utilisez la fonction *scatter(acp)*, concluez.