Fiche TD avec le logiciel : tdr51

Composantes principales

D. Chessel & A.B. Dufour & J.R. Lobry

Trois exemples numériques permettent d'aborder l'analyse en composantes principales par le biais de la fonction `dudi.pca()` du paquet `ade4`.

Table des matières

1 Deug : centrage et réduction

Considérons 104 étudiants ayant passé 9 matières pour leur examen. Chaque individu (ici un étudiant) est caractérisé par un point dans $\mathbb{R}^9$. Chaque variable (ici une matière) est caractérisée par un point dans $\mathbb{R}^{104}$. Nous allons avoir besoin ici très clairement de méthodes de réduction dimensionnelle, telles que l'ACP, pour comprendre la structure des données. Il y a ici au moins une variable d'intérêt évidente qui est la décision finale du jury d'examen : l'étudiant λ a-t-il été oui ou non reçu, et avec quels honneurs ? L'ACP nous permet-elle de prédire le résultat final à partir des données brutes ? Peut elle nous en dire plus ? Commencer par récupérer les données[1] :

```
library(ade4)
data(deug)
class(deug)
```

```
[1] "list"
```

```
names(deug)
```

```
[1] "tab"    "result" "cent"
```

Il s'agit donc d'une liste à trois composantes : `tab`, `result` et `cent`. Examinons les de plus près.

1.1 deug$tab

Ce sont les données brutes :

1. Algèbre et Analyse des données (sur 100)
2. Analyse (sur 60)
3. Probabilités (sur 80)
4. Informatique (sur 60)
5. Dominante (Sociologie ou Économie sur 120)
6. Options (sur 40)
7. Ouvertures (sur 40)
8. Anglais (sur 40)
9. Education physique et sportive (bonification <=15)

```
class(deug$tab)
```

```
[1] "data.frame"
```

```
dim(deug$tab)
```

```
[1] 104   9
```

```
names(deug$tab)
```

```
[1] "Algebra"    "Analysis"   "Proba"      "Informatic" "Economy"    "Option1"
[7] "Option2"    "English"    "Sport"
```

```
head(deug$tab)
```

```
  Algebra Analysis Proba Informatic Economy Option1 Option2 English Sport
1      40     26.0    26       26.0    51.9      17      24    19.0  11.5
2      37     34.5    37       32.0    72.0      24      22    26.0  11.5
3      37     41.0    29       34.5    72.0      24      27    19.6  11.5
4      63     37.5    57       35.5    77.4      23      23    21.0  14.0
5      55     31.5    34       36.0    57.9      19      24    24.0  11.5
6      50     38.0    32       20.0    66.9      20      15    22.2   0.0
```

[1] Il s'agit de données réelles concernant des étudiants en DEUG MASS. DEUG est un acronyme pour Diplôme d'Études Universitaires Générales, correspondant à un L2 dans le cadre du LMD, et MASS à un acronyme pour Mathématiques Appliquées aux Sciences Sociales

Ce `data.frame` contient les données brutes : combien de points chaque étudiant a-t-il obtenu pour chaque matière. Quel étudiant a obtenu le plus de points en algèbre ?

```
with(deug$tab, which.max(Algebra))
```
```
[1] 86
```

Donner l'étudiant qui a obtenu le plus de points en sport :

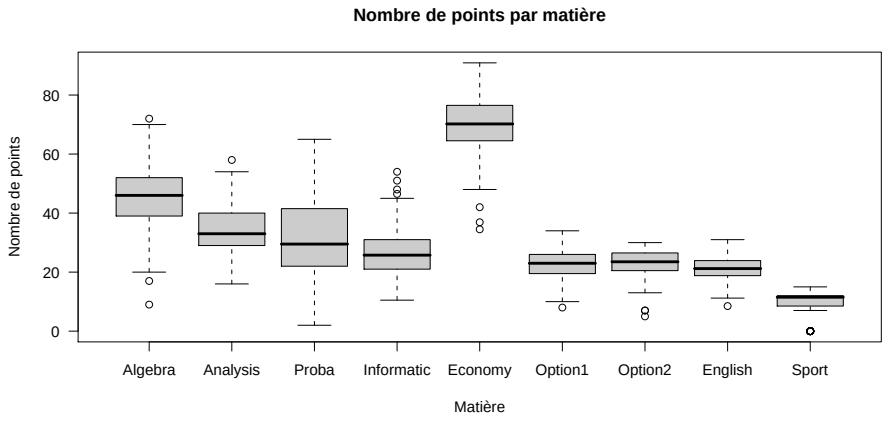
```
[1] 15
```

Donner l'étudiant qui a obtenu le moins de points en économie :

```
[1] 72
```

Représenter graphiquement les données :

```
boxplot(deug$tab, main = "Nombre de points par matière", las = 1,
    col = grey(0.8), xlab = "Matière", ylab = "Nombre de points")
```



1.2 deug$result

C'est la décision finale du jury :

D- Éliminé après les épreuves écrites

C- Éliminé après l'oral de rattrapage

B- Admis sans mention après l'oral de rattrapage

B Admis avec la mention Passable

A Admis avec la mention Assez Bien.

A+ Admis avec la mention B

```
deug$result
```
```
 [1] C- B  B  A  B  C- B  B  D  A  B  B  A  C- B  B  A  D  B  A+ A  B  B  B  B  B
[27] C- B  B  B  B  B  B  B  B  C- D  D  D  B  B  B  A  B  C- B  B  B  B  B  B  D
[53] A+ A  A  C- B  A  C- A  B  B  B  C- C- D  B- C- C- A  B  D  A  B  B  C- C- C-
[79] A  C- C- C- B  B  B  A  B  A  A  D  D  B- C- C- D  A  B  B  B  B  B  B  B  B
Levels: D A A+ C- B- B
```

C'est donc une variable qualitative non ordonnée. On décide de la tranformer en une variable qualitative ordonnée pour mettre un peu d'ordre.

```
deug$result <- factor(deug$result, levels = c("D", "C-", "B-", "B",
    "A", "A+"), order = TRUE)
deug$result
```

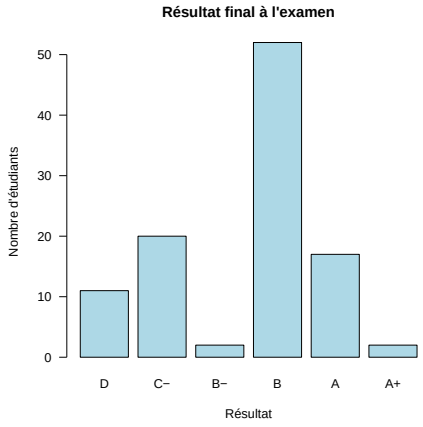
```
  [1] C- B  B  A  B  C- B  B  D  A  B  B  A  C- B  B  A  D  B  A+ A  B  B  B  B  B
 [27] C- B  B  B  B  B  B  B  B  C- D  D  D  B  B  B  A  B  C- B  B  B  B  B  B  D
 [53] A+ A  A  C- B  A  C- A  B  B  B  C- C- D  B- C- C- A  B  D  A  B  B  C- C- C-
 [79] A  C- C- C- B  B  B  A  B  A  A  D  D  B- C- C- D  A  B  B  B  B  B  B  B  B
Levels: D < C- < B- < B < A < A+
```

C'est bien plus clair ainsi, les plus mauvais résultats correspondent à la modalité D et les meilleurs à la modalité A+. Regardons comment se répartissent nos 104 étudiants.

```
table(deug$result)
```

```
 D C- B-  B  A A+
11 20  2 52 17  2
```

```
barplot(table(deug$result), lend = "butt", las = 1, main = "Résultat final à l'examen",
    ylab = "Nombre d'étudiants", xlab = "Résultat", col = "lightblue")
```



1.3 deug$cent

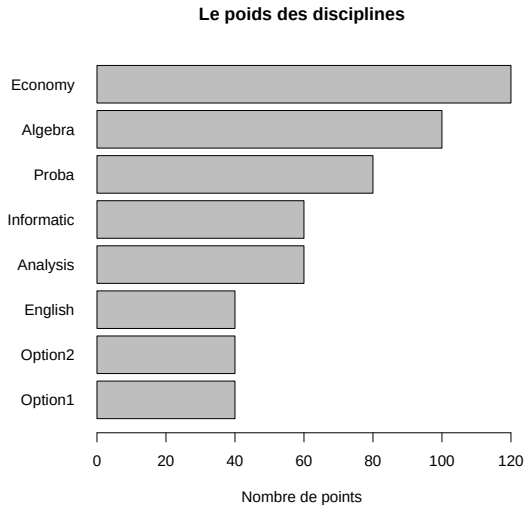
Encore faut-il avoir moyenne, `deug$cent` donne le nombre de points qu'il faut pour avoir la moyenne dans chaque matière :

```
deug$cent
```

```
  Algebra  Analysis     Proba Informatic   Economy   Option1   Option2
       50        30        40        30        60        20        20
  English     Sport
       20         0
```

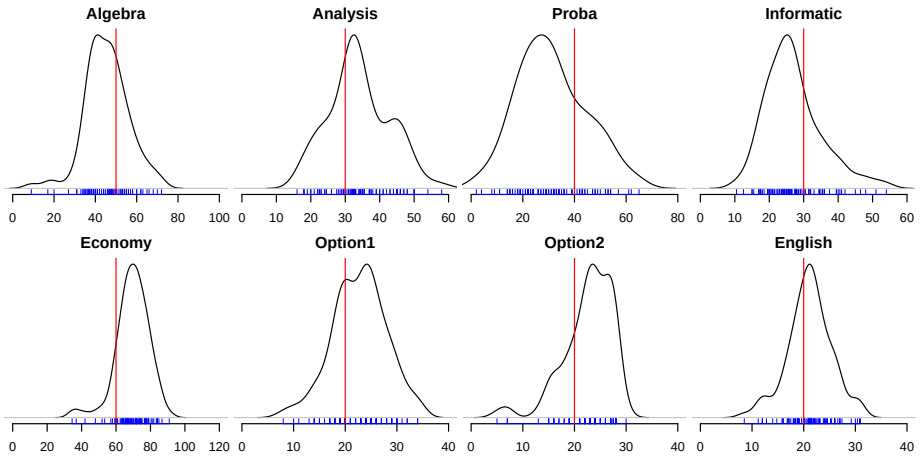
Ceci nous permet d'apprécier le poids des différentes disciplines, on ne regarde pas le sport qui a un rôle un peu à part.

```
coefs <- 2 * deug$cent[1:8]
par(mar = c(5, 5, 4, 2) + 0.1)
barplot(sort(coefs), horiz = T, las = 1, xlab = "Nombre de points",
    main = "Le poids des disciplines")
```



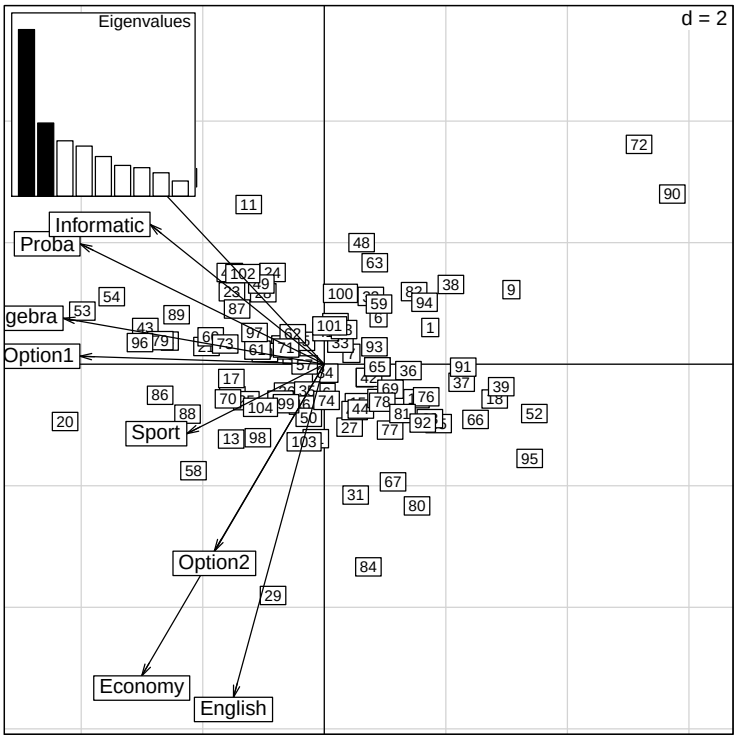
Nous pouvons maintenant examiner la distribution des notes pour chaque discipline en ajoutant en rouge la barre fatidique pour avoir la moyenne.

```
par(mfrow = c(2, 4), mar = c(2, 0, 2, 0.2) + 0.1)
for (i in 1:8) {
    plot(density(deug$tab[, i]), xlim = c(0, 2 * deug$cent[i]),
        main = colnames(deug$tab)[i], yaxt = "n", bty = "n")
    rug(deug$tab[, i], col = "blue")
    abline(v = deug$cent[i], col = "red")
}
```



1.4 Une ACP de deug$tab

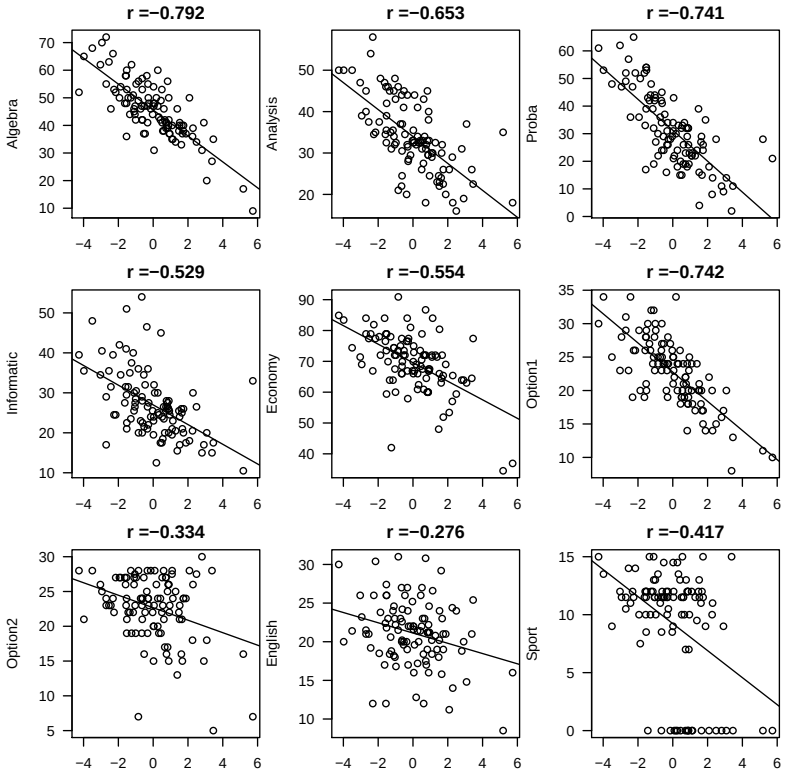
```
dudideug <- dudi.pca(deug$tab, scannf = F)
scatter(dudideug)
```



1.5 La cohérence du jury

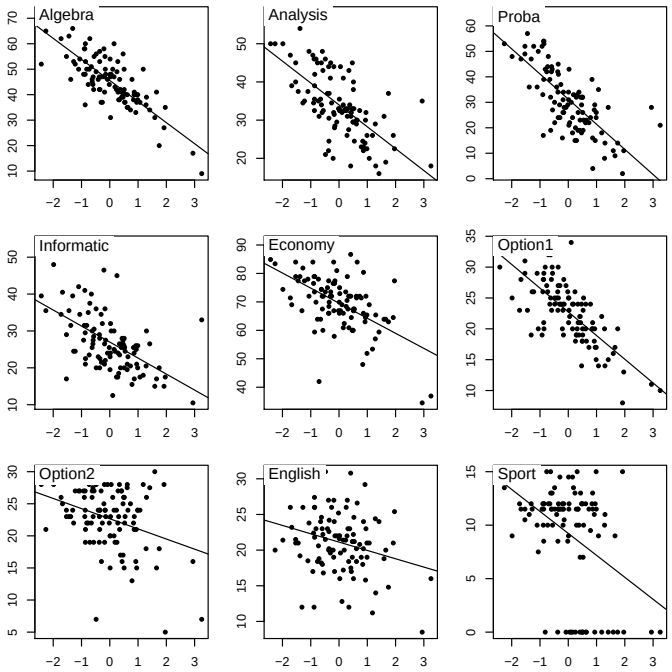
La question est : peut-on caractériser la cohésion des notations si tant est qu'elle existe. Exécuter et interpreter une analyse en composantes principales normée. Définir et interpréter le graphe canonique. Le raisonnement suppose l'existence d'une variable cachée (*latent variable*), la valeur de l'étudiant en général, qui explique une bonne part des résultats obtenus dans chaque matière. Les professeurs sont des variables d'enregistrement de cette variable cachée avec une certaine erreur admise pour chaque étudiant, erreur de mesure (le professeur se trompe dans l'addition, son sujet est plus ou moins pittoresque, il est plus ou moins indulgent suivant que ...) ou erreur propre à l'individu (les frites sont plus ou moins digestes, le rhum plus ou moins tenace, on aime plus ou moins l'analyse par rapport à l'algèbre !). La variable cachée est la première coordonnée des lignes. D'où le graphe canonique qui exprime ce point de vue : compare les coordonnées des individus sur le premier axe avec les variables de départ.

```
par(mfrow = c(3, 3), mar = c(2, 4, 2, 0) + 0.1)
for (i in 1:9) {
    x <- dudideug$li[, 1]
    y <- deug$tab[, i]
    titre <- paste("r =", round(cor(x, y), digits = 3), sep = "")
    plot(x, y, las = 1, ylab = colnames(deug$tab)[i], main = titre)
    abline(lm(y ~ x))
}
```



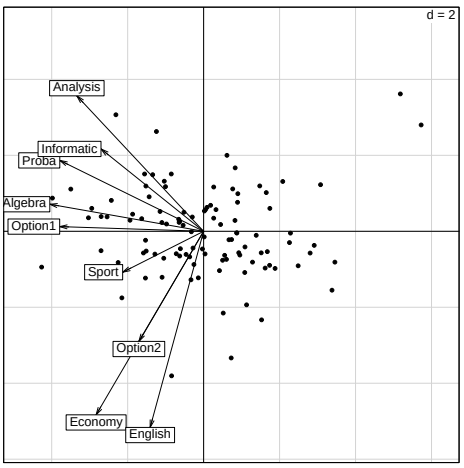
L'ACP donne le score qui maximise la somme des carrés de corrélation avec les variables. Ce graphe canonique est très utile pour aider à l'interprétation des axes, c'est pourquoi il existe une fonction, `score()`, pour faire ce graphe en routine :

```
score(dudideug)
```



Les deux graphes sont identiques. On ne travaille que sur les carrés de corrélation et le signe du score est aléatoire. On voit bien que tous les professeurs enregistrent une ordination générale des étudiants [2]. Autour de cette prédiction par la variable cachée (on appelle aussi cette opération une analyse factorielle, une recherche de facteur, dont le plus célèbre est le QI, par le biais de la qualité de la prédiction) la variabilité est encore forte. Pour le voir, on prendra le plan des deux premiers axes principaux.

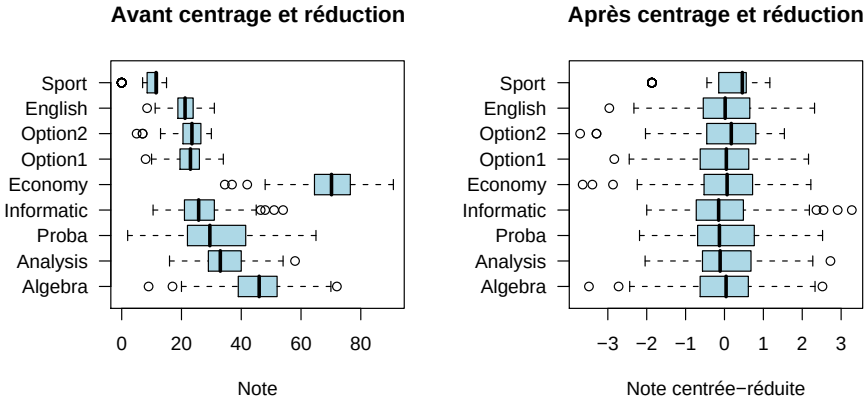
```
s.label(dudideug$li, clab = 0)
s.arrow(9 * dudideug$c1, add.p = T)
```



[2] même celui d'éducation physique qui ne met que des points de bonification sur la présence : les étudiants faibles en maths ne vont pas en gym, corrélation n'est pas causalité.

Identifier les points et expliquer l'information produite par ce graphe. On a projeté sur un plan 102 points de $\mathbb{R}^9$ et les 9 vecteurs de la base canonique. Le centrage a amené le nuage à l'origine. ***Les différences de moyenne entre matières et les différences de variance sont éliminées par la normalisation.***

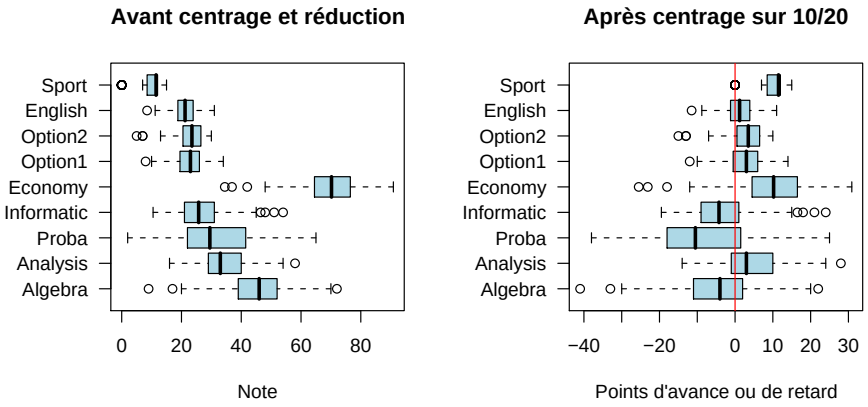
```
par(mfrow = c(1, 2), mar = c(5, 5, 4, 2) + 0.1)
boxplot(deug$tab, horizontal = T, las = 1, main = "Avant centrage et réduction",
    xlab = "Note", col = "lightblue")
boxplot(as.data.frame(scale(deug$tab)), horizontal = T, las = 1,
    main = "Après centrage et réduction", xlab = "Note centrée-réduite",
    col = "lightblue")
```



1.6 Un point de vue très différent

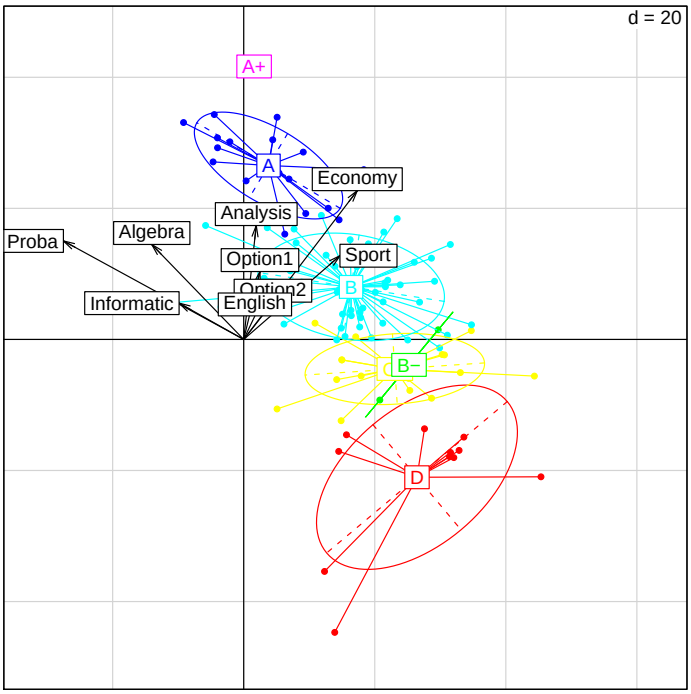
En général, les étudiants ne s'occupent que de leurs affaires et du nombre de points qui les séparent de la barre fixée *a priori*, en général 10/20. La composante `deug$cent` donne, pour chaque matière, le nombre de points correspondant à la moyenne de 10/20, toutes différences se comptant en points d'avance ou en points de retard.

```
par(mfrow = c(1, 2), mar = c(5, 5, 4, 2) + 0.1)
boxplot(deug$tab, horizontal = T, las = 1, main = "Avant centrage et réduction",
    xlab = "Note", col = "lightblue")
boxplot(as.data.frame(scale(deug$tab, center = deug$cent, scale = FALSE)),
    horizontal = T, las = 1, main = "Après centrage sur 10/20",
    xlab = "Points d'avance ou de retard", col = "lightblue")
abline(v = 0, col = "red")
```



Les arguments de la fonction `dudi.pca()` permettent de préciser le centrage et la réduction que l'on veut effecter.

```
dudidec <- dudi.pca(deug$tab, scal = FALSE, center = deug$cent,
    scan = FALSE)
s.class(dudidec$li, deug$result, col = rainbow(6))
s.arrow(40 * dudidec$c1, add.plot = TRUE)
```



On y gagne la distinction entre les matières qui donnent la posistion du nuage (premier axe) et celles qui donnent sa forme (deuxième axe). On retiendra l'existence de fonctions voisines avec un point de vue particulier.

1.7 Exercice

```
data(seconde)
head(seconde)
```

```
  HGEO FRAN PHYS MATH BIOL ECON ANGL ESPA
1 11.6  8.7  4.5  7.6  9.0  6.5   10 15.5
2 13.6 12.3  6.2  8.5 12.0 11.5    7 12.0
3 13.2 12.1  8.5  6.3 11.6 11.0   13 14.5
4  8.8  8.2  5.0  3.7 10.6 10.0   11 14.5
5 12.7  8.6 10.0  9.3 10.6 14.0    9 13.5
6 12.4 10.0  5.5  9.2 10.5 11.0   10 15.0
```

Analyser ce jeu de données.

2 Décathlon : le signe des corrélations

L'exemple porte le numéro 357 dans l'ouvrage de Hand et al. (1994) [1]. On a le résultat du décathlon masculin des jeux olympiques de 1988. Chaque individu (ici un athlète) est caractérisé par 10 variables correpondant à sa performance dans les 10 épreuves suivantes :

1. course de 100 mètres
2. saut en longueur
3. lancer du poids
4. saut en hauteur
5. course de 400 mètres
6. course du 110 m haies
7. lancer du disque
8. saut à la perche
9. lancer du javelot
10. course de 1500 mètres

Ces résultats sont utilisés pour calculer un score final en suivant le barème du décathlon, l'individu ayant le score le plus grand gagne la compétition. Importer les données dans R :

```
olympic <- read.table(url("http://pbil.univ-lyon1.fr/R/donnees/olympic.txt"))
names(olympic) <- c("100", "long", "poid", "haut", "400", "110",
    "disq", "perc", "jave", "1500", "score")
head(olympic)
```

```
    100 long  poid haut   400   110  disq perc  jave   1500 score
1 11.25 7.43 15.48 2.27 48.90 15.13 49.28  4.7 61.32 268.95  8488
2 10.87 7.45 14.97 1.97 47.71 14.46 44.36  5.1 61.76 273.02  8399
3 11.18 7.44 14.20 1.97 48.29 14.81 43.66  5.2 64.16 263.20  8328
4 10.62 7.38 15.02 2.03 49.06 14.72 44.80  4.9 64.04 285.11  8306
5 11.02 7.43 12.92 1.97 47.44 14.40 41.20  5.2 57.46 256.64  8286
6 10.83 7.72 13.58 2.12 48.34 14.18 43.06  4.9 52.18 274.07  8272
```

Sachant que le résultat des 10 épreuves est exprimé dans le système d'unité international, préciser pour chaque épreuve :

1. Les unités employées
2. Le domaine du possible ($\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{R}_-$, $\mathbb{R}^*$, $\mathbb{R}^*_+$, $\mathbb{R}^*_-$)
3. S'il vaut mieux avoir avoir une valeur grande ou petite pour augmenter le score final

Lisez la documentation de `?which.min` et `?which.max`. Que vous apprend le résultat suivant :

```
apply(olympic, 2, which.min)
```

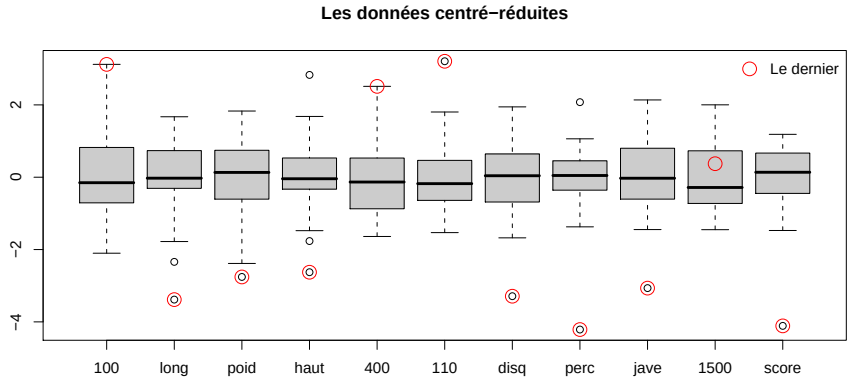
```
 100  long  poid  haut   400   110  disq  perc  jave  1500 score
   4    34    34    34     5     6    34    34    34     5    34
```

```
apply(olympic, 2, which.max)
```

```
 100  long  poid  haut   400   110  disq  perc  jave  1500 score
  34     6    18     1    34    34    17     7    17    28     1
```

Représenter les données après centrage et réduction par colonne :

```
dcr <- as.data.frame(scale(olympic))
bxr <- boxplot(dcr, main = "Les données centré-réduites", col = grey(0.8))
points(seq_len(ncol(olympic)), dcr[nrow(olympic), ], col = "red",
    cex = 2)
legend("topright", legend = "Le dernier", pch = 1, col = "red",
    pt.cex = 2, bty = "n")
```



Pourquoi faut-il enlever le dernier individu et supprimer la dernière colonne ?

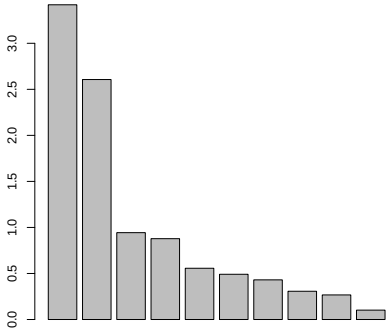
```
olympic <- olympic[-34, ]
olyres <- olympic$score
olympic$score <- NULL
names(olympic)
```

```
[1] "100"  "long" "poid" "haut" "400"  "110"  "disq" "perc" "jave" "1500"
```

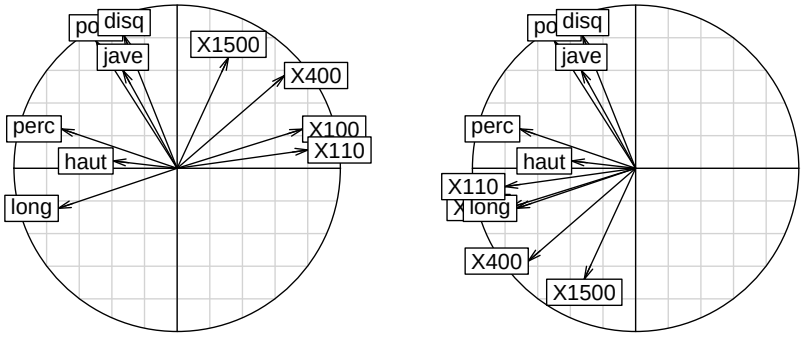
```
pca1 <- dudi.pca(olympic, scannf = FALSE)
```

On sélectionne le nombre d'axes à partir du graphe des valeurs propres

```
barplot(pca1$eig)
```



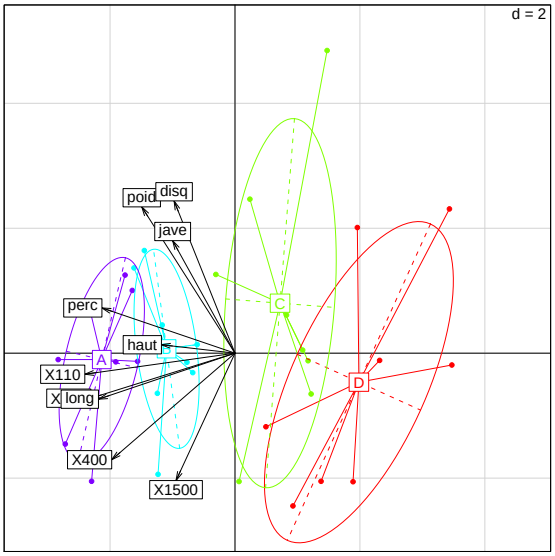

```
par(mfrow = c(1, 2))
s.corcircle(pca1$co)
olympic2 <- olympic
olympic2[, c(1, 5, 6, 10)] = -olympic2[, c(1, 5, 6, 10)]
pca2 <- dudi.pca(olympic2, scan = F)
s.corcircle(pca2$co)
```



L'image de gauche est particulièrement trompeuse ! Elle est statistiquement juste et expérimentalement fausse. La performance des athlètes augmente avec les distances des lancers, la hauteur et la longueur des sauts, elle décroît avec le temps des courses. L'image de droite est mathématiquement équivalente et expérimentalement correcte.

On peut utiliser le score comme variable illustrative en définissant des groupes de niveau :

```
groupe <- cut(olyres, quantile(olyres))
levels(groupe) <- c("D", "C", "B", "A", order = T)
s.class(pca2$li, groupe, col = rainbow(4))
s.arrow(3 * pca2$co, add.plot = T)
```



3 Rock

Récupérer le fichier `rock.txt`.

```
rock <- read.table(url("http://pbil.univ-lyon1.fr/R/donnees/rock.txt"))
rock
```

```
   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10
1   7  3  9 10  6  8  4  2  1   5
2   4  8  7  1  9  6  2  3  5  10
3   7  6  8  3  2  1  4  9 10   5
4   6  7  3  5  2  9 10  8  1   4
5   1  2  3  5  8  4  7 10  6   9
6   1  5  6  3  2  7  8  4  9  10
7   2  6 10  5  7  3  8  1  4   9
8   6  7  9  5  3  2  8 10  1   4
9   7  6  5  9  2  1  3  8 10   4
10  5  7  6  3  9  1  4 10  2   8
11  6  7  5  2  3  1  4 10  9   8
12  7  5  6  9  3  2 10  8  4   1
13  6  7  3  5  2  1  4  8  9  10
14  7  2 10  5  3  1  6  8  9   4
15  6  7 10  1  2  5  3  4  8   9
16  6 10  7  3  2  1  5  4  8   9
17  3  7 10  2  6  5  9  8  1   4
18  8  7  9  6  5  3  1  2  4  10
19  5  7  3  8  6  9  4  1  2  10
20  4  6  3  5  8  9  7  2  1  10
21  6  8  7  5  3  2  4  9  1  10
22  6  7  5  9  8  4  3  1  2  10
23  6  7  9  5  8  3  4  1  2  10
24  6  3  7  1  5  2  8  4  9  10
```

On a demandé à 24 étudiants de ranger par ordre de préférence 10 groupes de musique. La personne 1 (première ligne) préfère le 7, ensuite le 3, ensuite le 9, ensuite le 10, ... enfin le 5. Le code des groupes (figure 1) est dans l'ordre : *Metallica*, *Guns n' Roses*, *Nirvana*, *AC/DC*, *Noir Désir*, *U2*, *Pink Floyd*, *Led Zeppelin*, *Deep Purple* et *Bon Jovi*.

L'ordre de préférence n'est pas une donnée directement accessible à l'analyse statistique. Il convient d'en faire une note de qualité, prenant la valeur maximale pour le premier et minimale pour le dernier.

```
a <- c(2, 5, 3, 1, 4)
names(a) <- letters[a]
a
```

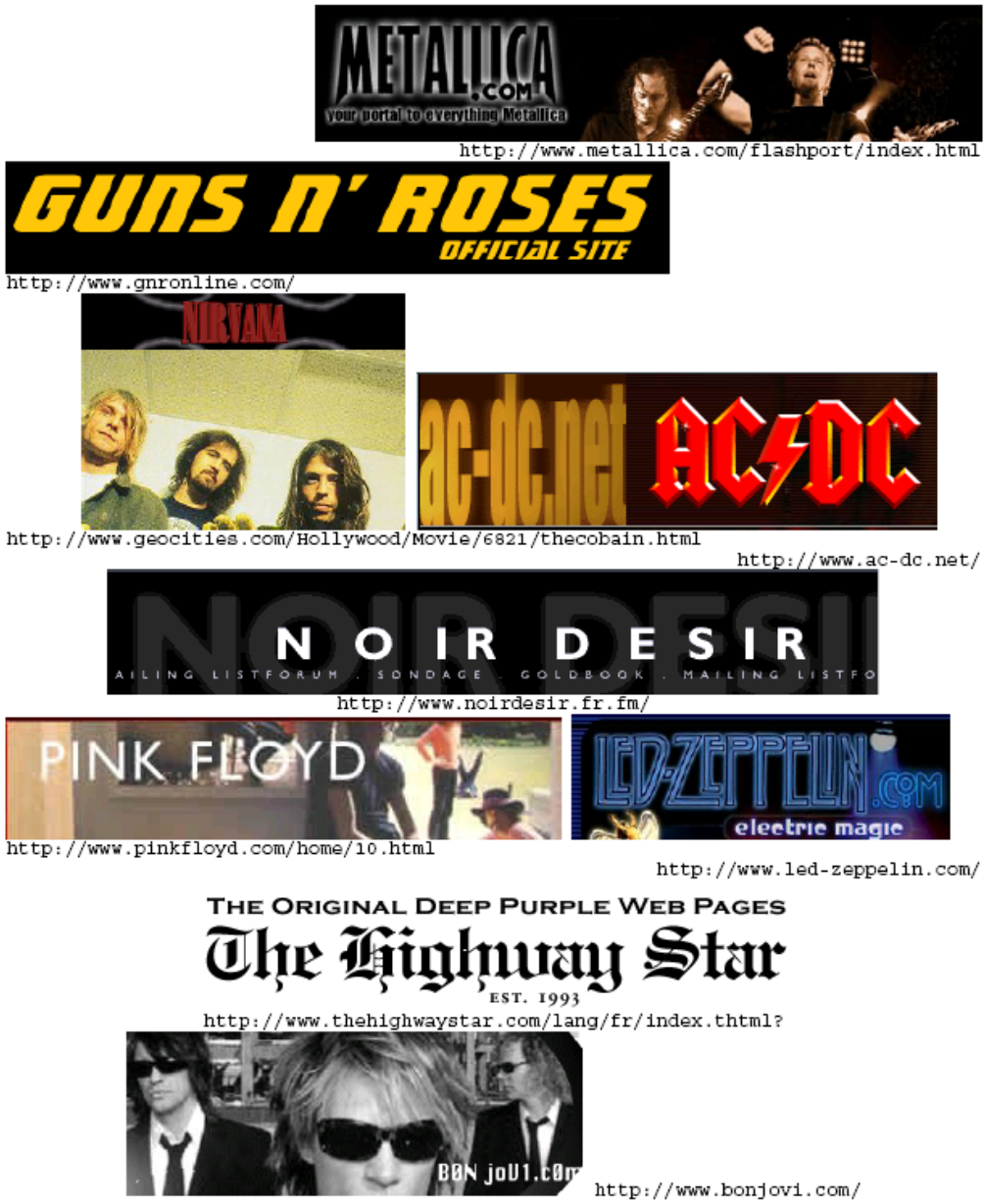


FIG. 1 – Quelques images des groupes disponibles au moment de la première version de la fiche. Document non contractuel !

```
b e c a d
2 5 3 1 4
```

```
order(a)
```

```
[1] 4 1 3 5 2
```

```
a[order(a)]
```

```
a b c d e
1 2 3 4 5
```

```
3 - order(a)
```

```
[1] -1  2  0 -2  1
```

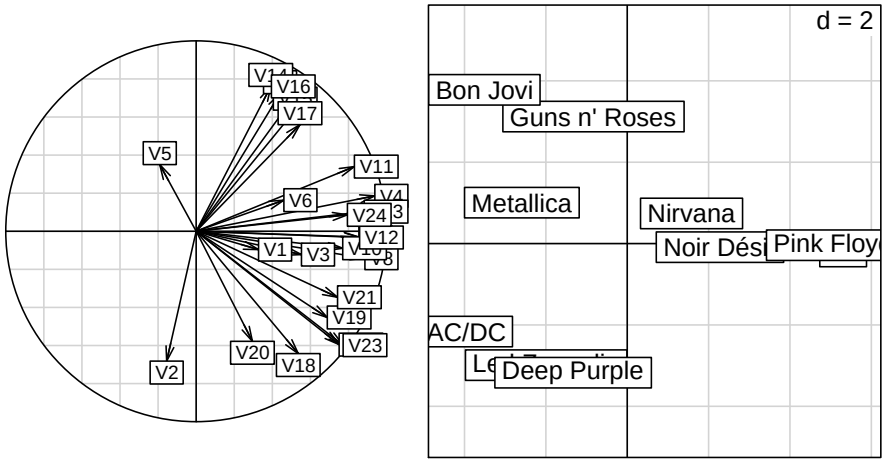
Le préféré 2 se retrouve avec la meilleure note 2 et le rejeté 4 se retrouve avec la plus mauvaise note (−2).

```
r <- as.data.frame(apply(rock, 1, function(x) (length(x) + 1)/2 -
    order(x)))
row.names(r) <- c("Metallica", "Guns n' Roses", "Nirvana", "AC/DC",
    "Noir Désir", "U2", "Pink Floyd", "Led Zeppelin", "Deep Purple",
    "Bon Jovi")
r
```

```
                V1   V2   V3   V4   V5   V6   V7   V8   V9  V10  V11  V12  V13  V14
Metallica     -3.5  1.5 -0.5 -3.5  4.5  4.5 -2.5 -3.5 -0.5 -0.5 -0.5 -4.5 -0.5 -0.5
Guns n' Roses -2.5 -1.5  0.5  0.5  3.5  0.5  4.5 -0.5  0.5 -3.5  1.5 -0.5  0.5  3.5
Nirvana        3.5 -2.5  1.5  2.5  2.5  1.5 -0.5  0.5 -1.5  1.5  0.5  0.5  2.5  0.5
AC/DC         -1.5  4.5 -1.5 -4.5 -0.5 -2.5 -3.5 -4.5 -4.5 -1.5 -1.5 -3.5 -1.5 -4.5
Noir Désir    -4.5 -3.5 -4.5  1.5  1.5  3.5  1.5  1.5  2.5  4.5  2.5  3.5  1.5  1.5
U2             0.5 -0.5  3.5  4.5 -3.5  2.5  3.5  4.5  3.5  2.5  4.5  2.5  4.5 -1.5
Pink Floyd     4.5  2.5  4.5  3.5 -1.5 -0.5  0.5  3.5  4.5  3.5  3.5  4.5  3.5  4.5
Led Zeppelin  -0.5  3.5  2.5 -2.5  0.5 -1.5 -1.5 -1.5 -2.5 -4.5 -4.5 -2.5 -2.5 -2.5
Deep Purple    2.5  0.5 -2.5 -0.5 -4.5 -3.5 -4.5  2.5  1.5  0.5 -3.5  1.5 -3.5 -3.5
Bon Jovi       1.5 -4.5 -3.5 -1.5 -2.5 -4.5  2.5 -2.5 -3.5 -2.5 -2.5 -1.5 -4.5  2.5
                V15  V16  V17  V18  V19  V20  V21  V22  V23  V24
Metallica       1.5 -0.5 -3.5 -1.5 -2.5 -3.5 -3.5 -2.5 -2.5  1.5
Guns n' Roses  0.5  0.5  1.5 -2.5 -3.5 -2.5 -0.5 -3.5 -3.5 -0.5
Nirvana       -1.5  1.5  4.5 -0.5  2.5  2.5  0.5 -1.5 -0.5  3.5
AC/DC         -2.5 -2.5 -4.5 -3.5 -1.5  4.5 -1.5 -0.5 -1.5 -2.5
Noir Désir    -0.5 -1.5 -0.5  0.5  4.5  1.5  1.5  2.5  1.5  0.5
U2             4.5  4.5  0.5  1.5  0.5  3.5  4.5  4.5  4.5  4.5
Pink Floyd     3.5  2.5  3.5  3.5  3.5 -1.5  2.5  3.5  3.5  2.5
Led Zeppelin  -3.5 -3.5 -2.5  4.5  1.5  0.5  3.5  0.5  0.5 -1.5
Deep Purple   -4.5 -4.5 -1.5  2.5 -0.5 -0.5 -2.5  1.5  2.5 -3.5
Bon Jovi       2.5  3.5  2.5 -4.5 -4.5 -4.5 -4.5 -4.5 -4.5 -4.5
```

Quel est le groupe le plus apprécié ?

```
dudi2 <- dudi.pca(r, scale = T, scannf = F)
par(mfrow = c(1, 2))
s.corcircle(dudi2$co, clab = 0.75)
s.label(dudi2$li)
```



Vous pouvez refaire cet exercice avec 51 étudiants (il s'agit des mêmes groupes) à partir de :

```
data(rankrock)
```

Pour un autre exemple et savoir comment mettre la photo de son groupe préféré sur la carte factorielle, voir : `http://pbil.univ-lyon1.fr/R/querep/qrg.pdf`

Références

[1] D.J. Hand, F. Daly, A.D. Lunn, K.J. McConway, and E. Ostrowski. *A handbook of small data sets*. Chapman & Hall, London, 1994.