Fiche de Biostatistique

# Analyses en composantes principales

D. Chessel, A.B. Dufour & J. Thioulouse

## Résumé

La fiche décrit le principe général d'un schéma de dualité et passe en revue quelques usages de l'analyse en composantes principales.

## Plan

1 SCHEMA DE DUALITE ..... 2
1.1 Rang d'un schéma ..... 3
1.2 Valeurs propres d'un schéma ..... 3
1.3 Vecteurs principaux d'un schéma ..... 4
1.4 Propriétés d'optimalité ..... 6
1.5 Approximations du tableau ..... 7
2 GRAPHES CANONIQUES ..... 8
2.1 Procédure centrale ..... 8
2.2 Décentrage ..... 11
2.3 Redondance ..... 17
2.4 Valeurs propres ..... 21
2.5 Profils et biplot ..... 25
2.6 Tableaux homogènes ..... 32
3 MODELES PROBABILISTES ..... 35

# 1 Schéma de dualité

On connaît la pratique de l'ACP centrée ou normée décrite dans le chapitre "Géométrie de l'espace des variables". Il s'agit d'un cas particulier d'une classe d'objets qu'on appelle schémas de dualité. On en donne ici une version simplifiée sans insister sur les démonstrations pour en faciliter l'usage [1]. Un schéma est constitué de trois éléments donnant un triplet $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$.

**X** est une matrice de données en général issue d'une matrice de données brutes **Y** à l'aide d'une transformation préalable. **X** a *n* lignes et *p* colonnes. Les *n* lignes de **X** sont des vecteurs de $\mathbb{R}^p$ tandis que les *p* colonnes de **X** sont des vecteurs de $\mathbb{R}^n$. $\mathbf{X}^t$ est la transposée de la matrice **X**, elle a *p* lignes et *n* colonnes.

**Q** est la matrice d'un produit scalaire de $\mathbb{R}^p$, soit une matrice carrée symétrique qui définit la fonction :

$$(\mathbf{x}, \mathbf{y}) = \left( \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix}, \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} \right) \in \mathbb{R}^p \times \mathbb{R}^p \mapsto \mathbf{x}^t \mathbf{Q} \mathbf{y} = \langle \mathbf{x} | \mathbf{y} \rangle_{\mathbf{Q}} \in \mathbb{R}$$

**D** est la matrice d'un produit scalaire de $\mathbb{R}^n$, soit une matrice carrée symétrique qui définit la fonction :

$$(\mathbf{x}, \mathbf{y}) = \left( \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \right) \in \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbf{x}^t \mathbf{D} \mathbf{y} = \langle \mathbf{x} | \mathbf{y} \rangle_{\mathbf{D}} \in \mathbb{R}$$

L'essentiel est que ces quatre matrices s'assemblent car les produits :

$$\mathbf{XQ}, \mathbf{QX}^t, \mathbf{X}^t\mathbf{D} \text{ et } \mathbf{DX}$$

ont un sens. On les dispose dans un schéma dit schéma de dualité :

$$\begin{array}{ccc} & \mathbf{Q} & \\ \mathbb{R}^p & \rightarrow & \mathbb{R}^{p^*} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X} \\ \mathbb{R}^{n^*} & \leftarrow & \mathbb{R}^n \\ & \mathbf{D} & \end{array}$$

Rigoureusement, $\mathbb{R}^{p^*}$ est le dual de $\mathbb{R}^p$ (ensemble des applications linéaires de $\mathbb{R}^p$ dans $\mathbb{R}$), $\mathbb{R}^{n^*}$ est le dual de $\mathbb{R}^n$ (ensemble des applications linéaires de $\mathbb{R}^n$ dans $\mathbb{R}$),

[1] Escoufier, Y. (1987) The duality diagramm : a means of better practical applications. In : Development in numerical ecology. Legendre, P. & Legendre, L. (Eds.) NATO advanced Institute , Serie G .Springer Verlag, Berlin. 139-156.

**Q** est vue comme la matrice d'une application linéaire définie par $\left(\mathbf{Q}(\mathbf{x})\right)(\mathbf{y}) = \langle \mathbf{x}|\mathbf{y} \rangle_{\mathbf{Q}}$, **D** est vue comme la matrice d'une application linéaire définie par $\left(\mathbf{D}(\mathbf{x})\right)(\mathbf{y}) = \langle \mathbf{x}|\mathbf{y} \rangle_{\mathbf{D}}$ et $\mathbf{X}^t$ est vue comme la matrice d'une application linéaire définie à l'aide du bidual. Pour l'utilisateur, la simplification :

$$(\mathbf{X},\mathbf{Q},\mathbf{D}) \Leftrightarrow \begin{array}{ccc} & \mathbf{Q} & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D} & \end{array}$$

suffit pour se souvenir que les produits de matrices :

$$\mathbf{X}^t\mathbf{DXQ},\ \mathbf{DXQX}^t,\ \mathbf{XQX}^t\mathbf{D} \text{ et } \mathbf{QX}^t\mathbf{DX}$$

ont un sens. On définit deux matrices $\mathbf{V} = \mathbf{X}^t\mathbf{DX}$ et $\mathbf{W} = \mathbf{XQX}^t$ qui s'insèrent par :

$$\begin{array}{ccc} & \mathbf{Q} & \\ \boxed{p} & \rightarrow & \boxed{p} \\ & \leftarrow & \\ & \mathbf{V} & \end{array} \qquad \begin{array}{ccc} & \mathbf{W} & \\ \boxed{n} & \rightarrow & \boxed{n} \\ & \leftarrow & \\ & \mathbf{D} & \end{array}$$

$\mathbf{VQ}$ et $\mathbf{WD}$ sont les opérateurs d'Escoufier. Le schéma a des propriétés théoriques très générales qui prennent des significations propres lorsqu'on utilise un ensemble de paramètres originaux. En particulier les ACP classiques, l'analyse des correspondances (AFC), les analyses de correspondances non symétriques (ANSC), les analyses discriminantes (AFD), l'analyse canonique (AC), les analyses de co-inertie (ACO), les analyses sur variables instrumentales (ACPVI), l'analyse des correspondances multiples (ACM), l'analyse factorielle multiple (AFM) et diverses extensions sont des conséquences directes des propriétés générales.

## 1.1 Rang d'un schéma

Le rang d'une matrice est la dimension du sous-espace engendré par ses lignes ou ses colonnes. Les matrices $\mathbf{X}$, $\mathbf{X}^t$, $\mathbf{VQ}$ et $\mathbf{WD}$ ont même rang, lequel est inférieur ou égal au nombre de lignes de $\mathbf{X}$ ($n$) et au nombre de colonnes de $\mathbf{X}$ ($p$). Appelons $r$ le rang commun ou rang du schéma.

## 1.2 Valeurs propres d'un schéma

Les matrices $\mathbf{VQ}$ et $\mathbf{WD}$ sont toujours diagonalisables. La première ( $p \times p$ ) a $p$ valeurs propres distinctes ou confondues dont les $r$ premières sont non nulles. Ces valeurs propres sont positives et rangées par ordre décroissant :

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_p = 0$$

La seconde ( $n \times n$ ) a $n$ valeurs propres distinctes ou confondues dont les $r$ premières sont non nulles. Ces valeurs propres sont positives et rangées par ordre décroissant :

$$\mu_1 \geq \mu_2 \geq \cdots \geq \mu_r > \mu_{r+1} = \cdots = \mu_n = 0$$

Les deux ensembles de valeurs propres non nulles sont identiques et on a toujours :

$$\mu_1 = \lambda_1, \mu_2 = \lambda_2, \cdots, \mu_r = \lambda_r$$

En toute généralité, il est possible d'avoir des valeurs propres non nulles multiples, mais dans la pratique, ne serait-ce que pour des raisons numériques, ce cas est exclu. On alors une seule série de valeurs propres non nulles qu'on appelle les valeurs propres du schéma :

$$\lambda_1 > \lambda_2 > \cdots > \lambda_r > 0$$

## 1.3 Vecteurs principaux d'un schéma

Les matrices **VQ** et **WD** sont toujours diagonalisables et ont même valeurs propres non nulles. La première est **Q**-symétrique (la matrice **QVQ** est égale à sa transposée) et à ce titre définit une base de vecteurs propres **Q**-orthonormée. Donc, il existe une matrice **A** telle que :

$$\mathbf{VQA} = \mathbf{A}\Lambda \text{ et } \mathbf{A}^t\mathbf{QA} = \mathbf{I}_p$$

$$\Lambda = Diag\left(\lambda_1, \lambda_2, \cdots, \lambda_p\right)$$

On ne s'intéresse qu'aux *r* premiers (valeurs propres non nulles) qui sont définis de façon unique (valeurs propres distinctes) :

$$\mathbf{VQA}_r = \mathbf{A}_r\Lambda_r \text{ et } \mathbf{A}_r^t\mathbf{QA}_r = \mathbf{I}_r$$

$\mathbf{A}_r$ est une matrice à *p* lignes et *r* colonnes, chacune de ses colonnes étant un vecteur propre **Q**-normé de $\mathbb{R}^p$. $\Lambda_r$ est la matrice diagonale qui contient les valeurs propres.

La seconde est **D**-symétrique (la matrice **DWD** est égale à sa transposée) et à ce titre définit une base de vecteurs propres **D**-orthonormée. Donc, il existe une matrice **B** telle que :

$$\mathbf{WDB} = \mathbf{B}\mathrm{M} \text{ et } \mathbf{B}^t\mathbf{DB} = \mathbf{I}_n$$

$$\mathrm{M} = Diag\left(\mu_1, \mu_2, \cdots, \mu_n\right)$$

On ne s'intéresse qu'aux *r* premiers (valeurs propres non nulles) qui sont définis de façon unique (valeurs propres distinctes) :

$$\mathbf{WDB}_r = \mathbf{B}\Lambda_r \text{ et } \mathbf{B}_r^t\mathbf{DB}_r = \mathbf{I}_r$$

$\mathbf{B}_r$ est une matrice à *n* lignes et *r* colonnes, chacune de ses colonnes étant un vecteur propre **D**-normé de $\mathbb{R}^n$. $\Lambda_r$ est la matrice diagonale qui contient les mêmes valeurs propres.

Les vecteurs de $\mathbf{A}_r$ sont les *r* **axes principaux** du schéma, les vecteurs de $\mathbf{B}_r$ sont les *r* **composantes principales** du schéma.

Les matrices $\mathbf{QV}$ et $\mathbf{DW}$ sont toujours diagonalisables et ont même valeurs propres non nulles que les précédentes. La première est $\mathbf{Q}^{-1}$-symétrique (la matrice $\mathbf{Q}^{-1}\mathbf{QV}$ est égale à sa transposée) et à ce titre définit une base de vecteurs propres $\mathbf{Q}^{-1}$-orthonormée. Donc, il existe une matrice $\mathbf{A}^*$ telle que :

$$\mathbf{QVA}^* = \mathbf{A}^*\Lambda \text{ et } \mathbf{A}^{*t}\mathbf{Q}^{-1}\mathbf{A} = \mathbf{I}_p$$

On ne s'intéresse qu'aux *r* premiers (valeurs propres non nulles) qui sont définis de façon unique (valeurs propres distinctes) :

$$\mathbf{QVA}_r^* = \mathbf{A}_r^*\Lambda_r \text{ et } \mathbf{A}_r^{*t}\mathbf{Q}^{-1}\mathbf{A}_r^* = \mathbf{I}_r$$

La seconde est $\mathbf{D}^{-1}$-symétrique (la matrice $\mathbf{D}^{-1}\mathbf{DW}$ est égale à sa transposée) et à ce titre définit une base de vecteurs propres $\mathbf{D}^{-1}$-orthonormée. Donc, il existe une matrice $\mathbf{B}^*$ telle que :

$$\mathbf{DWB}^* = \mathbf{B}^*\mathrm{M} \text{ et } \mathbf{B}^{*t}\mathbf{D}^{-1}\mathbf{B}^* = \mathbf{I}_n$$

On ne s'intéresse qu'aux *r* premiers (valeurs propres non nulles) qui sont définis de façon unique (valeurs propres distinctes) :

$$\mathbf{DWB}_r^* = \mathbf{B}_r^*\Lambda_r \text{ et } \mathbf{B}_r^{*t}\mathbf{D}^{-1}\mathbf{B}_r^* = \mathbf{I}_r$$

Les vecteurs de $\mathbf{A}_r^*$ sont les *r* **facteurs principaux** du schéma, les vecteurs de $\mathbf{B}_r^*$ sont les *r* **cofacteurs principaux** du schéma [1].

Il suffit pour retenir l'essentiel de mémoriser le schéma :

| | | |
|---|---|---|
| $\mathbf{A}_r = [\mathbf{a}_1, \cdots, \mathbf{a}_r]$<br>**Q**-orthonormés<br>axes principaux | $\overset{\mathbf{Q}}{\rightarrow}$ | $\mathbf{A}_r^* = [\mathbf{a}_1^*, \cdots, \mathbf{a}_r^*]$<br>$\mathbf{Q}^{-1}$-orthonormés<br>facteurs principaux |
| $\mathbf{X}^t \uparrow$ | $\Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_r \end{bmatrix}$ | $\downarrow \mathbf{X}$ |
| $\mathbf{B}_r^* = [\mathbf{b}_1^*, \cdots, \mathbf{b}_r^*]$<br>$\mathbf{D}^{-1}$-orthonormés<br>cofacteurs principaux | $\overset{\mathbf{D}}{\leftarrow}$ | $\mathbf{B}_r = [\mathbf{b}_1, \cdots, \mathbf{b}_r]$<br>**D**-orthonormés<br>composantes principales |

[1] Tenenhaus, M. & Young, F.W. (1985) An analysis and synthesis of multiple correspondence analysis, optimal scaling, dual scaling, homogeneity analysis and other methods for quantifying categorical multivariate data. Psychometrika : 50, 1, 91-119.

Enfin, il suffit de calculer un seul des quatre systèmes d'axes pour obtenir les trois autres. En effet, on a :

$$\boxed{1)\ \mathbf{A}_r^* = \mathbf{Q}\mathbf{A}_r \quad 2)\ \mathbf{B}_r = \mathbf{X}\mathbf{A}_r^*\Lambda_r^{-\frac{1}{2}} \quad 3)\ \mathbf{B}_r^* = \mathbf{D}\mathbf{B}_r \quad 4)\ \mathbf{A}_r = \mathbf{X}^t\mathbf{B}_r^*\Lambda_r^{-\frac{1}{2}}}$$

## 1.4 Propriétés d'optimalité

Ces vecteurs ont des propriétés fondamentales qui s'écrivent sous forme de théorèmes généraux et qui se retrouvent sous des formes pratiques variées dans la suite.

1) Si on cherche un vecteur $\mathbf{a}$ de $\mathbb{R}^p$ $\mathbf{Q}$-normé qui maximise $\|\mathbf{XQa}\|_{\mathbf{D}}^2$, la solution est unique. Elle est obtenue pour $\mathbf{a} = \mathbf{a}_1$ et le maximum atteint est $\lambda_1$. Si on cherche un nouveau vecteur $\mathbf{a}$ de $\mathbb{R}^p$ $\mathbf{Q}$-normé qui maximise à nouveau $\|\mathbf{XQa}\|_{\mathbf{D}}^2$ sous la contrainte $\langle \mathbf{a} | \mathbf{a}_1 \rangle_{\mathbf{Q}} = \mathbf{a}_1^t\mathbf{Q}\mathbf{a} = 0$, la solution est unique. Elle est obtenue pour $\mathbf{a} = \mathbf{a}_2$ et le maximum atteint est $\lambda_2$. Et ainsi de suite de proche en proche. Si on cherche un nouveau vecteur $\mathbf{a}$ de $\mathbb{R}^p$ $\mathbf{Q}$-normé qui maximise $\|\mathbf{XQa}\|_{\mathbf{D}}^2$ sous la contrainte $\langle \mathbf{a} | \mathbf{a}_1 \rangle_{\mathbf{Q}} = \cdots = \langle \mathbf{a} | \mathbf{a}_{j-1} \rangle_{\mathbf{Q}} = 0$, la solution est unique. Elle est obtenue pour $\mathbf{a} = \mathbf{a}_j$ et le maximum atteint est $\lambda_j$.

On résume en disant que les vecteurs $\mathbf{a}_1, \cdots, \mathbf{a}_j, \cdots, \mathbf{a}_r$ maximisent successivement sous contrainte d'orthonormalité au sens de $\mathbf{Q}$ la forme quadratique $\|\mathbf{XQa}\|_{\mathbf{D}}^2$.

2) Les vecteurs $\mathbf{a}_1^*, \cdots, \mathbf{a}_j^*, \cdots, \mathbf{a}_r^*$ maximisent successivement sous contrainte d'orthonormalité au sens de $\mathbf{Q}^{-1}$ la forme quadratique $\|\mathbf{Xa}^*\|_{\mathbf{D}}^2$.

3) Les vecteurs $\mathbf{b}_1, \cdots, \mathbf{b}_j, \cdots, \mathbf{b}_r$ maximisent successivement sous contrainte d'orthonormalité au sens de $\mathbf{D}$ la forme quadratique $\|\mathbf{X}^t\mathbf{Db}\|_{\mathbf{Q}}^2$.

4) Les vecteurs $\mathbf{b}_1^*, \cdots, \mathbf{b}_j^*, \cdots, \mathbf{b}_r^*$ maximisent successivement sous contrainte d'orthonormalité au sens de $\mathbf{D}^{-1}$ la forme quadratique $\|\mathbf{X}^t\mathbf{b}^*\|_{\mathbf{D}}^2$.

5) Si on cherche un couple formé d'un vecteur $\mathbf{a}$ de $\mathbb{R}^p$ $\mathbf{Q}$-normé et d'un vecteur $\mathbf{b}$ de $\mathbb{R}^n$ $\mathbf{D}$-normé qui maximise $\langle \mathbf{XQa} | \mathbf{b} \rangle_{\mathbf{D}} = \langle \mathbf{X}^t\mathbf{Db} | \mathbf{a} \rangle_{\mathbf{Q}}$, la solution est unique. Elle est obtenue pour $\mathbf{a} = \mathbf{a}_1$ et $\mathbf{b} = \mathbf{b}_1$ et le maximum atteint est $\sqrt{\lambda_1}$. Sous contrainte d'orthonormalité, le résultat s'étend aux couples suivants.

## 1.5 Approximations de tableau

On considère deux schémas de dualité de mêmes paramètres à l'exclusion du tableau :

$$\begin{array}{ccc} \boxed{p} & \overset{\mathbf{Q}}{\rightarrow} & \boxed{p} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X} \\ \boxed{n} & \underset{\mathbf{D}}{\leftarrow} & \boxed{n} \end{array} \quad \text{et} \quad \begin{array}{ccc} \boxed{p} & \overset{\mathbf{Q}}{\rightarrow} & \boxed{p} \\ \mathbf{Z}^t \uparrow & & \downarrow \mathbf{Z} \\ \boxed{n} & \underset{\mathbf{D}}{\leftarrow} & \boxed{n} \end{array}$$

On peut alors utiliser la configuration définie par :

$$\begin{array}{ccc} \boxed{p} & \overset{\mathbf{Q}}{\rightarrow} & \boxed{p} \\ \mathbf{Z}^t \uparrow & & \downarrow \mathbf{X} \\ \boxed{n} & \underset{\mathbf{D}}{\leftarrow} & \boxed{n} \end{array}$$

L'opérateur qui "fait le tour" $\mathbf{Z}^t\mathbf{DXQ}$ n'est pas diagonalisable mais cette matrice carrée a une trace (somme des éléments diagonaux) qui définit une application qui associe à un couple de tableaux un nombre :

$$(\mathbf{X},\mathbf{Z}) \mapsto Trace\left(\mathbf{Z}^t\mathbf{DXQ}\right) = Trace\left(\mathbf{XQZ}^t\mathbf{D}\right)$$

C'est un produit scalaire dans l'ensemble des tableaux :

$$\left\langle \mathbf{X} \middle| \mathbf{Z} \right\rangle_{\mathbf{D},\mathbf{Q}} = Trace\left(\mathbf{Z}^t\mathbf{DXQ}\right) = Trace\left(\mathbf{XQZ}^t\mathbf{D}\right)$$

On a donc une distance entre deux tableaux :

$$d^2_{\mathbf{D},\mathbf{Q}}(\mathbf{X},\mathbf{Z}) = \left\|\mathbf{X}-\mathbf{Z}\right\|^2_{\mathbf{D},\mathbf{Q}} = \left\langle \mathbf{X}-\mathbf{Z} \middle| \mathbf{X}-\mathbf{Z} \right\rangle_{\mathbf{D},\mathbf{Q}} = Trace\left((\mathbf{X}-\mathbf{Z})^t\,\mathbf{D}(\mathbf{X}-\mathbf{Z})\mathbf{Q}\right)$$

Par exemple, si

$$\mathbf{D} = \left(\frac{1}{n}\right)\mathbf{I}_n \quad \mathbf{Q} = \left(\frac{1}{p}\right)\mathbf{I}_p$$

$$d^2_{\mathbf{D},\mathbf{Q}}(\mathbf{X},\mathbf{Z}) = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}\left(x_{ij}-z_{ij}\right)^2$$

Le schéma $(\mathbf{X},\mathbf{Q},\mathbf{D})$ permet d'écrire la décomposition exacte :

$$\boxed{\mathbf{X} = \sum_{k=1}^{r}\sqrt{\lambda_k}\,\mathbf{b}_k\mathbf{a}_k^t}$$

Cela dérive simplement du fait que la ligne *i* du tableau, vecteur de $\mathbb{R}^p$, s'exprime par ses coordonnées dans la base des axes principaux, ou la colonne *j* du tableau, vecteur de

$\mathbb{R}^n$, s'exprime par ses coordonnées dans la base des composantes principales. Les formules de transition, déjà rencontrées, donnent :

$$\boxed{\mathbf{XQA}_r = \mathbf{B}_r \Lambda_r^{\frac{1}{2}}} \text{ et } \boxed{\mathbf{X}^t\mathbf{DB}_r = \mathbf{A}_r \Lambda_r^{\frac{1}{2}}}$$

On peut dire que les coordonnées des projections des lignes sur les axes principaux sont, à une constante près, les composantes des composantes principales et dualement que les coordonnées des projections des colonnes sur les composantes principales sont, à une constante près, les composantes des axes principaux .

Il s'en suit que la meilleure reconstitution de **X** par un tableau de rang 1 est :

$$\boxed{\hat{\mathbf{X}}_1 = \sqrt{\lambda_1}\, \mathbf{b}_1 \mathbf{a}_1^t}$$

que la meilleure reconstitution de **X** par un tableau de rang 2 est :

$$\boxed{\hat{\mathbf{X}}_2 = \sqrt{\lambda_1}\, \mathbf{b}_1 \mathbf{a}_1^t + \sqrt{\lambda_2}\, \mathbf{b}_2 \mathbf{a}_2^t}$$

que la meilleure reconstitution de **X** par un tableau de rang *m* est :

$$\boxed{\mathbf{X} = \sum_{k=1}^{m} \sqrt{\lambda_k}\, \mathbf{b}_k \mathbf{a}_k^t}$$

L'erreur de reconstitution est alors :

$$d^2_{\mathbf{D},\mathbf{Q}}\left(\mathbf{X}, \sum_{k=1}^{m} \sqrt{\lambda_k}\, \mathbf{b}_k \mathbf{a}_k^t\right) = \left\| \mathbf{X} - \sum_{k=1}^{m} \sqrt{\lambda_k}\, \mathbf{b}_k \mathbf{a}_k^t \right\|^2_{\mathbf{D},\mathbf{Q}} = \left\| \sum_{k=m+1}^{r} \sqrt{\lambda_k}\, \mathbf{b}_k \mathbf{a}_k^t \right\|^2_{\mathbf{D},\mathbf{Q}} = \sum_{k=m+1}^{r} \lambda_k$$

On dit qu'on a opéré la décomposition en valeurs singulières du tableau.

# 2 Graphes canoniques

## 2.1 Procédure centrale

On peut maintenant concevoir une routine unique de "diagonalisation d'un schéma de dualité" qui pour un triplet $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$ et le choix d'un entier *f* dit "nombres d'axes conservés" renvoie les *f* premières valeurs propres $\lambda_1 > \lambda_2 > \cdots > \lambda_f$, les *f* premiers axes principaux $\mathbf{a}_1, \cdots, \mathbf{a}_f$ et les *f* premières composantes principales $\mathbf{b}_1, \cdots, \mathbf{b}_f$. L'entier *f* peut être choisi à la lecture du graphe des valeurs propres.

Pour celui qui veut écrire le programme, deux remarques sont utiles. La première porte sur la procédure de calcul sous-jacente. On peut utiliser soit la diagonalisation d'une matrice soit la décomposition en valeurs singulières.

La documentation de R est un résumé parfait du problème :

**eigen package:base R Documentation**

Spectral Decomposition of a Matrix

```
Description:

     This function computes eigenvalues and eigenvectors by providing
     an interface to the EISPACK routines `RS', `RG', `CH' and `CG'.

Usage:

     eigen(x, symmetric, only.values=FALSE)

Arguments:

       x: a matrix whose spectral decomposition is to be computed.

symmetric: if `TRUE', the matrix is assumed to be symmetric (or
          Hermitian if complex) and only its lower triangle is used. If
          `symmetric' is not specified, the matrix is inspected for
          symmetry.

only.values: if `TRUE', only the eigenvalues are computed and returned,
          otherwise both eigenvalues and eigenvectors are returned.

Value:

     The spectral decomposition of `x' is returned as components of a
     list.

  values: a vector containing the p eigenvalues of `x', sorted in
          decreasing order, according to `Mod(values)' if they are
          complex.

 vectors: a p * p matrix whose columns contain the eigenvectors of `x',
          or `NULL' if `only.values' is `TRUE'.

Note:

     To compute the determinant of a matrix (do you really need it?),
     it is much more efficient to use the QR decomposition, see `qr'.

References:

     Smith, B. T, Boyle, J. M., Dongarra, J. J., Garbow, B. S.,
     Ikebe,Y., Klema, V., and  Moler, C. B. (1976). Matrix Eigensystems
     Routines - EISPACK Guide. Springer-Verlag Lecture Notes in
     Computer Science.

See Also:

     `svd', a generalization of `eigen'; `qr', and `chol' for related
     decompositions.

Examples:

> a <- cbind(c(1,-1),c(-1,1))
> a
     [,1] [,2]
[1,]    1   -1
[2,]   -1    1
> eigen(a)
$values
[1] 2 0

$vectors
        [,1]    [,2]
[1,] -0.7071 -0.7071
[2,]  0.7071 -0.7071
```

On aura soin de diagonaliser une matrice **symétrique** pour obtenir une base de vecteurs propres orthonormés. La procédure de base utilise une matrice symétrique et renvoie des vecteurs orthonormés dans la métrique canonique. On se contentera de "casser le schéma de dualité" du côté de la plus petite dimension en remplaçant **Q** ou **D** par une décomposition du type :

$$\begin{array}{ccc} & \mathbf{I}_p & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{H}\uparrow & & \downarrow\mathbf{H}^t \\ & \mathbf{Q} & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t\uparrow & & \downarrow\mathbf{X} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D} & \end{array}$$

Si **Q** est une matrice diagonale (ses termes diagonaux sont strictement positifs et les autres sont nuls) on se contente de noter :

$$\mathbf{H}=\mathbf{H}^t=\mathbf{Q}^{\frac{1}{2}}=Diag\left(\sqrt{q_{11}},...,\sqrt{q_{pp}}\right)\Rightarrow\mathbf{H}^{-1}=\mathbf{Q}^{-\frac{1}{2}}=Diag\left(\frac{1}{\sqrt{q_{11}}},...,\frac{1}{\sqrt{q_{pp}}}\right)$$

Si **Q** n'est pas diagonale, elle est symétrique (comme matrice d'un produit scalaire) et sa décomposition spectrale propose :

$$\mathbf{Q}=\mathbf{W}\Theta\mathbf{W}^t\Rightarrow\mathbf{H}=\Theta^{\frac{1}{2}}\mathbf{W}^t\Rightarrow\mathbf{H}^{-1}=\mathbf{W}\Theta^{-\frac{1}{2}}$$

On peut aussi utiliser la procédure de Choleski. Dans tous les cas, on diagonalisera alors, avec la procédure de base une matrice symétrique et on retrouvera les axes par :

$$\mathbf{H}\mathbf{X}^t\mathbf{D}\mathbf{X}\mathbf{H}^t=\mathbf{Z}\Lambda\mathbf{Z}^t\Rightarrow\mathbf{A}=\mathbf{H}^{-1}\mathbf{Z}$$

Les composantes principales sont enfin obtenues par les formules de transition. Si la petite dimension est de l'autre côté, on applique la même procédure dans l'autre sens.

On obtient le même résultat avec la décomposition en valeurs singulières :

**svd** **package:base** **R Documentation**

Singular Value Decomposition of a Matrix

Description:

Compute the singular-value decomposition of a rectangular matrix.

Usage:

```
svd(x, nu = min(n,p), nv = min(n,p))
```

Arguments:

| | |
|---|---|
| x: | a matrix whose SVD decomposition is to be computed. |
| nu: | the number of left eigenvectors to be computed. This must be one of `0', `nrow(x)' and `ncol(x)'. |
| nv: | the number of right eigenvectors to be computed. This must be one of `0', and `ncol(x)'. |

Details:

`svd' provides an interface to the LINPACK routine DSVDC. The singular value decomposition plays an important role in many statistical techniques.

```
Value:

     The SVD decomposition of the matrix as computed by LINPACK,

                              X = U D V',

     where U and V are orthogonal, V' means V transposed, and D is a
     diagonal matrix with the singular values D[i,i].  Equivalently, D
     = U' X V, which is verified in the examples, below.

     The components in the returned value correspond directly to the
     values returned by DSVDC.

       d: a vector containing the singular values of `x'.

       u: a matrix whose columns contain the left eigenvectors of `x'.

       v: a matrix whose columns contain the right eigenvectors of `x'.

References:

     Dongarra, J. J., Bunch, J. R., Moler, C. B. and Stewart, G. W.
     (1978) LINPACK Users Guide.  Philadelphia: SIAM Publications.

See Also:
```

Il conviendra alors de casser deux fois le schéma de départ, dans la même logique :

$$\begin{array}{ccc} & \mathbf{I}_p & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{H}\uparrow & & \downarrow\mathbf{H}^t \\ & \mathbf{Q} & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t\uparrow & & \downarrow\mathbf{X} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D} & \\ \mathbf{K}^t\uparrow & & \downarrow\mathbf{K} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{I}_n & \end{array}$$

On prendre la décomposition en valeurs singulières de :

$$\mathbf{KXH}^t = \mathbf{UDV}^t$$

On retrouvera les valeurs propres comme carrés des valeurs singulières et les axes et composantes par $\mathbf{B} = \mathbf{K}^{-1}\mathbf{U}$ et $\mathbf{A} = \mathbf{H}^{-1}\mathbf{V}$. On vérifie qu'on obtient ainsi des bases orthonormales pour les métriques choisies :

$$\mathbf{A}^t\mathbf{QA} = \mathbf{A}^t\mathbf{H}^t\mathbf{HA} = \mathbf{V}^t\mathbf{V} = \mathbf{I}_p$$

## 2.2 Décentrage

A partir du schéma général, on peut définir des pratiques dont les ACP centrées et normées sont des cas particuliers. Prenons l'exemple d'un tableau d'examen. On trouvera les données sur la carte Deug de la pile Data du logiciel ADE-4. 104 étudiants (Deug

Mass 2° année) ont leur notes pour 9 matières : 1- Algèbre et Analyse des données (sur 100), 2- Analyse (sur 60), 3-Probabilités (sur 80), 4-Informatique (sur 60), 5-Dominante (Sociologie ou Economie sur 120), 6-Options (sur 40), 7-Ouvertures (sur 40), 8-Anglais (sur 40) et 9-Education physique et sportive (bonification <=15).

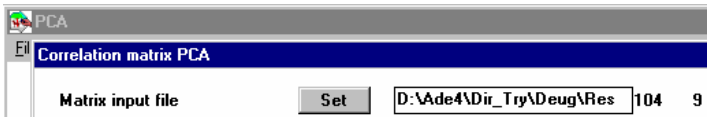
```
> res <- read.table("Res.txt")
> res
    V1   V2 V3   V4   V5 V6   V7   V8   V9
1   40 26.0 26 26.0 51.9 17 24.0 19.0 11.5
2   37 34.5 37 32.0 72.0 24 22.0 26.0 11.5
3   37 41.0 29 34.5 72.0 24 27.0 19.6 11.5
etc.
```

**prcomp package:mva R Documentation**

```
Principal Components Analysis

Description:

     Performs a principal components analysis on the given data matrix
     and returns the results as an object of class `prcomp'.

Usage:

     prcomp(x, retx = TRUE, center = TRUE, scale. = FALSE, tol = NULL)

Arguments:

       x: a matrix (or data frame) which provides the data for the
          principal components analysis.

    retx: a logical value indicating whether the rotated variables
          should be returned.

  center: a logical value indicating whether the variables should be
          shifted to be zero centered. Alternately, a vector of length
          equal the number of columns of `x' can be supplied. The value
          is passed to `scale'.

   scale: a logical value indicating whether the variables should be
          scaled to have unit variance before the analysis takes place.
          The default is `FALSE' for consistency with S, but in general
          scaling is advisable. Alternately, a vector of length equal
          the number of columns of `x' can be supplied.  The value is
          passed to `scale'.

     tol: a value indicating the magnitude below which components
          should be omitted. With the default null setting, no
          components are omitted.  Other settings for tol could be `tol
          = 0' or `tol = sqrt(.Machine$double.eps)'.
```

L'ACP normée est obtenue avec :

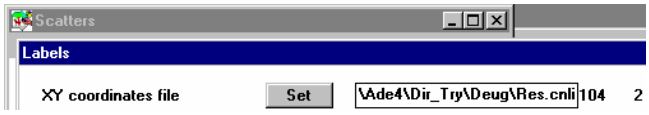
```
> pr0 <- prcomp(res,scale=T)
> pr0
Standard deviations:
[1] 1.7611 1.1675 1.0160 0.9665 0.8601 0.7581 0.7298 0.6615 0.5336
```

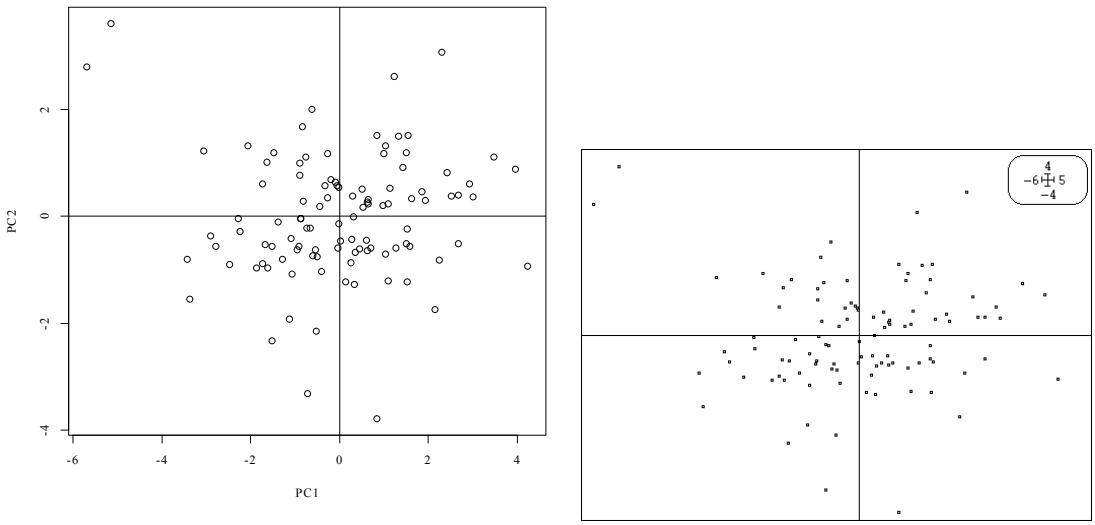



```
Num. Eigenval.   R.Iner.  R.Sum    |Num. Eigenval.   R.Iner.  R.Sum  |
01   +3.1014E+00 +0.3446 +0.3446   |02   +1.3630E+00 +0.1514 +0.4960 |
03   +1.0323E+00 +0.1147 +0.6107   |04   +9.3405E-01 +0.1038 +0.7145 |
05   +7.3975E-01 +0.0822 +0.7967   |06   +5.7467E-01 +0.0639 +0.8606 |
07   +5.3254E-01 +0.0592 +0.9197   |08   +4.3754E-01 +0.0486 +0.9684 |
09   +2.8478E-01 +0.0316 +1.0000
```

Le lien se fait par :

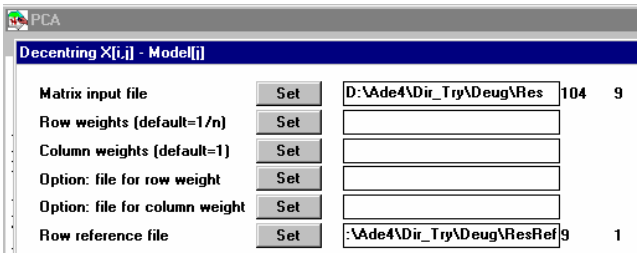
```
> pr0$sdev^2
[1] 3.1014 1.3630 1.0323 0.9341 0.7398 0.5747 0.5325 0.4375 0.2848
Référence: 50,30,40,30,60,20,20,20,0.
> names(pr0)
[1] "sdev"     "rotation" "x"
> plot(pr0$x[,1:2])
> abline(h=0)
> abline(v=0)
```




La typologie des étudiants se fait autour de l'étudiant moyen. Si on raisonne en termes ordinaires, on dit "j'ai tant de points d'avance ici, tant de points de retard là, …".

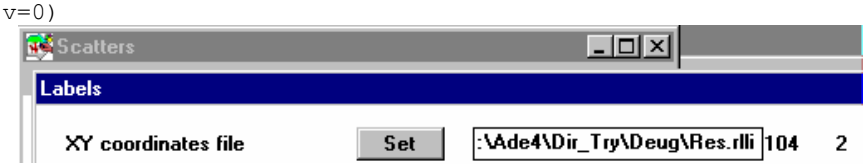
```
> ref <- c(50,30,40,30,60,20,20,20,0)
> ref
[1] 50 30 40 30 60 20 20 20  0
> pr1 <- prcomp(res,center=ref)
> pr1$sdev
[1] 20.271 16.320  8.428  7.716  6.877  5.873  4.563  3.931  3.219
> pr1$sdev^2
[1] 410.91 266.33  71.03  59.53  47.29  34.49  20.82  15.45  10.36
```

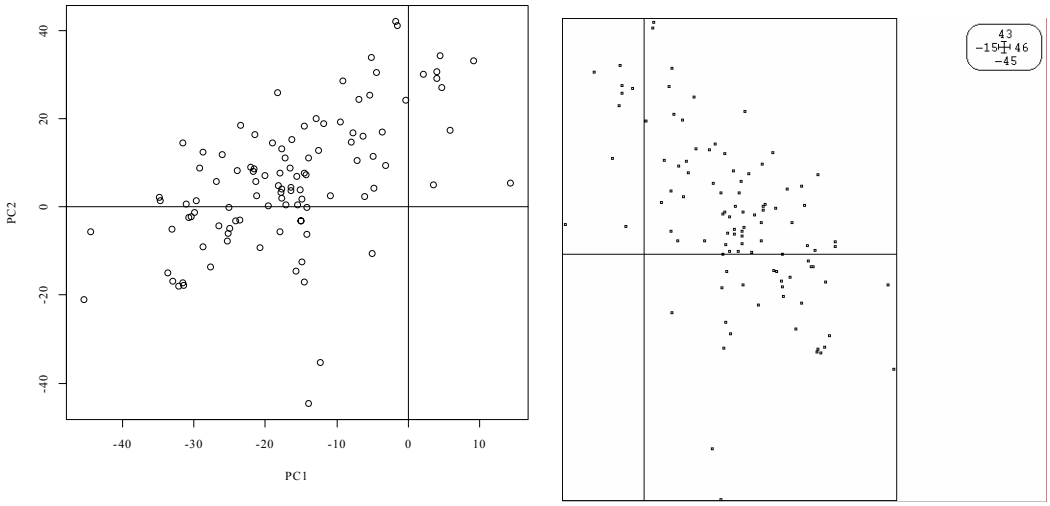



```
Total inertia:  927.216
-----------------------
Num. Eigenval.   R.Iner.  R.Sum    |Num. Eigenval.   R.Iner.  R.Sum  |
01   +4.0696E+02 +0.4389 +0.4389   |02   +2.6377E+02 +0.2845 +0.7234 |
03   +7.0346E+01 +0.0759 +0.7992   |04   +5.8958E+01 +0.0636 +0.8628 |
05   +4.6838E+01 +0.0505 +0.9133   |06   +3.4162E+01 +0.0368 +0.9502 |
07   +2.0617E+01 +0.0222 +0.9724   |08   +1.5302E+01 +0.0165 +0.9889 |
09   +1.0263E+01 +0.0111 +1.0000
```

Une toute petite difficulté :

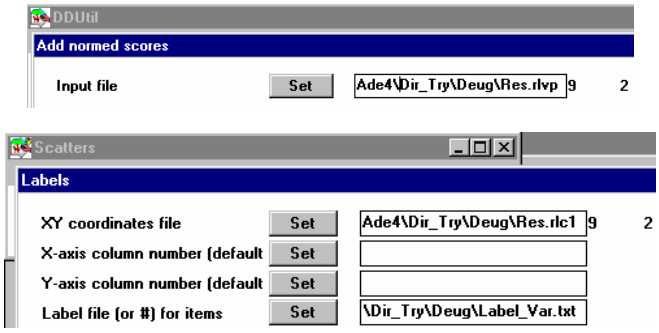
```
     The root-mean-square for a column is obtained by computing the
     square-root of the sum-of-squares of the non-missing values in the
     column divided by the number of non-missing values minus one.
> 406.96*104/103
[1] 410.9 (n-1) est une obligation, dommage !
> plot(pr1$x[,1:2])
> abline(h=0)
> abline(v=0)
```

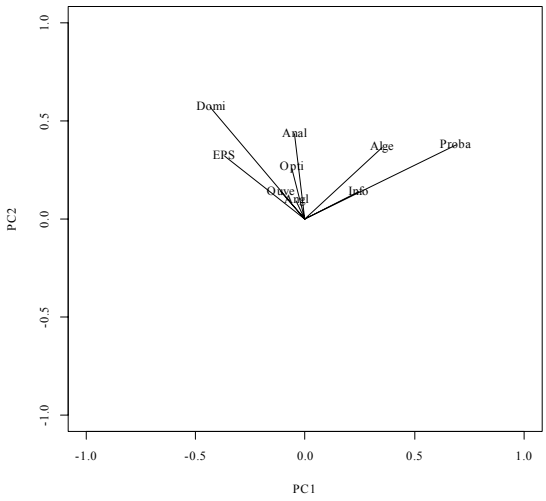


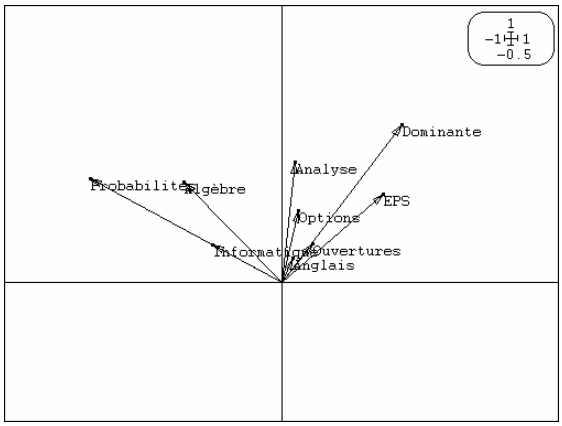




Bien noter que si **u** est propre **-u** aussi, donc que l'orientation d'un axe est aléatoire. Qu'est-ce qu'on apprend ?

```
> labelvar <-
c("Alge","Anal","Proba","Info","Domi","Opti","Ouve","Angl","EPS")
> plot(pr1$rotation[,1:2],type="n",xlim=c(-1,1),ylim=c(-1,1))
> segments(0,0,pr1$rotation[,1],pr1$rotation[,2])
> text(pr1$rotation[,1],pr1$rotation[,2],labelvar)
```

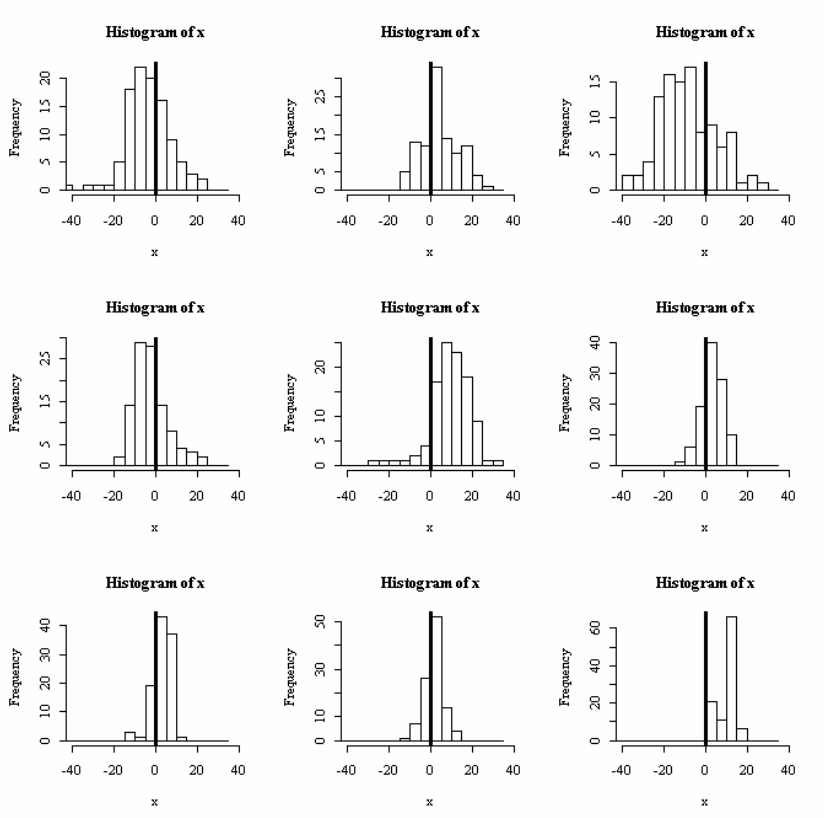





L'explication :

```
> apply(res0,2,hist,xlim=c(-40,40),breaks=seq(-45,35,by=5))
```




Il est bien connu que certains professeurs sont plus sympathiques que d'autres.

La procédure prcomp de R utilise le schéma de gauche, celle de PCA utilise celui de droite :

$$\boxed{p=9} \quad \mathbf{X}^t \uparrow \quad \boxed{n=104} \qquad \mathbf{Q}=\mathbf{I}_9 \rightarrow \qquad \boxed{p=9} \quad \downarrow \mathbf{X}=\left[y_{ij}-\alpha_j\right] \quad \boxed{n=104} \qquad \leftarrow \mathbf{D}=\frac{1}{(n-1)}\mathbf{I}_n$$

$$\boxed{p=9} \quad \mathbf{X}^t \uparrow \quad \boxed{n=104} \qquad \mathbf{Q}=\mathbf{I}_9 \rightarrow \qquad \boxed{p=9} \quad \downarrow \mathbf{X}=\left[y_{ij}-\alpha_j\right] \quad \boxed{n=104} \qquad \leftarrow \mathbf{D}=\frac{1}{n}\mathbf{I}_n$$

Les deux sont très voisins. Ce qui est plus variable est l'interprétation donnée aux calculs :

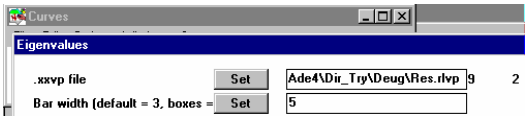
```
scale: a logical value indicating whether the variables should be
       scaled to have unit variance before the analysis takes place.
       The default is `FALSE' for consistency with S, but in general
       scaling is advisable. Alternately, a vector of length equal
       the number of columns of `x' can be supplied.  The value is
       passed to `scale'.
```

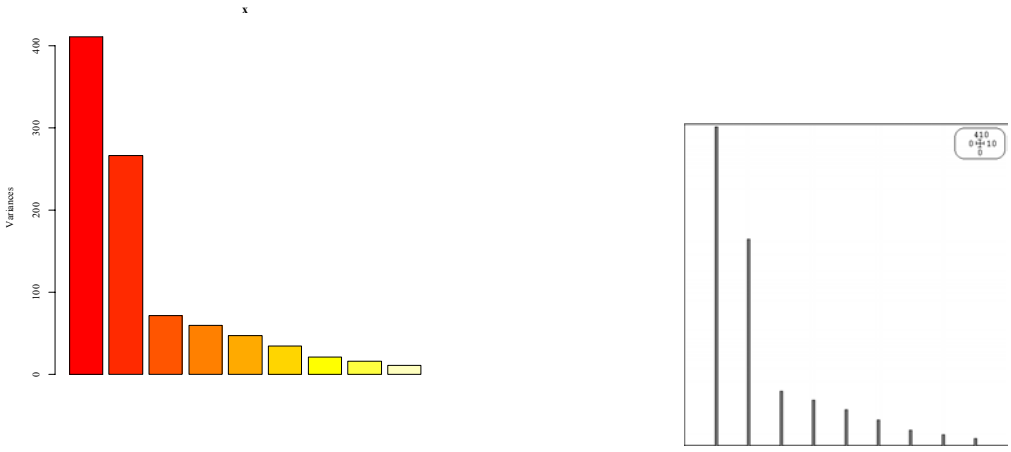
Ceci indique que l'interprétation en terme de modélisation des données est privilégiée. L'analyse fabrique un modèle des données par les données (l'article fondateur est [1]) :

$$\hat{\mathbf{X}}_2 = \sqrt{\lambda_1}\,\mathbf{b}_1\mathbf{a}_1^t + \sqrt{\lambda_2}\,\mathbf{b}_2\mathbf{a}_2^t$$

La note d'un étudiant dans une matière est le nombre de points nécessaires pour la moyenne corrigé par deux produits du type score du professeur x score de l'étudiant. En utilisant une représentation géométrique conservant les échelles, donc sans changement d'échelle ni explicite ni implicite, l'interprétation en termes de géométrie est privilégiée. Un nuage de 104 points est projeté sur un plan pour le voir.

```
> plot(pr1)
```




La dimension 2 s'impose par le graphe des valeurs propres.

[1] Eckart, C. & Young, G. (1936) The approximation of one matrix by another of lower rank. *Psychometrika* : 1, 211-218.

## 2.3 Redondance

Si on veut faire disparaître les différences (sensibles !) de système de notation (en moyenne et en amplitude) , l'ACP normée s'impose.

$$\begin{array}{ccc} & \mathbf{Q}=\mathbf{I}_p & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D}=\frac{1}{(n-1)}\mathbf{I}_n & \end{array} \qquad \begin{array}{ccc} & \mathbf{Q}=\mathbf{I}_p & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X} \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D}=\frac{1}{n}\mathbf{I}_n & \end{array}$$

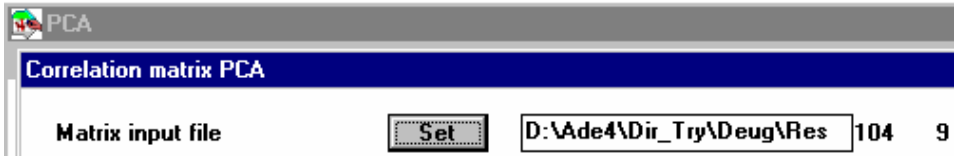
$$x_{ij}=\frac{y_{ij}-m_j}{\sqrt{\frac{1}{(n-1)}\sum_{i=1}^{n}\left(y_{ij}-m_j\right)^2}} \qquad x_{ij}=\frac{y_{ij}-m_j}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_{ij}-m_j\right)^2}}$$

Les deux schémas donnent des résultats identiques. Une composante principale est alors un score des lignes de norme 1, centré, qui maximise :

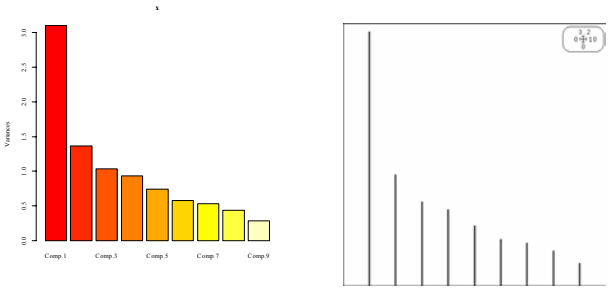
$$\left\|\mathbf{XDa}\right\|_{\mathbf{Q}}^{\mathbf{2}}=\sum_{j=1}^{p} corr^2\left(\mathbf{a},\mathbf{X}^j\right)$$

```
princomp                  package:mva                  R Documentation

Principal Components Analysis

Description:

     `princomp' performs a principal components analysis on the given
     data matrix and returns the results as an object of class
     `princomp'.

     `loadings' extracts the loadings.

     `screeplot' plots the variances against the number of the
     principal component. This is also the `plot' method.

Usage:

     princomp(x, cor = FALSE, scores = TRUE, covmat = NULL,
              subset = rep(TRUE, nrow(as.matrix(x))))
     loadings(x)
     plot(x, npcs = min(10, length(x$sdev)),
               type = c("barplot", "lines"), ...)
     screeplot(x, npcs = min(10, length(x$sdev)),
               type = c("barplot", "lines"), ...)

     print(x, ...)  summary(object)  predict(object, ...)

Arguments:

       x: a matrix (or data frame) which provides the data for the
          principal components analysis.

     cor: a logical value indicating whether the calculation should use
          the correlation matrix or the covariance matrix.

  scores: a logical value indicating whether the score on each
          principal component should be calculated.

  covmat: a covariance matrix, or a covariance list as returned by
          `cov.wt', `cov.mve' or `cov.mcd'. If supplied, this is used
```

```
           rather than the covariance matrix of `x'.

  subset: a vector used to select rows (observations) of the data
          matrix `x'.

x, object: an object of class `"princomp"', as from `princomp()'.

    npcs: the number of principal components to be plotted.

    type: the type of plot.

     ...: graphics parameters.

Details:

     The calculation is done using `eigen' on the correlation or
     covariance matrix, as determined by `cor'.  This is done for
     compatibility with the S-PLUS result.  A preferred method of
     calculation is to use svd on `x', as is done in `prcomp'.

     Note that the default calculation uses divisor `N' for the
     covariance matrix.

     The `print' method for the these objects prints the results in a
     nice format and the `plot' method produces a scree plot.

Value:

     `princomp' returns a list with class `"princomp"' containing the
     following components:

    sdev: the standard deviations of the principal components.

loadings: the matrix of variable loadings (i.e., a matrix whose columns
          contain the eigenvectors).

  center: the means that were subtracted.

   scale: the scalings applied to each variable.

   n.obs: the number of observations.

  scores: if `scores = TRUE', the scores of the supplied data on the
          principal components.

    call: the matched call.

References:

     Mardia, K. V., J. T. Kent and J. M. Bibby (1979). Multivariate
     Analysis, London: Academic Press.

     Venables, W. N. and B. D. Ripley (1997, 9). Modern Applied
     Statistics with S-PLUS, Springer-Verlag.

> pr2 <- princomp(res,cor=T)


> pr2
Call:
princomp(x = res, cor = T)

Standard deviations:
Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
1.7611 1.1675 1.0160 0.9665 0.8601 0.7581 0.7298 0.6615 0.5336

 9  variables and  104 observations.
```

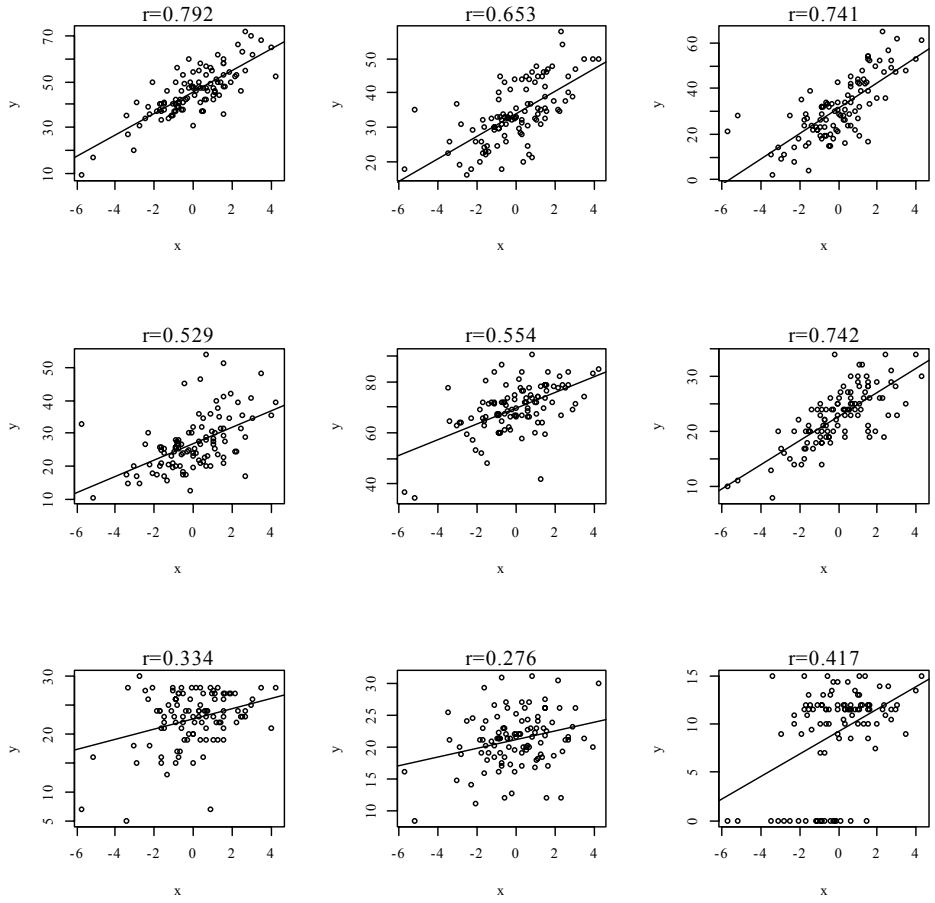


```
Total inertia:        9
-----------------------
Num. Eigenval.   R.Iner.  R.Sum    |Num. Eigenval.   R.Iner.  R.Sum  |
01   +3.1014E+00 +0.3446 +0.3446   |02   +1.3630E+00 +0.1514 +0.4960 |
03   +1.0323E+00 +0.1147 +0.6107   |04   +9.3405E-01 +0.1038 +0.7145 |
05   +7.3975E-01 +0.0822 +0.7967   |06   +5.7467E-01 +0.0639 +0.8606 |
07   +5.3254E-01 +0.0592 +0.9197   |08   +4.3754E-01 +0.0486 +0.9684 |
09   +2.8478E-01 +0.0316 +1.0000

> names(pr2)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
> pr2$sdev^2
Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
3.1014 1.3630 1.0323 0.9341 0.7398 0.5747 0.5325 0.4375 0.2848
```



Il reste un seul axe. Le graphe canonique, celui qui exprime le théorème cité, est alors :

```
> par(mfrow=c(3,3))
> plotreg
function(y,x){
   plot(x,y)
   abline(lm(y~x))
   mtext(paste("r=",round(cor(x,y),digits=3),sep=""))
}
> apply(res, 2, plotreg, x=pr2$scores[,1])
```

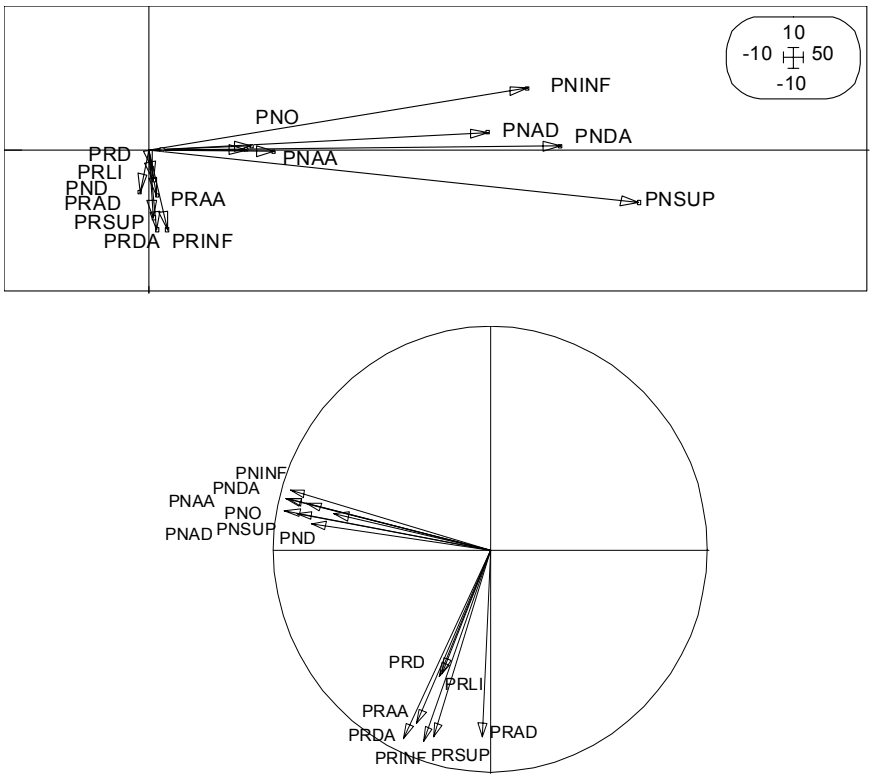


```
> pr2$loadings[,1]*pr2$sdev[1]
    V1     V2     V3     V4     V5     V6     V7     V8     V9
0.7925 0.6532 0.7410 0.5287 0.5539 0.7416 0.3336 0.2755 0.4172
```

Commenter.

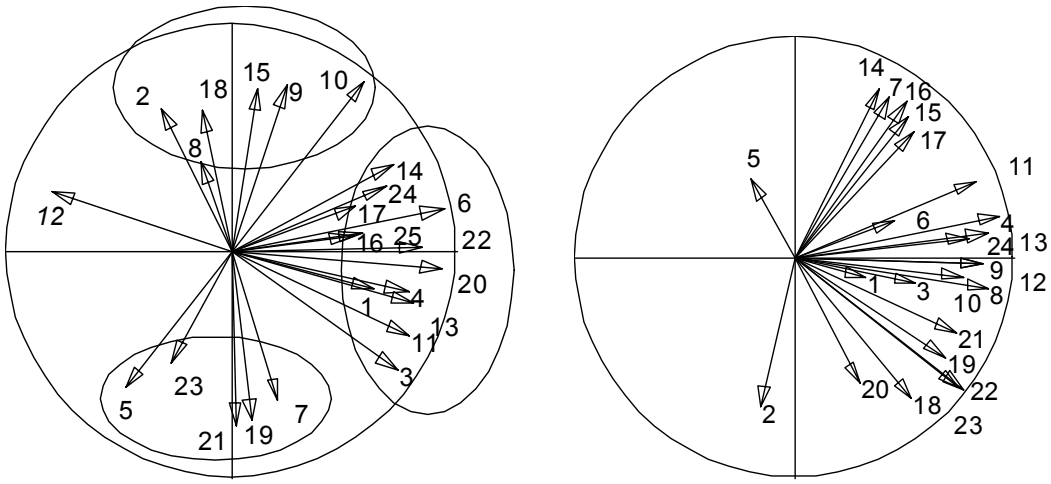
*Réduire ou éliminer la redondance est un des objectifs de l'analyse des données*. L'abondance de variables souvent redondantes est un obstacle majeur à la modélisation statistique. Les coordonnées factorielles lorsqu'elles sont interprétées sont de bons substituts aux mesures.

306 truites - 15 variables de coloration de la robe **1**. ACP centrée (en haut), normée (en bas) :

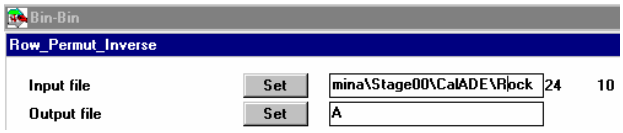



Il y a simplement deux variables : le plus simple est de sommer par paquets. Dans d'autre cas, la redondance n'est pas un parasite mais l'essentiel de la recherche. La même figure est utilisée pour caractériser la cohérence des jurys de classement : les juges sont alors les variables. Étudier les cartes de données Macon et Rock :

**1** Lascaux, J.M. (1996) *Analyse de la variabilité morphologique de la truite commune (Salmo trutta L.) dans les cours d'eau du bassin pyrénéen méditerranéen*. Thèse de doctorat en sciences agronomiques, INP Toulouse. 1-160.




Des goûts et des couleurs, il faut bien discuter. Pour le deuxième cas, on est passé par :




puis on a transposé pour avoir les juges en colonnes.

## 2.4 Valeurs propres

Quelle que soit la transformation préalable, on se retrouve avec un tableau **X** comportant *n* lignes et *p* colonnes. La géométrie du nuage des lignes est basée sur le produit scalaire :

$$\langle \mathbf{a} | \mathbf{b} \rangle = a_1 b_1 + ... + a_p b_p = 1 a_1 b_1 + ... + 1 a_p b_p$$

La géométrie du nuage des colonnes est basée sur le produit scalaire :

$$\langle \mathbf{a} | \mathbf{b} \rangle = \frac{1}{n}(a_1 b_1 + ... + a_n b_n) = \frac{1}{n} a_1 b_1 + ... + \frac{1}{n} a_n b_n$$

C'est l'option habituelle dans les cas simples. Mais en toute généralité, on peut choisir d'utiliser dans l'espace des lignes :

$$\langle \mathbf{a} | \mathbf{b} \rangle_\omega = \omega_1 a_1 b_1 + ... + \omega_p a_p b_p$$

et dans l'espace des colonnes :

$$\langle \mathbf{a} | \mathbf{b} \rangle_\pi = \pi_1 a_1 b_1 + ... + \pi_n a_n b_n$$

Ceci a un sens si et seulement si les coefficients $\omega_j$ et $\pi_i$ sont positifs. Il y a autant de coefficients, qui interviennent dans la géométrie de l'espace des lignes que de colonnes et réciproquement. Les $\omega_j$ sont dits poids des colonnes et les $\pi_i$ sont dits poids des lignes. L'analyse est caractérisée alors par un triplet formé du tableau transformé **Y** et de deux matrices diagonales (cette présentation étant faite pour la cohérence mathématique du total) qui s'écrivent :

$$\mathbf{Y} = \left[ y_{ij} \right] 1 \le i \le n, 1 \le j \le p$$

$$\mathbf{D}_\omega = \begin{bmatrix} \omega_1 & 0 & 0 \\ & \ddots & \\ 0 & \omega_j & 0 \\ & & \ddots \\ 0 & 0 & \omega_p \end{bmatrix} \text{ et } \mathbf{D}_\pi = \begin{bmatrix} \pi_1 & 0 & 0 \\ & \ddots & \\ 0 & \pi_i & 0 \\ & & \ddots \\ 0 & 0 & \pi_n \end{bmatrix}$$

L'analyse est entièrement définie par le triplet $(\mathbf{Y}, \mathbf{D}_\omega, \mathbf{D}_\pi)$ à deux métriques diagonales. La ligne *i* du tableau est un vecteur de $\mathbb{R}^p$ qui se trouve à une distance $d_i$ de l'origine avec :

$$d_i^2 = \sum_{j=1}^{p} \omega_j y_{ij}^2$$

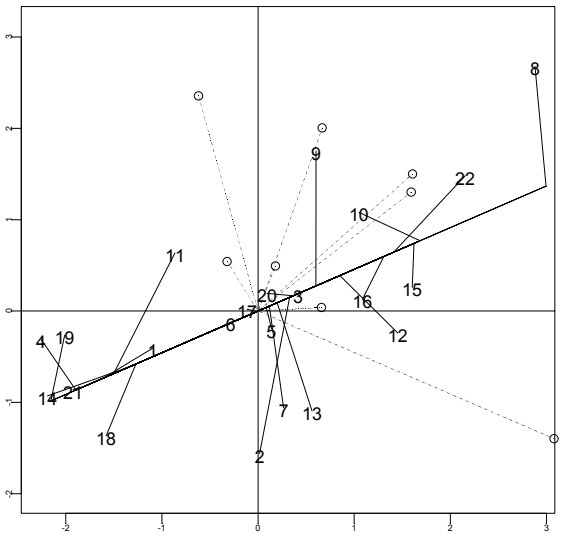
La colonne *j* du tableau est un vecteur de $\mathbb{R}^n$ qui se trouve à une distance $d_j$ de l'origine avec :

$$d_j^2 = \sum_{i=1}^{n} \pi_i y_{ij}^2$$

La ligne *i* du tableau est un vecteur de $\mathbb{R}^p$ qui a le poids $\pi_i$ et la colonne *j* du tableau est un vecteur de $\mathbb{R}^n$ qui a le poids $\omega_j$. Les quantités :

$$\sum_{i=1}^{n} \pi_i d_i^2 = \sum_{i=1}^{n} \pi_i \sum_{j=1}^{p} \omega_j y_{ij}^2 = \sum_{j=1}^{p} \omega_j \sum_{i=1}^{n} \pi_i y_{ij}^2 = \sum_{j=1}^{p} \omega_j d_j^2$$

sont les inerties communes des deux nuages associés au même triplet. Quand on projette les lignes sur un axe, l'inertie du nuage projeté est l'inertie projetée. De même pour les colonnes. Les opérations précédentes ont défini un axe principal **u** (nuage des lignes) ou une composante principale **v** (nuage des colonnes).




Minimiser le somme des carrés des distances entre les points et leur projection, c'est maximiser la somme des carrés des distances des projections à l'origine car la somme est constante (inertie totale).

$$\underset{\text{Minimum}}{\sum_{i=1}^{n} \pi_i d^2(M_i, m_i)} + \underset{\text{Maximum}}{\sum_{i=1}^{n} \pi_i d^2(m_i, O)} = \underset{\text{Constante}}{\sum_{i=1}^{n} \pi_i d^2(M_i, O)}$$

L'inertie projetée est maximum pour le premier axe principal. Ce maximum atteint est la première valeur propre ($\lambda_1$). On cherche un second axe, orthogonal au précédent qui maximise sous cette contrainte la même quantité. On trouve le deuxième axe principal. Le maximum atteint est la seconde valeur propre ($\lambda_2$). Et ainsi de suite. Si on raisonne sur le nuage des vecteurs colonnes, on a le même objectif et les mêmes résultats :

$$\underset{\text{Minimum}}{\sum_{j=1}^{p} \omega_j d^2\left(P_j, p_j\right)} + \underset{\text{Maximum}}{\sum_{j=1}^{p} \omega_j d^2\left(p_j, O\right)} = \underset{\text{Constante}}{\sum_{j=1}^{p} \omega_j d^2\left(P_j, O\right)}$$

La première composante principale donne le maximum $\lambda_1$, la seconde le nouveau maximum $\lambda_2$, etc. L'inertie totale du nuage des lignes comme l'inertie totale du nuage des colonnes se décompose de manière unique :

$$\sum_{i=1}^{n} \pi_i d^2\left(M_i, O\right) = \sum_{j=1}^{p} \omega_j d^2\left(P_j, O\right) = \lambda_1 + \lambda_2 + \ldots + \lambda_r$$

Le graphe des valeurs propres représente donc la manière dont se prend en compte la variabilité (progressivement jusqu'à 100%) ou comment on s'approche des données par automodélisation (jusqu'à 0%). Les cas typiques sont :

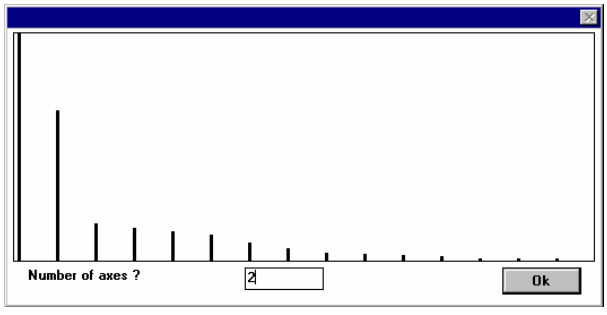
Carte Voiture (PCA: Correlation matrix PCA) :

```
Num. Eigenval.   R.Iner.  R.Sum    |Num. Eigenval.   R.Iner.  R.Sum  |
01   +4.6560E+00 +0.7760 +0.7760   |02   +9.1522E-01 +0.1525 +0.9285 |
03   +2.4043E-01 +0.0401 +0.9686   |04   +1.0271E-01 +0.0171 +0.9857 |
05   +6.4656E-02 +0.0108 +0.9965   |06   +2.0961E-02 +0.0035 +1.0000 |
```

93% de la variabilité totale s'exprime sur un plan (dimension 2).

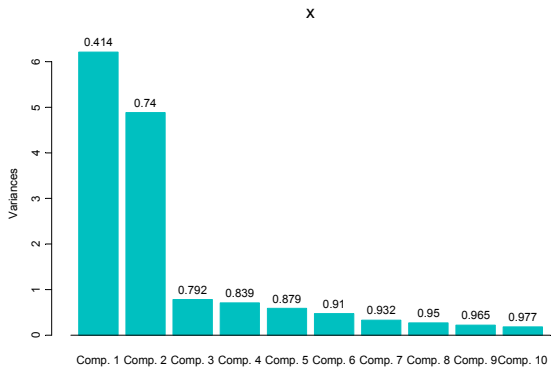
Carte Rhone (PCA: Correlation matrix PCA) :

```
Num. Eigenval.   R.Iner.  R.Sum    |Num. Eigenval.   R.Iner.  R.Sum  |
01   +6.2743E+00 +0.4183 +0.4183   |02   +4.1409E+00 +0.2761 +0.6943 |
03   +1.0082E+00 +0.0672 +0.7616   |04   +8.5920E-01 +0.0573 +0.8188 |
05   +7.6219E-01 +0.0508 +0.8697   |06   +6.6565E-01 +0.0444 +0.9140 |
07   +4.4427E-01 +0.0296 +0.9436   |08   +3.1199E-01 +0.0208 +0.9644 |
09   +1.7067E-01 +0.0114 +0.9758   |10   +1.4584E-01 +0.0097 +0.9855 |
11   +9.7799E-02 +0.0065 +0.9921   |12   +6.5818E-02 +0.0044 +0.9965 |
13   +2.9567E-02 +0.0020 +0.9984   |14   +2.3569E-02 +0.0016 +1.0000 |
15   +0.0000E+00 +0.0000 +1.0000
```




On garde 69% d'inertie sur un plan sans chercher à augmenter ce taux: le nuage est en dimension 2.

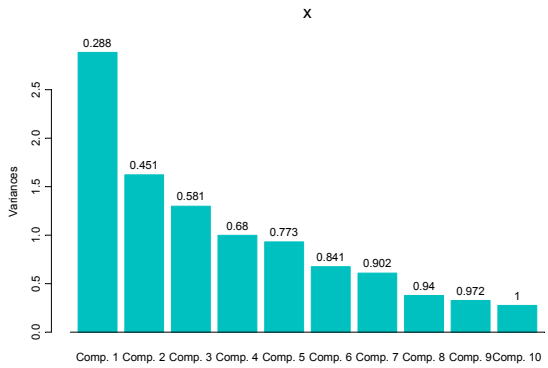
306 truites - 15 variables de coloration de la robe [1] :




```
plot(princomp(log(color+1),cor=T)
```

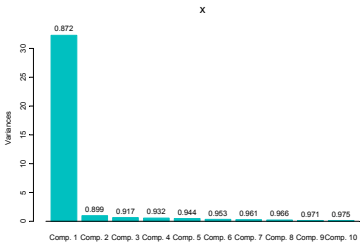
On garde 74% d'inertie sur un plan sans chercher à augmenter ce taux: le nuage est en dimension 2.

306 truites - 10 variables méristiques (ibidem):




On ne garde qu'un axe, l'analyse souligne la faible cohérence de ce type de variable. Ce n'est pas une mauvaise analyse, au contraire. Ces variables sont très peu corrélées et c'est un fait essentiel.

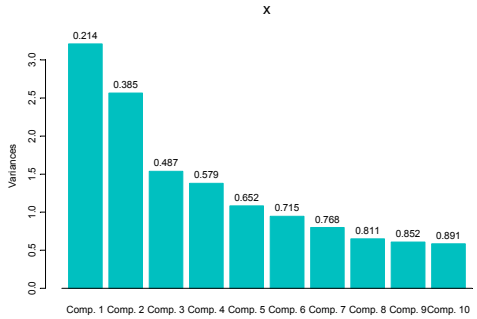
306 truites - 37 variables morphométriques (ibidem) :




[1] Lascaux, J.M. (1996) *Analyse de la variabilité morphologique de la truite commune (Salmo trutta L.) dans les cours d'eau du bassin pyrénéen méditerranéen*. Thèse de doctorat en sciences agronomiques, INP Toulouse. 1-160.

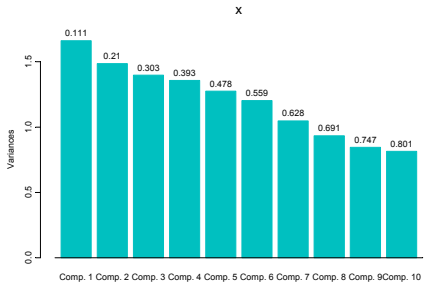
On ne garde qu'un axe, expression de l'effet taille. L'analyse illustre une trivialité et ce n'est pas la concentration de l'inertie dans peu de dimensions qui fait son intérêt.

306 truites - 15 variables ornementales binaires (ibidem) :




On garde deux facteurs.

```
plot(princomp(rmvnorm(n=100,d=15),cor=T))
```




L'ACP d'un jeu de données aléatoire ne donne aucune structure. Dès que les valeurs propres sont régulièrement décroissantes, on est dans la partie non organisée des données (pas forcément sans signification expérimentale par ailleurs).

## 2.5 Profils et biplot

Les données de profils sont une structure commune de données. Dans le tableau de départ **Y**, chaque ligne est une distribution de fréquences notée $y_{ij} = f_{j/i}$ (la fréquence de la colonne *j* sachant qu'on est sur la ligne *i*). Le terme de fréquence est pris au sens large, fréquence des électeurs du canton *i* ayant voté pour le candidat *j*, fréquence de l'allèle *j* dans la population *i*, fréquence des proies de la catégorie *j* dans l'estomac *i*, etc.

Dans tous les cas $\sum_{j=1}^{p} f_{j/i} = 1$. Les moyennes par colonnes pour la pondération uniforme $q_j = \frac{1}{n}\sum_{i=1}^{n} f_{j/i}$ vérifient également $\sum_{j=1}^{p} q_j = 1$. Le tableau **X** centré qui en découle $x_{ij} = y_{ij} - q_j$ a donc la propriété particulière :

$$\mathbf{X1}_p = 0$$

Le schéma suivant sera appelé ACP centrée sur profils :

$$\begin{array}{ccc} & \mathbf{Q}=\mathbf{I}_p & \\ \boxed{p} & \rightarrow & \boxed{p} \\ \mathbf{X}^t \uparrow & & \downarrow \mathbf{X}=\left[f_{j/i}-q_j\right] \\ \boxed{n} & \leftarrow & \boxed{n} \\ & \mathbf{D}=\frac{1}{n}\mathbf{I}_n & \end{array}$$

Les vecteurs $\mathbf{1}_p$ et $\mathbf{1}_n$ sont respectivement axe principal et composante principale. La somme des composantes d'un autre axe principal est donc nulle par orthogonalité (propriété particulière des données en profils) comme la somme des composantes d'une autre composante principale donc nulle par orthogonalité (propriété commune à toutes les ACP centrée). Les deux systèmes de coordonnées factorielles sont centrés.

La relation de transition prend alors une signification particulière.

$$\mathbf{L}_r=\mathbf{XQA}_r=\mathbf{B}_r\Lambda_r^{\frac{1}{2}} \Rightarrow L_k(i)=\sum_{j=1}^{p}\left(f_{j/i}-q_j\right)a_{jk}=\sum_{j=1}^{p}f_{j/i}a_{jk}+\sum_{j=1}^{p}q_j a_{jk}$$

$\mathbf{L}_r$ est le tableau des coordonnées factorielles des lignes sur les axes principaux. $L_k(i)$ est la coordonnée de l'individu *i* sur l'axe *k*. $a_{jk}$ est la composante de rang *j* de l'axe *k*. A une constante près ($\sum_{j=1}^{p}q_j a_{jk}$) le point *i* est positionné sur l'axe *k* à la moyenne des composantes ($\sum_{j=}^{p}f_{j/i}a_{jk}$) du vecteur. En dimension 3, l'opération donne *exactement* la représentation triangulaire.

Exemple (carte Europe de la pile Data de Ade-4 **1**)

| | 1978 | | | 1986 | | |
|---|---|---|---|---|---|---|
| | Primaire | Secondaire | Tertaire | Primaire | Secondaire | Tertaire |
| Belgique | 32 | 359 | 609 | 28 | 291 | 681 |
| Danemark | 79 | 319 | 602 | 59 | 282 | 659 |
| Espagne | 206 | 372 | 422 | 161 | 320 | 519 |
| France | 92 | 368 | 540 | 73 | 313 | 614 |
| Grèce | 320 | 297 | 383 | 285 | 281 | 434 |
| Irlande | 206 | 320 | 474 | 157 | 287 | 556 |
| Italie | 155 | 381 | 464 | 109 | 331 | 560 |
| Luxembourg | 62 | 392 | 546 | 40 | 330 | 630 |
| Pays-Bas | 54 | 330 | 616 | 49 | 255 | 696 |
| Portugal | 313 | 348 | 339 | 217 | 348 | 435 |
| Allemegne | 61 | 444 | 495 | 53 | 409 | 538 |
| Royaume-Uni | 28 | 390 | 582 | 25 | 309 | 666 |

Le premier tableau passé en pourcentage donne :

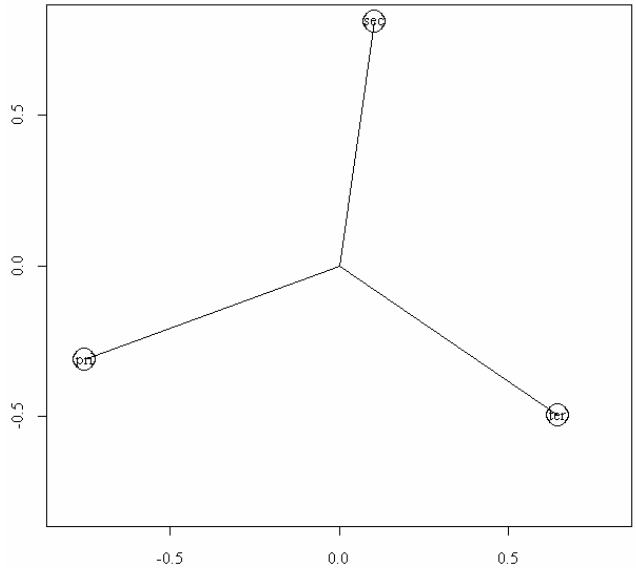
```
> euro78
```

**1** Collectif. (1989) *Encyclopaedia Universalis, Symposium, Les chiffres du Monde*. Encyclopaedia Universalis, Paris. 519 p.

```
              pri   sec   ter
Belgique    0.032 0.359 0.609
Danemark    0.079 0.319 0.602
Espagne     0.206 0.372 0.422
France      0.092 0.368 0.540
…
Pays-Bas    0.054 0.330 0.616
Portugal    0.313 0.348 0.339
Allemagne   0.061 0.444 0.495
Royaume-Uni 0.028 0.390 0.582

> pr0 <- princomp(euro78)
> pr0
Call:
princomp(x = euro78)

Standard deviations:
   Comp.1    Comp.2    Comp.3
1.318e-01 4.453e-02 4.715e-10

 3  variables and  12 observations.
> names(pr0)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
> pr0$loadings
     Comp.1  Comp.2 Comp.3
pri -0.7540 -0.3133 0.5774
sec  0.1057  0.8096 0.5774
ter  0.6483 -0.4963 0.5774 Donner la valeur exacte pour 0.5774
```

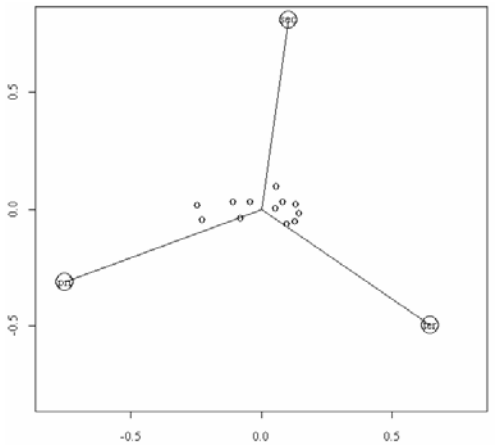
Les "loadings" sont les axes principaux.

```
> t(pr0$loadings)%*%pr0$loadings

             Comp.1     Comp.2     Comp.3
  Comp.1  1.000e+00 -5.573e-17  1.975e-16
  Comp.2 -5.573e-17  1.000e+00 -7.890e-17
  Comp.3  1.975e-16 -7.890e-17  1.000e+00

> plot(a1,a2,cex=3,xlim=c(-0.8,0.8),ylim=c(-0.8,0.8))
> segments(0,0,a1,a2)
> text(a1,a2,names(euro78))
```



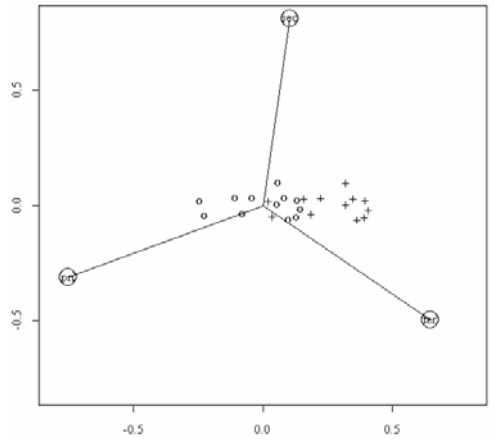

Le triangle est équilatéral. Les axes principaux sont orthogonaux à $\mathbf{1}_3$. Ils définissent le plan parallèle au plan $x+y+z=1$ et passant par l'origine. Les composantes des axes sont les produits scalaires avec les vecteurs de la base canonique et on voit la projection des vecteurs de la base canonique sur le plan des axes principaux.

```
> points(pr0$scores[,1:2])
```



Cette figure est un biplot (graphe double) superposant la projection des points du nuage des lignes du tableau centré et de la base canonique de l'espace. Il est ici logique de représenter les points avant le centrage.

```
> points(as.matrix(euro78)%*%pr0$loadings[,1:2],pch="+")
```



Les deux représentations sont décalées de :

```
> pr0$center
  pri   sec   ter
0.134 0.360 0.506
```

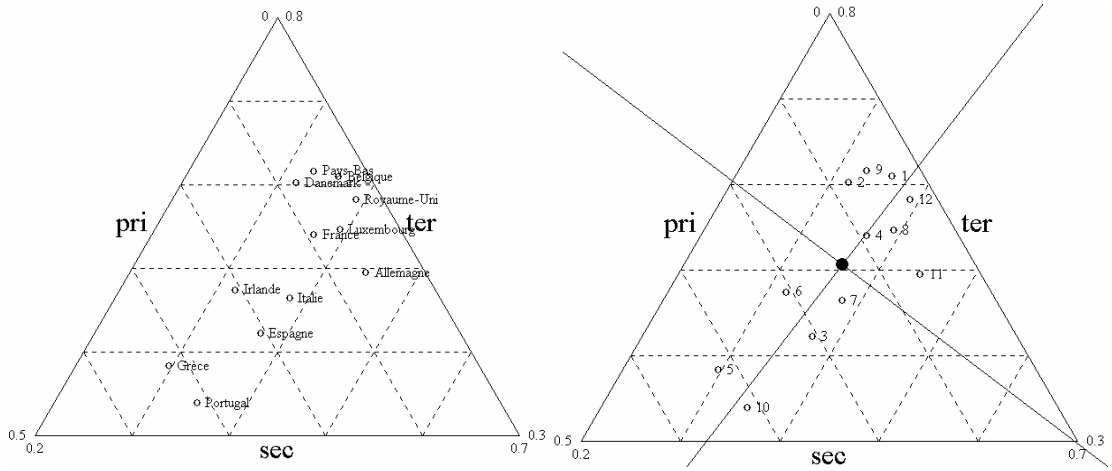
```
> pr0$center%*%pr0$loadings[, 1:2]
```

($\sum_{j=1}^{p} q_j a_{jk}$)

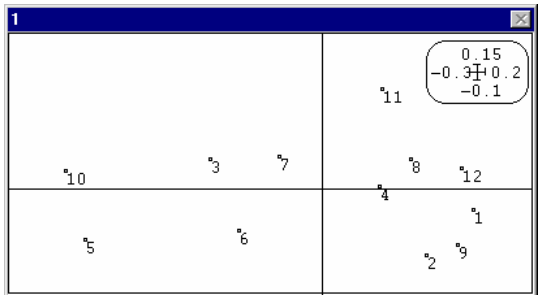
```
       Comp.1    Comp.2
  [1,] 0.2651 -0.001656
```

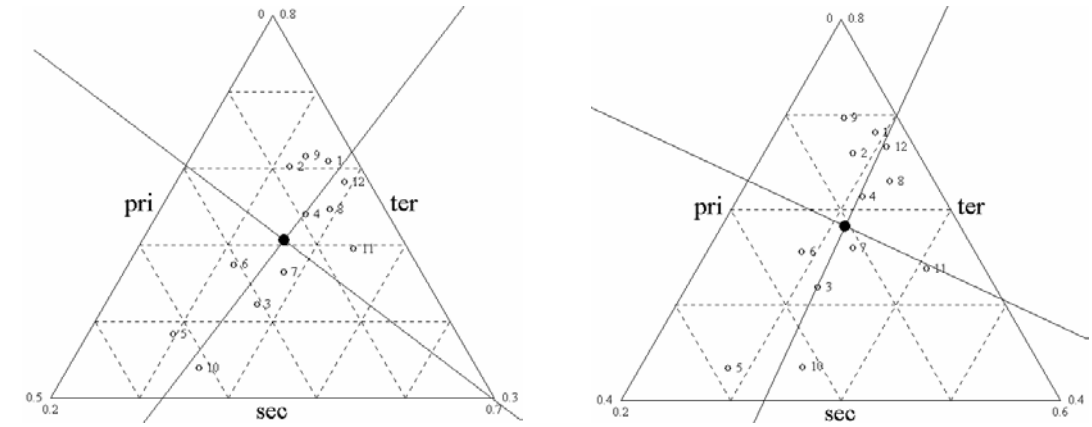
En dimension 3 (3 catégories, les données sont dans un plan), la représentation triangulaire s'impose. Un exercice tout à fait excentrique : placer sur ce dessin les axes principaux.

```
> plot.triangle(euro78,row.names(euro78))
> plot.triangle(euro78,addaxes=T)
```




On fait ici le contraire du standard : projeter sur le plan des données les axes principaux au lieu de projeter les données sur les axes principaux. L'avantage est de "voir" ce que l'on fait dans une carte factorielle :



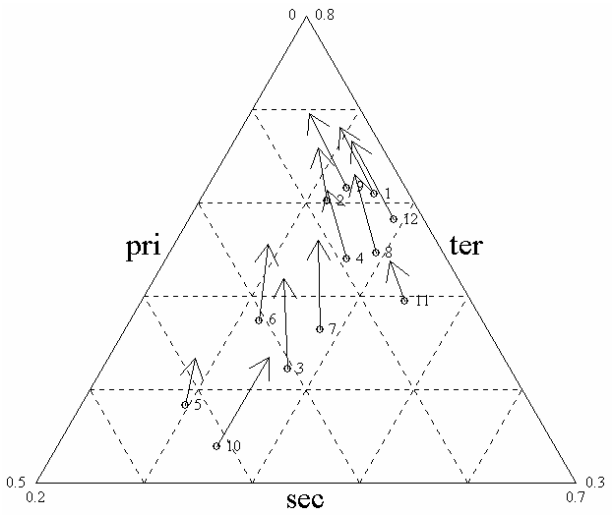

```
> plot.triangle(euro78,add=T)
> plot.triangle(euro86,add=T)
```

On peut alors poser la question du multi-tableau.

*Ce qui reste stable ou évolue concerne chaque point, le point moyen, l'hétérogénéité du nuage de points, les positions relatives entre points, les axes principaux, la typologie.*

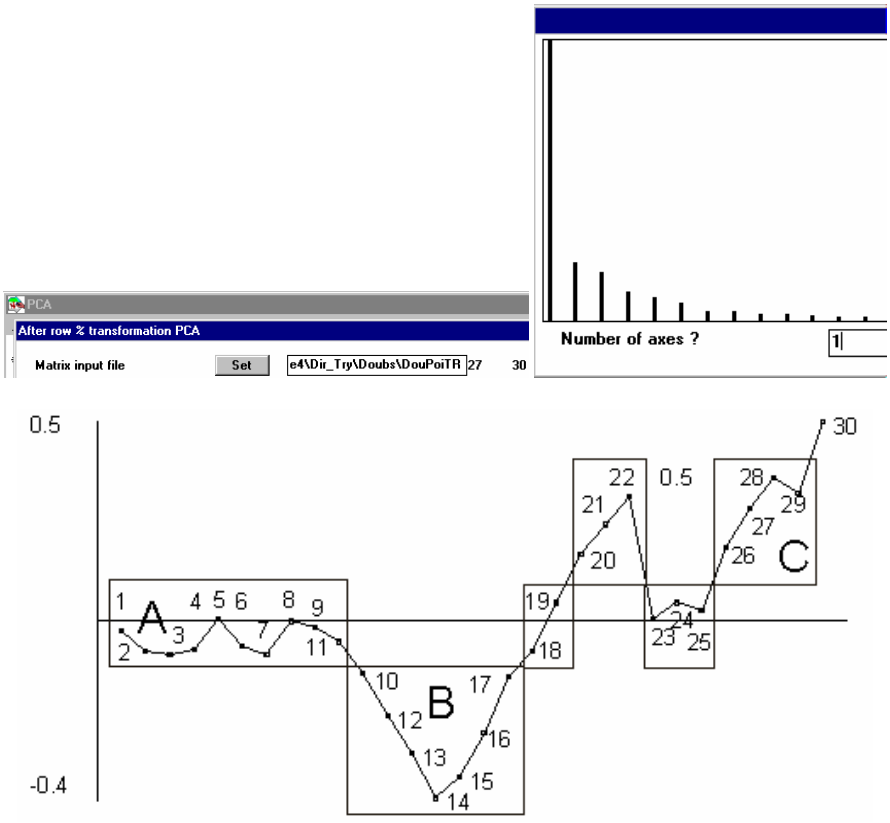
Question de fond : typologie d'évolutions ou évolution d'une typologie ?

```
> plot.bitriangle(euro78,euro86)
```



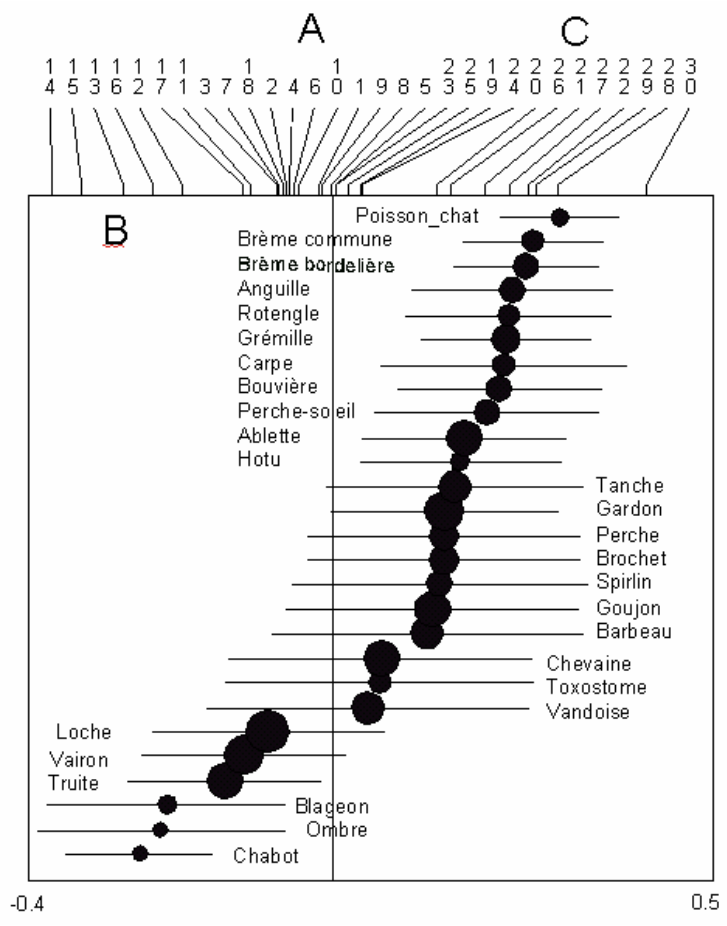

Les méthodes développées autour de ces questions sont nombreuses. On retiendra pour l'instant que la représentation triangulaire s'étend en dimension quelconque par le biais des relations d'averaging (lignes à la moyenne des catégories colonnes).

Exemple : fiche BS6 (p. 20). Carte Doubs [1] de la pile Data du logiciel ADE-4. On considère les profils de distribution de 27 espèces dans 30 stations, on garde un axe.



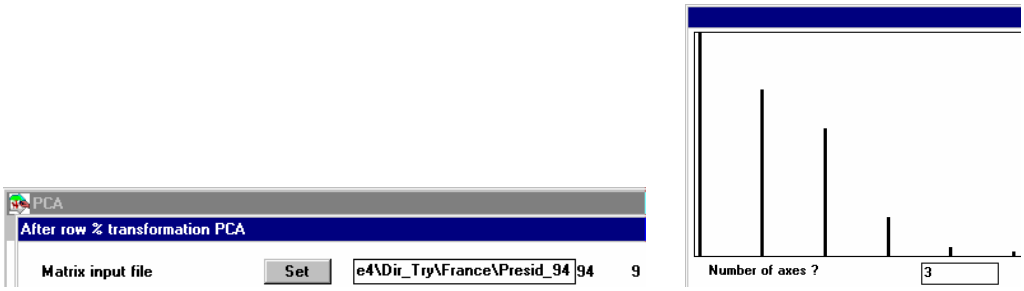

*Scores des catégories obtenu par les composantes du premier axe principal représentées dans l'espace (ordre d'apparition sur le gradient amont-aval).*

[1] Verneaux, J. (1973) Cours d'eau de Franche-Comté (Massif du Jura). Recherches écologiques sur le réseau hydrographique du Doubs. Essai de biotypologie. Thèse d'état, Besançon. 1-257.

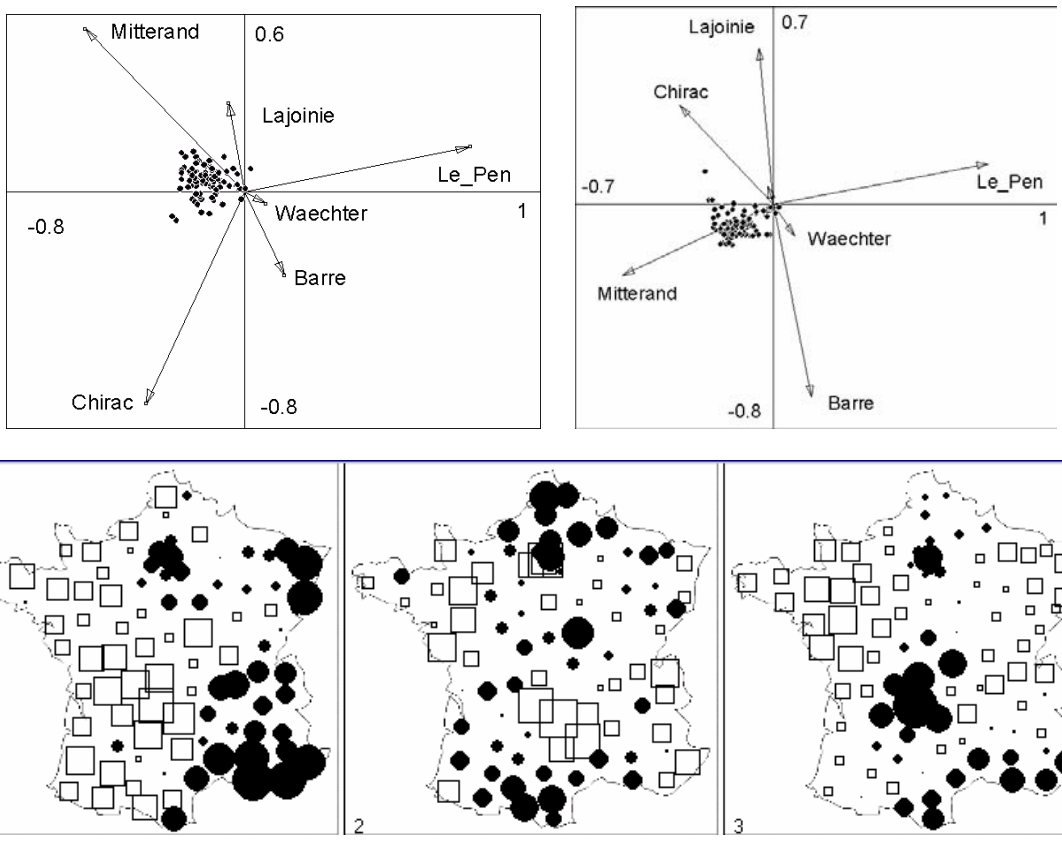



*Scores des catégories obtenus par les composantes du premier axe principal. Scores des profils (espèces) représentés par leur moyenne (cercle) et leur variance (un écart-type) rangés par moyenne croissante. Le 0 donne la position de l'espèce indifférente (profil uniforme). Le score des catégorie maximise la variance des moyennes par espèce.*

Exemple : Carte France de la pile Data du logiciel ADE-4. On considère les profils de distribution de 94 départements pour l'élection présidentielle (Premier tour, avril 1988), on garde trois axes.



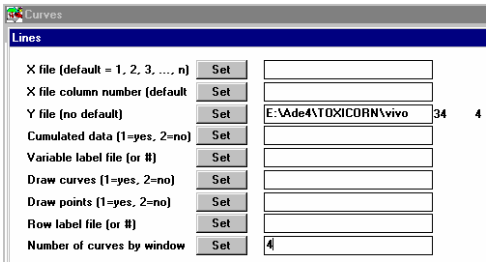

Les axes principaux sont des vecteurs à 9 composantes (candidats). Ils permettent de représenter les départements à la moyenne des distributions de fréquences (averaging). A gauche plan des axes 1 et 2, à droite plan des axes 1 et 3. L'expression d'un maximum de la variabilité ne cache pas la présence d'un compromis.

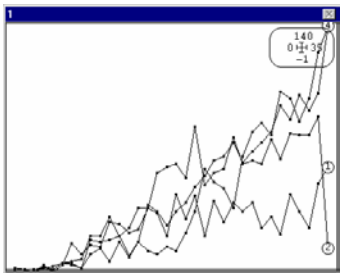



*Cartographie des coordonnées centrées (expression des composantes régionales de l'opinion).*

## 2.6 Tableaux homogènes

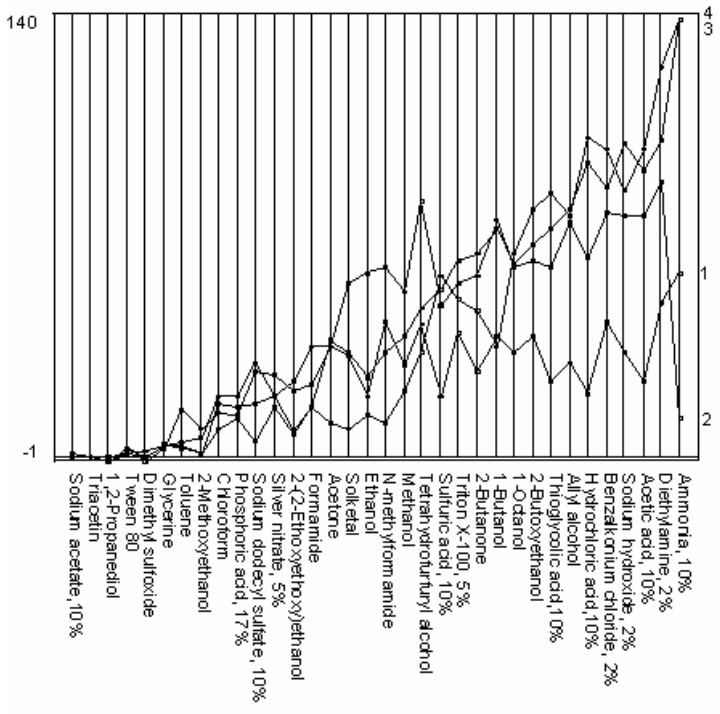
Ils comportent dans chaque cellule du tableau la mesure d'une même variable. L'ACP peut être vue comme un moyen de modéliser ces données. La question se comprend bien en toxicologie. Utiliser la carte Toxicorn **1** :



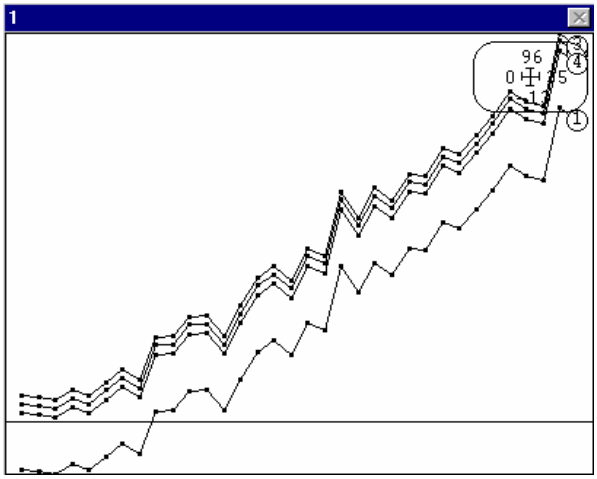




Compléter avec Graph1D: Labels et assembler :

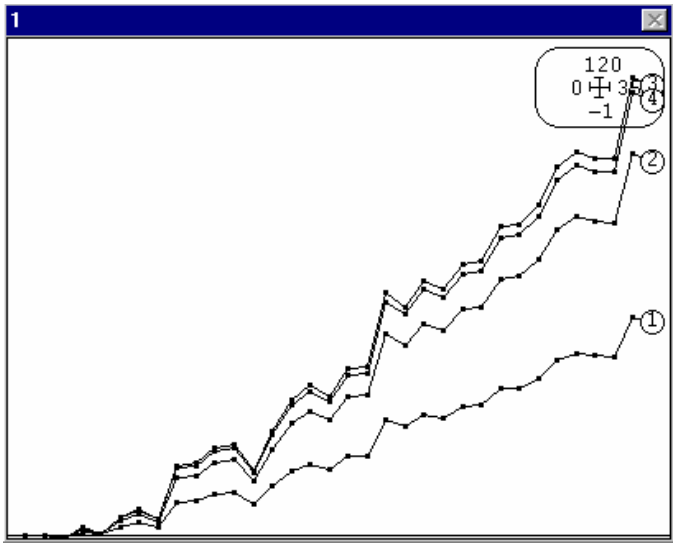
**1** Jacobs, G.A. & Martens, M.A. (1990) Quantification of eye irritation based upon in vitro changes of corneal thickness. *ATLA* : 17, 255-262.8.



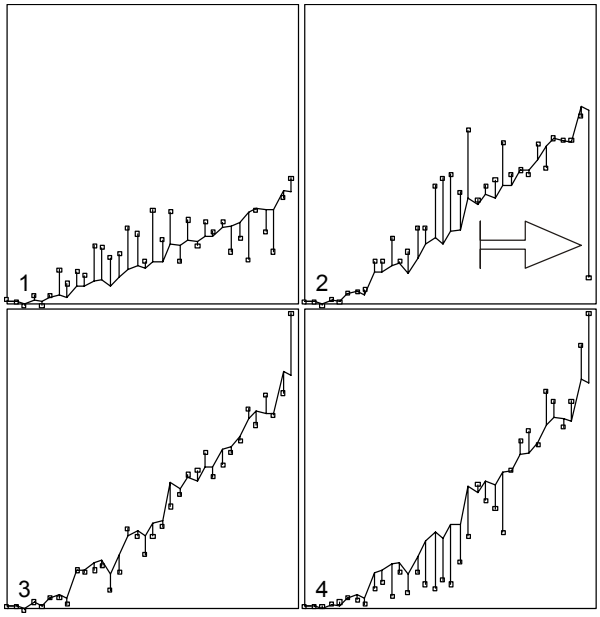

Chaque produit a un niveau de toxicité et chaque durée d'exposition (4, 24, 48 et 72 heures noté 1, 2, 3 et 4) a un effet, les deux manifestement se multiplient l'un l'autre. En effet un modèle additif donne (HTA: Double centring additive) :




On veut trouver un score des lignes $\alpha_i$ et un score des colonnes $\beta_j$ pour que l'écart au modèle $\sum_{i=1}^{n}\sum_{j=1}^{p}\left(x_{ij}-\alpha_i\beta_j\right)^2$ soit minimum. Comme le modèle $\frac{\alpha_i}{k}k\beta_j$ a les mêmes performances que le modèle $\alpha_i\beta_j$, on impose la recherche du modèle $k\alpha_i\beta_j$ avec les contraintes $\sum_{j=1}^{p}\beta_j^2=1$ et $\frac{1}{n}\sum_{i=1}^{n}\alpha_i^2=1$. $k$ est alors un facteur d'échelle libre et la solution est unique. La solution provient directement de l'ACP sans modification préalable du tableau brut (HTA: Double centring multiplicative) :




On peut alors superposer données et modèle :




Une des expériences donne un résultat qui nuit fortement à la modélisation de l'ensemble. A retenir : l'équation fondamentale est :

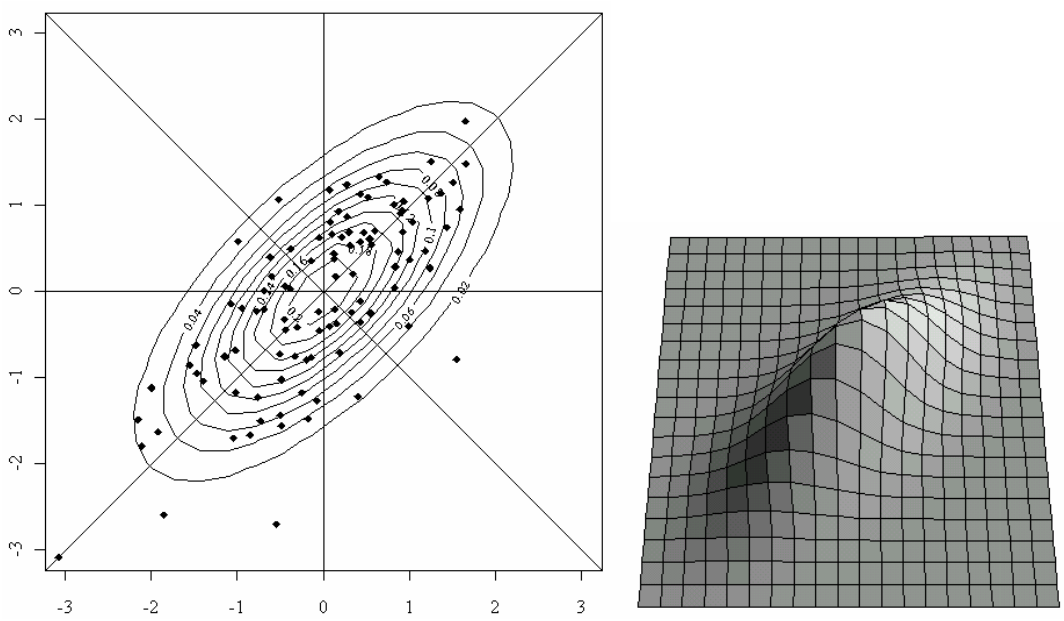
**Données = Evidences + Structures + Erreur**

Les évidences sont les propriétés des données immédiatement accessibles : certaines espèces sont rares et d'autres sont abondantes. Les individus les plus grands ont de plus grandes oreilles. Si on laisse les évidences dans les données, on dira que l'analyse enfonce des portes ouvertes. Les structures sont des propriétés associées aux comparaisons multiples entre éléments. Une technique comme l'utilisation du premier facteur de la décomposition en valeurs singulières participe selon les cas de l'élimination d'une évidence ou de la définition d'un fait peu apparent.

# 3 Modèles probabilistes

## 3.1 Loi normale bivariée

```
echa<-rmvnorm(100,mu=c(0,0),sigma=matrix(c(1,0.7,0.7,1),2,2))
echa<-as.data.frame(echa)
names(echa)<-c("x","y")
attach(echa)
par(mai=c(1,1,1,1))
plot(x,y,xlim=c(-3,3),ylim=c(-3,3),type="n")

xg<-seq(-3,3,le=20)
yg<-seq(-3,3,le=20)
xyg<-expand.grid(x=xg,y=yg)
xyg[1:5,]
z<-dmvnorm(xyg,mu=c(0,0),sigma=matrix(c(1,0.7,0.7,1),2,2))
z<-matrix(z,nrow=20)

contour(xg,yg,z)
points(x,y,pch=18)
abline(v=0)
abline(h=0)
abline(0,1)
abline(0,-1)
> persp(xg,yg,z,box=F,theta=0,phi=70,expand = 0.5, ltheta = 120, shade = 0.75)
```



On a créé un échantillon aléatoire simple d'une loi normale multivariée de moyenne (0,0) et de matrice de variances-covariances :

$$\mathbf{C} = \begin{bmatrix} 1 & 0.7 \\ 0.7 & 1 \end{bmatrix}$$

La densité s'écrit : $g(x,y) = \frac{1}{\sqrt{2\pi^2 \det(\mathbf{C})}} \exp\left( -\frac{1}{2} \begin{bmatrix} x & y \end{bmatrix} \mathbf{C}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} \right)$

```
> C <- matrix(c(1,0.7,0.7,1),2,2)
> C
     [,1] [,2]
[1,]  1.0  0.7
[2,]  0.7  1.0
> Cm1 <- solve(C,diag(1,2))%*%C
```

```
> Cm1%*%C
     [,1] [,2]
[1,]  1.0  0.7
[2,]  0.7  1.0
> det(C)
[1] 0.51


> (1/sqrt(2*pi*pi*0.51)*exp(-1*c(1,2)%*%Cm1%*%c(1,2)/2))
        [,1]
[1,] 0.02587
> dmvnorm(c(1,2),c(0,0),C)
[1] 0.02578
```

On estime le vecteur des moyennes (vraies valeurs 0 et 0) :

```
> apply(echa,2,mean)
       x        y
-0.02606 -0.07586
```

On estime la matrice des variances-covariances (vraie valeur **C**) :

```
> var(echa)
       x      y
x 0.9016 0.7003
y 0.7003 1.0303
```

On estime les vecteurs propres normés de **C**. Les vraies valeurs sont :

```
> eigen(C)
$values
[1] 1.7 0.3

$vectors
       [,1]    [,2]
[1,] 0.7071  0.7071
[2,] 0.7071 -0.7071
```

Les valeurs estimées sont :

```
> eigen(var(echa))
$values
[1] 1.6692 0.2627

$vectors
       y       x
x 0.6740  0.7387
y 0.7387 -0.6740
```

L'ACP centrée est la méthode d'estimation de ces axes principaux :

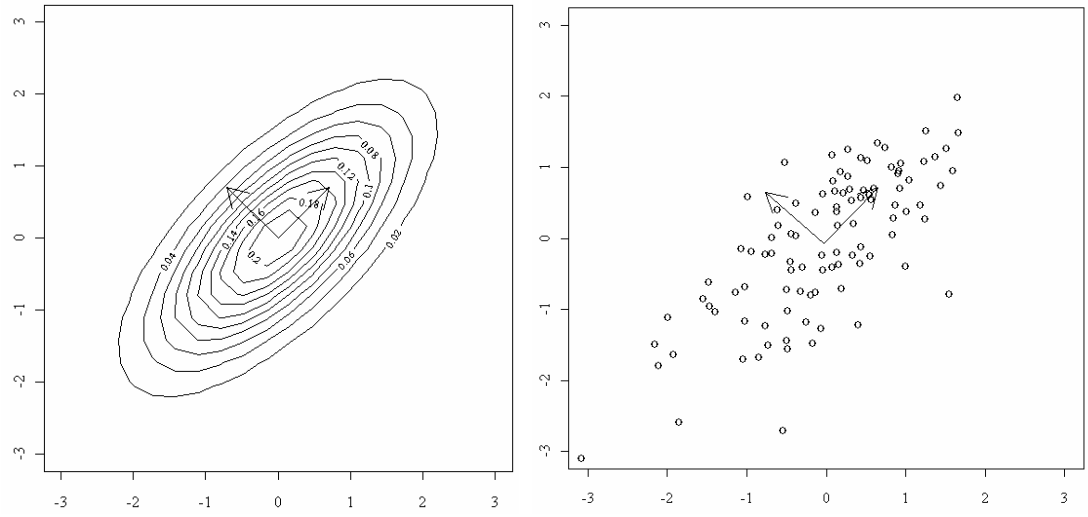
```
> prcomp(echa)
Standard deviations:
[1] 1.2920 0.5126

Rotation:
     PC1     PC2
x 0.6740 -0.7387
y 0.7387  0.6740

> pca0$sdev
 Comp. 1 Comp. 2
    1.31  0.5514
> pca0$loadings
  Comp. 1 Comp. 2
x -0.744   0.668
y -0.668  -0.744
```

La population :

```
> par(mai=c(1,1,1,1))
> plot(x,y,xlim=c(-3,3),ylim=c(-3,3),type="n")
> contour(xg,yg,z)
> arrows(0,0,1/sqrt(2),1/sqrt(2))
> arrows(0,0,-1/sqrt(2),1/sqrt(2))
```

L'échantillon :

```
> pca0 <- prcomp(echa)
> plot(echa,xlim=c(-3,3),ylim=c(-3,3))
> moy <- apply(echa,2,mean)
> arrows(moy[1],moy[2],moy[1]+pca0$rotation[1,1],moy[1]+pca0$rotation[2,1])
> arrows(moy[1],moy[2],moy[1]+pca0$rotation[1,2],moy[1]+pca0$rotation[2,2])
```



*Population* *Echantillon*

La même procédure (diagonalisation d'une matrice de covariances) prend une toute autre signification. C'est l'origine de conflits d'écoles.

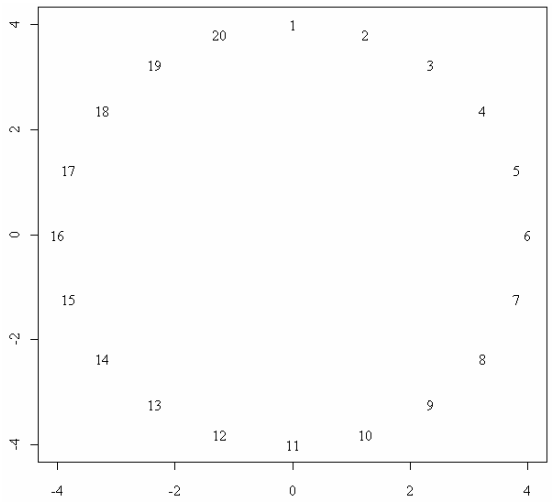
## 3.2 Erreur sphérique

Une autre manière [1] de voir l'ACP peut se présenter par l'exemple suivant.

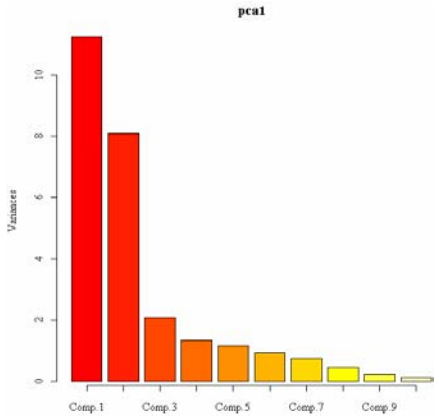
```
> x0 <- 4 * sin(seq(0, 1.9 * pi, le = 20))
> y0 <- 4 * cos(seq(0, 1.9 * pi, le = 20))
> plot(x0, y0, type = "n")
> text(x0, y0)
```

20 points sont sur un cercle.

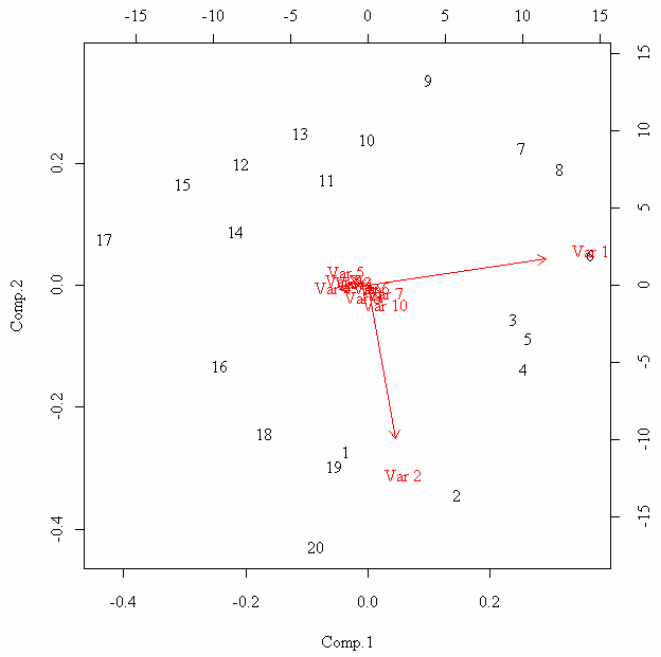
[1] Besse, P., Caussinus, H., Ferre, L. & Fine, J. (1986) Some guidelines for principal component analysis. In : COMPSTAT 1986. International Association for Statistical Computing. (Ed.) Physica-Verlag, Heidelberg. 23-30.




```
> error <- rmvnorm(20, s = diag(10))

> error
          [,1]     [,2]     [,3]       [,4]    [,5]     [,6]    [,7]     [,8]
 [1,] -1.05375 -0.52481 -0.28698  0.3521068  1.4085 -1.52848 -0.5539 -1.17653
 [2,]  0.23963  0.74546  1.13050 -1.5022218 -0.4505  0.41332 -0.7328  0.61716
 [3,]  1.13614 -2.15204  0.41437  0.0800738 -0.9318  0.05744 -0.7718 -0.06062
 [4,] -0.05693 -0.06618 -2.30162 -2.2500917  0.0465 -0.04287 -0.6095 -0.35151
 [5,] …
> error[, 1] <- error[, 1] + x0
> error[, 2] <- error[, 2] + y0
> error[1:4,  ]
         [,1]  [,2]     [,3]     [,4]   [,5]     [,6]     [,7]
> error[, 1] <- error[, 1] + x0
> error[, 2] <- error[, 2] + y0
> error[1:4,  ]
       [,1]  [,2]    [,3]     [,4]    [,5]     [,6]    [,7]     [,8]    [,9]
[1,] -1.054 3.475 -0.2870  0.35211  1.4085 -1.52848 -0.5539 -1.17653 -1.4154
[2,]  1.476 4.550  1.1305 -1.50222 -0.4505  0.41332 -0.7328  0.61716  1.0869
[3,]  3.487 1.084  0.4144  0.08007 -0.9318  0.05744 -0.7718 -0.06062  0.2953
[4,]  3.179 2.285 -2.3016 -2.25009  0.0465 -0.04287 -0.6095 -0.35151  0.7131
…
> pca1 <- princomp(error)
> pca1
Call:
princomp(x = error)

Standard deviations:
 Comp.1   Comp.2   Comp.3   Comp.4   Comp.5   Comp.6   Comp.7   Comp.8   Comp.9
Comp.10
 3.3523   2.8438   1.4376   1.1588   1.0798   0.9698   0.8639   0.6635   0.4648
0.3705

 10  variables and  20 observations.
>plot(pca1)
```




```
>biplot(pca1)
```

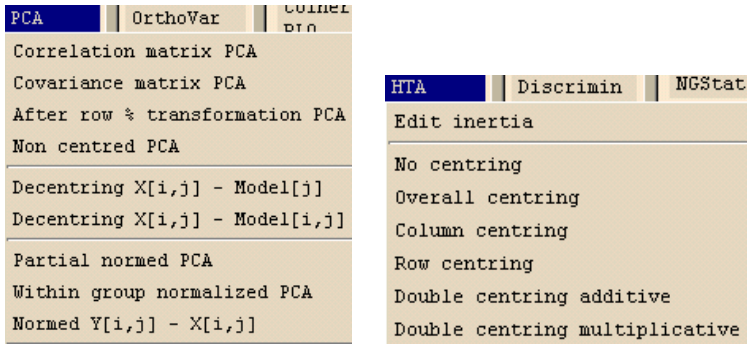



La ligne *i* du tableau est un échantillon aléatoire d'une loi normale de moyenne $\mu_i$ et de variance $\sigma^2\mathbf{I}_p$. Le nuage des moyennes est un modèle du nuage observé, chaque point se trouvant autour de sa position avec une erreur gaussienne de même variance dans toutes les directions. Les moyennes se trouvent dans un sous-espace de dimension donnée (ici un plan de dimension 2). Il convient d'estimer la dimension du sous-espace, les positions des points modèles dans ce sous-espace et le paramètre $\sigma^2$. On voit sur le graphe des valeurs propres l'estimation de la dimension, sur le biplot l'estimation du plan (ici le plan des deux premières variables) et sur ce plan l'estimation de la position des points (20 points sur un cercle). La métrique identité utilisée est alors un modèle de l'erreur (écart entre le point théorique et sa réalisation). Changer de métrique, c'est changer le modèle de l'erreur. De multiples développements sont associés à ce point de vue.

En conclusion, on peut noter ceci. L'utilisateur dit "le tableau est soumis à une analyse en composantes principales" veut dire que le tableau a été envoyé dans une procédure :

```
> prcomp
function (x, retx = TRUE, center = TRUE, scale. = FALSE, tol = NULL)
{
    x <- as.matrix(x)
    x <- scale(x, center = center, scale = scale.)
    s <- svd(x, nu = 0)
    if (!is.null(tol)) {
        rank <- sum(s$d > (s$d[1] * tol))
        if (rank < ncol(x))
            s$v <- s$v[, 1:rank, drop = FALSE]
    }
    s$d <- s$d/sqrt(max(1, nrow(x) - 1))
    dimnames(s$v) <- list(colnames(x), paste("PC", seq(len = ncol(s$v)),
        sep = ""))
    r <- list(sdev = s$d, rotation = s$v)
    if (retx)
        r$x <- x %*% s$v
    class(r) <- "prcomp"
    r
}
```

La procédure donne des valeurs propres (ou des valeurs singulières), des vecteurs et des scores. Le modèle conceptuel qui justifie cette procédure est multiple (algébrique : changement de base, numérique : modèle multiplicatif du tableau, probabiliste : estimation de modèles, géométrique : projections d'inertie maximale). Dans le résumé de l'opération par le triplet $(\mathbf{X},\mathbf{Q},\mathbf{D})$, on peut faire varier **X** (centrage, non centrage, décentrage, normalisation), on peut faire varier **Q** et **D** (pondérations, normes non diagonales) :




On peut enfin exprimer les résultats de multiples façons. Dans la suite, nous passerons en revue des "réglages" de ce modèle général utiles dans des circonstances variées.