Études des données Eaux 2018

Éva GUEGANO
Maëlle BRASSIER

Novembre 2018

* * *

* * *

Introduction

Ce projet s'insère dans l'enseignement Analyse des données et consiste à étudier un jeu de données en mettant à profit les différentes connaissances du cours. L'objectif de ce travail sera de décrire et d'analyser nos données sous plusieurs angles, par différentes représentations de données et par le biais de classifications. Il faudra par la suite les analyser et en tirer des conclusions pertinentes. Pour ce faire, nous utiliserons une ACP, que nous coderons à la main sur R et que nous comparerons ensuite à la fonction incluse dans R. Nous effectuerons également une description univariée de chaque variable, bivariée grâce à une matrice de corrélation ainsi qu'une description multivariée. Enfin nous conclurons en fonction des différentes observations que nous avons dressées et essayerons d'ouvrir une perspective plus large de ce projet.

Une Analyse en Composantes Principales est une méthode d'analyse de données qui peut se décomposer en trois grands axes. Le premier consiste à produire une synthèse des relations entre variables soit leur corrélation. On fera ensuite une topologie des individus qui revient à les regrouper en plusieurs classes homogènes ou bien encore détecter des individus isolés. Enfin, on transformera des variables corrélées entre elles en de nouvelles variables non-corrélées, appelées composantes principales et qui synthétisent l'information, la rendant moins redondante.

1. Traitement du jeu de données

Notre jeu de données se trouve dans `Eaux2018FM.txt` qui est une version mise à jour d'un ancien fichier. Il décrit 95 eaux différentes, décrites par 12 variables. Ces variables sont les suivantes :

1. Nom
2. Nature : plat (eau plate), gaz (eau gazeuse)
3. Ca : Calcium (en mg/l)
4. Mg : Magnesium (en mg/l)
5. Na : Sodium (en mg/l)
6. K : Potassium (en mg/l)
7. Cl : Chlorures (en mg/l)
8. NO3 : Nitrates (en mg/l)
9. SO4 : Sulfates (en mg/l)
10. HCO3 : bicarbonates (en mg/l)

11. PH
12. Pays : France ou Maroc

La plupart des variables sont quantitatives -toutes continues- et quelques unes qualitatives, à savoir le **Nom**, la **Nature** et le **Pays.**

Cependant, le jeu de données tel qu'il nous a été fourni ne nous permettait pas d'effectuer une ACP correctement. En effet, les données qualitatives se retranscrivent sous forme de chaînes de caractères et de ce fait sont non traitables tandis que certaines lignes comportent des valeurs Non Applicable (NA). En premier lieu, nous avons fait le choix de supprimer lesdites colonnes et lignes, afin de s'assurer d'avoir des données cohérentes. Néanmoins, la perte de ces lignes nous a semblé dommageable pour notre analyse. C'est pourquoi nous avons opté finalement pour une imputation des données manquantes. Pour ce faire, il a fallu modifier les NAs par la médiane correspondante à la variable. Nous n'avons pas utilisé la moyenne puisqu'elle est sensible à la présence de valeurs extrêmes.

À l'aide de cette commande

```
avecmediane_NA <- round(apply(data_eaux_avec_NA, 2, function(x) ifelse(is.na(x), median(x,na.rm = TRUE), x)),2)
```

nous obtenons un nouveau jeu de données, `Eaux2018FM_maj.txt` (disponible en Annexe).

Le cas de la variable **PH** a été source de questions puisque si l'on se tient à la définition du cours, une variable qualitative a un ensemble fini de valeurs. Or, on peut considérer que le PH répond à ce critère puisque son échelle varie de 1 à 14. Nous décidons tout de même de le garder puisque nous verrons par la suite qu'il joue un rôle important dans nos études.

Bien que nous ayons mis les variables qualitatives de côté, nous pouvons tout de même représenter leur distribution.

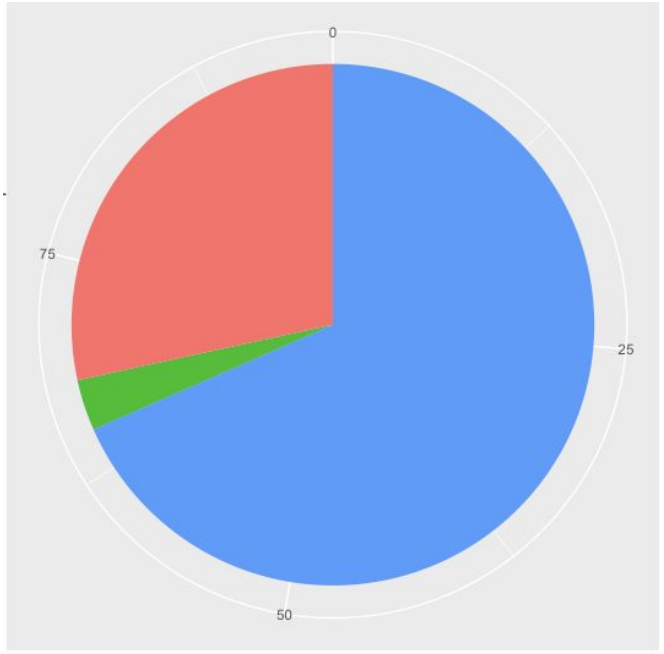

Figure 1. Distribution de la nature des eaux

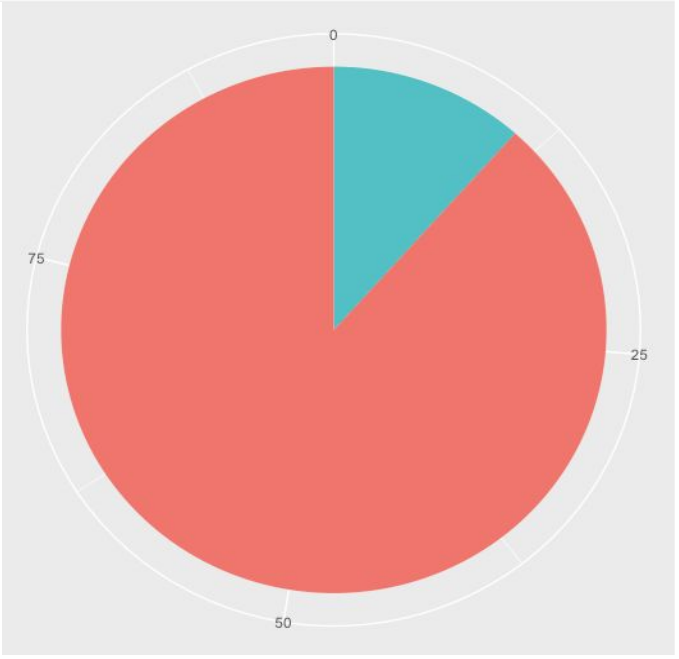

Figure 2. Distribution des pays des eaux

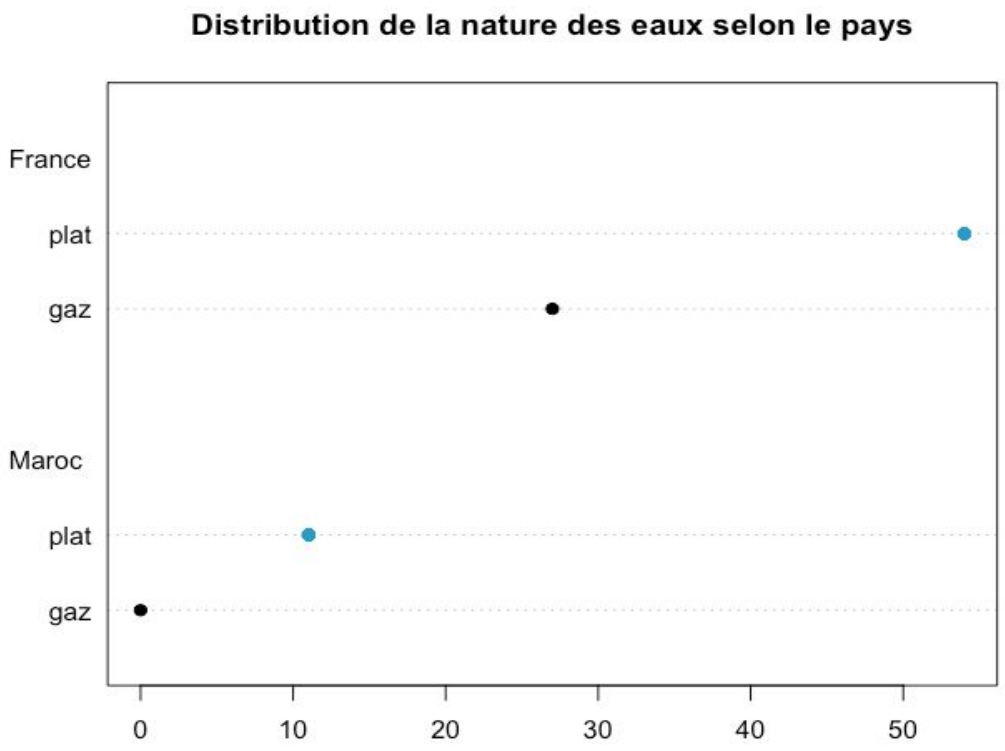

Figure 3. Distribution de la natures des eaux selon le pays

On remarque facilement que la majorité des eaux sont de nature plate et provenant de France. Il est à noter également que la totalité des eaux du Maroc sont de nature plate.

2. Description univariée des variables

Avant de commencer toute fonction ou calcul, il nous est apparu important de réaliser une description univariée de chaque variable pour pouvoir les décrire et comprendre plus en détails. En outre, cela permet de vérifier la fiabilité des caractères et détecter des valeurs extrêmes ou aberrantes. Comme nous sommes en présence de données quantitatives continues, nous pouvons aussi vérifier la distribution des données, à savoir si elles suivent une loi normale. Celles-ci peuvent se décrire par des :

- paramètres de position : moyenne, percentile, mode, médiane..
- paramètres de dispersion : variance, écart-type, amplitude, skweness..

Pour des raisons de lisibilité et d'économie, nous n'étudierons pas tous ces paramètres mais uniquement quelques uns.

2.1 Histogrammes : normalité des variables ?

Bien que le graphique de densité -qui est une version lissée de l'histogramme- soit recommandé, nous avons préféré les histogrammes afin de pouvoir tester la normalité des variables. Ci-dessous se trouve le condensé de ces histogrammes. La ligne bleue correspond à la norme.

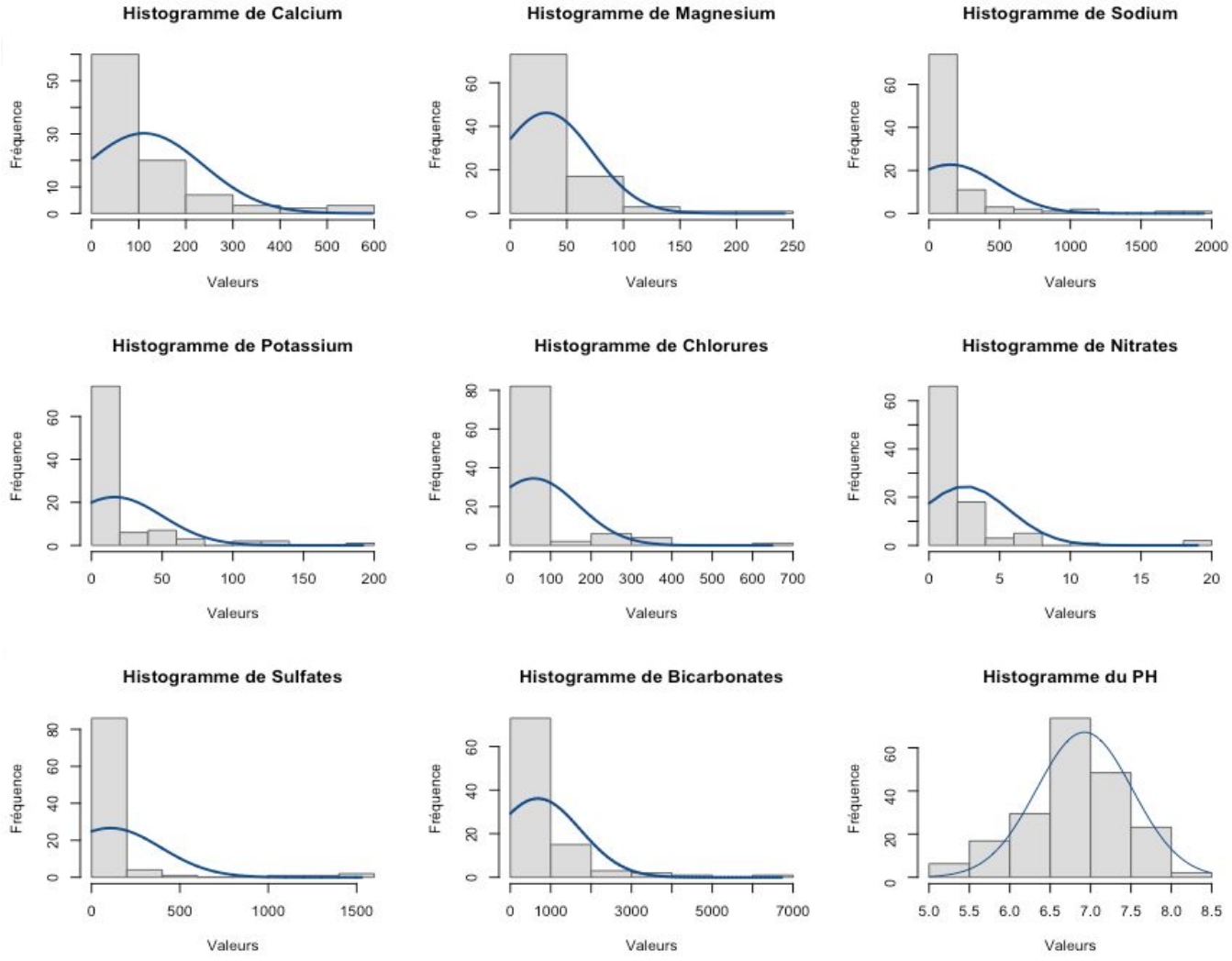

Figure 4. Ensemble des histogrammes

On constate tout de suite et à vue d'oeil que les variables ne suivent pas une loi normale à l'exception du **PH**. En effet, il suffit de comparer la forme de distribution des données à la courbe représentant une loi normale et l'on voit que la première classe de chaque histogramme dépasse largement cette courbe. Seul **PH** semble donc suivre une loi normale, avec une distribution en forme de "cloche".

Nous remarquons également que (toujours excepté PH) chaque variable a une étendue assez dispersée, allant jusque dans les milliers (**Sodium, Sulfates, Bicarbonates**..) mais se concentrant très majoritairement dans la toute première classe ; celle-ci contenant plus de 90% des effectifs.

Ces histogrammes nous ont permis d'avoir une première approche sur notre jeu de données. Mais concernant le test de normalité, il nous faut effectuer un test plus pertinent. C'est pourquoi nous allons utiliser un graphique de normalité.

2.2 Graphique de normalité : qqplot

Le graphique de normalité permet, de façon plus précise et formelle de vérifier la normalité d'une variable. Plus les points sont proches de la droite, plus la distribution empirique est normale. En exploitant les commandes `qqnorm` et `qqline` de R, nous obtenons ces graphiques suivants :

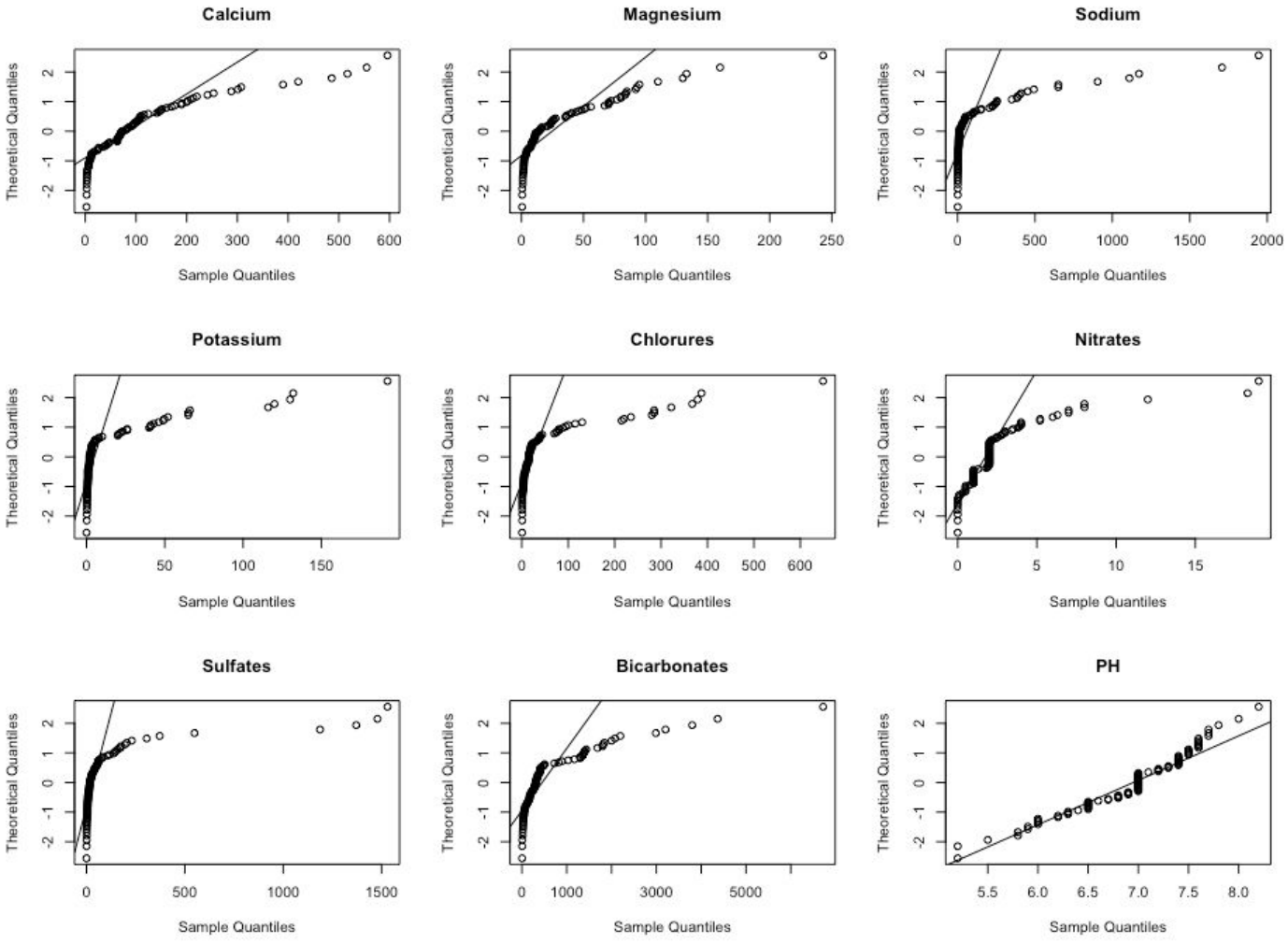

Figure 5. Ensemble des graphiques qqplot

Encore une fois, nous observons le même phénomène. Bien que les points semblent suivre dans un premier temps la droite, ils finissent par s'en éloigner à mesure que le seuil des valeurs augmente. Cela prouve bien notre hypothèse comme quoi les variables ne suivent pas une loi normale, excepté **PH.**

2.3 Boîte à moustache : médiane, quartiles, valeurs extrêmes…

Les boîtes à moustaches sont des outils intéressants et particulièrement utiles car elles représentent une distribution empirique en nous fournissant la médiane, les quartiles et les valeurs extrêmes en même temps. Pour rappel, la médiane est la valeur qui divise nos données en 2 parties égales. Les 3 quartiles quant à eux sont la représentation des données de la série qui séparent respectivement les 25, 50 et 75% inférieurs des données. Nous avons aussi les valeurs minimum et maximum ainsi que les valeurs extrêmes (outliers). Ces dernières peuvent être enlevées mais comme elles nous paraissent toutes sémantiquement plausibles (mise à part peut-être la valeur 6722,2 de **Bicarbonate** qui est le double de la 2ème valeur la plus extrême), nous préférons les conserver.

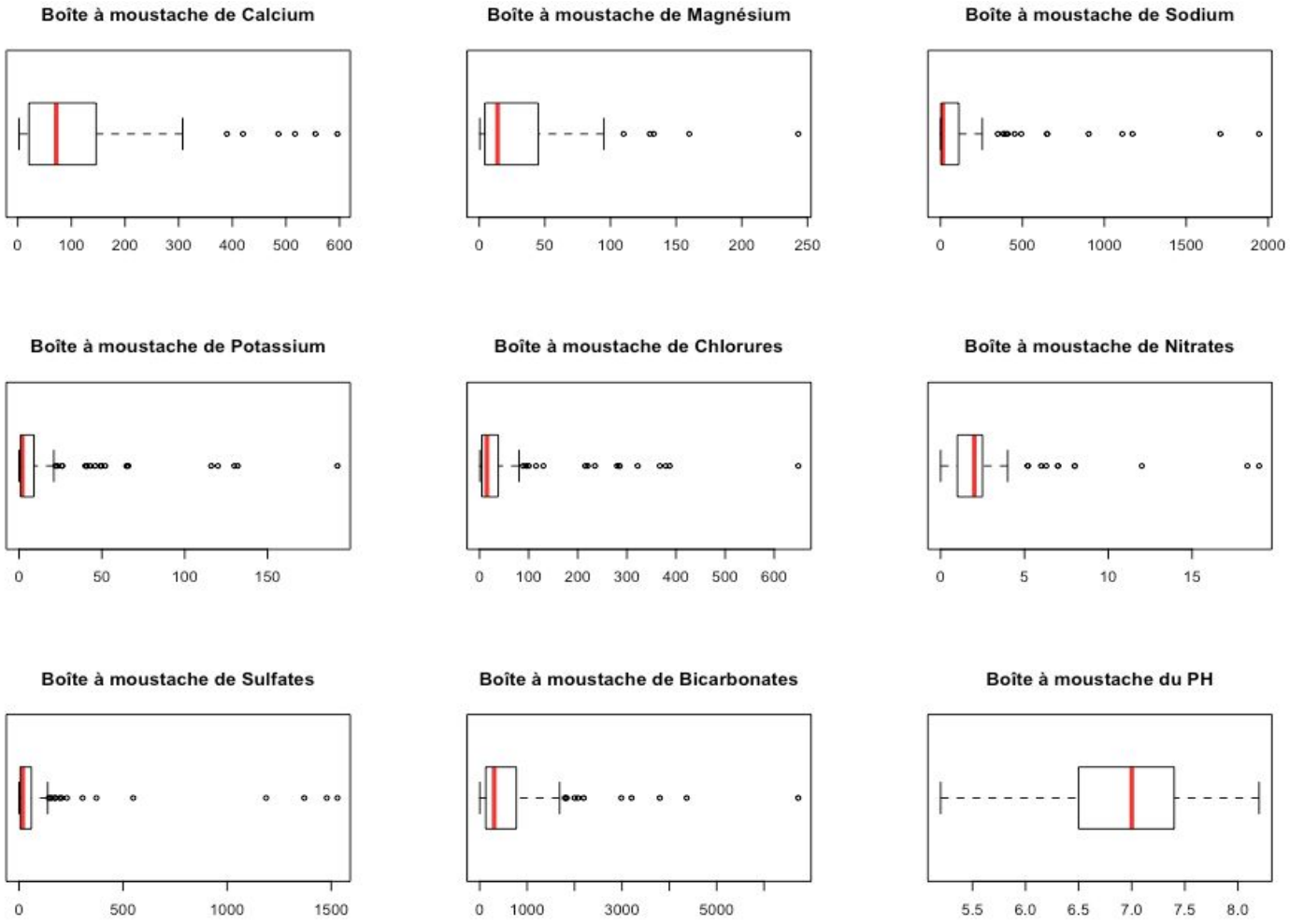

Figure 6. Ensemble des boîtes à moustache

Grâce à cette représentation graphique, nous pouvons observer la symétrie des distributions. Une symétrie n'affirme pas toujours une normalité mais une distribution normale est forcément symétrique. Cela (re)confirme ainsi notre hypothèse précédente

puisque seul **PH** a une médiane qui se situe au centre de la boîte et deux moustaches symétriques.

De plus, nous pouvons ajouter qu'il se crée deux groupes de variables : celles avec une forte présence de valeurs extrêmes dont l'étendue est importante (ex : **Bicarbonates, Sulfates, Sodium**…) et celles qui à l'inverse ont relativement moins de outliers et qui se dispersent moins (ex : **Calcium, Nitrates**...).

2.4 Coefficient de variation

Maintenant que nous avons abordé les paramètres de position, nous allons pouvoir passer à ceux de dispersion. Ignorant les variances ou écart interquartile car peu interprétables, nous optons pour le coefficient de variation.

Sa formule est telle que : $\frac{ecart-type}{moyenne}$

Plus ce coefficient est grand, plus la dispersion des valeurs autour de la moyenne est grande. Nous obtenons les résultats suivants :

Calcium : 1.142305	**Magnésium** : 1.103855	**Sodium** : 1.820392
Potassium : 1.710712	**Chlorures** : 1.851776	**Nitrates** : 1.265329
Sulfates : 2.418556	**Bicarbonates** : 1.251005	**PH** : 0.102531

Tableau 1. Résultats des coefficients de variation

On remarque que, bien que relativement proches, certaines variables possèdent un coefficient supérieur à d'autres. C'est le cas de **Sodium, Potassium, Chlorures** et **Sulfates**. On peut donc en conclure que la majorité des valeurs de ces variables se concentrent autour de la moyenne. Cela peut se remarquer par la concentration de points autour de la moyenne dans les graphiques de normalité (cela se traduit visuellement par une ligne verticale de points) et une absence presque totale de répartition dans les classes autres que la première dans les histogrammes. **PH** est bien entendu le plus faible puisque ces valeurs sont réparties de manière variée.

Cette première description des variables univariée nous a donné un premier aperçu des variables que nous allons manipuler par la suite. Elle nous a permis en outre de détecter la présence de quelques variables extrêmes, notamment celle de "6722,2" qui pourra avoir des conséquences par la suite.

3. Description multivariée des variables

3.1 Matrice de corrélation

Nous allons désormais décrire nos variables de façon multivariée, c'est-à-dire en étudiant leur lien de corrélation. Une matrice de corrélation sert alors à représenter l'ensemble des liens de corrélation entre chaque variable. Une corrélation a une valeur comprise entre -1 et +1. Plus la valeur absolue de cette corrélation est haute, plus la relation linéaire est forte. Une valeur de 1 indique une relation linéaire parfaite tandis qu'une de 0 signifie l'absence de lien de corrélation.

En utilisant une fonction sous R nous obtenons la matrice suivante :

	Ca	Mg	Na	K	Cl	NO3	SO4	HCO3	PH
Ca	1.00	0.66	0.08	0.13	0.11	0.02	0.85	0.28	-0.10
Mg	0.66	1.00	0.44	0.58	0.62	-0.03	0.46	0.58	-0.38
Na	0.08	0.44	1.00	0.87	0.63	-0.13	-0.08	0.91	-0.31
K	0.13	0.58	0.87	1.00	0.61	-0.18	-0.09	0.88	-0.44
Cl	0.11	0.62	0.63	0.61	1.00	-0.02	-0.05	0.53	-0.24
NO3	0.02	-0.03	-0.13	-0.18	-0.02	1.00	-0.06	-0.08	-0.18
SO4	0.85	0.46	-0.08	-0.09	-0.05	-0.06	1.00	-0.05	0.09
HCO3	0.28	0.58	0.91	0.88	0.53	-0.08	-0.05	1.00	-0.41
PH	-0.10	-0.38	-0.31	-0.44	-0.24	-0.18	0.09	-0.41	1.00

Figure 6. Matrice de corrélation des variables

Afin d'afficher une représentation plus pertinente, nous modifions cette matrice comme tel :

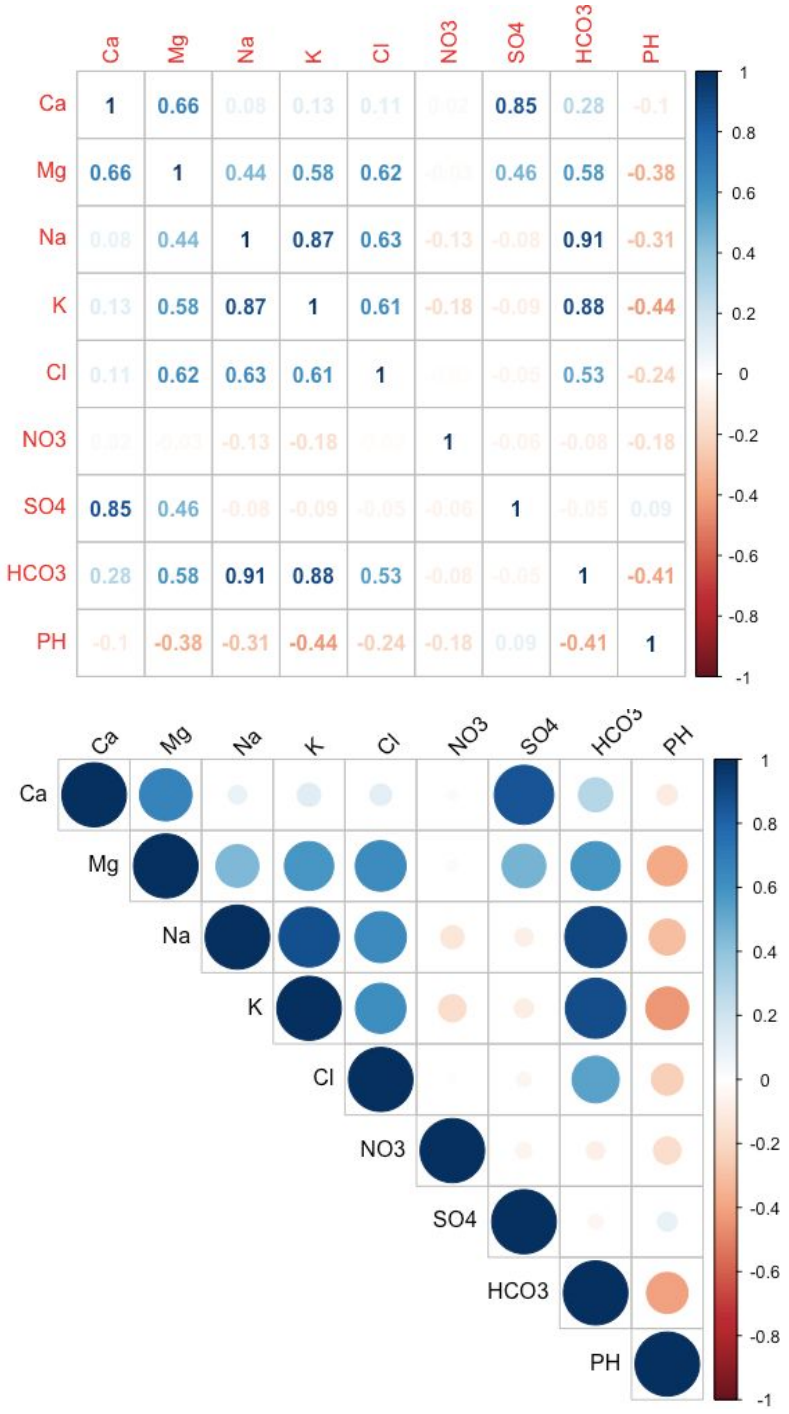

Figure 7. Différentes représentations de la matrice de corrélation

Comme nous l'avons mentionné, plus la valeur d'une relation s'approche de 0, plus leur lien est faible. À l'inverse, plus sa valeur absolue est grande, plus il est fort. Cependant, il faut également prendre en compte le signe. S'il est négatif, alors les deux caractères varient dans le sens inverse, s'il est positif ils iront dans le même sens.

Ici, on observe facilement (grâce aux couleurs) que les variables le plus fortement corrélées sont **Ca/SO4**, **Na/ HCO3** et **K/HCO**3. Au contraire, **PH** est corrélé négativement

avec presque tous les autres, notamment avec **K, Mg** et **HCO3**. Excepté légèrement avec **PH, NO3** n'est corrélé avec aucune autre variable.

Cette analyse reste à confirmer avec l'ACP.

3.2 Significativité du coefficient de corrélation

Maintenant que nous avons nos coefficients de corrélation, il s'agit maintenant de vérifier s'ils sont significatifs ou non. Cela revient à chercher si cette corrélation a été obtenu par hasard ou non. Nous allons calculer des p-values qui représentent les probabilités d'obtenir une valeur théorique supérieure à une valeur t observée (avec un nombre n d'individus et le coefficient de corrélation). Si la p-value est inférieure à $\alpha = 0.05$, alors on peut considérer la relation comme significative. Et à l'inverse, si elle est supérieure, alors on ne pourra pas.

Grâce à la commande rcorr, on a :

	Ca	Mg	Na	K	Cl	NO3	SO4	HCO3	PH
Ca		0.0000	0.5427	0.3012	0.3870	0.9059	0.0000	0.0282	0.4190
Mg	0.0000		0.0004	0.0000	0.0000	0.8107	0.0002	0.0000	0.0026
Na	0.5427	0.0004		0.0000	0.0000	0.3075	0.5194	0.0000	0.0132
K	0.3012	0.0000	0.0000		0.0000	0.1632	0.4667	0.0000	0.0004
Cl	0.3870	0.0000	0.0000	0.0000		0.8582	0.7163	0.0000	0.0620
NO3	0.9059	0.8107	0.3075	0.1632	0.8582		0.6428	0.5474	0.1702
SO4	0.0000	0.0002	0.5194	0.4667	0.7163	0.6428		0.7147	0.5072
HCO3	0.0282	0.0000	0.0000	0.0000	0.0000	0.5474	0.7147		0.0008
PH	0.4190	0.0026	0.0132	0.0004	0.0620	0.1702	0.5072	0.0008	

Figure 8. Résultats du test de significativité

Si l'on compare ces résultats à la représentation graphique de la matrice de corrélation, le lien se fait facilement. Tous les coefficients se rapprochant de 0 (et donc peu visibles) ont une significativité largement supérieure à 0,05. Les coefficients dont la valeur absolue se rapproche de 0 (et dont les points sont importants) ont une significativité inférieure à α.

Cette description multivariée, à l'instar de l'univariée, nous a permis de mieux comprendre et connaître nos variables. Les liens de corrélation entre celles-ci ont été mis en avant et nous retrouverons sûrement ces relations lors de notre ACP.

4. Réalisation d'une ACP via R

Grâce aux informations que nous avons extraites de chaque variable et de leur corrélation, nous allons pouvoir effectuer une description multivariée à l'aide d'une Analyse de Composantes Principales sous R. Il est important de préciser que nous manipulons uniquement les données françaises puisque nous considérons les eaux marocaines comme des

individus supplémentaires. En utilisant la librairie ade4 et la commande dudi.pca, nous pouvons dès lors étudier les résultats fournis.

```
acp_result_francaise=dudi.pca(matrice_quant,
scannf=FALSE, scale = TRUE, center = TRUE)
```

scale et center centrent et réduisent notre matrice de base et scannf cache le scree plot.

Nous avons également chargé la librairie **factoextra** afin d'illustrer les différents résultats de l'acp.

4.1 Nombre de facteurs retenus

Tout d'abord, intéressons nous aux valeurs propres. Elles nous apparaissent ainsi, via la commande `get_eigenvalue.`

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	4.177517914	46.4168657	46.41687
Dim.2	1.901640525	21.1293392	67.54620
Dim.3	1.176163938	13.0684882	80.61469
Dim.4	0.717272966	7.9696996	88.58439
Dim.5	0.478397037	5.3155226	93.89992
Dim.6	0.290716540	3.2301838	97.13010
Dim.7	0.189166110	2.1018457	99.23194
Dim.8	0.065923069	0.7324785	99.96442
Dim.9	0.003201903	0.0355767	100.00000

Figure 9. Valeurs propres et leurs pourcentages

On peut remarquer, à vue d'oeil, que la première dimension comporte déjà 46% et que c'est à partir de la dimension 4 que la progression ralentit avec déjà un pourcentage de variance cumulé de 88%. Un graphique nous le confirme.

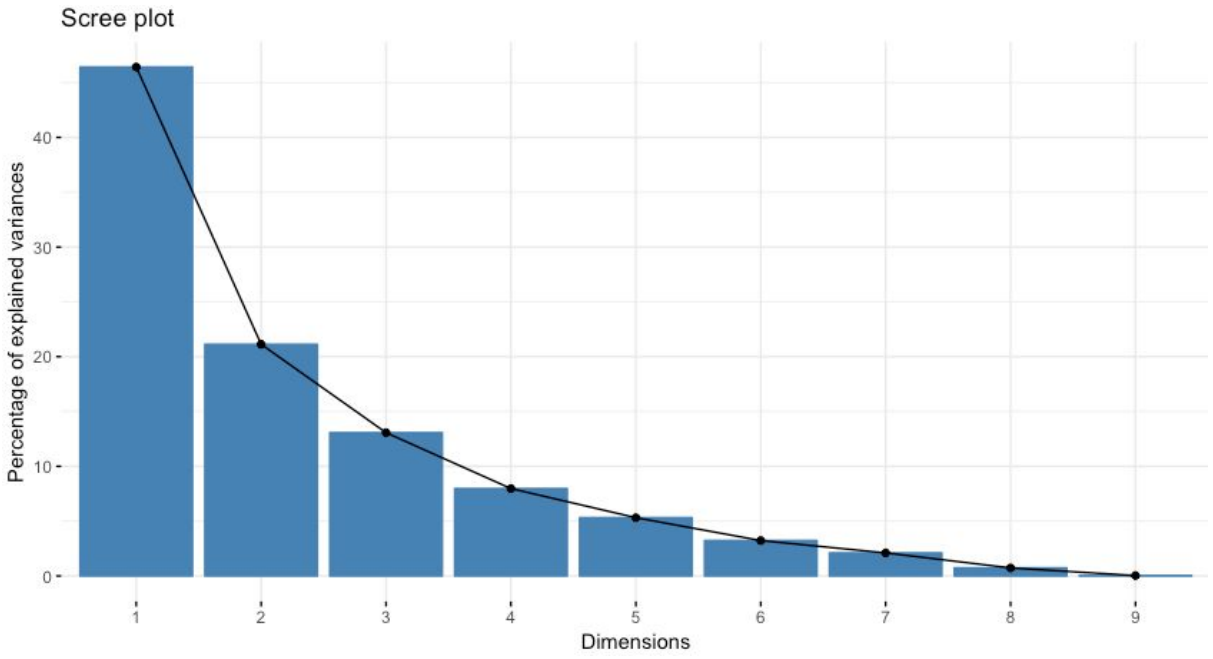

Figure 10. Visualisation des valeurs propres et le pourcentage de variances expliquées

Il y a effectivement un décrochage entre la première et deuxième dimensions. Mais nous ne pouvons pas uniquement sélectionner la première dimension car son pourcentage cumulé est inférieur à 50%. En prenant la deuxième dimension, le pourcentage cumulé passe à 70% (**règle de l'inertie minimale**), les deux dimensions possèdent des valeurs propres supérieures à 1 (**règle de Kaiser**) et il y a un écart significatif entre la deuxième et troisième dimensions. C'est pour ces raisons que nous choisissons donc de conserver les 2 premiers axes. On ne s'intéressera donc uniquement à ces composantes car elles ont la plus forte variance et on construira des nuages de points d'individus en fonction d'elles.

4.2 Analyse des variables

L'analyse des variables peut se faire de façon numérique en calculant l'inertie. Grâce à elle, nous connaissons les coordonnées des variables, leur contribution et leur qualité de représentation (en %). Mais un graphique étant toujours plus représentatif, nous optons pour ce choix visuel qui sera d'autant plus pertinent pour l'analyse.

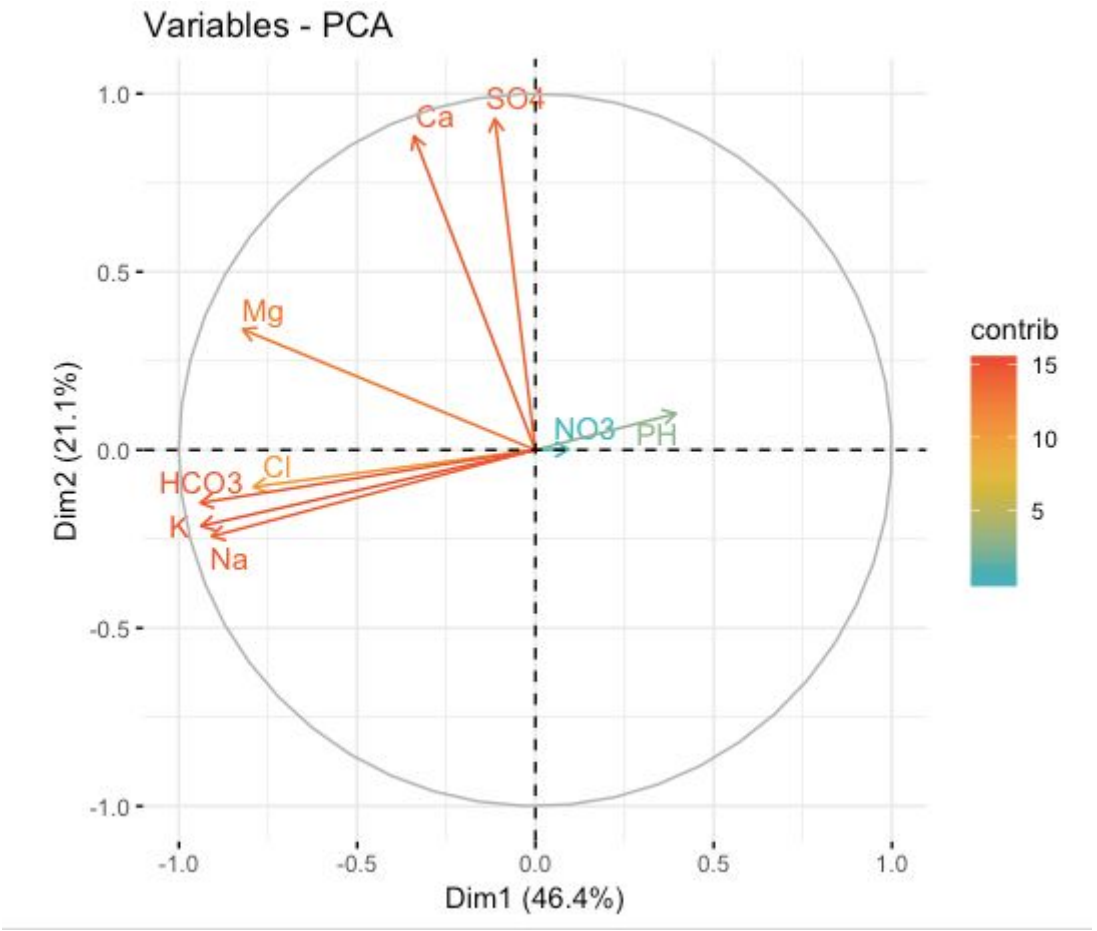

Figure 11. Représentation des variables

On sait qu'une variable est bien représentée lorsqu'elle se situe proche du cercle. À partir de cette définition, toutes les variables peuvent être décrites comme bien représentées exceptées **NO3** et **PH** qui elles s'en retrouvent éloignées.

Au niveau de la contribution des variables sur les axes, le graphique l'indique par un jeu de couleur : les variables ayant le plus contribué sont en rouge tandis qu'à l'inverse, celles qui ont le moins contribué sont en bleu ou vert. Mais on peut très bien déterminer cet indice par une autre manière. En effet, une variable qui contribue fortement à l'axe 1 est une variable qui se situe le plus aux extrémités (gauche ou droite) de l'axe des abscisses. Ici, ce sera donc **Na, K et HCO3** qui contribuent le plus à l'axe 1 négativement.

De même pour l'axe 2 qui se base sur l'axe des ordonnées. Dans ce cas, on aura **SO4** et **Ca** mais cette fois ci de façon positive.

Que ce soit pour l'axe 1 ou 2, **NO3** et **PH** sont ceux qui contribuent le moins, d'où leur couleur bleu et verte. On ne s'intéresse normalement qu'aux variables à forte contribution.

Enfin, ce graphique permet de percevoir le lien de corrélation mentionné précédemment entre les variables. Les variables qui sont corrélées positivement se situent du

même côté du graphique et partagent le même sens et la même direction alors que les variables corrélées négativement sont à l'opposé. Si les vecteur sont perpendiculaires, les variables n'ont pas de lien de corrélation. Par exemple, **Ca** et **SO4** sont côte à côte (corrélés positivement) tandis que **PH** et **K** sont à l'opposé (corrélés négativement).

Les valeurs des coordonnées des variables sont à retrouver dans l'Annexe.

4.3 Analyse des individus

De la même manière, nous allons afficher les individus sur un graphique.

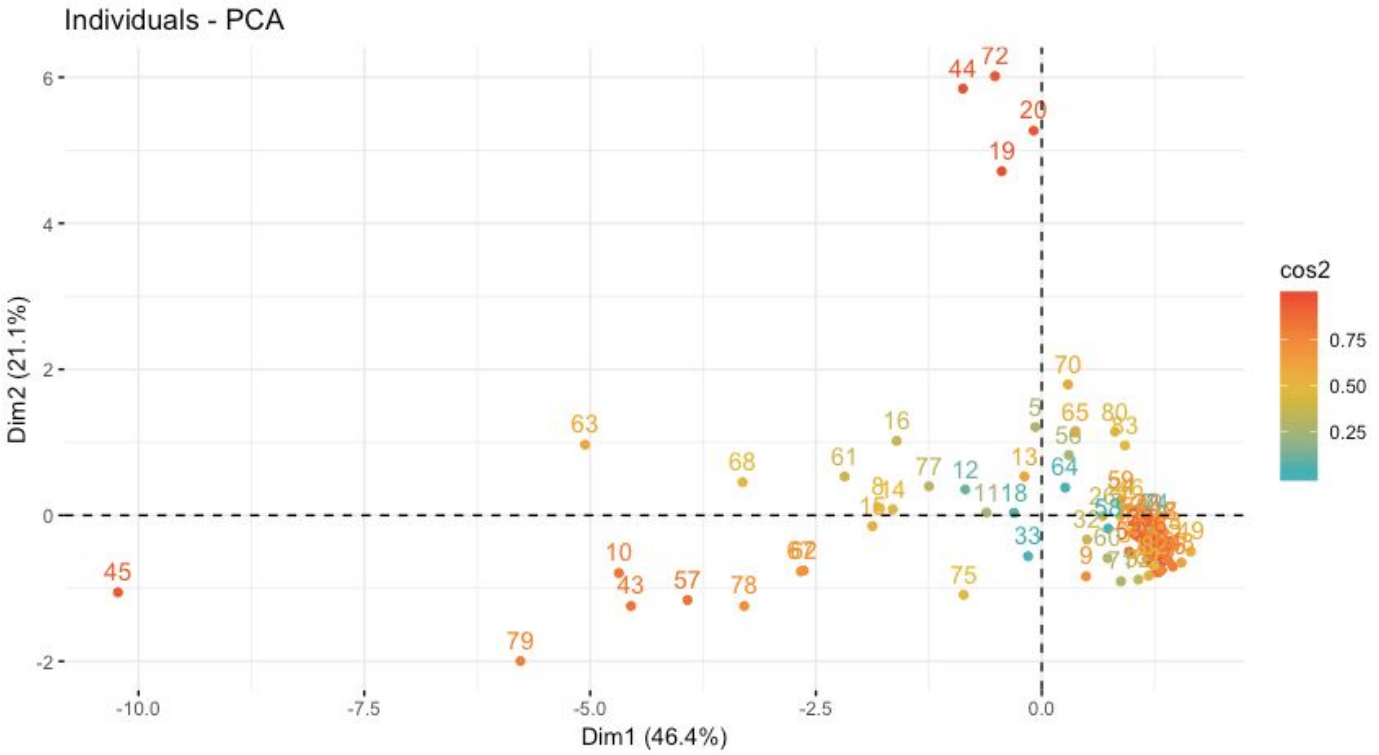

Figure 12. Représentation des individus

À l'inverse des variables, la qualité de représentation des individus se fait par un calcul qui est celui de la somme des cos^2. Plus cette valeur se rapproche de 1, plus l'individu est bien représenté. Visuellement, cela se décrit toujours par une échelle de couleurs. Les individus les mieux représentés sont **44, 72, 20** et **19** qui sont tout en haut à gauche. Les moins bien représentés sont **12**, **18**, **33** et **64** qui eux se regroupent autour de l'origine.

Dans la même logique, **44, 72, 20** et **19** seront les individus qui contribueront le plus à l'axe 2 positivement et **45** pour l'axe 1 négativement.

À travers ces représentations, nous pouvons composer 4 groupes d'eaux :

- le 1er, celui comportant le plus grand nombre d'individus qui se regroupent dans la partie centrale droite. Ce sont ceux ayant un fort taux de **Nitrates** et de **pH**. Problème avec le pH : tous les individus ont un pH compris entre 5 et 8. Avoir un fort taux de pH ne signifie rien ?.. Mais comme nous avons vu que le Nitrates et le pH ont une contribution faible, nous pouvons plutôt définir ce groupe comme celui incluant tous les individus qui ont un taux bas de toutes les autres variables. Cela se constate avec l'individu **32 (Christalline St Sophie)** qui a des valeurs faibles (63 26 99 21 33 2 60 493 7.4).
- le 2ème, avec les individus **44, 72, 20** et **19** qui ont donc un fort taux de **Calcium** mais surtout de **Sulfates**
- le 3ème, représentant le reste des individus, plutôt dans la partie inférieure gauche, se concentrent autour des variables de **Potassium, Chlorures, Bicarbonates, Sodium** et **Magnésium**
- l'individu **45 ("Hydroxydase")**, qui se retrouve seul et peut être considérée comme un individu extrême ? Il est intéressant de noter que c'est cet individu qui contient la valeur aberrante "6722.2"

Afin d'avoir une vue d'ensemble sur notre ACP, nous avons décidé de joindre la représentation des variables et des individus sur un seul graphique et également de différencier nos individus par la nature de leur eau (plate ou gazeuse). Le code correspondant se trouve dans l'Annexe.

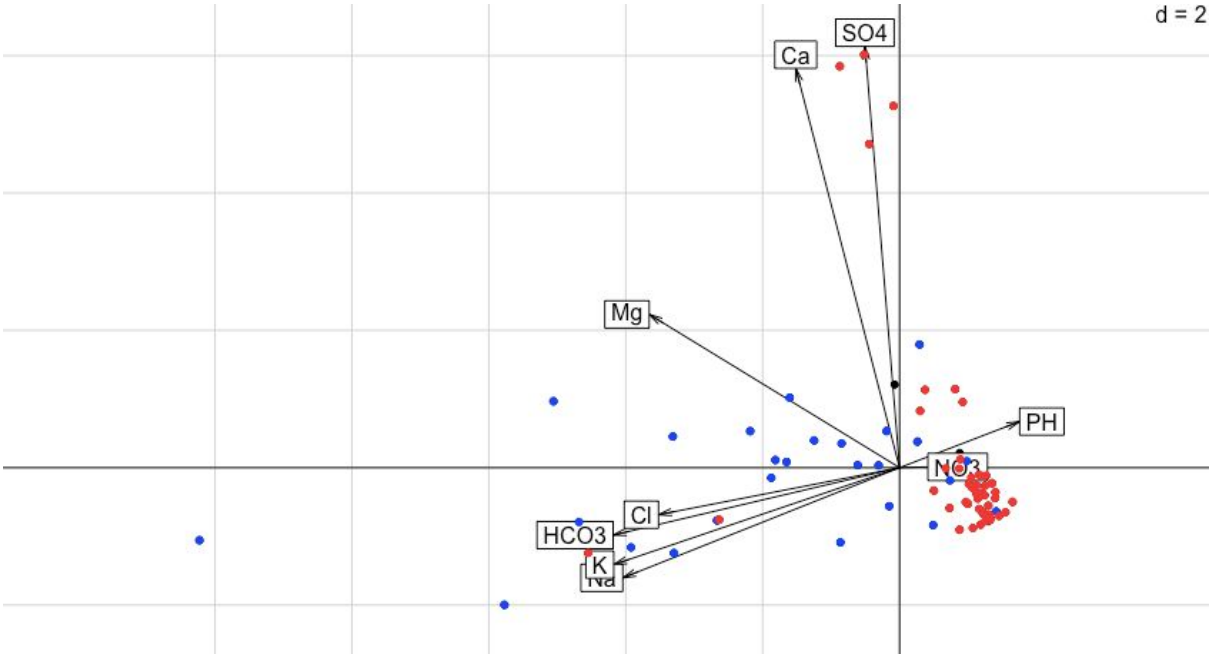

Figure 13. Représentation des variables et individus selon la nature de l'eau (rouge : plate, bleu : gazeuse)

Les deux points noirs représentent les deux individus qui n'avaient pas de renseignement sur leur nature.

On constate que les eaux plates et gazeuses correspondent aux différents groupes que nous venons de constituer. Les plates s'apparentent au tout premier groupe qui a un taux faible pour chaque variable en général et au second qui se focalise sur le **Sulfate**. Les gazeuses représentent le reste, à savoir le troisième groupe qui combine les variables **Potassium, Chlorures, Bicarbonates, Sodium** et **Magnésium** et l'individu isolé **45.**

Les valeurs des coordonnées des individus sont à retrouver dans l'Annexe.

4.4 Individus supplémentaires

Comme nous l'avons vu précédemment, les eaux marocaines sont considérées comme des individus supplémentaires. Cela signifie qu'elles n'ont pas contribué à l'ACP. Nous allons maintenant prédire les coordonnées de ces individus en utilisant uniquement les informations fournies par l'ACP que nous venons d'effectuer.

```
individus_supp <- matrice_base[85:95, 1:9]
          individus_supp[, 1:9]
```

On récupère les individus supplémentaires présents dans la matrice de base pour les stocker dans une nouvelle matrice. Puis en appliquant les résultats de l'ACP et la commande suivante

```
ind_sup_coord <- suprow(acp_result_francaise,
        individus_supp)%>%.$lisup
```

nous obtenons cette prédiction :

	Axis1	Axis2
1	0.97321468	-0.6647606
2	-0.06990465	-0.3845149
3	0.69772996	-0.3547405
4	-0.98372541	-0.1520314
5	0.98541233	-0.2065259
6	1.01149012	-0.6602196
7	0.79528806	-0.2371270
8	0.91921441	-0.2816947
9	1.04152603	-0.7430760
10	-0.08048458	-0.8630077
11	0.78230052	-0.7589264

Figure 14. Prédiction des individus supplémentaires

Nous pouvons donc placer les nouveaux individus grâce à leurs coordonnées.

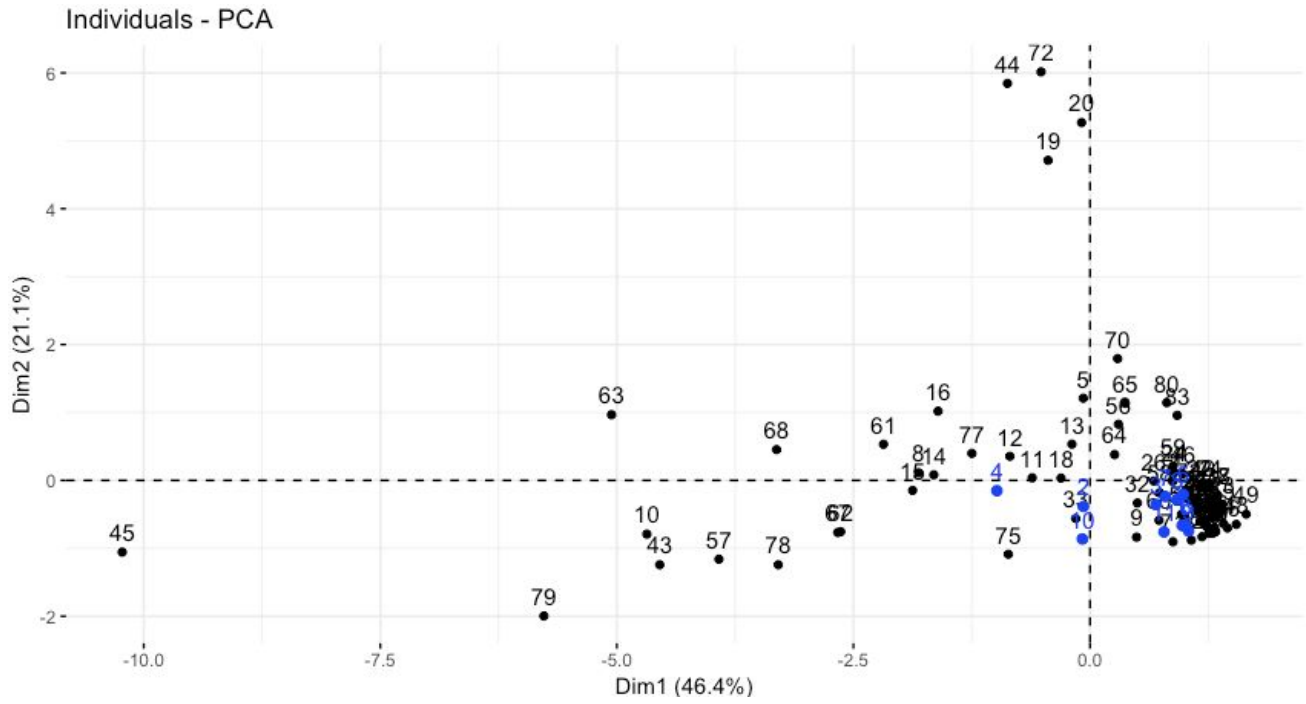

Figure 15. Représentation des individus supplémentaires

Puisque toutes les eaux marocaines sont de nature plate, il semble cohérent qu'elles se retrouvent dans le cluster des eaux plates, celles se rapprochant du **PH** et du **Nitrates.**

5. Création d'une fonction ACP

Suite à tout cela, nous avons créé à la main une fonction pour réaliser l'ACP des eaux françaises (voir le code en annexe). Nous trouvons les mêmes valeurs propres qu'avec l'ACP de R :

4.177518 1.901641 1.176164 0.717273 0.478397 0.2907165 0.1891661 0.06592307 0.003201903

Nous trouvons également les mêmes coordonnées pour les individus qu'avec l'ACP de R, les résultats étant nombreux et identiques nous ne les avons mis qu'une seule fois en annexe.

Nous avons ensuite représenté sur des graphiques, les coordonnées des variables des composantes principales que nous trouvons :

```
Coordonnées des variables :
             [,1]         [,2]        [,3]         [,4]        [,5]
[1,] -0.34054435  0.880750826  0.06250876 -0.006267927  0.10165540
[2,] -0.82108380  0.338074398  0.04390016 -0.016128125 -0.16119308
[3,] -0.90780833 -0.243004483 -0.12632194  0.165736472  0.17593199
[4,] -0.93744524 -0.213813735 -0.08208560  0.020653784  0.13175708
[5,] -0.79069230 -0.103774835  0.02843822  0.044728260 -0.56984060
[6,]  0.09279902  0.001674034  0.84761719  0.521033766  0.02092047
[7,] -0.11364931  0.929289436 -0.06425527  0.012673226  0.02321136
[8,] -0.93950988 -0.148798598 -0.06127234  0.113797869  0.24971409
[9,]  0.39376182  0.101807761 -0.64845001  0.634421813 -0.07560520
             [,6]        [,7]         [,8]          [,9]
[1,]  0.06042116  0.29968157  0.020523538  0.0120871728
[2,]  0.38431876 -0.18758665 -0.019902052  0.0084145369
[3,] -0.17042007 -0.03449270 -0.104875492  0.0349766967
[4,] -0.05545025 -0.03787110  0.215512647  0.0009487424
[5,] -0.15899783  0.10507312 -0.011838161 -0.0069284635
[6,] -0.01105002 -0.02527264  0.015997718 -0.0002111136
[7,] -0.27075923 -0.21253103 -0.006988557 -0.0121665024
[8,]  0.05787459  0.06708163 -0.081886603 -0.0395469032
[9,]  0.07200518  0.01399654  0.022602371 -0.0008363515
```

Il s'agit des mêmes coordonnées que celles trouvées avec l'ACP de R, sauf que nous avons fait le choix de garder les 9 dimensions (avec l'ACP de R nous ne gardons que les 2 premières, voir en annexe).

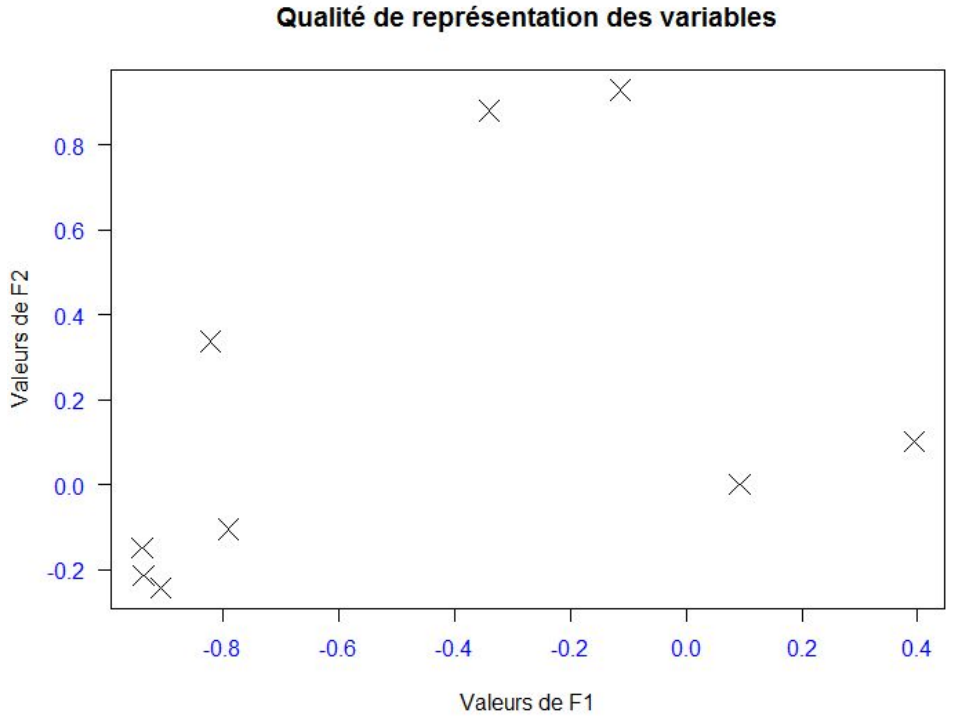

Figure 16. Représentation des variables des composantes F1 et F2

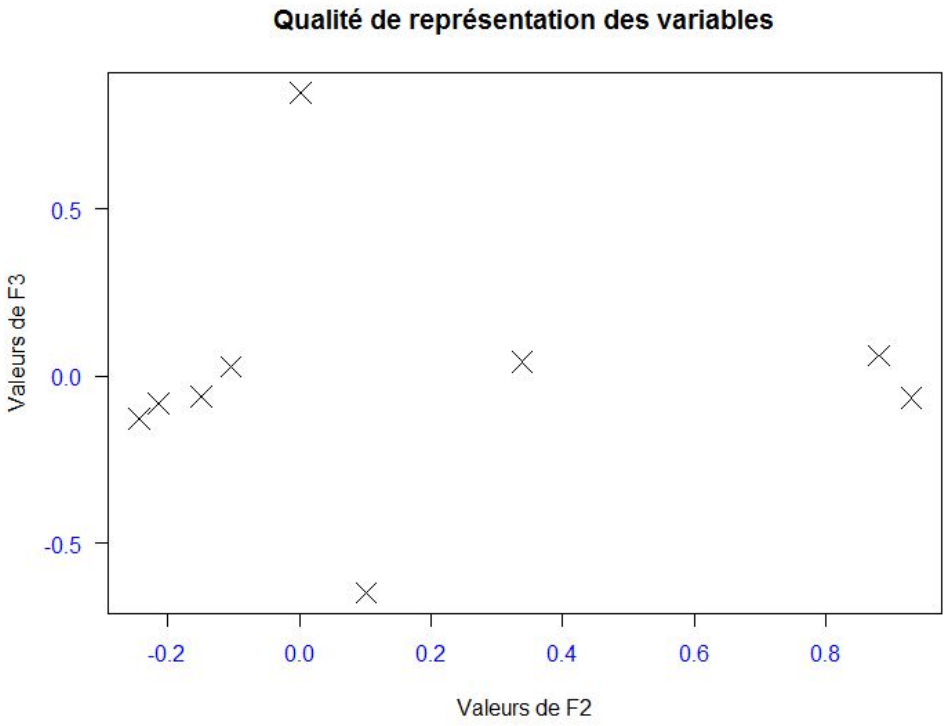

Figure 17. Représentation des variables des composantes F2 et F3

Nous pouvons également remarquer que la matrice de contribution des variables est identique à celle trouvée avec R :

```
Contribution des variables :
             [,1]         [,2]        [,3]         [,4]        [,5]
 [1,] -0.34054435  0.880750826  0.06250876 -0.006267927  0.10165540
 [2,] -0.82108380  0.338074398  0.04390016 -0.016128125 -0.16119308
 [3,] -0.90780833 -0.243004483 -0.12632194  0.165736472  0.17593199
 [4,] -0.93744524 -0.213813735 -0.08208560  0.020653784  0.13175708
 [5,] -0.79069230 -0.103774835  0.02843822  0.044728260 -0.56984060
 [6,]  0.09279902  0.001674034  0.84761719  0.521033766  0.02092047
 [7,] -0.11364931  0.929289436 -0.06425527  0.012673226  0.02321136
 [8,] -0.93950988 -0.148798598 -0.06127234  0.113797869  0.24971409
 [9,]  0.39376182  0.101807761 -0.64845001  0.634421813 -0.07560520
             [,6]         [,7]         [,8]          [,9]
 [1,]  0.06042116  0.29968157  0.020523538  0.0120871728
 [2,]  0.38431876 -0.18758665 -0.019902052  0.0084145369
 [3,] -0.17042007 -0.03449270 -0.104875492  0.0349766967
 [4,] -0.05545025 -0.03787110  0.215512647  0.0009487424
 [5,] -0.15899783  0.10507312 -0.011838161 -0.0069284635
 [6,] -0.01105002 -0.02527264  0.015997718 -0.0002111136
 [7,] -0.27075923 -0.21253103 -0.006988557 -0.0121665024
 [8,]  0.05787459  0.06708163 -0.081886603 -0.0395469032
 [9,]  0.07200518  0.01399654  0.022602371 -0.0008363515
```

Grâce à cette ACP à la main, nous avons bien réussi à retranscrire les mêmes résultats obtenus avec celle de R. Les calculs et différentes étapes que nous avons suivis nous ont appris à mieux comprendre le fonctionnement de l'ACP d'un point de vue technique.

6. Classification non-supervisée : k-means

Une classification non-supervisée est une classification qui ne se base sur aucun échantillon d'apprentissage, en soi on fait une recherche "à l'aveugle". Il s'agit de définir une partition qui respecte au mieux les similarités entre les objets. L'approche envisagée pour cette étude est une classification par partition : on considère un ensemble *n* d'objets dont les variables permettent de calculer une dissimilarité entre les objets. On optera donc pour la méthode des k-means.

Sur R, on dispose de la commande kmeans() qui nous donne les 2 clusters suivants :

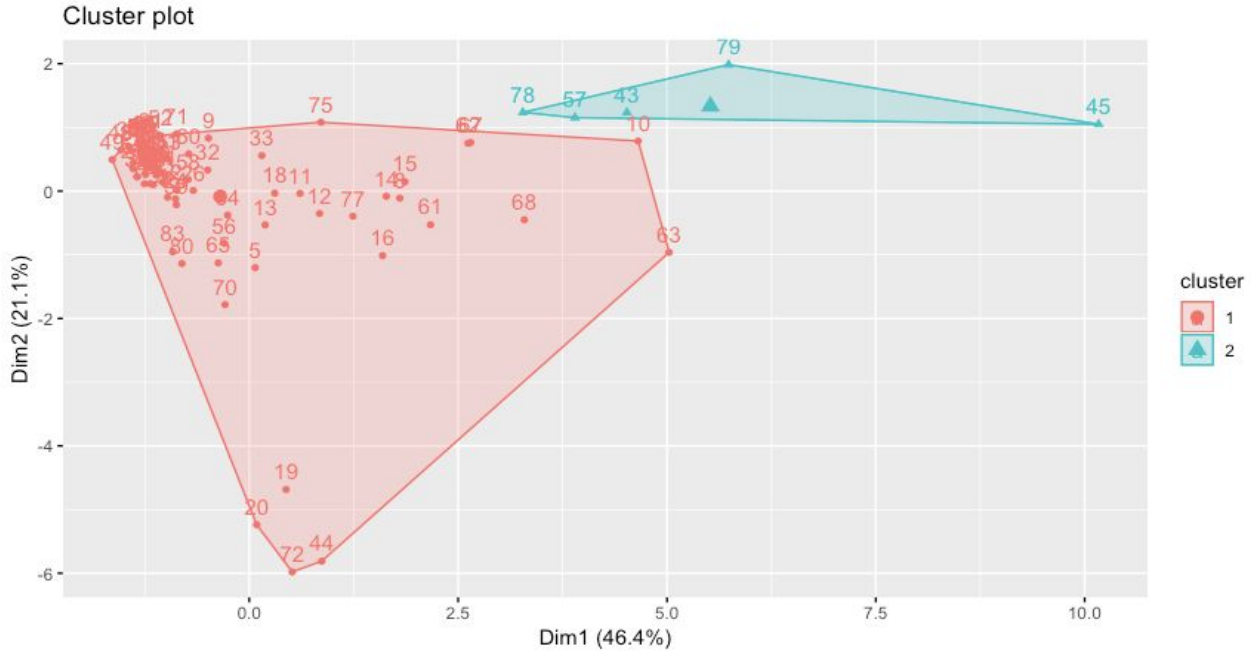

Figure 18. Représentation des 2 clusters

Cependant, ce graphique représente 2 clusters car nous avons indiqué nous même le nombre, comment-savoir quel est le nombre optimal de clusters à représenter ? Pour répondre à cette question, il suffit d'utiliser la "Average Silhouette Method" qui va calculer la qualité du clustering.

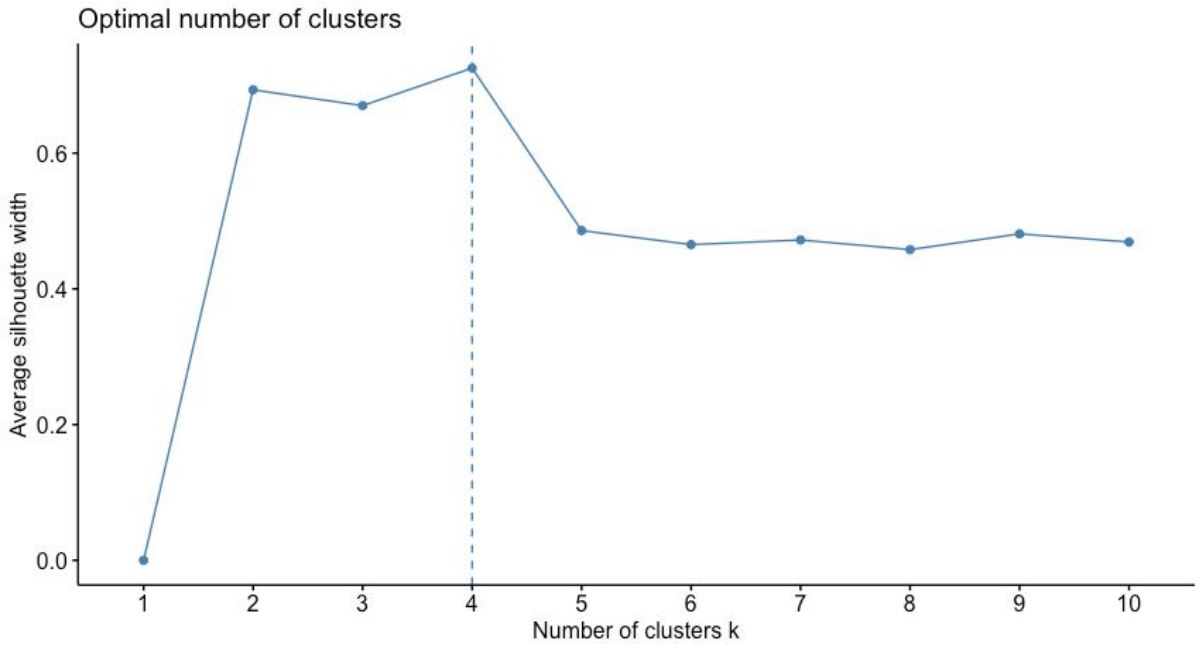

Figure 19. Recherche du nombre optimal de clusters

On voit ici que la réponse est 4.

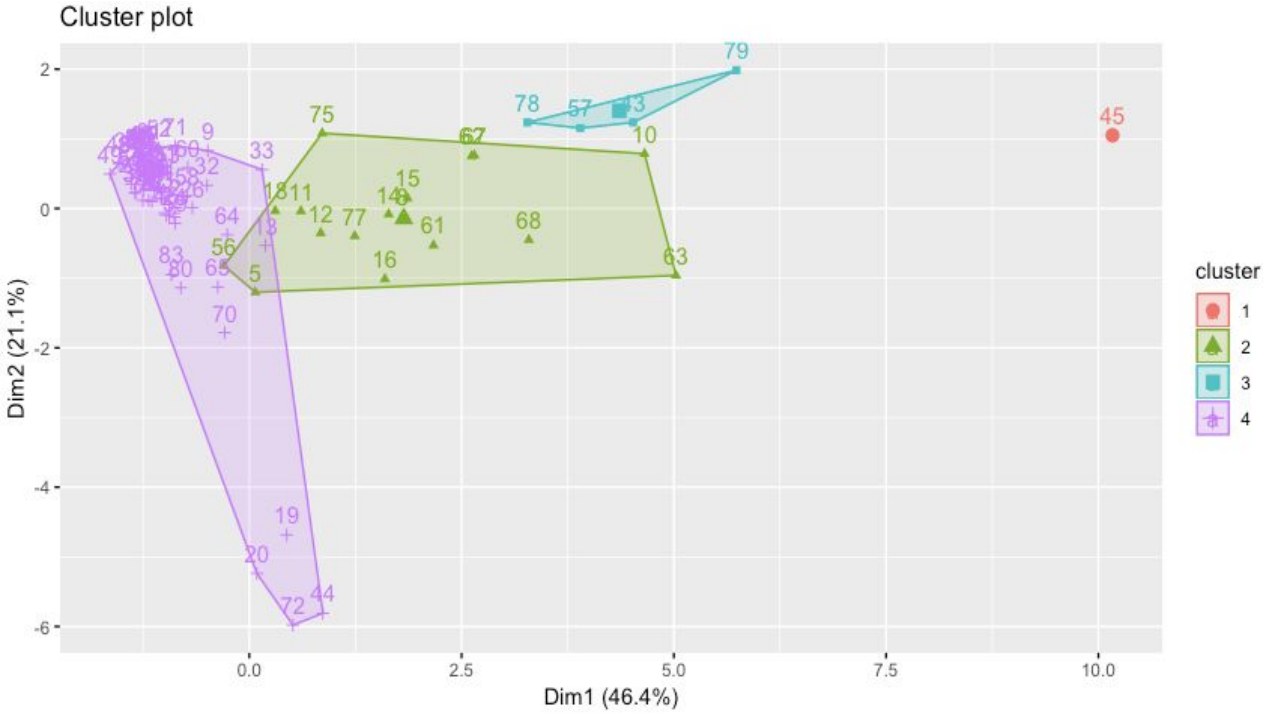

Figure 20. Représentation des 4 clusters

On remarque aisément que ces clusters sont semblables aux groupes que nous avions définis précédemment. La classification non-supervisée par k-means représente donc bien les mêmes résultats que notre ACP, bien qu'elle adopte des méthodes différentes.

7. Classification supervisée : AFD

Au contraire de la précédente, une classification supervisée est une méthode de classification qui utilise un échantillon d'apprentissage et de test afin de prédire ses classes.

Nous partageons alors notre jeu de données en échantillon d'apprentissage (80%) et échantillon de test (20%). Nous normalisons les données, puis créons un modèle de prédiction qui est le suivant :

```
Call:
lda(colonne_nature ~ ., data = train_transformed)

Prior probabilities of groups:
      gaz      plat
0.2972973 0.7027027

Group means:
             Ca         Mg         Na          K         Cl         NO3         SO4        HCO3          PH
gaz   0.2527082  0.7431893  0.8775457  0.8700875  0.6933011 -0.05155985 -0.02339041  0.9385802 -0.6217223
plat -0.1406472 -0.2923137 -0.3512772 -0.3450131 -0.2749186  0.03064040  0.02072168 -0.3932427  0.2565881

Coefficients of linear discriminants:
              LD1
Ca   -0.116254328
Mg   -0.002868868
Na    0.195177010
K     0.648465669
Cl   -0.232737673
NO3   0.104317219
SO4   0.074891577
HCO3 -1.650123012
PH    0.519942727
```

Figure 21. Modèle de prédictions

Nous avons tout en haut le pourcentage des observations d'apprentissage dans chaque groupe. Ici, nous avons 30% dans "gaz" et 70% dans "plat". Group means équivaut à la moyenne de chaque variable pour chaque groupe. Et enfin nous avons les combinaisons linéaires des variables de prédication qui sont utilisés pour former la règle de décision de LDA. On peut afficher ces informations comme tel :

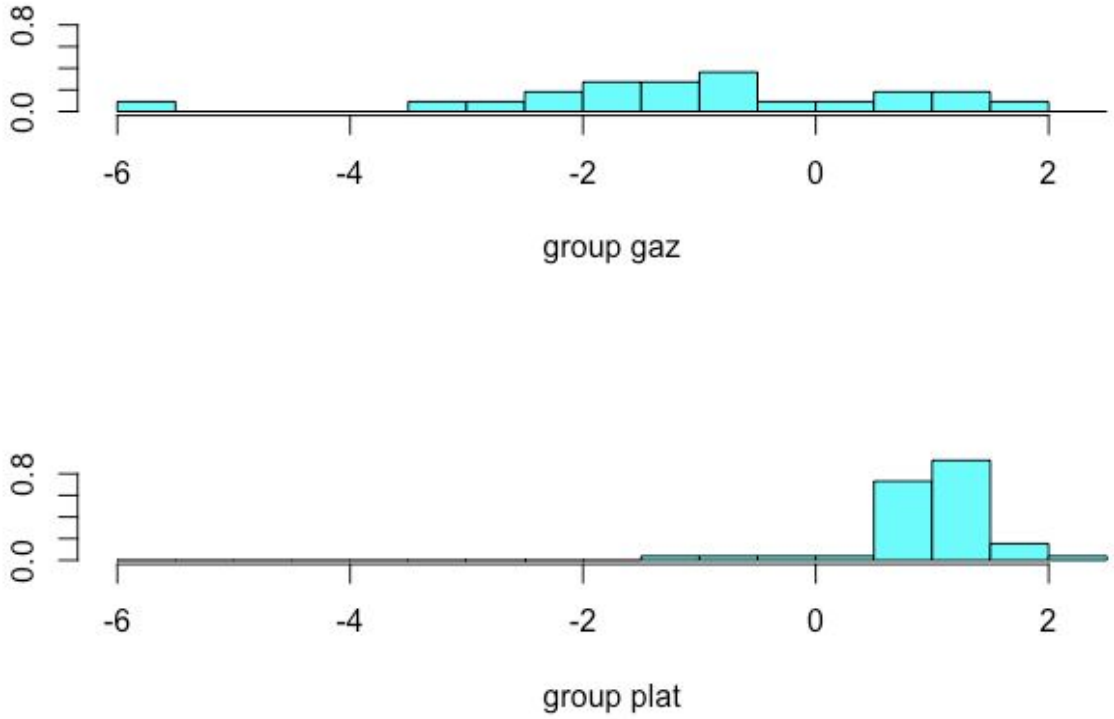

Figure 22. Représentation des prédictions

Grâce à notre modèle entraîné, nous pouvons maintenant faire des prédictions. Elles permettent de fournir les classes prédites par les informations et une matrice dont les colonnes sont les groupes, les lignes les individus et les valeurs sont les probabilités que les observations correspondantes appartiennent à tel groupe. Voici cette matrice pour quelques variables de test :

	gaz	plat
1	0.008850069	0.99114993
8	0.815278204	0.18472180
16	0.965485958	0.03451404
17	0.041204902	0.95879510
26	0.076641651	0.92335835
32	0.041181004	0.95881900

Figure 23. Matrice de prédictions

Aucun graphique n'est malheureusement possible puisque nous possédons seulement LD1, or il nécessite au moins trois groupes pour ajouter LD2.

8. Conclusion

Dans ce sujet, nous avons réalisé une étude de données sur un ensemble d'eaux françaises et marocaines, caractérisées par plusieurs variables. Nous avons au préalable analysé différentes descriptions de variables qui nous ont permis de synthétiser de nombreuses informations. Puis, en combinant et comparant une fonction écrite à la main et une disponible sur R, nous avons exécuté une ACP. À partir de celle ci, 2 composantes principales ont été créées, réduisant ainsi la perte d'informations. Une fois ceci fait, les variables et individus ont été projetés dans un plan donné que nous avons ensuite interprétés. Nous avons retenu 4 grands groupes d'individus qui correspondent chacun à un liaison avec une ou plusieurs variables. Cela a également mis en valeur la présence d'un individu aberrant, l'eau Hydroxydase que nous pourrions ou aurions pu enlever.

Ce projet nous a donc permis d'assimiler et approfondir de nombreuses connaissances sur l'analyse de données, notamment avec la manipulation d'une ACP. On peut s'interroger sur l'utilisation d'une Analyse factorielle des correspondances qui sert à trouver des liens entre deux variables qualitatives. Cela pourrait être le cas de nos deux variables Nature et Pays, si l'on rajoutait des eaux marocaines de nature gazeuse.

Annexe

Code de l'ACP à la main :

```
ACP <- function (m){

  LIG <- dim(m)[1]
  COL <- dim(m)[2]

  # On calcule une matrice avec la moyenne de chaque colonne (de même taille que m)
  mMoy <- matrix(c(colMeans(m)), nrow=LIG, ncol=COL)
  mC <- scale(m, center=TRUE, scale=FALSE)

  # On calcule une matrice où on a 1 sur les racines carrées des variances des colonnes
  mVarEnt <- 1/(sqrt(t(matrix(apply(m, 2, var) * ((LIG-1)/LIG), nrow=COL, ncol=COL))))

  # On calcule la matrice diagonale 1/s
  d <- diag(x = 1, nrow=COL, ncol=COL)* mVarEnt

  # On calcule la matrice centrée-réduite
  mCR <- mC%*%d

  # On calcule la matrice d'inertie
  matInertie <- (t(mCR)%*%mCR)/LIG

  # On calcule une matrice ligne des valeurs propres
  matValP <- eigen(matInertie)$values
  cat("Valeurs propres :", "=", matValP, "\n")

  # On calcule une matrice avec les vecteurs propres normés en colonnes
  matVecP <- eigen(matInertie)$vectors
  cat("Vecteurs propres normés : \n")
  print(matVecP)

  # On calcule une matrice avec les composantes principales en colonnes
  matCompPr <- mCR%*%matVecP
```

```
  cat("Composantes principales : \n")
  print(matCompPr)

  # On calcule la matrice qualité
  matQ <- matCompPr

  # On calcule la matrice des sommes des lignes au carré
  sumLignCar <- rowSums(matCompPr^2)

  for(i in seq(1, LIG))
  {
    for(j in seq(1, COL))
    {
      matQ[i,j] <- ((matCompPr[i,j]^2)/sumLignCar[i])
    }
  }

  # On calcule la matrice de contribution
  matContr <- matQ

  for(i in seq(1, LIG))
  {
    for(j in seq(1, COL))
    {
      matContr[i,j] <- ((matCompPr[i,j]^2)/matValP[j]) * (1/LIG)
    }
  }

  # On calcule la matrice des coordonnées des variables
  lambda <- eigen(matInertie)$values
  u <- eigen(matInertie)$vectors
  matCoordVar <- u%*%diag(lambda^0.5)

  # On calcule la contribution des variables (matrice de même taille
que la matrice des coordonnées)
  matContrVar <- matCoordVar

  for(i in seq(1, COL))
  {
    for(j in seq(1, COL))
    {
      matContrVar[i,j] <- ((matCoordVar[i,j]^2)/matValP[j])
    }
  }
```

```
  # On fait les graphes de la qualité de la représentation des
variables
    # Ici on fait F1 en abscisse et F2 en ordonnée. Les points x
et y se confondent
    graph1 <- plot(matCoordVar[,1], matCoordVar[,2], type="p",
pch=4, cex=2, col.axis="blue", las=1,
                   main="Qualité de représentation des variables",
                   xlab="Valeurs de F1",
                   ylab="Valeurs de F2")

    # Ici on fait F2 en abscisse et F3 en ordonnÃ©e
    graph2 <- plot(matCoordVar[,2], matCoordVar[,3], type="p",
pch=4, cex=2, col.axis="blue", las=1,
                   main="Qualité de représentation des variables",
                   xlab="Valeurs de F2",
                   ylab="Valeurs de F3")

}
```

N'ayant pas gardé trace de tous nos programmes R pour nos graphiques, nous mettrons en Annexe les plus importants.

Code de la figure des histogrammes :

Pour les exemples suivants, nous prendrons le cas du PH :

```
h <- hist(new_data_eaux_imputation_mediane[,9],
          col = "gainsboro", border = "dimgray",xlab = "Valeurs",
ylab = "Fréquence", main = "Histogramme du PH", prob =TRUE)
xfit <- seq(min(new_data_eaux_imputation_mediane[,9]),
max(new_data_eaux_imputation_mediane[,9]))
yfit <- dnorm(xfit, mean =
mean(new_data_eaux_imputation_mediane[,9]), sd =
sd(new_data_eaux_imputation_mediane[,9]))
yfit <- yfit * diff(h$mids[1:2]) *
length(new_data_eaux_imputation_mediane[,9])
curve(dnorm(x, mean=mean(new_data_eaux_imputation_mediane[,9]),
sd=sd(new_data_eaux_imputation_mediane[,9])), add=TRUE, col
="dodgerblue4")
```

Code de la figure des boîtes à moustaches :

```
> boxplot(new_data_eaux_imputation_mediane[,9], main="Boîte à
moustache du PH", medcol="red")
```

Code de la figure de qqplot :

```
> qqnorm(new_matrice_imputation[,9], datax=TRUE, main="PH")
> qqline(new_matrice_imputation[,9],datax=TRUE)
```

Code coefficient de variation :

```
> sd(new_data_eaux_quant[,9]) / mean(new_data_eaux_quant[,9])
```

Code matrice de corrélation : Ici, on arrondit à 2 chiffres après la virgule.

```
> matrice_correlation_eaux <- round(cor(data_des_eaux4),2)
```

Code représentations de la matrice de corrélation :

```
> library(corrplot)
> corrplot(matrice_correlation_eaux, type="upper", tl.col="black",
tl.srt=45)
```

Code significativité :

```
> rcorr(matrice_correlation_eaux[,1:9])
```

Code ACP, valeurs propres et visulation des valeurs propres :

```
> acp_result_francaise = dudi.pca(new_matrice_imputation,
scannf=FALSE, scale = TRUE, center = TRUE)
> get_eigenvalue(acp_result_francaise)
> fviz_eig(acp_result_francaise)
```

Code graphique des variables :

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

Code graphique des individus :

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

Code représentation des variables et individus selon la nature de l'eau :

```
> nature_test <- data_frame_eaux$Nature
> nature_test_col <- ifelse(nature_test== "plat", "blue", "red")
> scatter(acp_result_francaise, clab.row = 0, posieig = "none")
s.class(acp_result_francaise$li, fac=as.factor(nature_test), col =
gcol, add.plot = TRUE, cstar = 0, clabel = 0,
        cellipse = 0)
```

Code 1er clustering :

```
> cluster_2 <- kmeans(data_frame_eaux, centers = 2, nstart = 25)
> fviz_cluster(k2, data = df)
```

Code Average Silhouette Method :

```
> fviz_nbclust(data_frame_eaux, kmeans, method = "silhouette")
```

Code modèle et prédiction AFD :

```
> echantillon_apprentissage <- data_frame_eaux$Nature %>%
  createDataPartition(p = 0.8, list = FALSE)
train_eau <- matrice_eaux[echantillon_apprentissage, ]
test_eau <- matrice_eaux[-echantillon_apprentissage, ]

>preproc_param <- train_eau %>%
  preProcess(method = c("center", "scale"))

> train_transformed <- preproc_param %>% predict(train_eau)
test_transformed <- preproc_param %>% predict(test_eau)

> library(MASS)
model_eau <- lda(Nature~., data = train_transformed)
predictions_eau <- model_eau %>% predict(test_transformed)
```

Code matrice prédictions :

```
> head(predictions$posterior, 6)
```

Résultats :

Coordonnées des individus via la fonction R dudi :

```
      Dim.1       Dim.2
1    1.16334679 -0.60169765
2    0.87272807 -0.01325688
3    1.05929993 -0.29024646
4    1.29650731 -0.55400682
5   -0.07110439  1.20948765
6    1.18263594 -0.40452472
7    1.30913640 -0.76636843
8   -1.81283775  0.11145430
9    0.49169453 -0.83606968
10  -4.68290016 -0.79122756
11  -0.61131557  0.03782297
12  -0.84630844  0.35439864
13  -0.19121877  0.53333693
14  -1.65075705  0.08247795
15  -1.87459709 -0.14741489
16  -1.60589786  1.01899345
17   1.31547477 -0.69035330
18  -0.30674360  0.03366530
19  -0.44316796  4.71249978
20  -0.08921983  5.26851246
21   1.12318314 -0.25645156
22   1.15461093 -0.10114646
23   1.39965431 -0.35534984
24   0.88810917  0.12372990
25   1.09515378 -0.31059158
26   0.67436470 -0.00896278
27   1.32125203 -0.75110478
28   1.12634460 -0.38181274
29   1.19540004 -0.36792086
30   1.24889317 -0.26199638
31   0.99725666 -0.52589058
32   0.50104515 -0.33301178
33  -0.15140820 -0.56133034
34   1.25759121 -0.77870321
```

```
35   1.21121957 -0.65274924
36   1.45214090 -0.70189422
37   1.34902155 -0.22612341
38   1.35102845 -0.22791850
39   1.09515378 -0.31059158
40   1.19292024 -0.11709680
41   1.01273811 -0.23896435
42   1.13969308 -0.44975413
43  -4.54717852 -1.24236324
44  -0.87228477  5.84512726
45 -10.22755992 -1.05648446
46   0.98190094  0.09612516
47   1.29542321 -0.77896546
48   1.54632242 -0.64863491
49   1.65058503 -0.49719433
50   1.32654857 -0.75151884
51   1.32372410 -0.75097530
52   1.07085922 -0.88004138
53   0.96703181 -0.49887087
54   1.23655430 -0.68486879
55   1.39658874 -0.44320793
56   0.29983998  0.82615421
57  -3.92177537 -1.16045527
58   0.73594795 -0.18258631
59   0.87785644  0.21449797
60   0.72952336 -0.58797374
61  -2.18229316  0.53143772
62  -2.63461225 -0.75586894
63  -5.05643558  0.96695797
64   0.26031252  0.37987612
65   0.37313889  1.13435982
66   1.28500615 -0.74458504
67  -2.66601913 -0.76790452
68  -3.31239284  0.45321154
69   1.32431155 -0.75083740
70   0.29164704  1.79339334
71   0.87668167 -0.90375913
72  -0.51687558  6.01500172
73   1.04477605 -0.14335408
74   1.26263772 -0.11523275
75  -0.86322069 -1.08808673
76   1.24217165 -0.40114796
77  -1.24846512  0.39743218
78  -3.29455257 -1.24257493
79  -5.77170978 -1.99677752
80   0.81102816  1.14411323
```

```
81   1.18420059 -0.82963564
82   1.24824496 -0.68583823
83   0.92171773  0.95691662
84   1.41067291 -0.63670896
```

Coordonnées des variables via la fonction R dudi :

```
            Dim.1        Dim.2
Ca    -0.34054435  0.880750826
Mg    -0.82108380  0.338074398
Na    -0.90780833 -0.243004483
K     -0.93744524 -0.213813735
Cl    -0.79069230 -0.103774835
NO3    0.09279902  0.001674034
SO4   -0.11364931  0.929289436
HCO3  -0.93950988 -0.148798598
PH     0.39376182  0.101807761
```