TD n°1 – Analyse en Composantes Principales

Rudy LOMBARD
Nono Armel TCHIASSO

5A CFA IA & DATA

I. Chargement des données

1) L'une des premières conditions est que les variables liées au test soient quantitatives. Ces variables étudiées doivent suivre une loi normale.

2) Pour utiliser la PCA, on s'assure que nos variables soient quantitatives. D'autre part, la PCA est appliquée à des faits linéaires. Afin de vérifier si des variables sont linéaires, on détermine leur coefficient de corrélation compris entre -1 et 1. Selon le summary, on s'aperçoit qu'on a une majorité de variables quantitatives. Pour déterminer la linéarité de nos données, on applique un test de corrélation entre deux variables.

```
    Calories      Calories.from.Fat   Total.Fat
Min.   :   0.0   Min.   :   0.0    Min.   :  0.000
1st Qu.: 210.0   1st Qu.:  20.0    1st Qu.:  2.375
Median : 340.0   Median : 100.0    Median : 11.000
Mean   : 368.3   Mean   : 127.1    Mean   : 14.165
3rd Qu.: 500.0   3rd Qu.: 200.0    3rd Qu.: 22.250
Max.   :1880.0   Max.   :1060.0    Max.   :118.000
```

3) Nous devons effectuer un test de Pearson afin de déterminer la corrélation linéaire de nos variables. Avant d'effectuer cela, il faut s'assurer que notre couple de variable suit une loi normale bivariée et donc utilisation du package MVN.

```
> result$multivariateNormality
             Test         Statistic               p value Result
1 Mardia Skewness 224.229958917198 2.30464057341325e-47     NO
2 Mardia Kurtosis 22.0280366224199                     0     NO
3             MVN              <NA>                  <NA>     NO
```

Le test de normalité bivarié est négatif donc le couple de variable « Calories » et « Total.Fat » ne suivent pas une loi normale bivariée. On ne va pas se baser sur la p-value pour déterminer la corrélation linéaire de nos variables.

```
> cor.test(menu$Calories, menu$Total.Fat)

        Pearson's product-moment correlation

data:  menu$Calories and menu$Total.Fat
t = 34.048, df = 258, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8795250 0.9243604
sample estimates:
      cor
0.9044092
```

Par conséquent, on regarde le coefficient de corrélation obtenu grâce au test de corrélation linéaire. Nous avons une corrélation de 90% entre « Calories » et « Total.Fat ». Nous pouvons constater que pour cet échantillon, les calories et le gras total dans un menu mcdo sont majoritairement corrélés.

4) Pour tester l'indépendance de variables explicatives quantitatives deux à deux, il faut construire une matrice de corrélation. Cette matrice de corrélation comportera dans chaque cellule, le coefficient de corrélation correspondant à la variable en ligne et celle en colonne. Nous avons donc généré la matrice de corrélation concernant les variables « Calories », « Total.Fat », « Cholesterol », « Sodium », « Sugars » et « Protein » avec la fonction cor(). Ensuite, nous avons copié les données de cette première matrice dans une matrice vide 6x6.

```
#Affichage de la matrice de corrélation
correlation_mat <- cor(menu[,indices])

da <- c("Calories", "Total.Fat","Cholesterol","Sodium","Sugars","Protein")
m1 <- matrix(nrow = 6, ncol = 6)
dimnames(m1) <- list(da, da)
m1

#Remplissage de la matrice de corrélation
for (i in 0:36) {
  m1[i] <- correlation_mat[i]
}
m1
```

La matrice de corrélation est stockée dans la variable « correlation_mat ». On initialise une nouvelle matrice « m1 » 6x6. Enfin, on fait une boucle for qui copie le contenu de « correlation-mat » dans « m1 ». Ci-dessous, la matrice de corrélation « m1 ».

```
> m1
             Calories  Total.Fat Cholesterol     Sodium     Sugars    Protein
Calories    1.0000000  0.9044092   0.5963992  0.7123087  0.2595981  0.7878475
Total.Fat   0.9044092  1.0000000   0.6805474  0.8461584 -0.1154457  0.8077730
Cholesterol 0.5963992  0.6805474   1.0000000  0.6243619 -0.1355183  0.5615614
Sodium      0.7123087  0.8461584   0.6243619  1.0000000 -0.4265355  0.8698016
Sugars      0.2595981 -0.1154457  -0.1355183 -0.4265355  1.0000000 -0.1799396
Protein     0.7878475  0.8077730   0.5615614  0.8698016 -0.1799396  1.0000000
```

II. Corrélation linéaire entre deux variables quantitatives

1. En regardant la matrice de corrélation, on distingue que le groupe de variables « Calories » et « Total.Fat » présente un coefficient de corrélation à 0.90. Ensuite, nous avons le groupe de variables « Protein » et « Sodium » qui possède un coefficient de corrélation de 0.87.

2. On sait qu'une ACP se fait sur des variables quantitatives continues dans le but de réduire la dimension du jeu de données. Elle consiste à remplacer un ensemble de variables initiales, corrélées entre elles tout en conservant la meilleure représentation du jeu de données. Ici, nous avons des variables quantitatives continues corrélées entre elle et nous souhaitons réduire la dimension du jeu de données donc nous utilisons l'ACP.

3. Une ACP normée est une ACP réalisée sur des données centrées et réduites. Le centrage est réalisé de manière systématique en ACP (qu'elle soit normée ou non normée) et permet de translater le centre de gravité du nuage sur l'origine. L'ACP normée intervient impérativement lorsque les unités de mesures des variables sont hétérogènes. L'ACP non normée, quant à elle, intervient quand les variables sont mesurées dans la même unité.

4. Nos variables ne présentent pas la même unité de mesures. Par exemple, la variable « Calories » peut se mesurer en kilocalorie alors que la variable « Protein » se mesurera en gramme. Donc nous décidons donc d'appliquer l'ACP normée.

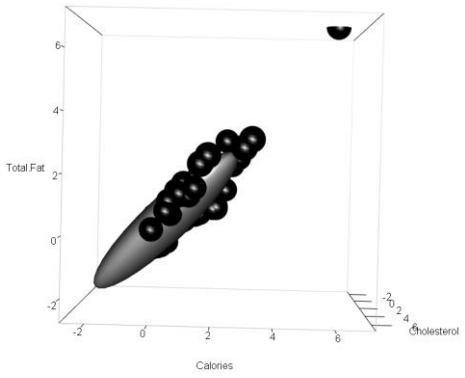
III. Représentation en trois dimensions

1. L'option « type=s » sert à afficher nos données sous forme de sphère.

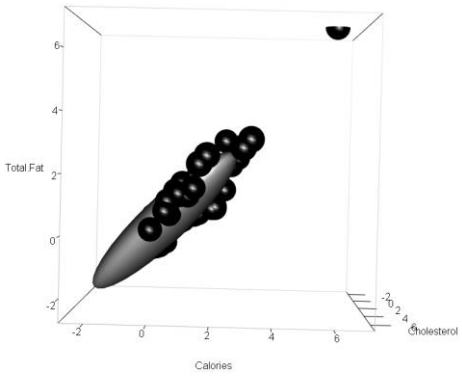
2. Les axes sont tous mis à la même échelle. Cependant, nous n'observons pas de changements significatifs quant à la répartition des données.

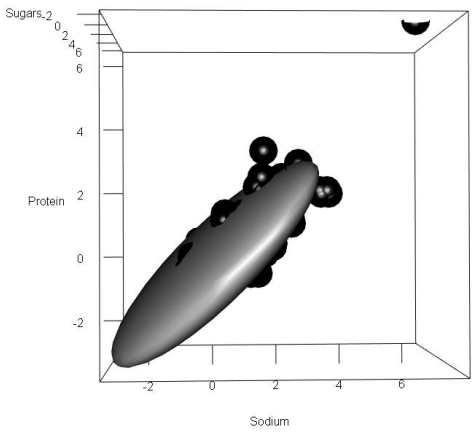
3. L'ellipse correspond à une région de confiance. On remarque que la majorité des points sont dans cette région, cela signifie que nous conservons un grand nombre d'informations. L'ellipse évalue la qualité de la combinaison des trois variables. On suppose que si on applique une ACP, il est probable que celle-ci utilise cette combinaison linéaire.

L'orientation de l'ellipse correspond au signe de la corrélation entre deux variables. Par exemple, ci-dessous nous avons l'ellipse en position ascendante (entre les variables « Calories » et « Total.Fat ») ce qui correspond à une corrélation positive :



4. On remarque une meilleure couverture de points concernant la seconde ellipse.





5. On observe qu'une plus grande concentration de points à l'intérieur de notre seconde ellipse. Cela signifie qu'on perd moins d'informations pour cette combinaison linéaire. Par conséquent, pour une ACP, on pense que la seconde combinaison linéaire est la plus adaptée.

IV. Analyse en Composantes Principales

1. Le dataframe « tab » est un tableau à 260 lignes et 3 colonnes qui contient nos trois variables « Calories », « Total.Fat » et « Cholesterol » centrées et réduites.

2. La petite différence se remarque à l'utilisation d'une variance en 1/n dans dudi.pca() contre une variance en 1/(n-1) dans scale().

3. Dans dudi.pca(), on utilise une variance en 1/n, nous devons donc appliqué cette même variance avec un centrage et réduction de toutes les valeurs du tableau data_macdo.cr (renommée menu.cr dans notre code) pour retrouver exactement les mêmes que dans dudi.pca().

```
> head(acp$tab)
      Calories  Total.Fat Cholesterol
1 -0.284683606 -0.0821929  2.35424457
2 -0.493184275 -0.4348357 -0.34376431
3  0.007217331  0.6230927 -0.11414653
4  0.340818401  0.9757356  2.64126679
5  0.132317732  0.6230927 -0.05674209
6  0.257418134  0.6230927  2.81348012
> head(apply(menu.cr, 2, scale))
         Calories  Total.Fat Cholesterol
[1,] -0.284683606 -0.0821929  2.35424457
[2,] -0.493184275 -0.4348357 -0.34376431
[3,]  0.007217331  0.6230927 -0.11414653
[4,]  0.340818401  0.9757356  2.64126679
[5,]  0.132317732  0.6230927 -0.05674209
[6,]  0.257418134  0.6230927  2.81348012
```

V. Informations associées à une ACP

1. Dans notre exemple, le premier axe factoriel extrait 82.1% de l'inertie totale, le second axe factoriel 14.9% et le troisième 2.9%. Au cumulé, nous avons 100% de l'inertie totale avec les trois axes.

```
> pve
[1] 82.082065 14.986348  2.931587
> cumsum(pve)
[1]  82.08206  97.06841 100.00000
```

2. Dans notre cas, cela représente le nombre de variables indépendantes (rang de la matrice diagonalisée).

3. Cela donne le nombre de facteurs concernés dans l'analyse.

4. Il donne les coordonnées des variables (colonnes). Les vecteurs sont de norme unité.

5. Il donne les coordonnées des individus (lignes).

6. Il donne les coordonnées des variables (colonnes). Les vecteurs sont normés à la racine carrée de la valeur propre correspondante.

7. « call » garde une trace de la façon dont ont été conduits les calculs lors de la fonction dudi.pca().

8. Le vecteur « cent » donne les moyennes des variables analysées.

9. Le vecteur « norm » donne les écarts-types (sur racine carré de n) des variables analysées.

10. Commande acp$nf

```
> acp$nf
[1] 3
```

➔ 3 facteurs sont retenus

VI. Analyse des variables

1. Les données sont toutes centrées-réduites car les variables sont toutes de longueur 1. De plus, on sait qu'il y a une forte corrélation linéaire positive entre « Calories » et « Total.Fat ». Sur la figure, on voit que l'angle entre les deux vecteurs représentant ces deux variables est très petit, ce qui signifie une forte corrélation linéaire positive.

2. L'inertie de l'axe 1 est la plus grande. L'attribut « Cholesterol » a le plus grand angle par rapport à l'axe 1 donc on en déduit que cet attribut est le moins bien représenté dans le cercle de corrélation linéaire.

3. L'angle formé entre « Calories » et « Total.Fat » est très faible, ce qui signifie une très forte corrélation linéaire positive entre les deux variables.

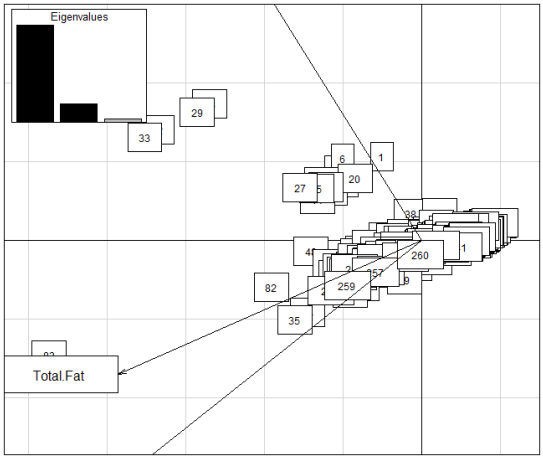
4. D'après le tableau des sommes cumulatives de la contribution relative en pourcentage de nos colonnes, on peut indiquer que « Calories » et « Total.Fat » contribuent respectivement à 86.33% et 91.87% contre 68.05% pour le « Cholesterol ». On en conclut que les variables « Calories » et « Total.Fat » sont les attributs qui ont contribué à la construction de l'axe F1. De plus, sur la figure, on observe que les angles de ces deux attributs par rapport à l'axe 1 est relativement faible comparé à celui du « Cholesterol ».

5. L'axe 1 est l'axe d'allongement maximal donc celui qui a l'inertie maximale.

6. D'après le tableau des sommes cumulatives de la contribution relative en pourcentage de nos colonnes, on peut indiquer que « Cholesterol » contribuent à 33.8% contre 9.89% pour « Calories» et 3.25% pour « Total.Fat ». On en conclut que la variable « Cholesterol » contribue assez fortement à la construction de l'axe F2.

7. L'axe 2 est la seconde direction d'allongement maximal orthogonal à l'axe 1. C'est la deuxième combinaison linéaire avec le plus d'inertie et représentant aussi la seconde composante principale.

8. La sortie de scatter(acp) qui nous montre les individus représentés sur les 3 axes correspondant aux composantes principales.



Représentation des points en fonction du plan avec affichage de l'inertie totales sur chaque axe

VII. Analyse des variables : Seconde partie

1. Remise à niveau des données scale et acp$tab

```
> head(new_menu)
        Sodium     Sugars    Protein
[1,] 0.4414709 -0.9230901 0.32107066
[2,] 0.4761983 -0.9230901 0.40875802
[3,] 0.4935619 -0.9580251 0.05800856
[4,] 0.6324712 -0.9580251 0.67182011
[5,] 0.6671985 -0.9580251 0.67182011
[6,] 0.8061077 -0.9230901 1.11025694
> head(acp$tab)
     Sodium     Sugars    Protein
1 0.4414709 -0.9230901 0.32107066
2 0.4761983 -0.9230901 0.40875802
3 0.4935619 -0.9580251 0.05800856
4 0.6324712 -0.9580251 0.67182011
5 0.6671985 -0.9580251 0.67182011
6 0.8061077 -0.9230901 1.11025694
```

2. Mesure de l'inertie pour nos 3 nouvelles variables

```
> #Mesure de l'inertie totale
> pve <- 100*acp$eig/sum(acp$eig)
> pve
[1] 68.26610 28.65429  3.07961
> cumsum(pve)
[1]  68.26610  96.92039 100.00000
```

Dans l'exemple, le premier axe factoriel extrait 68.27 % de l'inertie totale, le deuxième axe factoriel 28.65 % de l'inertie totale et le troisième axe extrait 3.08% de l'inertie totale. Le premier plan factoriel constitué des 3 premiers axes représente donc 100 % de l'inertie totale. Ceci signifie que lorsque nous projetons le nuage de points initial dans le plan défini par les 3 premiers axes factoriels, il n'y a pas de perte d'informations.

3. **Acp$rank**

```
> acp$rank
[1] 3
```

➔ Signifie qu'il y a 3 variables indépendantes, soit 3 rangs dans la matrice diagonalisée

4. **Acp$nf**

```
> acp$nf
[1] 3
```

➔ Nous avons 3 facteurs qui sont conservé pour notre analyse.

➔

5. **Acp$c1**

```
> acp$c1
                CS1       CS2        CS3
Sodium    0.6772419 0.1179021  0.7262524
Sugars   -0.3834587 0.8989821  0.2116377
Protein   0.6279353 0.4218178 -0.6540390
```

➔ Nous renvoie les coordonnées des variables.

6. **Acp$l1**

```
> head(acp$l1)
        RS1        RS2        RS3
1 0.5971453 -0.6928218 -0.2787724
2 0.6520555 -0.6485118 -0.3844796
3 0.5157300 -0.8397526  0.3874145
4 0.8507985 -0.5428310 -0.6014623
5 0.8672329 -0.5384149 -0.5184868
6 1.1159888 -0.2874079 -1.1056736
```

➔ Nous renvoie les coordonnées des individus

7. **Acp$co**

```
> acp$co
             Comp1      Comp2       Comp3
Sodium    0.9691857 0.1093143  0.22074763
Sugars   -0.5487592 0.8335018  0.06432823
Protein   0.8986242 0.3910933 -0.19879805
```

➔ Nous renvoie les coordonnées avec des vecteurs normés à la racine carré de la valeur propre correspondante.

8. **Acp$call**

```
> acp$call
dudi.pca(df = menu[, list], center = TRUE, scale = TRUE, scan
nf = FALSE,
```

➔ Trace de la façon dont ont été conduits les calculs lors de l'appel de la fonction dudi.pca()

9. **Acp$cent**

```
> acp$cent
   Sodium    Sugars   Protein
495.75000  29.42308  13.33846
```

➔ Nous renvoie les moyennes des variables analysées

10. **Acp$norm**

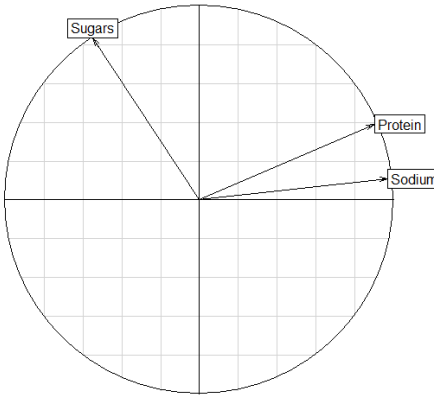
```
> acp$norm
   Sodium    Sugars   Protein
575.91559  28.62459  11.40415
```

➔ Ecarts-types des variables analysées

11. **Acp$nf**

```
> acp$nf
[1] 3
```

-> Nous avons 3 facteurs qui ont été retenu pour notre analyse.



Cercle des corrélations linéaires pour les variables « Sugars », « Protein » et « Sodium »

12. Un attribut est bien représenté lorsque sa longueur est égale au rayon du cercle des corrélations.

13. L'angle entre 2 variables nous aide à déterminer la corrélation linéaire entre ces 2 variables, plus il est faible, plus les variables sont corrélées positivement, ici on observe que l'angle entre « Protein » et « Sodium » est très faible, nous en déduisons que l'attribut le plus corrélé positivement à « Protein » est l'attribut Sodium.

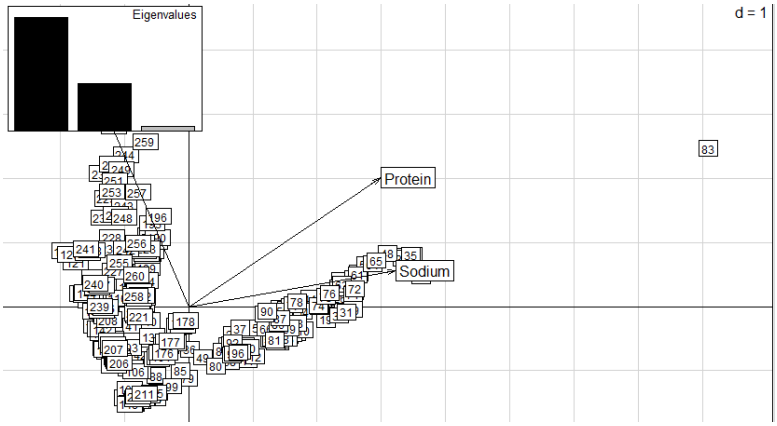
14.

```
Cumulative sum of column relative contributions (%):
          Axis1 Axis1:2 Axis1:3    Axis4:3
Sodium    93.93   95.13     100  1.110e-13
Sugars    30.11   99.59     100 -2.665e-13
Protein   80.75   96.05     100  0.000e+00
```

A l'aide de la somme cumulative des colonnes de contributions relatives, on observe que les variables « Sodium » et « Protein » ont largement contribué à former l'axe F1, pour des valeurs respectives de 94 % et 81%. Cet axe représente la plus grande inertie totale (allongement maximal), ce qui signifie que c'est sur cet axe que la perte d'information est moindre.

Toujours à l'aide du tableau, on observe que la variable « Sugars » a nettement contribué à la construction de l'axe F2, à hauteur de 66.5%. Cet axe représente la deuxième plus grande inertie totale orthogonale à l'axe 1, c'est sur ce deuxième axe que la perte d'information sera moindre.

15.



Représentation des points en fonction du plan avec affichage de l'inertie totales sur chaque axe

VIII. Conclusion

Nous observons que les données que nous avons traitées sont dans l'ensemble corrélées linéairement entres elles, les variables choisies pour ce Dataframe nous laissent difficilement le choix de manger régulièrement dans les restaurants MacDo, cependant, à consommation parcimonieuse... 😊