Faculté d'Ingénierie et Management de la Santé

Méthodes d'Analyse

Rapport de TP : ACP (Analyse en Composantes Principales)

Allan De Clercq[1]

[1]Université de Lille

6 décembre 2024

Encadrant : Assi N'Guessan Assi.Nguessan@polytech-lille.fr

Abstract :

Ce document présente le rapport d'un travail pratique réalisé dans le cadre de l'unité d'enseignement *Méthodes d'Analyse* de l'Université de Lille. Ce travail a pour objectif de mettre en pratique les notions d'Analyse en Composantes Principales (ACP) vues en cours.

Cette analyse a permis de mieux comprendre les caractéristiques des différents modèles de voitures et de les regrouper en fonction de leurs caractéristiques. Cette analyse pourrait être utile pour les constructeurs automobiles qui souhaitent mieux comprendre les caractéristiques des différents modèles de voitures. Mais elle pourrait également être utile pour les consommateurs qui souhaitent comparer les caractéristiques des différents modèles de voitures avant d'acheter.

Table des matières

Introduction

Ce projet utilise le dataset *mtcars* de R pour illustrer les concepts de l'Analyse en Composantes Principales (ACP). Le dataset *mtcars* contient les caractéristiques de 32 voitures et est composé de 11 variables :
- *mpg* : Miles/(US) gallon
- *cyl* : Nombre de cylindres
- *disp* : Déplacement (pouces cubes)
- *hp* : Puissance (chevaux)
- *drat* : Rapport de transmission
- *wt* : Poids (1000 livres)
- *qsec* : Temps de quart de mile
- *vs* : Moteur (0 = V-shaped, 1 = straight)
- *am* : Transmission (0 = automatique, 1 = manuelle)
- *gear* : Nombre de vitesses
- *carb* : Nombre de carburateurs

```
> head(df_mtcars)
                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4          21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag      21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710         22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive     21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant            18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
> str(df_mtcars)
'data.frame': 32 obs. of  11 variables:
$ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
$ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
$ disp: num  160 160 108 258 360 ...
$ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
$ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
$ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
$ qsec: num  16.5 17 18.6 19.4 17 ...
$ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
$ am  : num  1 1 1 0 0 0 0 0 0 0 ...
$ gear: num  4 4 4 3 3 3 3 4 4 4 ...
```

1 Matrice de corrélation avec R : Analyse et visualisation

Une matrice de corrélation est utilisée pour évaluer la dépendence entre plusieurs variables en méme temps. Le résultat est une table contenant les coefficients de corrélation entre chaque variable et les autres. Il existe différentes méthodes de test de corrélation : Le test de corrélation de Pearson, la corrélation de Kendall et de Spearman qui sont des tests basés sur le rang. Ces méthodes sont discutées dans les sections suivantes. La matrice de corrélation peut étre visualisée en utilisant un corrélogramme. Léobjectif de cet article est de vous montrer comment calculer et visualiser une matrice de corrélation dans R.

1.1 Matrice de corrélation

La fonction `cor()` est utilisée pour calculer la matrice de corrélation. La syntaxe de base est la suivante :

```
> cor(df_mtcars)
       mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
mpg   1.00 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
cyl  -0.85  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
disp -0.85  0.90  1.00  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
hp   -0.78  0.83  0.79  1.00 -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
drat  0.68 -0.70 -0.71 -0.45  1.00 -0.71  0.09  0.44  0.71  0.70 -0.09
```

```
wt   -0.87  0.78  0.89  0.66 -0.71  1.00 -0.17 -0.55 -0.69 -0.58  0.43
qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00  0.74 -0.23 -0.21 -0.66
vs    0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74  1.00  0.17  0.21 -0.57
am    0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17  1.00  0.79  0.06
gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79  1.00  0.27
carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27  1.00
```

Par défaut, la fonction `cor()` utilise la méthode de corrélation de Pearson [1]. Le résultat de la fonction est une table de coefficients de corrélation entre chaque variable et les autres. On peut remarquer que la diagonale de la matrice est composée de 1, car une variable est toujours corrélée à elle méme. Les valeurs de la matrice de corrélation varient entre -1 et 1.

— Une valeur de 1 indique une corrélation positive parfaite.
— Une valeur de -1 indique une corrélation négative parfaite.
— Une valeur de 0 indique qu'il n'y a pas de corrélation entre les variables.

Malheureusement, cette fonction n'affiche pas la significativité de la corrélation (p-value).

1.2 Test de significativité de la corrélation(p-value)

Dans cette section, nous utilisons le package **Hmisc** de R et la fonction `rcorr()` pour calculer le niveau de significativité (p-value) de la corrélation. En utilisant cette fonction le coefficient de corrélation de (Pearson ou rho de Spearman) est calculer pour toutes les paires de variables possibles dans la table de donnée.

```
> rcorr(as.matrix(df_mtcars), type="pearson")
       mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
mpg   1.00 -0.85 -0.85 -0.78  0.68 -0.87  0.42  0.66  0.60  0.48 -0.55
cyl  -0.85  1.00  0.90  0.83 -0.70  0.78 -0.59 -0.81 -0.52 -0.49  0.53
disp -0.85  0.90  1.00  0.79 -0.71  0.89 -0.43 -0.71 -0.59 -0.56  0.39
hp   -0.78  0.83  0.79  1.00 -0.45  0.66 -0.71 -0.72 -0.24 -0.13  0.75
drat  0.68 -0.70 -0.71 -0.45  1.00 -0.71  0.09  0.44  0.71  0.70 -0.09
wt   -0.87  0.78  0.89  0.66 -0.71  1.00 -0.17 -0.55 -0.69 -0.58  0.43
qsec  0.42 -0.59 -0.43 -0.71  0.09 -0.17  1.00  0.74 -0.23 -0.21 -0.66
vs    0.66 -0.81 -0.71 -0.72  0.44 -0.55  0.74  1.00  0.17  0.21 -0.57
am    0.60 -0.52 -0.59 -0.24  0.71 -0.69 -0.23  0.17  1.00  0.79  0.06
gear  0.48 -0.49 -0.56 -0.13  0.70 -0.58 -0.21  0.21  0.79  1.00  0.27
carb -0.55  0.53  0.39  0.75 -0.09  0.43 -0.66 -0.57  0.06  0.27  1.00

n= 32

P
     mpg    cyl    disp   hp     drat   wt     qsec   vs     am     gear   carb
mpg         0.0000 0.0000 0.0000 0.0000 0.0000 0.0171 0.0000 0.0003 0.0054 0.0011
cyl  0.0000        0.0000 0.0000 0.0000 0.0000 0.0004 0.0000 0.0022 0.0042 0.0019
disp 0.0000 0.0000        0.0000 0.0000 0.0000 0.0131 0.0000 0.0004 0.0010 0.0253
hp   0.0000 0.0000 0.0000        0.0100 0.0000 0.0000 0.0000 0.1798 0.4930 0.0000
drat 0.0000 0.0000 0.0000 0.0100        0.0000 0.6196 0.0117 0.0000 0.0000 0.6212
wt   0.0000 0.0000 0.0000 0.0000 0.0000        0.3389 0.0010 0.0000 0.0005 0.0146
qsec 0.0171 0.0004 0.0131 0.0000 0.6196 0.3389        0.0000 0.2057 0.2425 0.0000
vs   0.0000 0.0000 0.0000 0.0000 0.0117 0.0010 0.0000        0.3570 0.2579 0.0007
am   0.0003 0.0022 0.0004 0.1798 0.0000 0.0000 0.2057 0.3570        0.0000 0.7545
gear 0.0054 0.0042 0.0010 0.4930 0.0000 0.0005 0.2425 0.2579 0.0000        0.1290
carb 0.0011 0.0019 0.0253 0.0000 0.6212 0.0146 0.0000 0.0007 0.7545 0.1290
```

On remarque que la fonction `rcorr()` affiche la matrice de corrélation et la matrice des p-values. Les p-values sont affichées en bas de la matrice. Les p-values sont utilisées pour tester l'hypothèse nulle H_0 selon laquelle il n'y a pas de corrélation entre les variables.

$$H_0 \text{ : Il n'y a pas de corrélation entre les variables}$$

$$H_1 \text{ : Il y a une corrélation entre les variables}$$

Si la p-value est inférieure à l'**erreur de première espèce** α (généralement 5%), on rejette H_0 et on conclut qu'il y a une corrélation significative entre les variables. Ou plus simplement dit, **tant que la p-value est inférieure à 0.05, on rejette** H_0.

Par exemple : la p-value de la corrélation entre *qsec* et *wt* est : $p = 0.3389 > 0.05$, on ne rejette pas H_0 et on conclut qu'il n'y a pas de corrélation significative entre ces deux variables.

1.3 Corrélogramme

Un corrélogramme est un graphique qui affiche une matrice de corrélation sous forme de diagramme de points. Il existe plusieurs façon de visualiser une matrice de corrélation sous forme de corrélogramme.

Dans un premier temps, nous allons utiliser la fonction `symnum()` pour transformer les coefficients de corrélation en symboles.Elle prend la matrice de corrélation comme argument

```
> symnum(mcor, abbr.colnames=FALSE)
    mpg cyl disp  hp drat wt qsec vs am gear carb
mpg  1
cyl  +   1
disp +   *   1
hp   ,   +   ,    1
drat ,   ,   ,    .  1
wt   +   ,   +    ,  ,    1
qsec .   .   .    ,          1
vs   ,   +   ,    ,  .    .  ,    1
am   .   .   .       ,    ,          1
gear .   .   .       ,    .          ,  1
carb .   .   .    ,       .  ,    .          1
attr(,"legend")
[1] 0 ' ' 0.3 '.' 0.6 ',' 0.8 '+' 0.9 '*' 0.95 'B' 1
```

Comme indiqué dans la légende, les symboles sont définis comme suit :

- 0 : Pas de corrélation ( )
- 0.3 : Corrélation faible ($\cdot$)
- 0.6 : Corrélation modérée (,)
- 0.8 : Corrélation forte ($+$)
- 0.9 : Corrélation très forte ($*$)
- 0.95 : Corrélation presque parfaite (B)
- 1 : Corrélation parfaite (1)

Ensuite, nous allons utiliser la fonction `corrplot()` du package **corrplot** pour visualiser la matrice de corrélation sous forme de corrélogramme. Cette représentation graphique permet de visualiser rapidement les variables qui sont fortement corrélées entre elles.

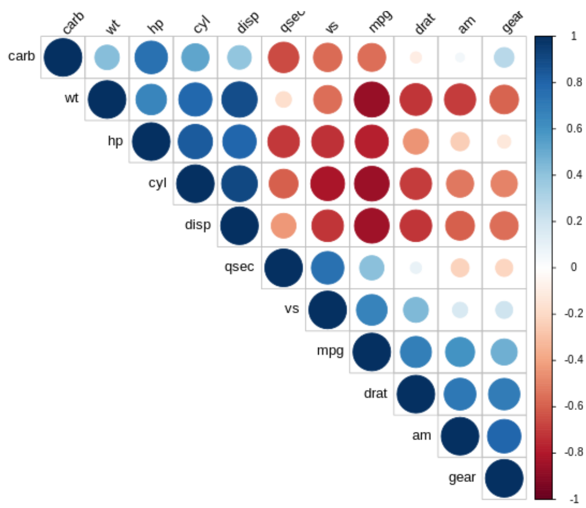

FIGURE 1 – Corrélogramme de la matrice de corrélation

Les variables qui sont positivementcorrélées entre elles sont représentées par des carrés bleus. En revanche, les variables qui sont négativement corrélées entre elles sont représentées par des carrés rouges. L'intensité de la couleur est proportionnelle à la valeur absolue de la corrélation. Plus la couleur est foncée, plus la corrélation est forte.

2 Analyse en Composantes Principales

2.1 Sélection du nombre de composantes principales (dimensions)

Dans un premier temps, il nous faut déterminer le nombre de composantes principales à retenir. Il exite 2 principales méthodes pour ce faire.

— La méthode de Kaiser : On retient les composantes principales dont les valeurs propres sont supérieures à 1.
— La méthode des 80% : On retient les composantes principales qui expliquent 80% de la variance totale.

Nous allons opter pour la méthode des 80% de variance expliquée. Pour cela, nous allons utiliser la fonction `PCA()` du package **FactoMineR** qui permet de calculer les composantes principales. Mais avant cela les données doivent être **centré-réduites** (standardisées) à l'aide de la fonction `scale()`. Avec la fonction `sapply()` il est facile de vérifier la standardisation des données.

```
> df_mtcars_scaled <- as.data.frame(scale(df_mtcars))

> sapply(df_mtcars_scaled, sd, na.rm = TRUE)
mpg  cyl disp   hp drat   wt qsec   vs   am gear carb
1    1    1    1    1    1    1    1    1    1    1

> sapply(df_mtcars_scaled, mean, na.rm = TRUE)
mpg  cyl disp   hp drat   wt qsec   vs   am gear carb
0    0    0    0    0    0    0    0    0    0    0
```

La fonction `PCA()` prend en paramètre les données standardisées et le paramètre `graph=FALSE` pour ne pas afficher le graphique. Ensuite, nous allons afficher les valeurs propres de chaque composante principale.

```
> pca_r <- PCA(df_mtcars_scaled, graph=FALSE)

> pca_r$eig
        eigenvalue percentage of variance cumulative percentage of variance
comp 1  6.60840025             60.0763659                          60.07637
comp 2  2.65046789             24.0951627                          84.17153
comp 3  0.62719727              5.7017934                          89.87332
comp 4  0.26959744              2.4508858                          92.32421
comp 5  0.22345110              2.0313737                          94.35558
comp 6  0.21159612              1.9236011                          96.27918
comp 7  0.13526199              1.2296544                          97.50884
comp 8  0.12290143              1.1172858                          98.62612
comp 9  0.07704665              0.7004241                          99.32655
comp 10 0.05203544              0.4730495                          99.79960
comp 11 0.02204441              0.2004037                         100.00000
```

Nous pouvons remarquer que les deux premières composantes principales expliquent 84.17% de la variance totale. Nous allons donc retenir les deux premières composantes principales pour la suite de l'analyse. A noter que la **somme des valeurs propres est égale au nombre de variables**.

Il est également possible de visualiser les valeurs propres de chaque composante principale en utilisant la fonction `fviz_eig()` du package **factoextra** en rentrant en paramètre l'objet `pca_r`.

```
> fviz_eig(pca_r, addlabels = TRUE)
```

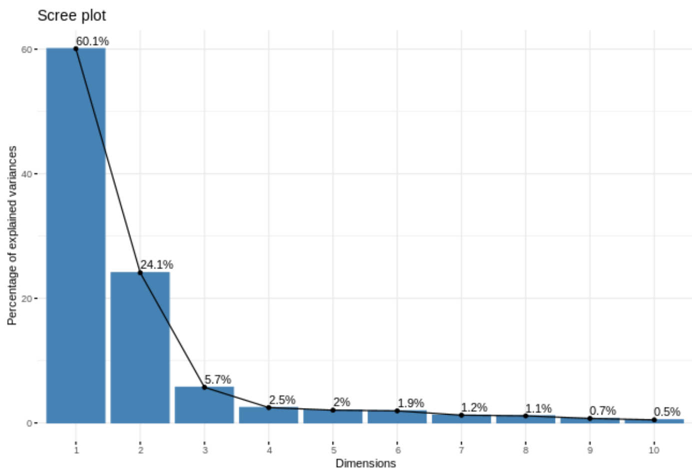

FIGURE 2 – Valeurs propres de chaque composante principale

2.2 Interprétation des composants principales (dimensions) retenues

Ce dataset contient 11 variables, nous chercheons à réduire la dimensionnalité en conservant le maximum d'information. Nous avons décidé de retenir les deux premières composantes principales qui expliquent 84.17% de la variance totale. Nous allons maintenant interpréter ces deux composantes principales pour comprendre les relations entre les variables et déterminer les variables qui contribuent le plus à chaque composante.

Pour établir le seuil de contribution des variables qui est jugé significatif nous allons utiliser la formule suivante :

$$\text{Seuil de contribution} = \frac{100\%}{\text{Nombre de variables}} = \frac{1}{11} \approx 9.1\%$$

100% car c'est la somme des contributions des variables pour chaque composante principale.

2.2.1 Première composante

Nous allons dans un premier temps fixer la première dimension et l'orienter. Pour rappel, la première dimension explique 60.08% de la variance totale.

Nous allons analyser ses contributions pour chaque variable. Nous ne sélectionnerons que les variables dont la contribution est significative et donc supérieure à $9,1$.

```
 > pca_r$var$contrib[, 1]
  mpg     cyl    disp     hp   drat      wt     qsec     vs      am     gear    carb
 13.143 13.981 13.556 10.894  8.653 11.979  4.018  9.395  5.520  4.281  4.580
```

Nous pouvons représenter graphiquement les contributions des variables pour la première dimension en utilisant la fonction `fviz_contrib()` du package **factoextra**.

```
> fviz_contrib(pca_r, choice = "var", axes = 1:1, top = 11)
```

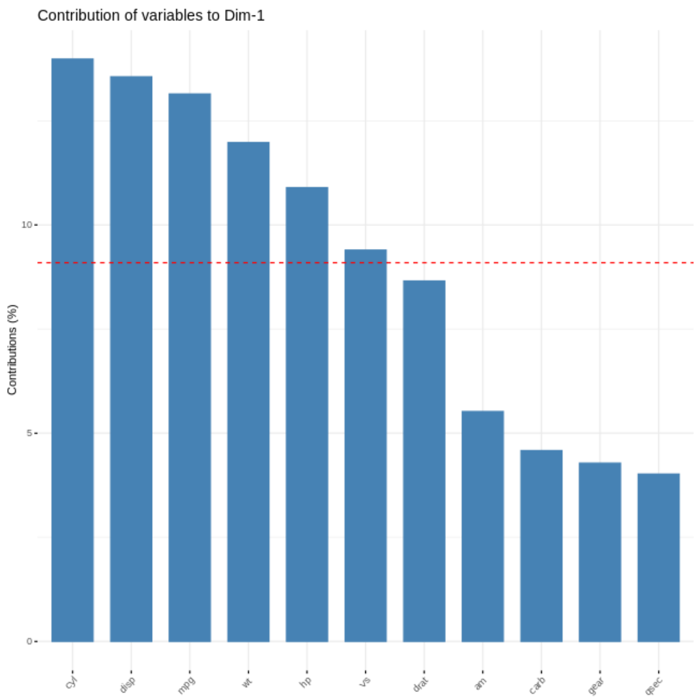

FIGURE 3 – Contribution des variables pour la première dimension

Nous ne retenons que 6 variables dont la contribution est supérieure à 9.1 : $cyl, disp, mpg, wt, hp, vs$. En analysant `pca_r`, nous pouvons extraire les informations importantes des contributeurs significatifs :

— Contribution : La contribution des variables à la dimension.
— Cos2 : La qualité de représentation des variables sur la dimension.
— Corrélation : La corrélation entre les variables et la dimension (signe positif ou négatif).

variables	contribution(%)	cos2	corrélation
cyl	13.98	0.924	+
disp	13.56	0.896	+
mpg	13.14	0.869	-
wt	11.98	0.792	+
hp	10.89	0.720	+
vs	9.39	0.621	-
total	73.94		

la première composante principale de 60.08% issue de l'acp normée est expliquée à hauteur de $73,94\%$ par les variables $cyl, disp, wt$ et hp qui se projettnet positivement sur cette dimension. Les variables mpg et vs se projettent négativement sur cette dimension.

Du fait de cette projection, cette tendance oppose le groupe de variables $cyl, disp, wt$ et hp au groupe mpg et vs.

2.2.2 Deuxième composante

Nous allons maintenant analyser la deuxième dimension qui explique 24.09% de la variance totale. Nous allons analyser ses contributions pour chaque variable. Nous ne sélectionnerons que les variables dont la contribution est significative et donc supérieure à $9, 1$.

```
> pca_r$var$contrib[, 2]
  mpg    cyl   disp     hp   drat     wt   qsec     vs     am   gear   carb
0.026  0.191  0.243  6.189  7.546  2.046 21.472  5.366 18.440 21.377 17.104
```

```
> fviz_contrib(pca_r, choice = "var", axes = 2:2, top = 11)
```

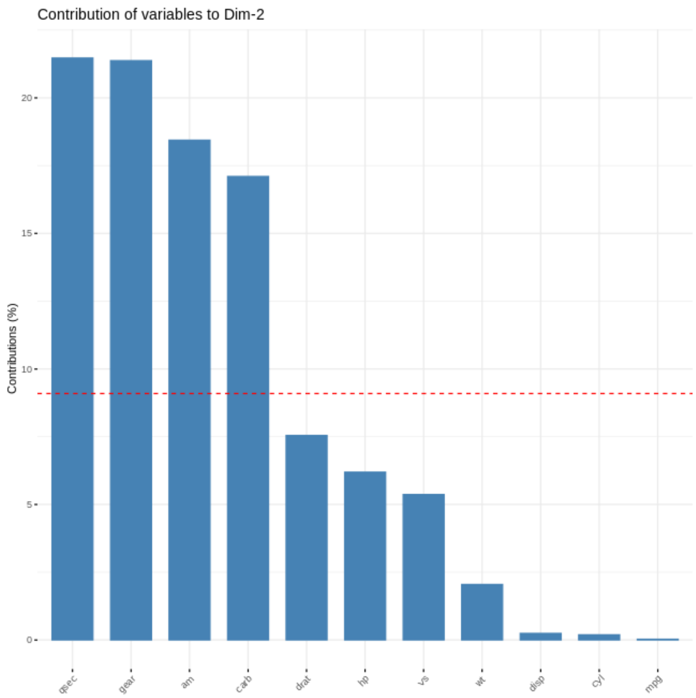

FIGURE 4 – Contribution des variables pour la deuxième dimension

Nous ne retenons que 6 variables dont la contribution est supérieure à 9.1 : $qsec, gear, am, carb$. En analysant `pca_r`, nous pouvons extraire les informations importantes des contributeurs significatifs :

variables	contribution(%)	cos2	corrélation
qsec	21.47	0.569	-
gear	21.38	0.567	+
am	18.44	0.489	+
carb	17.10	0.453	+
total	78.39		

la deuxième composante principale de 24.09% issue de l'acp normée est expliquée à hauteur de 78.39% par les variables $gear, am$ et $carb$ qui se projettnet positivement sur cette dimension. La variable $qsec$ se projette seule négativement sur cette dimension.

Du fait de cette projection, cette tendance oppose le groupe de variables $gear, am$ et $carb$ à la variable $qsec$.

2.3 Analyse des corrélations entre les composantes principales

Nous allons maintenant représenter graphiquement les cos2 des variables pour les deux premières composantes principales. Le package **factoextra** propose la fonction `fviz_cos2()` pour visualiser les cos2 des variables.

Sous forme de cercle de corrélation, les variables sont représentées par des flèches. En fonction de l'axe (composante) sur lequel elles se projettent, les variables sont colorées en fonction de leur cos2. Plus le cos2 est proche de 1, plus la variable est bien représentée.

Si l'angle entre deux flèches (variables) est :

- Aigu : Les variables sont positivement corrélées.
- Droit : Les variables sont indépendantes.

— Obtu : Les variables sont négativement corrélées.

```
> fviz_pca_var(pca_r,
+              axes = c(1, 2),
+              col.var = "cos2",
+              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"))
```

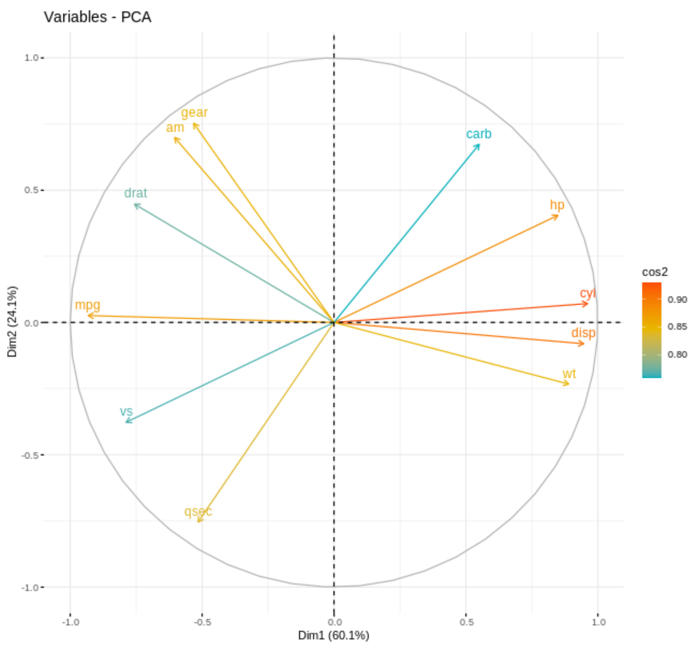

FIGURE 5 – Cercle de corrélation des variables

On remarque les variables cyl, $disp$, wt et hp sont bien représentées sur la première dimension.

Si nous souhaitons analyser $carb$, on remarque qu'il est indépendante de $gear, am, etdart$. Si on reprends leur signification, on peut dire que le nombre de carburateurs est indépendant du nombre de vitesses, du type de transmission et du rapport de transmission tandis que ceux-ci sont fortement et positivement corrélés entre eux. nous pouvons égalment remarquer que $qsec$ est fortement négativement corrélé avec $carb$. Ce qui signifie plus le nombre de carburateurs est élevé, plus le temps de quart de mile est faible.

En analysant la variable cyl, qui représente le nombre de cylindres, on peut dire qu'elle est positivement corrélée avec $disp$ et hp et négativement corrélée avec mpg. Ce qui signifie que plus le nombre de cylindres est élevé, plus le déplacement et la puissance sont élevés et plus le nombre de miles par gallon est faible. Elle semble également être indépendante de $qsec$, c'est-à-dire que le nombre de cylindres n'a pas d'impact sur le temps de quart de mile.

2.4 Analyse des individus (modèles de voitures)

Nous allons maintenant analyser les modèles de voitures en utilisant la fonction `fviz_pca_ind()` du package **factoextra**. Cette fonction permet de visualiser les individus sur les deux premières composantes principales. Les individus sont représentés par des points. Plus les points sont proches, plus les individus sont similaires.

```
> fviz_pca_ind(pca_r,
+              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
+              repel = TRUE)
```

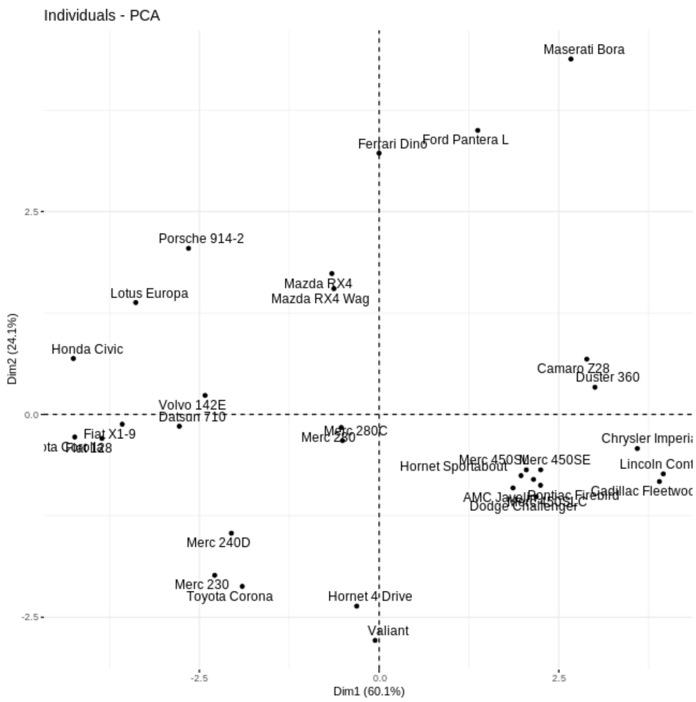

FIGURE 6 – Nuage de points des individus en fonction des deux premières composantes principales et leurs corrélations

En analysant le graphique, nous pouvons voir que les modèles de voitures se regroupes en fonction de leurs caractéristiques. Certains groupes se démarquent mais rien n'est encore clair. c'est pour cela qu'en dernière partie de ce travail, nous allons utiliser une méthode de classification hiérarchique pour regrouper les modèles de voitures en fonction de leurs caractéristiques.

3 Classification hiérarchique

La classification hiérarchique est une méthode de classification non supervisée qui permet de regrouper des individus en fonction de leurs caractéristiques. Il existe deux types de classification hiérarchique : la classification ascendante hiérarchique (CAH) et la classification descendante hiérarchique (CDA). Dans ce travail, nous allons utiliser la méthode de classification ascendante hiérarchique (CAH) pour regrouper les modèles de voitures en fonction de leurs caractéristiques.

Pour ce faire nous allons utiliser la foncton `HCPC()` du package **FactoMineR** qui permet de réaliser une classification hiérarchique sur les composantes principales. Nous allons utiliser les deux premières composantes principales pour réaliser la classification comme déterminé précédemment. Nous devons donc relancer la fonction `PCA()` en spécifiant le nombre de composantes principales à retenir : `ncp=2`.

```
> pca_r2 <- PCA(df_mtcars_scaled, ncp=2, graph=FALSE)
> hcpc <- HCPC(pca_r2, graph = FALSE)
> summary(hcpc)
           Length Class      Mode
data.clust 12     data.frame list
desc.var    3     catdes     list
desc.axes   3     catdes     list
desc.ind    2     -none-     list
call        8     -none-     list
```

La fonction `HCPC()` retourne un objet de type `HCPC` qui contient plusieurs informations :
— `data.clust` : Les données des individus avec les groupes auxquels ils appartiennent.
— `desc.var` : Les informations sur les variables.
— `desc.axes` : Les informations sur les axes.

— `desc.ind` : Les informations sur les individus.

Nous allons maintenant visualiser les groupes obtenus en utilisant la fonction `fviz_cluster()` du package **factoextra**.

```
> fviz_cluster(hcpc, show.clust.cent = T, main = "Factor map", labelsize = 10, palette
    = "Dark2")
```

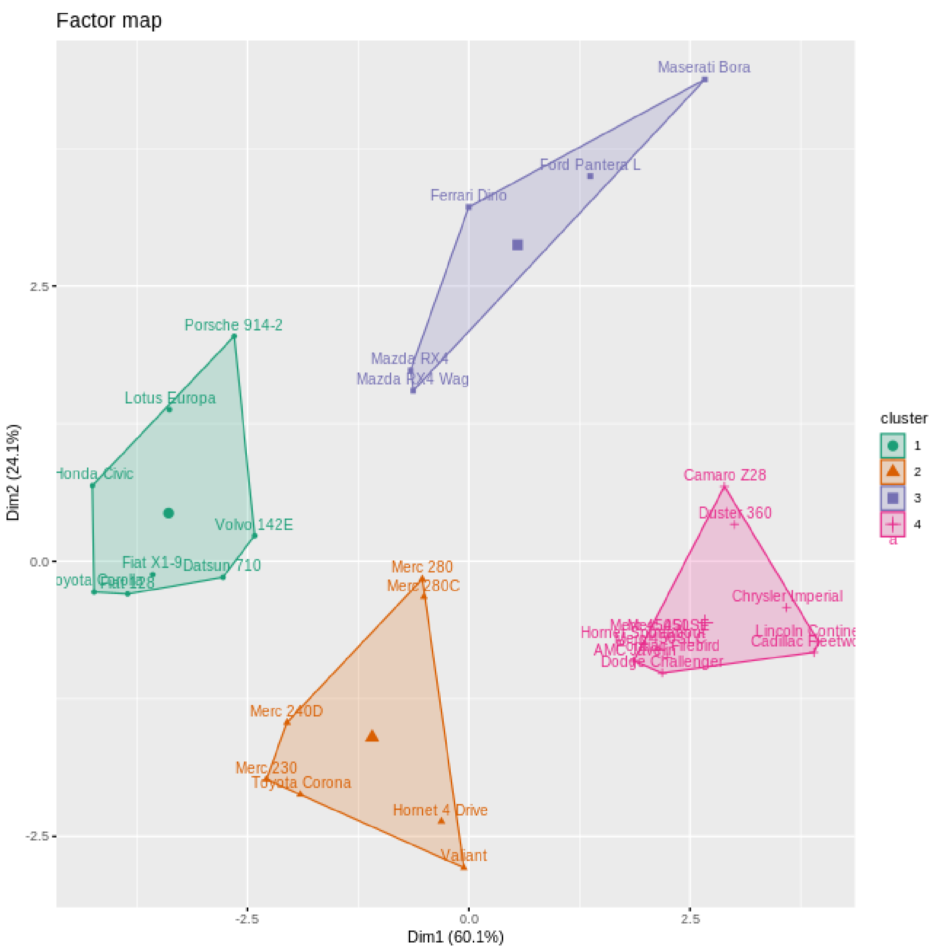

FIGURE 7 – Classification hiérarchique des modèles de voitures

Nous pouvons voir que 4 groupes ont été formés. Les groupes sont représentés par des couleurs différentes. Les centres de gravités des groupes sont les plus éloiignés possibles les uns des autres (distance euclidienne) tandis que les individus d'un même groupe sont les plus proches possibles les uns des autres. **L'intragroupe est minimisé tandis que l'intergroupe est maximisé.**

Il est cependant difficle d'identifier visuelement les différents individus d'un groupe surtout s'ils sont très similaires. C'est donc pourquoi nous allons utiliser la fonction `fviz_dend()` du package **factoextra** pour extraire les groupes et les individus qui les composent sous forme de dendrogramme.

```
> fviz_dend(hcpc, k = 4, cex = 0.5, rect = TRUE, rect_fill = TRUE, rect_border = "
   black")
```

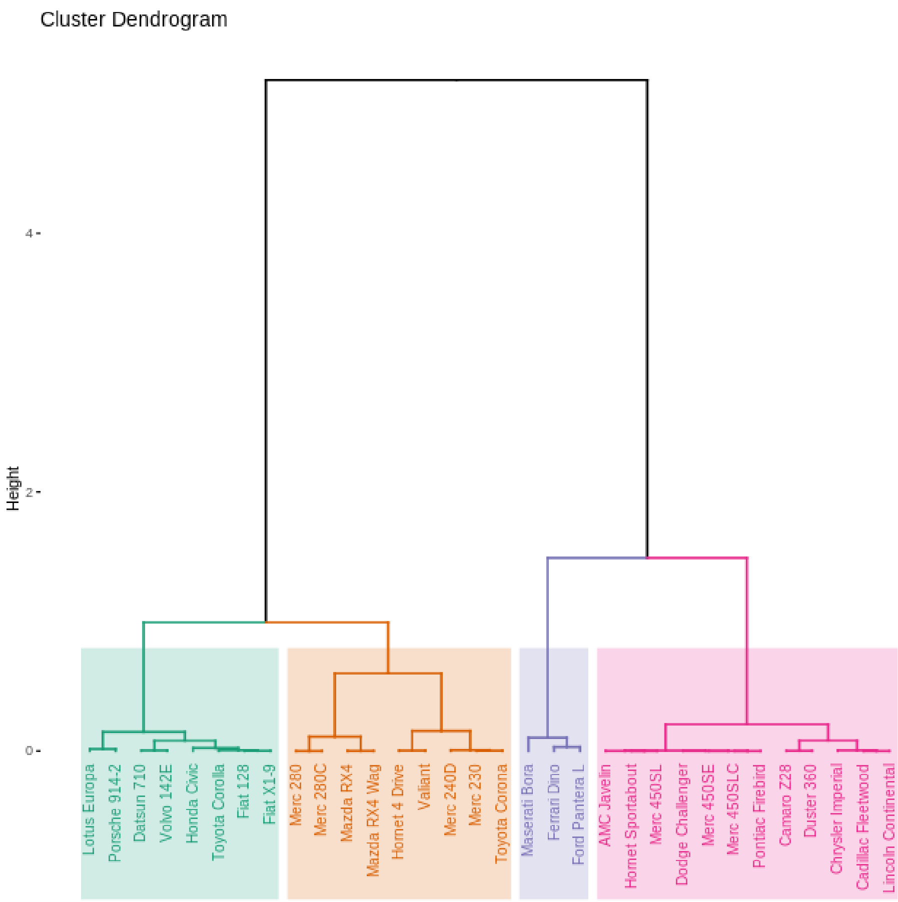

FIGURE 8 – Dendrogramme des groupes de modèles de voitures

Le dendrogramme permet de visualiser les groupes de modèles de voitures et les individus qui les composent. Plus les individus sont proches, plus ils sont similaires. Les groupes sont représentés par des couleurs différentes. Il est également possible de visualiser les sous-groupes ou encore les sur-groupes que composent les données.

C'est le principe du classement **hiérarchique** c'est à dire que les groupes sont imbriqués les uns dans les autres partant d'un groupe global pour arriver à des groupes plus spécifique à l'individu.

Nous pouvons désormais bien distinguer les différents groupes de modèles de voitures et les individus qui les composent.

Groupe 1	Groupe 2	Groupe 3	Groupe 4
Datsun 710	Hornet 4 Drive	Mazda RX4	Hornet Sportabout
Fiat 128	Valiant	Mazda RX4 Wag	Duster 360
Honda Civic	Merc 240D	Ford Pantera L	Merc 450SE
Toyota Corolla	Merc 230	Ferrari Dino	Merc 450SL
Fiat X1-9	Merc 280	Maserati Bora	Merc 450SLC
Porsche 914-2	Merc 280C		Cadillac Fleetwood
Lotus Europa	Toyota Corona		Lincoln Continental
Volvo 142E			Chrysler Imperial
			Dodge Challenger
			AMC Javelin
			Camaro Z28
			Pontiac Firebird

4 Conclusion

Ce travail avait pour objectif d'analyser les caractéristiques de différents modèles de voitures et de les regrouper en fonction de leurs caractéristiques. Nous avons commencé par une analyse de la corrélation entre les variables pour comprendre les relations entre elles. Ensuite, nous avons réalisé une analyse en composantes principales pour réduire la dimensionnalité des données et interpréter les composantes principales. Enfin, nous avons réalisé une classification hiérarchique pour regrouper les modèles de voitures en fonction de leurs caractéristiques.

Nous avons pu identifier 4 groupes de modèles de voitures différents. Les groupes ont été formés en fonction des caractéristiques des modèles de voitures. Les groupes ont été visualisés sous forme de dendrogramme pour mieux comprendre les relations entre les groupes et les individus qui les composent.

Cette analyse a permis de mieux comprendre les caractéristiques des différents modèles de voitures et de les regrouper en fonction de leurs caractéristiques. Cette analyse pourrait être utile pour les constructeurs automobiles qui souhaitent mieux comprendre les caractéristiques des différents modèles de voitures. Mais elle pourrait également être utile pour les consommateurs qui souhaitent comparer les caractéristiques des différents modèles de voitures avant d'acheter.

Références

1. BLYTH, Stephen. Karl Pearson and the Correlation Curve. *International Statistical Review / Revue Internationale de Statistique* [en ligne]. 1994, t. 62, nº 3, p. 393-403 [visité le 2024-12-04]. ISSN 03067734, ISSN 17515823. Disp. à l'adr. : http://www.jstor.org/stable/1403769.