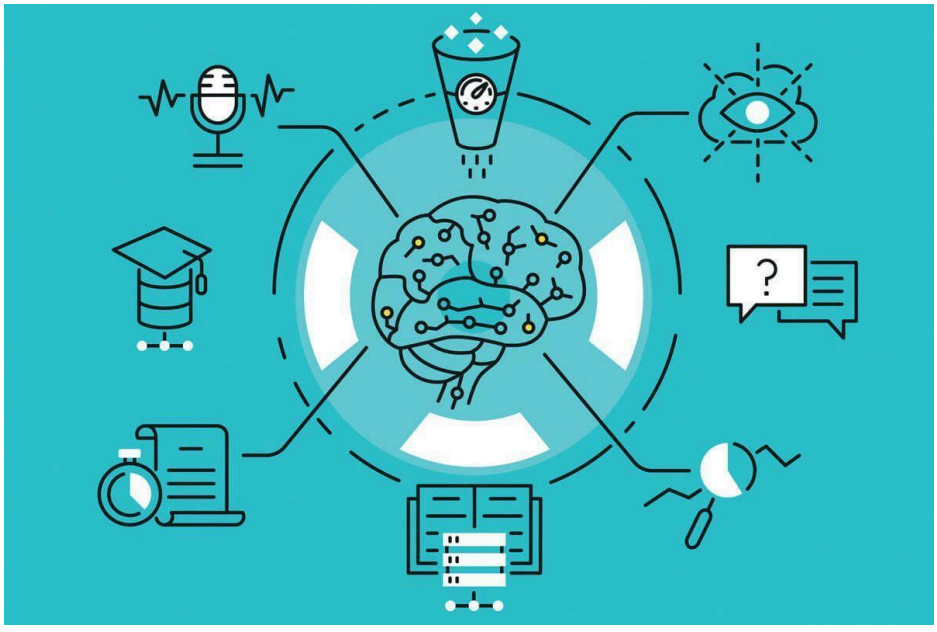
UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Master 1 Économétrie- Statistiques

Projet de Machine Learning et Analyse de données



Sous la direction de Monsieur Joseph RYNKIEWICZ

Samanta LAMOUR

Manal GHORAFI

Charlotte CEGARRA

Saikou BAH

Sommaire

1.-Régression binaire

L'objectif de cet exercice est de parvenir dans un premier temps à déterminer le meilleur modèle afin de modéliser P (Y |X1, X2), puis par la suite pouvoir prédire 1000 observations de Y.

Pour ce faire, nous allons utiliser les fichiers en format de texte ***simu.text*** et ***xsimutest.***

Notre fichier ***simu.text*** contient une simulation d'un modèle binaire avec 2000 observations, deux variables explicatives X1 et X2 et une variable à expliquer Y qui est une variable binaire à valeurs dans {1,2}.

Dans un premier temps, nous allons importer toutes les bibliothèques nécessaires dans le cadre de notre travail comme pandas, scickit-learn etc...

Afin de déterminer le meilleur modèle possible pour modéliser la probabilité conditionnelle de Y sachant X1 et X2 (P (Y |X1, X2)), nous allons suivre une série d'étapes assez importante impliquant l'analyse exploratoire des données, la préparation des donnés, le choix du modèle, la formation du modèle et l'évaluation de ce dernier.

1.1.-Analyse exploratoire des données

1.1.1.-Comprendre les données

Pour comprendre les données, il est nécessaire d'effectuer une analyse des statistiques descriptives de X1, X2 et Y.

Une fois nos données chargées et stockés dans un Dataframe, nous pouvons alors procéder à l'analyse de ces dernières.

Un affichage des premières lignes du Dataframe est important afin d'avoir un aperçu des données.

```
In [60]: df.head()
```

Out[60]:

	X1	X2	Y
0	-1.681427	-1.534811	1
1	-0.690532	0.710814	1
2	4.676125	-1.624768	2
3	0.211525	3.657683	2
4	0.387863	0.522408	2

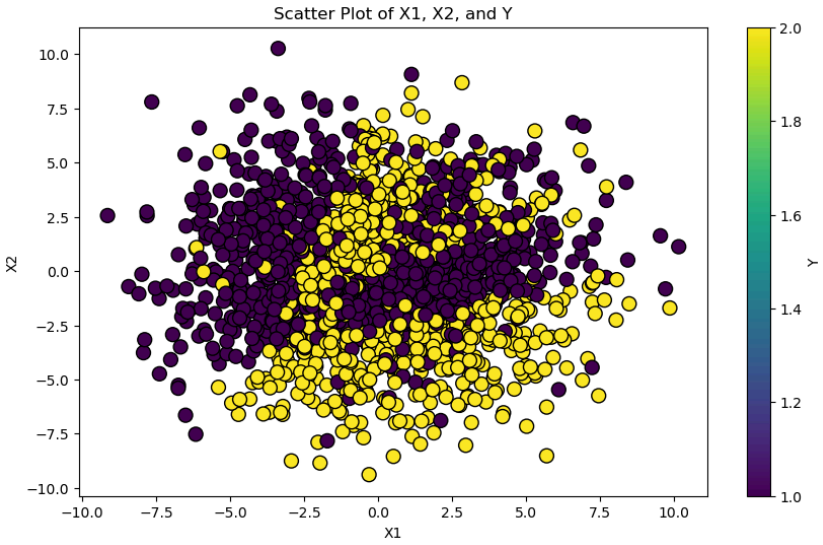
Afin d'effectuer l'analyse descriptive des données, nous avons utilisé la fonction *describe*

```
df.describe() #resume statistiques descriptives
```

	X1	X2	Y
count	2000.000000	2000.000000	2000.000000
mean	0.087890	-0.043909	1.414000
std	3.001684	2.955444	0.492672
min	-9.143583	-9.387265	1.000000
25%	-1.931862	-1.944058	1.000000
50%	0.086439	-0.074521	1.000000
75%	2.125775	1.945073	2.000000
max	10.171112	10.263284	2.000000

Count étant égale à 2000 partout, cela nous montre qu'il n'y a pas de valeur manquante dans les données (Ce qui a été confirmé grâce à la fonction *info*). Mean indique la moyenne des valeurs et donne une idée de la tendance centrale des données. La moyenne étant légèrement différente de la médiane pour toutes les variables, cela indique une certaine distribution asymétrique des données, mais qui n'est pas pour autant très visible sur le graphe.

Une visualisation des données à travers le scatter plot nous permet également qu'il n'y a pas de relation linéaire simple entre X1 et X2.



1.2.- Sélection et construction du modèle

1.2.1.-Choix des modèles

Quelques modèles couramment utilisés pour la classification binaire incluent la régression logistique, les SVM (Support Vector Machines), les forêts aléatoires, les réseaux neuronaux, etc.

Dans notre exercice, nous allons utiliser la régression logistique, les SVM, les forêts aléatoires, le boosting, le gradientboost, l'adaboost et le bagging.

1.3.-Entraînement, formation et évaluation du modèle

Dans cette partie, nous allons diviser les données en ensembles d'entraînement et de test.

Nous avons avant tout normaliser les données. Cette standardisation est nécessaire car elle permet la comparabilité des caractéristiques et éviter que les caractéristiques avec de grandes valeurs dominent celles avec les plus petites valeurs les plus grandes. De plus, beaucoup de modèle de Machine Learning comme SVM fonctionnent mieux avec les données standardisées.

1.3.1.-Entraînement des différents modèles

Puis nous avons entraîné le modèle sur l'ensemble d'entraînement et nous avons évaluer les performances du modèle sur l'ensemble de test en utilisant des métriques appropriées comme l'exactitude (Accuracy), le F1-score etc…

Pour ce faire, nous avons stocké les différents modèles et leur nom dans un dictionnaire afin de pouvoir par la suite itérer sur chaque modèle et effectuer les opérations comme l'entrainement et l'évaluation de la performance sur un ensemble de test.

Nous avons par la suite utilisé une boucle pour itérer sur chaque modèle du dictionnaire créé préalablement, puis nous avons ajusté le modèle sur l'ensemble d'entraînement, fait des prédictions sur l'ensemble de test, calcule la précision et imprime le rapport de classification.

1.4.- Choix du meilleur modèle

Une fois les modèles entrainés et évalués, nous allons pouvoir choisir le meilleur modèle en se basant sur l'exactitude (Accuracy). Le modèle ayant l'Accuracy le plus élevé est le meilleur modèle et dans notre cas, c'est le Gradient Boosting.

1.5.- Test

Pour déterminer de manière plus robuste quel modèle est le meilleur, on a effectué une validation croisée (cross-validation) afin d'évaluer les performances des modèles.

Ce code fournit un cadre solide pour évaluer et comparer les différents modèles de manière exhaustive et objective, en tenant compte de plusieurs aspects cruciaux de la performance du modèle. Cela aide à identifier le modèle le plus performant pour l'ensemble de données spécifique en nous renvoyant la moyenne de l'Accuracy et le meilleur modèle est celui ayant la moyenne de l'Accuracy le plus élevé à savoir le Gradient Boosting.

1.6.- Optimisation des paramètres

L'optimisation des paramètres est une étape très importante afin de trouver la meilleure combinaison de paramètres pour le modèle, ce qui peut amener à de meilleure performance. Cette approche utilise la validation croisée pour une évaluation plus robuste et moins biaisée des performances du modèle assurant ainsi que ce dernier fonctionne à son niveau optimal.

Partie II

1.7.-Prédictions de Y

Dans cette partie, il est demandé de prédire les 1000 observations de la variable Y. Le code fournit prépare les données de test en les mettant à l'échelle de la même manière que les données d'entraînement, puis utilise un modèle entraîné pour faire des prédictions sur ces données transformées.

2.-Analyse en composantes principales

On dispose des mesures suivantes sur plusieurs types de voitures vendues en 2015 dans le fichier "voitures". On doit effectuer une analyse en composantes principales à l'aide de toutes les variables.

Fichier voiture :

	CYL	PUIS	LON	LAR	POIDS	VITESSE	ACCEL
ALPHAMITO	875	105	406	172	1130	184	11.4
AUDIA1	999	95	397	174	1065	186	10.9
CITROENC4	1199	130	442	182	1280	196	10.1
JAGUARF	2995	340	447	192	1587	260	5.7
PEUGEOTRCZ	1997	160	428	184	1370	220	8.2
LANDROVER	2993	256	483	191	2570	180	9.3
RENAULTCLIO	898	90	406	173	1092	182	12.2
BMWS3	1995	116	462	181	1570	198	11.1
DACIA	898	90	406	173	962	175	11.1
HYUNDAI	1995	136	447	185	1751	184	10.9
LANCIA	2776	177	522	200	2315	193	11.5
RENAULTCAPTUR	898	90	412	178	1180	171	13.0
FORDMUSTANG	4951	421	272	192	1720	250	4.8
FIAT500	1242	69	355	163	905	160	12.9
HONDA	2199	150	472	184	1632	212	9.4
FERRARI	6262	660	491	195	1880	335	4.1
SUBARU	1998	147	445	178	1440	198	9.3
MAZDA	1560	115	458	175	1490	180	13.7
VOLKSWAGEN	1598	105	425	179	1220	192	10.7
JAGUARPACE	1999	180	473	194	1775	208	8.7

	CO2
ALPHAMITO	98
AUDIA1	103
CITROENC4	115
JAGUARF	234
PEUGEOTRCZ	130
LANDROVER	203
RENAULTCLIO	105
BMWS3	109
DACIA	116
HYUNDAI	139
LANCIA	207
RENAULTCAPTUR	113
FORDMUSTANG	299
FIAT500	115
HONDA	138
FERRARI	380
SUBARU	141
MAZDA	138
VOLKSWAGEN	99
JAGUARPACE	139

2.1.-Question 1

Quel est le pourcentage d'inertie expliquée par les trois premiers facteurs ? Par le premier plan factoriel ?

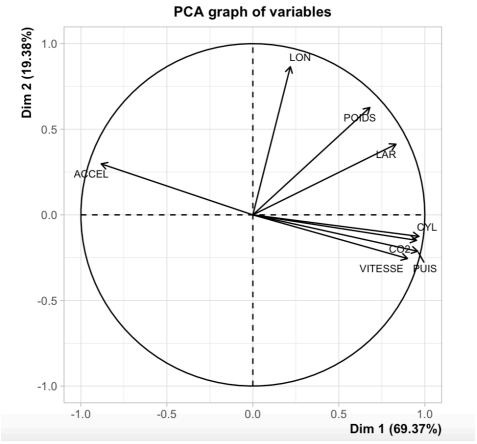
	eigenvalue	percentage of variance
comp 1	5.549271024	69.36588781
comp 2	1.550202725	19.37753406
comp 3	0.480655817	6.00819771
comp 4	0.280682369	3.50852962
comp 5	0.084751837	1.05939796
comp 6	0.034770185	0.43462731
comp 7	0.013450458	0.16813073
comp 8	0.006215585	0.07769482

	cumulative percentage of variance
comp 1	69.36589
comp 2	88.74342
comp 3	94.75162
comp 4	98.26015
comp 5	99.31955
comp 6	99.75417
comp 7	99.92231
comp 8	100.00000

Le pourcentage d'inertie expliquée par les trois premiers facteurs est la somme de la variance expliquée par les trois premiers facteurs soit environ 95%.
Le pourcentage d'inertie expliquée par le premier plan factoriel est la somme de la variance expliquée par les deux premiers facteurs soit environ 89%.

2.2- Question 2

Interpréter les 2 axes principaux à partir des corrélations des variables avec ces axes

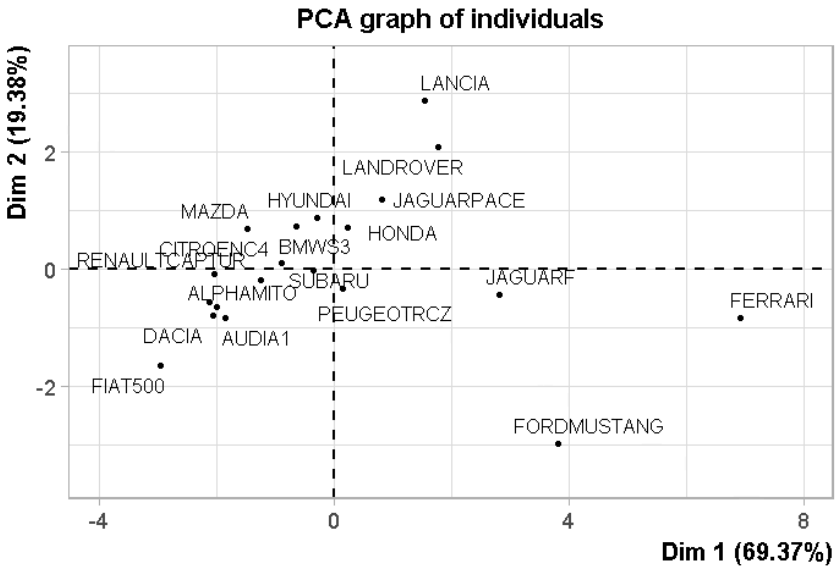


L'axe 1 est l'axe horizontal et l'axe 2 est l'axe verticale. Au centre se trouve l'individu moyen.
Pour répondre à cette question, on regarde le cercle de corrélation des variables.
On observe que la quasi-totalité des variables sont corrélées positivement avec le 1er axe (sauf la variable accel) : les coordonnées sur le 1er axe de toutes les variables sont positives donc elles se retrouvent toutes à droite.

On a donc un effet taille (toutes les variables sont corrélées avec un axe). Si on est très à droite cela montre que toutes les valeurs des variables sont plutôt grandes.
La variable “accel” représente l’accélération des véhicules et elle est mesuré en termes de temps. Sa corrélation négative avec l’axe 1 montre que plus le temps est petit et plus une voiture est rapide. Si la valeur de “accel” est faible, cela voudrait dire que la voiture accélère rapidement. À l’inverse, si la valeur de “accel” est élevée, cela voudrait dire que la voiture prend du temps pour accélérer.
Les deux axes ne sont pas représentés parfaitement par une variable spécifique car aucune des variables n’est couchée sur un axe.
L’axe 1 est bien représenté par le cylindre, cependant, ce n’est pas la seule variable qui peut l’expliquer. Mais c’est celle qui explique le mieux l’axe étant donné que c’est la plus proche. En effet, cet axe est le mieux représenté par le cylindre mais peut être aussi expliqué par l’émission de C02, la puissance ou encore la vitesse.
La cylindrée est une caractéristique importante des moteurs de véhicules, et elle est souvent associée à la puissance du moteur. En général, une cylindrée plus importante peut indiquer un moteur plus puissant.
Ainsi, le premier axe pourrait nous indiquer les voitures les plus performantes. À droite on aura les meilleures voitures (avec une valeur importante pour des variables comme “cyl”, “CO2”, “vitesse” et “puissance” et une valeur faible pour la variable “accel”) et à gauche les moins bonnes (les valeurs des variables “cyl”, “CO2”, “vitesse” et “puissance” sont plus faibles sauf la variable “accel” qui sera plus élevé). L’axe 1 opposerait donc les voitures de sport, des voitures utilitaires.
Par construction les axes sont orthogonaux et si on a un effet taille sur le premier axe, on aura un effet forme sur le deuxième.
L’axe 2 est plutôt bien représenté par la longueur, en effet, c’est la variable la plus proche de cet axe.
Étant donné les performances de chaque voiture, l’axe 2 pourrait séparer les grandes et longues voitures (qui sont plus lourdes) de celles qui sont plus légères et rapides.

2.3.-Question 3

Représentez les individus sur le premier plan factoriel et répondez aux questions suivantes *:*



1. **Les individus sont-ils bien représentés sur le premier plan factoriel ?**

Pour savoir si les individus sont bien représentés sur le premier plan factoriel, on regarde les cosinus carrés des individus sur les 2 premiers axes.

	Dim.1	Dim.2
ALPHAMITO	0.88418157	0.0882481546
AUDIA1	0.82447953	0.1603240825
CITROENC4	0.63441116	0.0097161853
JAGUARF	0.87295591	0.0211260379
PEUGEOTRCZ	0.02354612	0.1090728501
LANDROVER	0.31747599	0.4453076078
RENAULTCLIO	0.91588474	0.0619189342
BMWS3	0.34739866	0.4385043460
DACIA	0.83548059	0.1227757130
HYUNDAI	0.07691485	0.7124226564
LANCIA	0.20567125	0.7152890894
RENAULTCAPTUR	0.91218646	0.0011120940
FORDMUSTANG	0.52868209	0.3233183930
FIAT500	0.70301792	0.2181054338
HONDA	0.05681670	0.5506712131
FERRARI	0.92712779	0.0130358167
SUBARU	0.26365614	0.0002406284
MAZDA	0.57219677	0.1274534969
VOLKSWAGEN	0.85640405	0.0198445442
JAGUARPACE	0.21914850	0.4760572094

Il faut faire la somme des cosinus au carré (de chaque axe) pour avoir la qualité de la projection sur le premier plan factoriel.

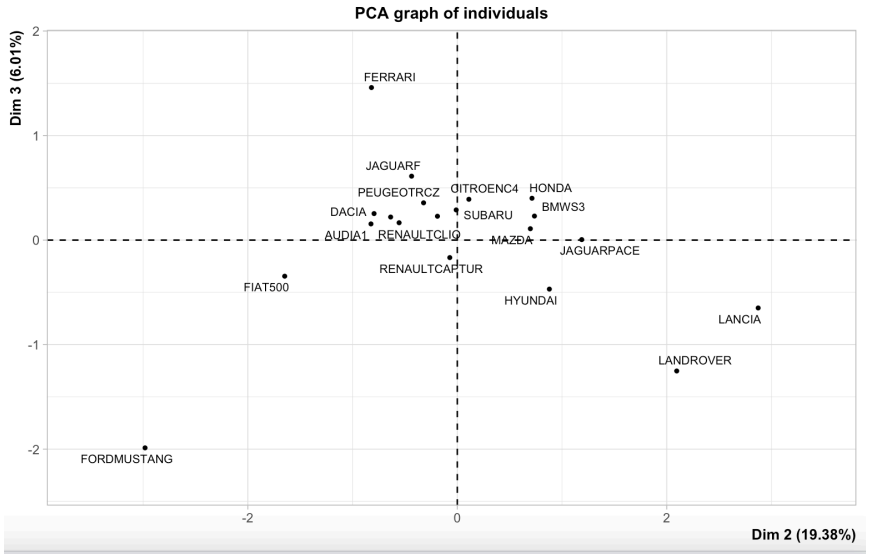
Les 3 individus les mieux représentés sont AUDIA1, avec une somme des cosinus carré d'environ 0,98, puis nous avons ALPHAMITO avec une somme des cosinus carré d'environ 97%. Enfin, nous avons la DACIA avec une somme des cosinus carré d'environ 0,95.
On constate que tous les individus semblent être bien représentés sur le premier plan factoriel c'est-à-dire qu'ils ont presque tous une somme des cosinus carré supérieur à 0,5 sauf la PEUGEOTRCZ et la SUBARU.
En effet, La somme des cosinus au carré pour l'individu PEUGEOTRCZ est d'environ 0,13 et pour l'individu SUBARU est d'environ 0,26, ce qui est largement en dessous de 0,5.
Ainsi, ils se sont fait écraser par la projection et se retrouvent donc au centre comme nous pouvons le voir sur le graphique.

	eigenvalue	percentage of variance
comp 1	5.549271024	69.36588781
comp 2	1.550202725	19.37753406
comp 3	0.480655817	6.00819771
comp 4	0.280682369	3.50852962
comp 5	0.084751837	1.05939796
comp 6	0.034770185	0.43462731
comp 7	0.013450458	0.16813073
comp 8	0.006215585	0.07769482

On peut essayer d'aller plus loin et voir ce qu'il se passe sur d'autres axes. Cependant, on peut déjà voir que la part d'inertie expliquée par chaque axe sera beaucoup plus basse.

	Dim.2	Dim.3
ALPHAMITO	0.0882481546	1.059039e-02
AUDIA1	0.1603240825	5.639548e-03
CITROENC4	0.0097161853	1.215544e-01
JAGUARF	0.0211260379	4.142085e-02
PEUGEOTRCZ	0.1090728501	1.335421e-01
LANDROVER	0.4453076078	1.591429e-01
RENAULTCLIO	0.0619189342	5.517074e-03
BMWS3	0.4385043460	4.263289e-02
DACIA	0.1227757130	1.250491e-02
HYUNDAI	0.7124226564	2.027533e-01
LANCIA	0.7152890894	3.655778e-02
RENAULTCAPTUR	0.0011120940	6.056581e-03
FORDMUSTANG	0.3233183930	1.437252e-01
FIAT500	0.2181054338	9.588572e-03
HONDA	0.5506712131	1.734503e-01
FERRARI	0.0130358167	4.129523e-02
SUBARU	0.0002406284	1.728306e-01
MAZDA	0.1274534969	3.104131e-03
VOLKSWAGEN	0.0198445442	2.868204e-02
JAGUARPACE	0.4760572094	7.096227e-06

Si on regarde les axes 2 et 3, les individus PEUGEOTRCZ et SUBARU ne sont toujours pas bien représentés. En effet, la somme des cosinus au carré pour l'individu PEUGEOTRCZ est d'environ 0,23<0,5 et pour l'individu SUBARU elle est d'environ 0,17<0,5, ils sont donc écrasés par la projection et vont se retrouver au centre du graphique.



A l'inverse des individus LANDROVER, HYUNDAI, LANCIA et HONDA, tous les autres individus se retrouvent au centre, écrasés par la projection. En effet, leur somme des cosinus carré est inférieure à 0,5.

2. **Quelles sont les caractéristiques des individus en haut du graphe ?**

Il n'y a aucune variable couchée sur l'axe 2, donc ce qui représenterait le plus cet axe serait la longueur. Ainsi, plus une voiture est située vers le haut et plus elle est grande par rapport à la moyenne (en termes de poids, de longueur et de largeur).
Ainsi, tout en bas du graphique on retrouve le Fordmustang qui est une voiture de sport donc elle sera moins longue que la moyenne. À l'inverse, Lancia est situé tout en haut du graphique. Certains modèles qu'elle a produits comme les SUV ou les Berlines sont plus grands.

3. **Quelles sont les caractéristiques des individus à droite du graphe ?**

De même pour l'axe 1, aucune variable n'est couchée dessus. Ainsi on pourrait penser que l'axe 1 est représenté par le cylindre de la voiture étant donné que cette variable est la plus proche de l'axe. En effet, plus les voitures se situent vers la droite, plus elles ont une grosse cylindrée. Par exemple, Ferrari a une plus grosse cylindrée que la Renault.

Mais il est également représenté par les variables “CO2”, “PUIS” et “VITESSE”. Une voiture située à droite du graphique regroupe des valeurs élevées pour chacune de ces variables. Or, une voiture qui regroupe une valeur élevée pour ces 4 variables au profil d’une voiture de sport. On peut voir par exemple que la voiture qui est située le plus à droite est la Ferrari qui est une voiture de sport. La deuxième voiture située la plus à droite est la Ford Mustang. Ensuite, on retrouve la Jaguarf. Les modèles de ces marques sont souvent associés à des performances élevées, offrant des moteurs puissants, une accélération rapide et des caractéristiques sportives.

4. **Quelles sont les caractéristiques des individus en bas à gauche du graphe ?**

Sur le graphique des individus, on peut voir que les voitures citadines sont concentrées sur la gauche.
Les voitures citadines sont généralement compactes et légères, adaptées à la conduite en milieu urbain. Les individus situés à gauche auront des valeurs faibles pour les variables “CYL”, “CO2”, “PUIS” et “VITESSE”. À l’inverse, plus une voiture est située à gauche et plus sa valeur pour la variable “ACCEL” sera élevée. Ainsi, les voitures situées à gauche polluent moins, ne sont pas très rapides et ont des moteurs souvent de faible puissance.
Si une voiture est située à gauche mais également vers le bas, cela veut dire qu’elle est plus petite, plus légère et moins longue que la moyenne.
Ainsi, les individus situés en bas à gauche ont tendance à être plus petits en termes de dimensions, plus légers et avec des moteurs moins puissants, ce qui est caractéristique des voitures citadines.
La voiture qui est le plus en bas à gauche est la FIAT500. Ceci est cohérent avec ce qui a été dit au-dessus. C’est une voiture citadine, qui est adaptée à une utilisation en milieu urbain, avec une émission réduite de CO2. Elle est conçue pour être légère, maniable et économique.

5. **Peut-on dire que les individus PEUGEOTRCZ et JAGUARF ont un profil semblable ? Si oui quel est-il ?**

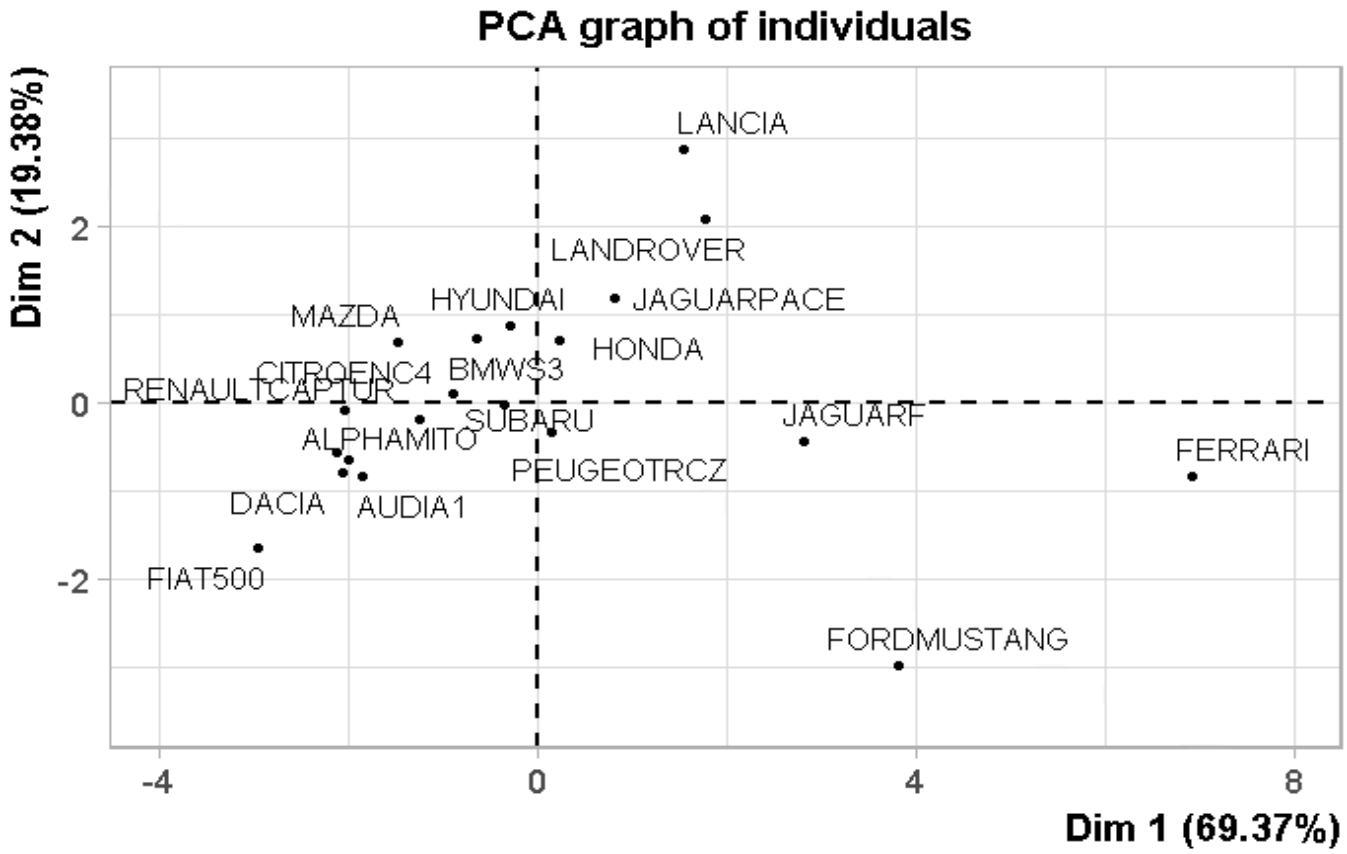
À la question (1), nous avons vu que tous les individus sauf la PEUGEOTRCZ et la SUBARU semblent bien représentés sur le premier plan factoriel. La somme des cosinus au carré pour

l'individu PEUGEOTRCZ est d'environ 0,13. Ce qui est largement en dessous de 0,5. Ainsi, il s'est fait écrasé par la projection. Lorsqu'on regarde le graphique des individus, on voit qu'il se retrouve au centre.
L'individu PEUGEOTRCZ étant mal représenté, on ne peut pas interpréter les proximités.
Donc on ne peut pas conclure sur le fait qu'ils aient un profil semblable ou non.

6. **Peut-on dire que les individus LANCIA et LANDROVER ont un profil semblable ? Si oui quel est-il ?**

La proximité entre LANCIA et LANDROVER est justifiée car la somme des cos2 est supérieure à 0.5.
Ce sont les deux individus les plus hauts sur le graphique des individus. Ainsi, ce sont donc les deux marques de voitures les plus longues et les plus lourdes. Elles ont des moteurs puissants (leur cylindre est similaire). Elles polluent plus que les voitures situées à gauche, et vont plus vite également (mais moins que la FERRARI ou la JAGUARF par exemple). Par exemple, LANDROVER produit des grosses voitures de luxe telles que les Range Rover.

7. **Interpréter la représentation graphique des individus.**



Dans un premier temps, on peut voir que l'axe 1 oppose les marques de voitures de sport (à droite) aux marques de voitures citadines (à gauche). Ainsi, les marques situées le plus à droite telles que FERRARI ou encore FORDMUSTANG, produisent des voitures avec des caractéristiques similaires. C'est-à-dire des voitures rapides, qui polluent beaucoup et avec des moteurs très puissants.

FERRARI a la cylindrée la plus élevée (6262 cm³), suivi par la FORDMUSTANG (4951 cm³) donc elles ont des moteurs avec une grande capacité. La vitesse maximale que peut atteindre la FERRARI est de 335 km/h et celle de la FORDMUSTANG est de 250 km/h, ce qui montre qu'elles sont rapides. La variable "ACCEL" mesuré en seconde représente le temps nécessaire pour que la voiture atteigne une vitesse de 100km/h. Ainsi, la FERRARI et la FORDMUSTANG ont les valeurs les plus basses pour cette variable. Cependant, leur émission de C02 sont les plus élevée (380 g/km pour la FERRARI et 299 g/km pour la FORDMUSTANG).

À l'inverse, les marques situées le plus à gauche telles que FIAT500 ou encore DACIA, produisent des voitures avec des caractéristiques similaires c'est-à-dire pas très rapides, qui polluent peu et avec des moteurs peu puissants.
La FIAT500 a une cylindrée de 1242 cm^3 et la DACIA a une cylindrée de seulement 89, ce qui est vraiment peu élevé en comparaison avec la FERRARI et la FORDMUSTANG. La vitesse maximale que chaque voiture peut atteindre est également beaucoup plus faible. Pour la FIAT500, elle est de 160 km/h et pour la DACIA, elle est de 175 km/h. Enfin, leur émission de CO2 par kilomètre est relativement faible. Leur valeur est de 115 g/km pour la FIAT500 et de 116 g/km pour la DACIA.

Ensuite, l'axe 2 oppose les marques des voitures selon la longueur des voitures produites. Ainsi, les marques situées en haut telles que LANCIA ou LANDROVER produisent des voitures plus longues mais également plus lourdes et larges que les marques situées tout en bas telles que FORDMUSTANG ou FIAT500 par exemple.
En effet, LANCIA mesure 522cm de long et 200cm de large ce qui lui mène à un poids total de 2315kg, de même pour LANDROVER qui pèse 2570kg, et mesure 483cm de long et 191cm de large. Ces deux marques représentent donc les voitures les plus massives et les plus lourdes.
A l'inverse, FORDMUSTANG et FIAT500 sont beaucoup plus petites et légères. On l'observe avec les données, FIAT500 se caractérise par son poids léger soit 905kg et sa petite taille soit 355cm de long sur 163cm de large. De même pour FORDMUSTANG qui mesure 272cm de long sur 192cm de large.

On observe une proximité entre la DACIA, la ALPHAMITO et l'AUDIA1. Ceux sont les 3 individus les mieux représentés avec une somme des cosinus carré beaucoup plus élevé que 0,5. Ainsi, leur proximité est justifiée. Ce sont trois marques de voitures qui produisent des voitures citadines (faible émission de CO2, faible capacité du moteur, vitesse maximale pas très élevée).

On peut aussi s'intéresser aux individus opposés comme FIAT500 et FERRARI sur le premier axe.

FERRARI présente toutes les caractéristiques d’une voiture de sport : elle a la plus grosse cylindrée soit 6262cm³ et la plus grosse puissance soit 660km/h avec un faible temps d’accélération soit 4,1s pour atteindre 100km/h.
A l'inverse, FIAT 500 est considéré comme une voiture citadine notamment grâce à sa taille compacte. En effet, celle-ci mesure 355cm de long et 163cm de large avec un temps d’accélération de 12,9 secondes pour atteindre les 100 km/h, soit 3 fois plus que la FERRARI.

3.-Classification

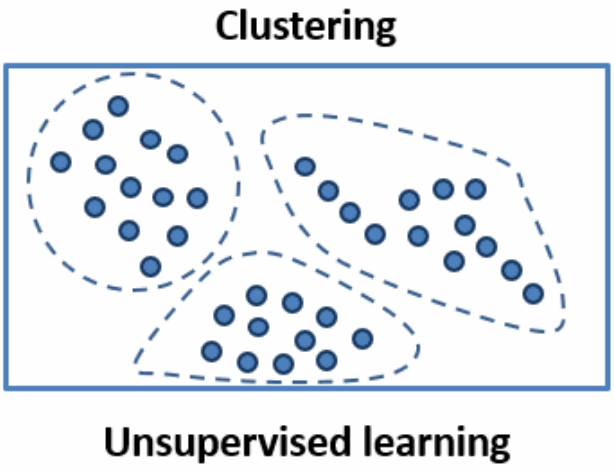
3.1.-Classification avec l'algorithme K-means

1.Importation des données et visualisation :

Dans notre base de données, nous avons 8 variables indépendantes qui représentent les caractéristiques de chaque marque. Nous constatons que ces variables ne sont pas exprimées dans la même unité de mesure. Il est important de résoudre ce problème, car l'algorithme k-means est sensible aux différences d'unités entre les variables.

2.Algorithme k-means :

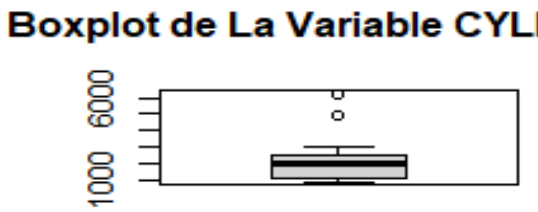
C'est une méthode de clustering non supervisée qui permet de regrouper des données similaires. Elle nous permet de former K clusters distincts à partir de nos observations, ainsi on aura un regroupement des données similaires au sein d'un même cluster.

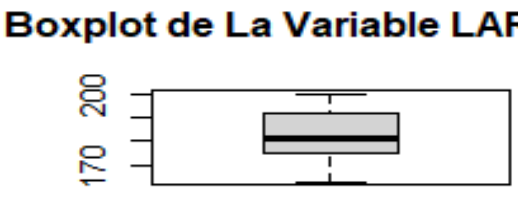


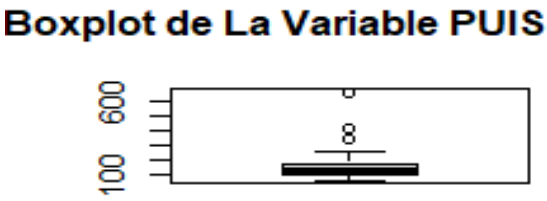
3.2.-Analyse descriptive et Nettoyage des donnés

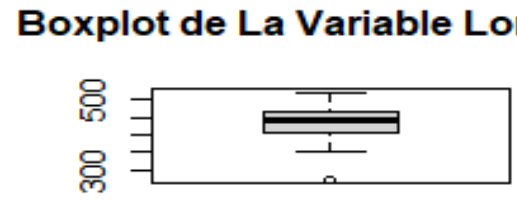
Vérification des valeurs aberrantes :

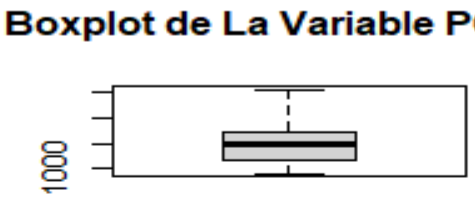
Dans notre base de données, on constate une absence de données manquantes . Néamoins, il est important de verifier la présence de valeurs aberrantes, car l'algorithme est sensible à ces valeurs. La présence de valeurs aberrantes pourrait influencer les centres des clusters et donc directement impacter la qualité de notre partitionnement.

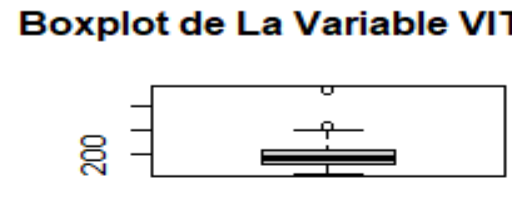


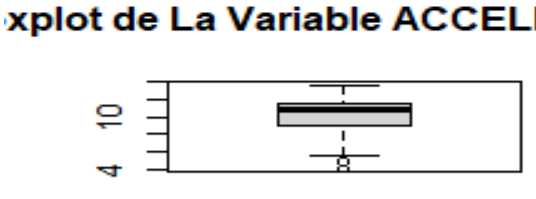


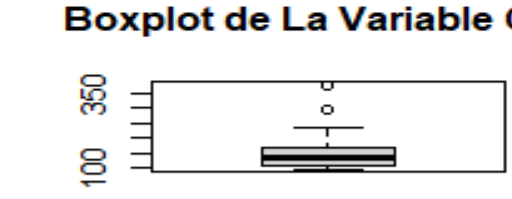








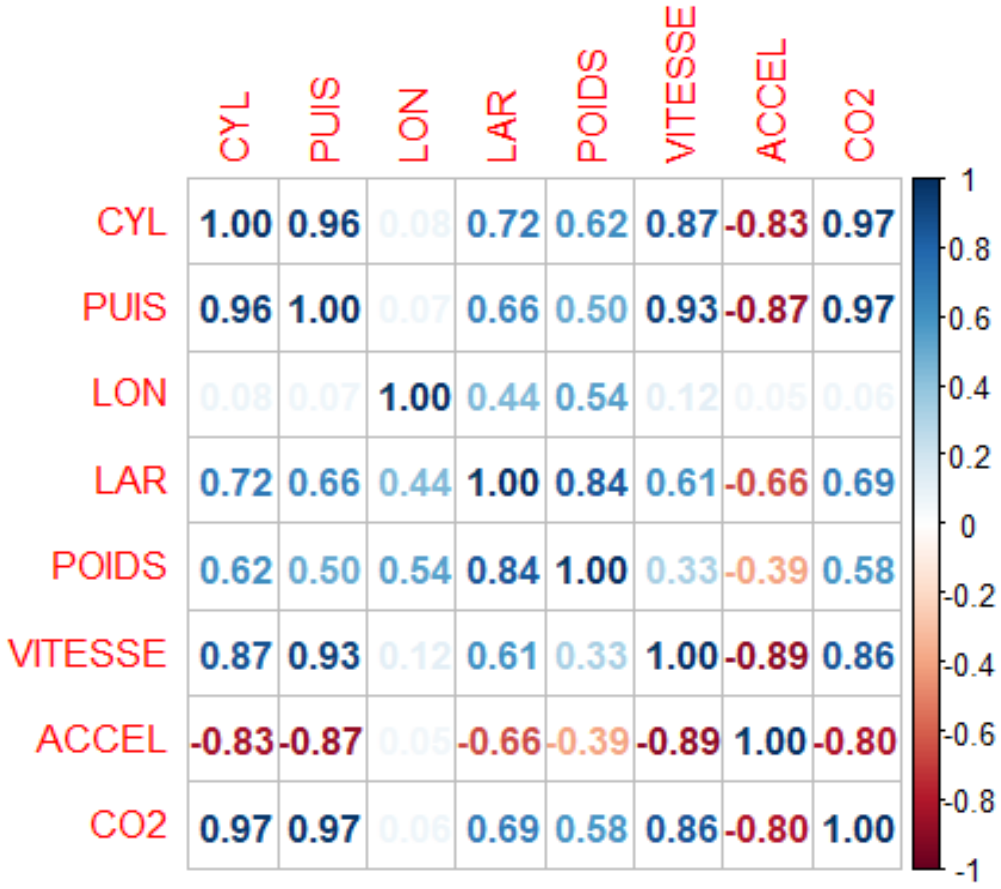




On constate la présence de valeurs aberrantes pour plusieurs de nos variables, cela pourrait avoir des répercussions sur les centres des clusters ainsi que sur la distance euclidienne utilisée par notre algorithme. Elles risquent également de biaiser le choix optimal de nos clusters. Pour résoudre ce problème, une solution consiste à effectuer une transformation logarithmique. Toutefois, nous allons travailler sur nos données brutes.

Matrice de corrélation

Nous allons analyser la corrélation entre les différentes variables pour identifier d'éventuelles relations. On a effectué cette analyse uniquement sur les variables numériques, car la fonction ***cor ()*** ne fonctionne qu'avec ce type de variables.



Nous constatons une corrélation de 0,96 entre le nombre de cylindres d'une voiture et sa puissance, cela suggère une forte corrélation positive entre ces deux variables. Cela signifie qu'en général, lorsque le nombre de cylindres augmente, la puissance du véhicule a tendance à augmenter également. On a une corrélation similaire qui existe entre le nombre de cylindres et

les émissions de CO2, suggérant que les véhicules avec un nombre plus élevé de cylindres ont tendance à émettre davantage de CO2.

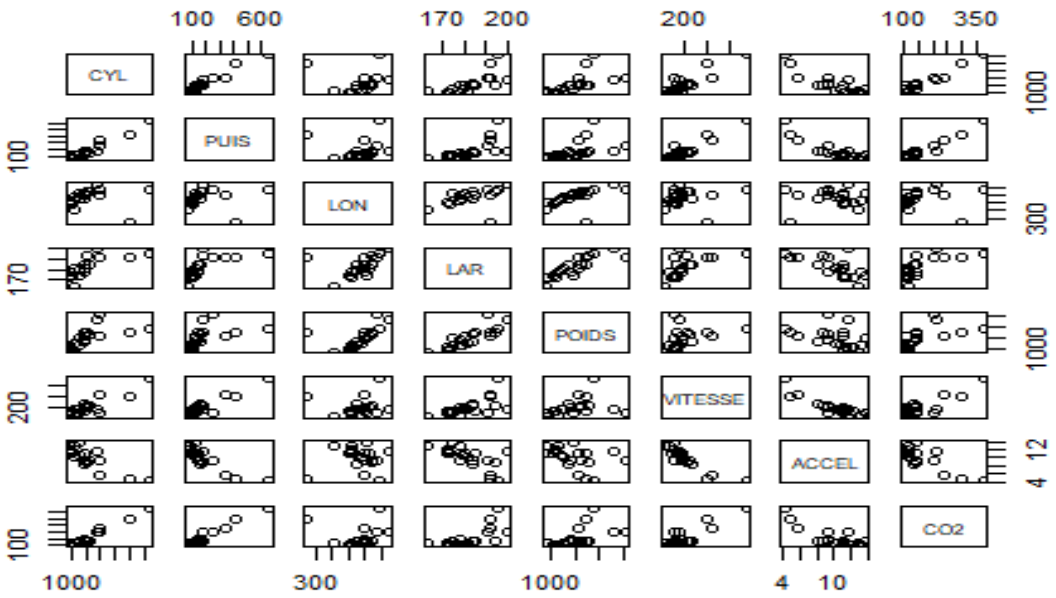
Cependant, une corrélation de -0,83 est observée entre le nombre de cylindres et l'accélération (temps nécessaire pour atteindre 100 km/h). Ainsi, les véhicules avec un nombre plus élevé de cylindres ont tendance à avoir un temps d'accélération plus court.

Il est essentiel de visualiser ces corrélations, car une forte corrélation peut introduire une redondance, ce qui entraîne une attribution de poids excessive par notre algorithme aux variables fortement corrélées.

```
#Variance des variables

##        Variable   Variance
## CYL         CYL 1931706.34
## PUIS       PUIS   20529.31
## LON         LON    2894.79
## LAR         LAR      90.09
## POIDS     POIDS  187162.33
## VITESSE VITESSE    1566.48
## ACCEL     ACCEL       6.91
## CO2         CO2    5509.94
```

Nous constatons que la variance de chaque variable est non nulle, ce qui est positif. Néanmoins, il est important de noter que les variances diffèrent énormément. Les variables avec des variances plus élevées auront un impact plus significatif sur la mesure de la distance dans nos analyses.

```
#Nuages par pairs
```



L'objectif de cette étape est de visualiser les relations entre différentes variables afin de détecter d'éventuels clusters naturels, des regroupements potentiels de points qui présentent des tendances similaires. Par exemple, quand on examine la variable CO2, nous identifions clairement cinq observations qui se distinguent des autres, une observation similaire est également notée pour la variable puissance. Cependant, avec la variable longueur, la détection de tels regroupements devient plus complexe. Lorsqu'on a un grand nombre de variables, l'analyse visuelle des relations devient difficile à l'œil nu. C'est pourquoi il est essentiel d'utiliser un algorithme pour une analyse plus approfondie.

Modification des données

Comme l'algorithme repose sur la mesure des distances et que nos variables ont des unités différentes, cela risque de biaiser nos résultats. Afin d'éviter ce problème, on standardise les variables pour les ramener à une échelle commune. Cette démarche contribue à rendre l'algorithme plus cohérent et améliore l'interprétation des résultats de clustering.

```
#Moyenne des variables

##     CYL    PUIS     LON     LAR   POIDS VITESSE   ACCEL     CO2
## 2116.35  181.60  432.45  182.25 1496.70  203.20    9.95  156.05
```

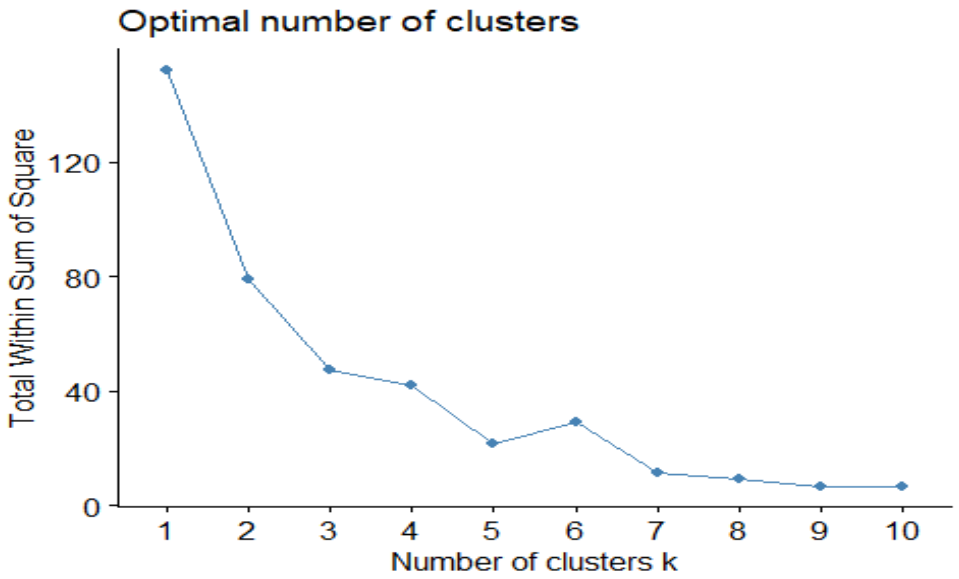
Nous constatons des valeurs moyennes très différentes, indiquant que nos variables sont mesurées dans des unités différentes. Lors de l'application de l'algorithme, les variables avec une variance élevée auront une influence très élevée sur les résultats.

3.3.-Détermination des clusters

Il est difficile de choisir un nombre de cluster intuitivement, la stratégie simple est de faire évoluer le nombre de classes et surveiller l'évolution de l'inertie intra-classe. Dans notre cas, on observe une augmentation significative jusqu'à 0.49(classe2), le reste des augmentations est moins importantes, comme dans l'analyse k-means nous recherchons le point ou l'augmentation ralentit, considéré comme le point optimal pour le nombre de classes. On pourrait aussi être tenté de dire qu'on a plutôt deux solutions, une solution à 2 classes ou 3.

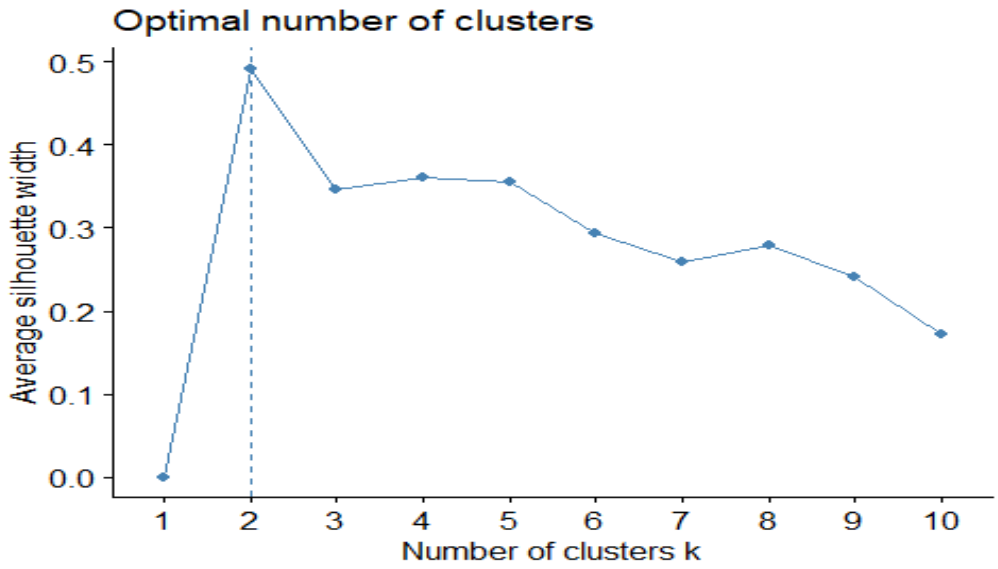
```
##   Nombredeclusters Inertie
## 1                1    0.00
## 2                2    0.48
## 3                3    0.69
## 4                4    0.72
## 5                5    0.80
## 6                6    0.88
## 7                7    0.90
## 8                8    0.94
```

Une autre méthode couramment utilisée est la méthode du coude (Elbow Method), qui repose sur une approche graphique. Pour chaque valeur de k, on mesure la somme des carrés des distances intra-cluster, puis on trace le graphique. Le point du coude, représente le nombre de clusters à partir duquel la réduction de la variance n'est plus significative. C'est ce point qui est généralement choisi comme nombre optimal de clusters.



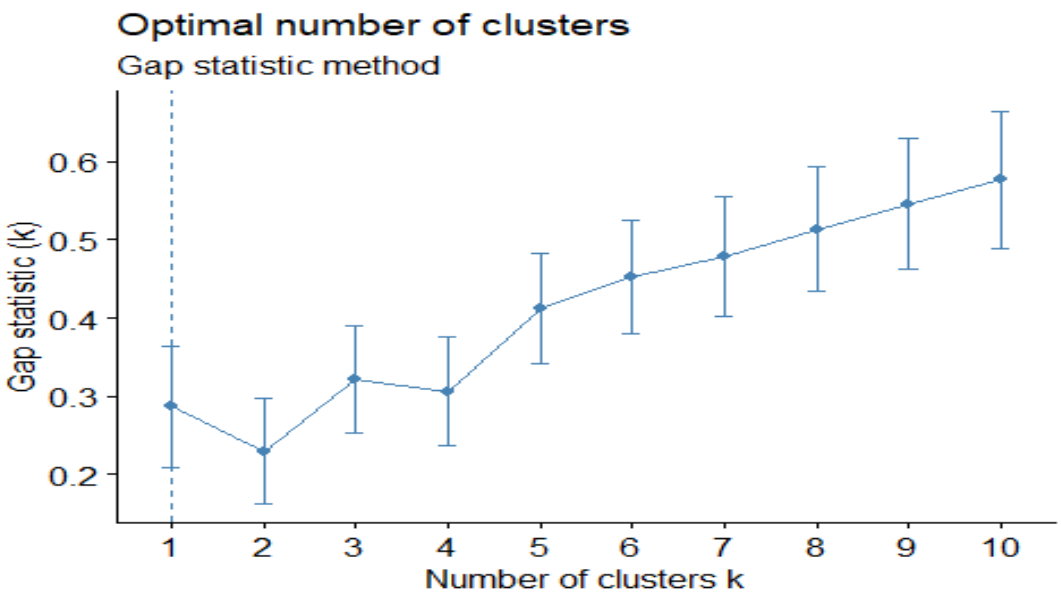
Ici, le point du coude identifié est au nombre de clusters = 3, avec une chute significative de la courbe entre 1 et 3. Néanmoins, l'interprétation peut varier entre différentes personnes.

Une autre méthode est l'utilisation du score de silhouette pour évaluer la qualité des clusters, qui varie entre [-1, 1]. On considère la valeur de "k" ayant le score de silhouette le plus proche de 1 comme le nombre optimal de clusters.



Les deux méthodes fournissent des résultats divergents. La méthode de la silhouette, qui mesure la qualité de la séparation entre les clusters, suggère un nombre optimal de clusters égal à 2.

Dans ce cas, nous allons prendre en compte les deux suggestions et évaluer la pertinence des clusters générés pour chaque configuration (2 et 3 clusters) afin de prendre une décision éclairée.



Cette méthode suggère un cluster optimal avec k=1, ce qui pourrait indiquer que la structure de nos données est homogène et ne justifie pas la création de clusters. La statistique du gap compare la dispersion intra-cluster réelle avec celle générée aléatoirement.

Procédure K-means

L'algorithme k-means est un processus itératif visant à minimiser la somme des distances entre chaque observation et le centroïde.

Pour un début, nous fixons le nombre de clusters à k=2.

```
K-means clustering with 2 clusters of sizes 17, 3
##
## Cluster means:
##          CYL       PUIS         LON        LAR       POIDS    VITESSE
## 1 -0.3326175 -0.3597222  0.09550051 -0.1998653 -0.09475722 -0.3498602
## 2  1.8848323  2.0384256 -0.54116956  1.1325701  0.53695760  1.9825411
##        ACCEL        CO2
## 1  0.3413093 -0.3525259
## 2 -1.9340858  1.9976469
```

```
##
## Clustering vector:
##     ALPHAMITO        AUDIA1     CITROENC4       JAGUARF    PEUGEOTRCZ
##             1             1             1             2             1
##     LANDROVER   RENAULTCLIO         BMWS3         DACIA       HYUNDAI
##             1             1             1             1             1
##        LANCIA RENAULTCAPTUR   FORDMUSTANG       FIAT500         HONDA
##             1             1             2             1             1
##       FERRARI        SUBARU         MAZDA    VOLKSWAGEN    JAGUARPACE
##             2             1             1             1             1
##
## Within cluster sum of squares by cluster:
## [1] 57.11066 19.94784
##  (between_SS / total_SS =  49.3 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

L'algorithme k-means a formé 2 clusters, le premier avec 17 Marques et le second avec 3 marques. A travers la moyenne des clusters on obtient pour chaque classe la moyenne des variables de nos observations, si on regarde plus en detail, on observe que Les voitures du Cluster 1 ont quelques caractéristiques spécifiques. En moyenne, elles sont :

- **Plus longues que la moyenne.**
- **Moins larges que la moyenne.**
- **Un nombre de cylindres légèrement inférieur à la moyenne.**
- **Une puissance légèrement inférieure à la moyenne.**
- **Une vitesse légèrement inférieure à la moyenne.**
- **Une accélération légèrement plus élevée que la moyenne.**
- **Émettent légèrement moins de CO2 que la moyenne.**

  Les voitures du Cluster 2 ont quelques caractéristiques spécifiques. En moyenne, elles sont :

  - **Plus de cylindres que la moyenne.**

- **Avec une puissance plus élevée que la moyenne.**
- **Plus courtes que la moyenne.**
- **Plus larges que la moyenne.**
- **Plus lourdes que la moyenne.**
- **Avec une vitesse plus élevée que la moyenne.**
- **Avec une accélération nettement inférieure à la moyenne.**
- **Émettant nettement plus de CO2 que la moyenne.**

Le vecteur de clustering indique à quel cluster chaque observation appartient. Au sein du cluster 2, on a les marques suivantes : FERRARI JAGUARF FORDMUSTANG. On a donc dans ce cluster des voitures de sport avec des caractéristiques : cylindrées élevées, une puissance importante, une accélération rapide et des émissions de CO2 relativement élevées peuvent être regroupées ensemble. La qualité du partitionnement est de 49,3%

```
## K-means clustering with 3 clusters of sizes 3, 8, 9
##
## Cluster means:
##           CYL       PUIS        LON        LAR      POIDS    VITESSE
## 1  1.88483231  2.0384256 -0.5411696  1.1325701  0.5369576  1.9825411
## 2  0.09184389 -0.1141118  0.6328617  0.5136074  0.7077184 -0.1029591
## 3 -0.70991645 -0.5780425 -0.3821539 -0.8340633 -0.8080689 -0.5693279
##         ACCEL         CO2
## 1 -1.93408579  1.99764687
## 2 -0.05707138 -0.07140066
## 3  0.69542538 -0.60241503
##
## Clustering vector:
##     ALPHAMITO        AUDIA1     CITROENC4       JAGUARF    PEUGEOTRCZ
##             3             3             3             1             2
##     LANDROVER   RENAULTCLIO         BMWS3         DACIA       HYUNDAI
##             2             3             2             3             2
##        LANCIA RENAULTCAPTUR   FORDMUSTANG       FIAT500         HONDA
##             2             3             1             3             2
##       FERRARI        SUBARU         MAZDA    VOLKSWAGEN    JAGUARPACE
##             1             2             3             3             2
##
## Within cluster sum of squares by cluster:
## [1] 19.947844 17.929894  9.246285
```

```
##  (between_SS / total_SS =  69.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

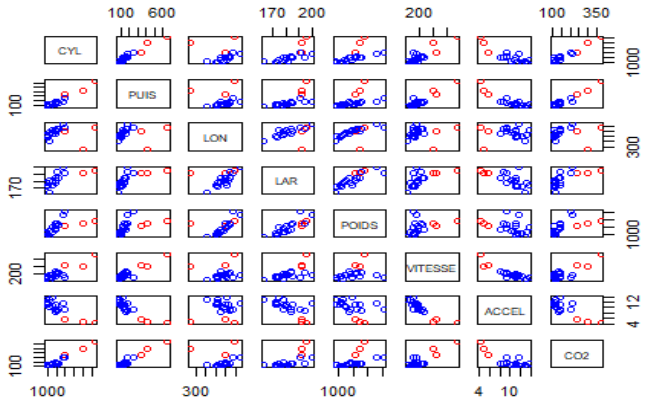
On observe une augmentation de la qualité du partitionnement, ce qui est cohérent puisque nous augmentons le nombre de clusters. Le cluster qui comprenait les 3 voitures reste inchangé, tandis que l'algorithme a partitionné le deuxième cluster en deux sous-clusters distincts. Cette division plus fine nous fournit des informations plus détaillées sur les similarités et les différences entre les observations au sein du cluster qui comprenait 17 marques.

```
##                CYL PUIS LON LAR POIDS VITESSE ACCEL CO2
## FIAT500       1242   69 355 163   905     160  12.9 115
## RENAULTCAPTUR  898   90 412 178  1180     171  13.0 113
## DACIA          898   90 406 173   962     175  11.1 116
## RENAULTCLIO    898   90 406 173  1092     182  12.2 105
## AUDIA1         999   95 397 174  1065     186  10.9 103
## ALPHAMITO      875  105 406 172  1130     184  11.4  98
## VOLKSWAGEN    1598  105 425 179  1220     192  10.7  99
## MAZDA         1560  115 458 175  1490     180  13.7 138
## BMWS3         1995  116 462 181  1570     198  11.1 109
```

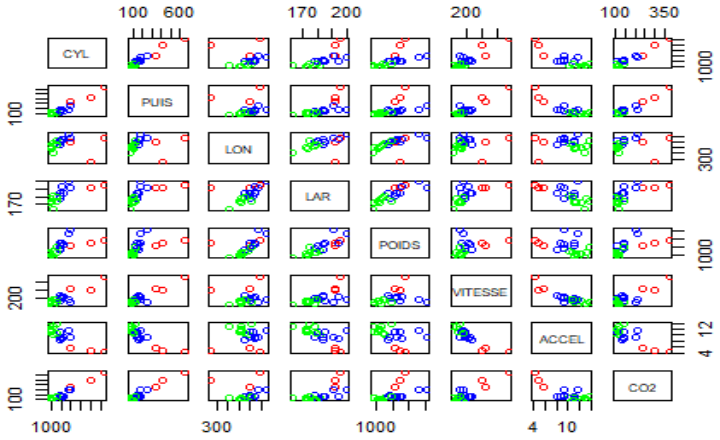
Dans ce cluster, on observe qu'il regroupe les voitures peu puissantes, avec une vitesse modérée et émettant peu de CO2. Cependant, il y a une erreur concernant la BMW S3, classée à tort dans le cluster 3.

Groupe d'appartenance

```
# coloriage selon le groupe d'appartenance pour k=2
```




```
#coloriage selon le groupe d'appartenance pour k=3
```



En observant les résultats avec k=2, on constate que l'algorithme k-means parvient bien à capturer les relations que nous avions identifiées avec les variables CO2, vitesse, et puissance. Cependant, avec k=3, on observe des chevauchements pour l'ensemble des variables, on en conclut que certaines marques sont difficiles à distinguer en fonction de nos variables. Cela met en évidence la difficulté du choix partitionnement quand certaines observations présentent des caractéristiques similaires importantes.

```
#calcul des moyennes conditionnels
```

```
##   Group.1      CYL     PUIS      LON      LAR    POIDS  VITESSE     ACCEL
## 1       1 1654.059 130.0588 437.5882 180.3529 1455.706 189.3529 10.847059
## 2       2 4736.000 473.6667 403.3333 193.0000 1729.000 281.6667  4.866667
```

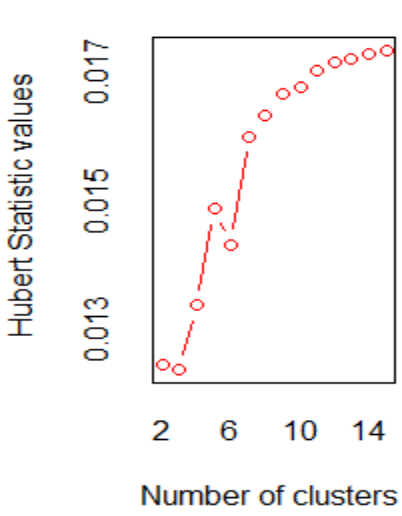
```
##        CO2
## 1 129.8824
## 2 304.3333
```

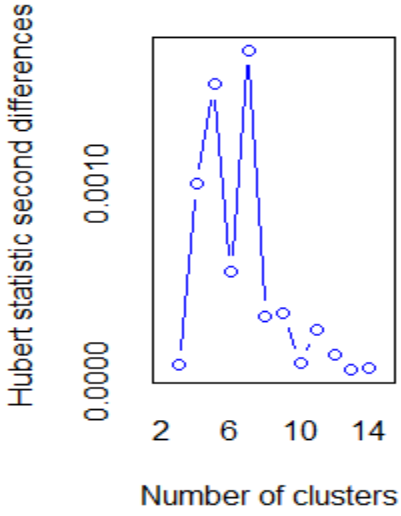
```
print(aggregate(x=vt,by=list(km1$cluster),FUN=mean))
```

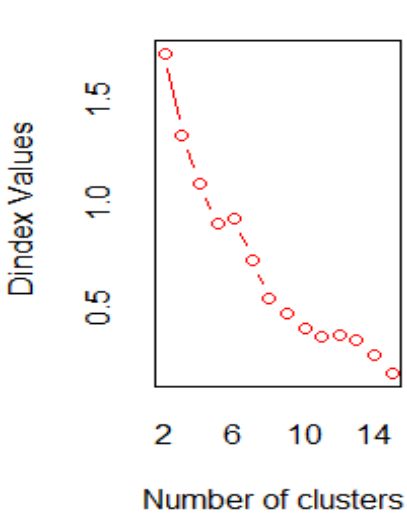
```
##   Group.1      CYL      PUIS      LON      LAR    POIDS  VITESSE     ACCEL
## 1       1 4736.000 473.66667 403.3333 193.0000 1729.000 281.6667  4.866667
## 2       2 2244.000 165.25000 466.5000 187.1250 1802.875 199.1250  9.800000
## 3       3 1129.667  98.77778 411.8889 174.3333 1147.111 180.6667 11.777778
##        CO2
## 1 304.3333
## 2 150.7500
## 3 111.3333
```

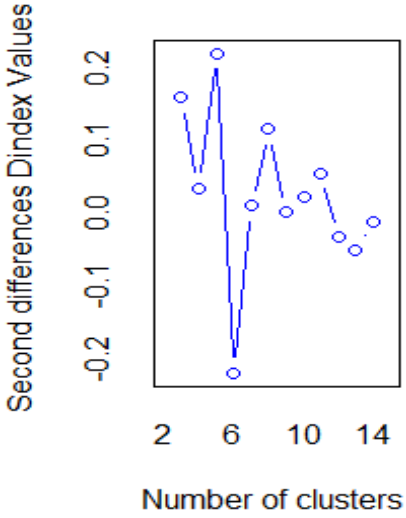
Pour k=2, on observe que le groupe 2 a une vitesse moyenne de 281.6667, tandis que le groupe 1 a une vitesse moyenne de 189.3529. Si on regarde la longueur et la largeur, on constate que les moyennes entre les deux groupes sont proches. Les moyennes conditionnelles donnent des informations sur les caractéristiques moyennes des observations de chaque cluster, cela permet de mettre en évidence les différences spécifiques entre les groupes.

```
library(NbClust)
NbClust( data=vt_stan, distance="euclidean", method="kmeans")
```










```
## *** : The D index is a graphical method of determining the number of
clusters.
##                 In the plot of D index, we seek a significant knee (the
significant peak in Dindex
##                 second differences plot) that corresponds to a significant
increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
```

```
## * 6 proposed 2 as the best number of clusters
## * 4 proposed 3 as the best number of clusters
## * 1 proposed 5 as the best number of clusters
## * 2 proposed 6 as the best number of clusters
## * 1 proposed 8 as the best number of clusters
## * 3 proposed 10 as the best number of clusters
## * 1 proposed 11 as the best number of clusters
## * 1 proposed 12 as the best number of clusters
## * 1 proposed 13 as the best number of clusters
## * 4 proposed 15 as the best number of clusters
##
##                    ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  2
##
##
## *******************************************************************

## $All.index
##         KL      CH Hartigan     CCC      Scott   Marriot   TrCovW  TraceW
## 2   1.8163 16.6248  12.1827 -1.0680   45.5611 1990.6343 236.2514 79.0184
## 3   2.3824 18.9170   6.3078 -0.4674   91.4126  452.3972  42.2141 47.1240
## 4   0.8064 18.2526   9.7171 -0.6093  121.1439  181.8825  27.0387 34.3708
## 5   1.2593 22.9055  -3.8893  0.7892  201.3869    5.1423  16.0159 21.3840
## 6   0.6826 11.9422  11.4475 -3.1378  152.4523   85.5303  21.4281 28.8695
## 7   1.7866 18.5687  10.1430 -0.3428  253.4969    0.7445  10.7787 15.8827
## 8   3.1771 27.4923   3.9916  2.1311  320.2627    0.0345   2.2551  8.9217
## 9   1.4840 29.8434   3.0832  2.4794  354.6873    0.0078   1.5102  6.6948
## 10  2.3281 31.1868   1.4567  2.4836  426.6564    0.0003   1.0647  5.2291
## 11  2.1700 29.0722  -0.2791  1.1124  500.7594    0.0000   0.8527  4.5642
## 12 16.3270 22.7416   0.1877 -1.0997  542.0485    0.0000   0.7277  4.7103
## 13  0.0110 18.6824   3.0436 -3.0846 1050.7277    0.0000   0.7119  4.6023
## 14  0.9192 21.4095   4.9866 -2.8959       NaN    0.0000   0.4480  3.2076
## 15  2.4387 30.6324   2.5610 -1.3588       NaN    0.0000   0.1506  1.7517
##         Friedman   Rubin Cindex     DB Silhouette    Duda Pseudot2   Beale
## 2   1.947515e+02  1.9236 0.2744 1.0307     0.4914  0.7694   2.6970  1.1878
## 3   3.194506e+02  3.2255 0.3009 0.9206     0.3454  1.4024  -3.7303 -1.3270
## 4   3.490288e+02  4.4224 0.2371 0.7930     0.3616  1.3411  -0.2543 -0.6721
## 5   9.177279e+02  7.1081 0.2977 0.6909     0.4047  1.2464  -1.1859 -0.8954
## 6   4.550759e+02  5.2651 0.2111 0.6142     0.3928 18.3750  -0.9456  0.0000
## 7   1.374556e+03  9.5702 0.2458 0.5598     0.4359  4.4688   0.0000  0.0000
## 8   1.736627e+03 17.0372 0.4525 0.4235     0.5296  4.0065  -3.0016 -3.1729
## 9   2.354206e+03 22.7043 0.4512 0.5531     0.4906  1.5881  -1.1109 -1.4678
## 10  4.310240e+03 29.0682 0.3871 0.5157     0.5273  2.6393  -1.8633 -2.4620
## 11  2.925813e+04 33.3024 0.3752 0.4901     0.5310  1.6518  -0.7892 -1.3903
## 12  1.642826e+05 32.2698 0.4861 0.6126     0.4179 44.6638   0.0000  0.0000
## 13  4.969007e+15 33.0269 0.5320 0.5134     0.4956  3.9975  -0.7498 -1.9815
## 14 -1.566666e+16 47.3872 0.4366 0.4432     0.5703  5.9147   0.0000  0.0000
## 15  3.794026e+16 86.7708 0.6461 0.3939     0.6437  4.1879  -0.7612 -2.0116
##    Ratkowsky    Ball Ptbiserial     Frey McClain   Dunn Hubert SDindex
```

```
Dindex
## 2     0.4648 39.5092     0.7108   3.3525  0.2618 0.3386 0.0125  6.4995
1.7239
## 3     0.4744 15.7080     0.5311   0.4238  0.7845 0.1827 0.0124  4.9452
1.3329
## 4     0.4368  8.5927     0.5348  -0.0775  0.9470 0.1293 0.0134  3.8002
1.1017
## 5     0.4134  4.2768     0.5567 -42.4903  0.9104 0.1779 0.0149  2.9160
0.9047
## 6     0.3658  4.8116     0.4735  -0.0649  1.3219 0.1293 0.0143  3.1898
0.9283
## 7     0.3571  2.2690     0.4997   0.0166  1.2222 0.1779 0.0160  2.7579
0.7313
## 8     0.3429  1.1152     0.5070   1.2704  1.1912 0.3250 0.0163  1.6966
0.5448
## 9     0.3259  0.7439     0.4190   0.6323  1.8106 0.5055 0.0167  2.2900
0.4742
## 10    0.3107  0.5229     0.3915   3.8837  2.0584 0.5556 0.0168  2.7302
0.4054
## 11    0.2969  0.4149     0.3484  -0.6662  2.6306 0.4197 0.0170  3.1004
0.3599
## 12    0.2841  0.3925     0.2547  -0.2724  5.2436 0.2278 0.0172  6.8880
0.3691
## 13    0.2731  0.3540     0.2346   0.1668  6.2421 0.2278 0.0172  6.8822
0.3458
## 14    0.2644  0.2291     0.2173   0.0755  7.1978 0.2278 0.0173  6.9519
0.2716
## 15    0.2567  0.1168     0.2111   0.2530  7.3468 0.3609 0.0173  6.6535
0.1863
##      SDbw
## 2  0.9164
## 3  0.8279
## 4  0.5141
## 5  0.2823
## 6  0.2703
## 7  0.1872
## 8  0.0383
## 9  0.0387
## 10 0.0284
## 11 0.0236
## 12 0.0298
## 13 0.0268
## 14 0.0180
## 15 0.0085
##
## $All.CriticalValues
##    CritValue_Duda CritValue_PseudoT2 Fvalue_Beale
## 2          0.2421            28.1750       0.3467
## 3          0.4148            18.3382       1.0000
## 4          0.0447            21.3664       1.0000
```

```
## 5          0.3842             9.6185       1.0000
## 6         -0.1521            -7.5745          NaN
## 7         -0.1521             0.0000          NaN
## 8          0.3012             9.2802       1.0000
## 9          0.2421             9.3917       1.0000
## 10         0.2421             9.3917       1.0000
## 11         0.1620            10.3432       1.0000
## 12        -0.1521             0.0000          NaN
## 13         0.0447            21.3664       1.0000
## 14        -0.1521             0.0000          NaN
## 15         0.0447            21.3664       1.0000
##
## $Best.nc
##                       KL      CH Hartigan     CCC    Scott  Marriot   TrCovW
## Number_clusters 12.000 10.0000   6.0000 10.0000  13.0000    3.000   3.0000
## Value_Index     16.327 31.1868  15.3368  2.4836 508.6792 1267.723 194.0373
##                   TraceW     Friedman   Rubin Cindex      DB Silhouette
Duda
## Number_clusters  5.0000 1.500000e+01 11.0000 6.0000 15.0000    15.0000
2.0000
## Value_Index     20.4724 5.360692e+16 -5.2669 0.2111  0.3939     0.6437
0.7694
##                 PseudoT2  Beale Ratkowsky    Ball PtBiserial   Frey
McClain
## Number_clusters    2.000 2.0000    3.0000  3.0000     2.0000 2.0000
2.0000
## Value_Index        2.697 1.1878    0.4744 23.8012     0.7108 3.3525
0.2618
##                     Dunn Hubert SDindex Dindex    SDbw
## Number_clusters 10.0000      0  8.0000      0 15.0000
## Value_Index      0.5556      0  1.6966      0  0.0085
##
## $Best.partition
##     ALPHAMITO        AUDIA1     CITROENC4       JAGUARF    PEUGEOTRCZ
##             2             2             2             1             2
##     LANDROVER   RENAULTCLIO         BMWS3         DACIA       HYUNDAI
##             1             2             2             2             2
##        LANCIA RENAULTCAPTUR   FORDMUSTANG       FIAT500         HONDA
##             1             2             1             2             2
##       FERRARI        SUBARU         MAZDA    VOLKSWAGEN    JAGUARPACE
##             1             2             2             2             2
```

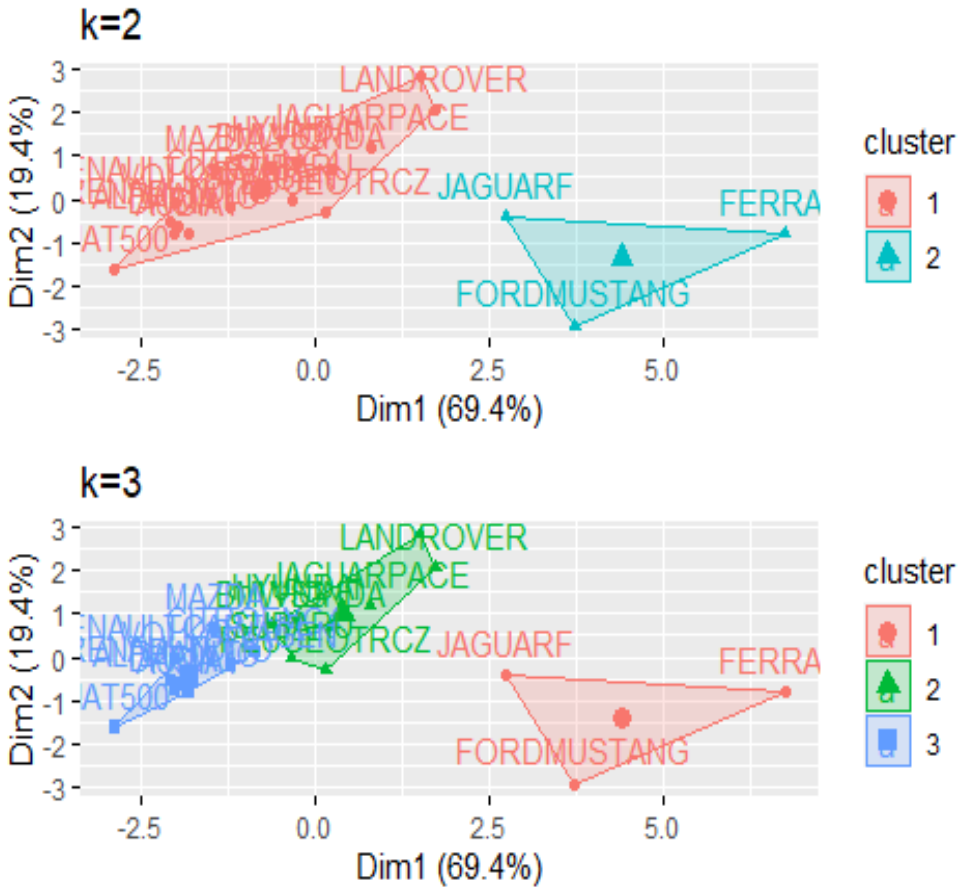
Pour déterminer explicitement le nombre optimal de clusters, on peut utiliser plusieurs indices pour différents nombres de clusters. Dans notre cas, la majorité d'entre eux suggèrent que 2 est le nombre optimal, avec une partition de 5 pour le groupe 1 et 15 pour le groupe 2.

Vérification de la qualité de la segmentation

```
metric<-silhouette(km$cluster,dist(vt_stan, "euclidean"))

## Silhouette of 20 units in 2 clusters from silhouette.default(x =
km$cluster, dist = dist(vt_stan, "euclidean")) :
##  Cluster sizes and average silhouette widths:
##        17         3
## 0.6010531 0.2187474
## Individual silhouette widths:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.01243  0.44381  0.64808  0.54371  0.68995  0.70953

metric1<-silhouette(km1$cluster,dist(vt_stan, "euclidean"))

## Silhouette of 20 units in 3 clusters from silhouette.default(x =

km1$cluster, dist = dist(vt_stan, "euclidean")) :

##  Cluster sizes and average silhouette widths:

##         3         8         9

## 0.1093000 0.2495165 0.5094129

## Individual silhouette widths:

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.

## -0.2045  0.2019  0.3860  0.3454  0.5298  0.6736
```

Le coefficient de silhouette varie entre [-1, 1], et plus la moyenne du coefficient est proche de 1, meilleure est la qualité de la segmentation. On observe que pour k=2, nous avons une meilleure séparation avec une moyenne de 0.54, tandis que pour k=3, la moyenne est de 0.3454. Ces résultats confirment la tendance observée précédemment, indiquant que k=2 est une segmentation de meilleure qualité pour notre ensemble de données.

GRAPHE



Avec k=2, on observe une séparation claire des groupes dans l'espace des variables. Cependant, pour k=3, on constate un chevauchement, indiquant une certaine difficulté dans la classification.

Maintenant, on va associer chaque marque à son groupe respectif.

Comme le nombre de clusters qu'il faut utiliser n'est pas clair, on a utilisé un autre algorithme qui est une variante du kmeans qui attribue chaque observation un degré d'appartenance à tous les clusters, le degré est compris entre [0,1].

```
## Iteration:   1, Error:  3.8146194187
## Iteration:   2, Error:  3.7626422366
## Iteration:   3, Error:  3.6387150193
## Iteration:   4, Error:  3.3512232820
## Iteration:   5, Error:  3.1263953478
## Iteration:   6, Error:  3.0604687202
## Iteration:   7, Error:  3.0363238490
## Iteration:   8, Error:  3.0220100084
## Iteration:   9, Error:  3.0127547089
## Iteration:  10, Error:  3.0068151979
```

```
## Iteration:  11, Error:  3.0030851015
## Iteration:  12, Error:  3.0007957621
## Iteration:  13, Error:  2.9994197457
## Iteration:  14, Error:  2.9986072095
## Iteration:  15, Error:  2.9981342713
## Iteration:  16, Error:  2.9978621196
## Iteration:  17, Error:  2.9977068940
## Iteration:  18, Error:  2.9976189603
## Iteration:  19, Error:  2.9975694049
## Iteration:  20, Error:  2.9975415873
## Iteration:  21, Error:  2.9975260183
## Iteration:  22, Error:  2.9975173240
## Iteration:  23, Error:  2.9975124768
## Iteration:  24, Error:  2.9975097779
## Iteration:  25, Error:  2.9975082765
## Iteration:  26, Error:  2.9975074419
## Iteration:  27, Error:  2.9975069781
## Iteration:  28, Error:  2.9975067206
## Iteration:  29, Error:  2.9975065776
## Iteration:  30, Error:  2.9975064982
## Iteration:  31 converged, Error:  2.9975064541
```

```
# les centres des clusters
```

```
##          CYL      PUIS          LON        LAR      POIDS    VITESSE
ACCEL
## 1 -0.4322953 -0.423642 -0.062313558 -0.4114578 -0.3666628 -0.3544864
0.3903694
## 2  1.3665413  1.443939  0.004785344  1.1226731  0.8398773  1.3026945
-1.3473986
##          CO2
## 1 -0.4533454
## 2  1.4775028
```

```
#  les degrés d'appartenance
```

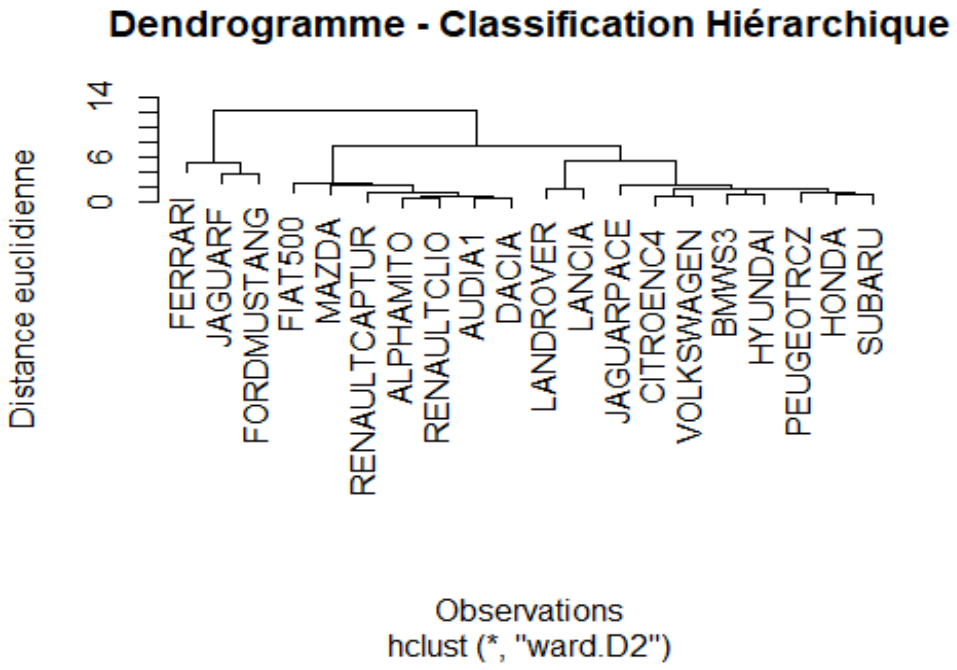
```
##                        1           2
## ALPHAMITO     0.95887976 0.041120244
## AUDIA1        0.95775467 0.042245326
## CITROENC4     0.97762186 0.022378137
## JAGUARF       0.08321521 0.916784786
## PEUGEOTRCZ    0.83286910 0.167130895
## LANDROVER     0.41908122 0.580918779
## RENAULTCLIO   0.95634995 0.043650051
## BMWS3         0.94948236 0.050517636
## DACIA         0.94760937 0.052390627
```

```
## HYUNDAI        0.89806387 0.101936132
## LANCIA         0.47646018 0.523539823
## RENAULTCAPTUR 0.95774902 0.042250984
## FORDMUSTANG    0.22241788 0.777582118
## FIAT500        0.86058246 0.139417539
## HONDA          0.82425498 0.175745021
## FERRARI        0.18597205 0.814027946
w## SUBARU          0.94665284 0.053347155
## MAZDA          0.93713090 0.062869096
## VOLKSWAGEN     0.99004507 0.009954934
## JAGUARPACE     0.63378705 0.366212950
```

On conclut donc il serait judicieux de prendre k=2 et avec 5 marques pour le groupe 1 et 15 pour le second

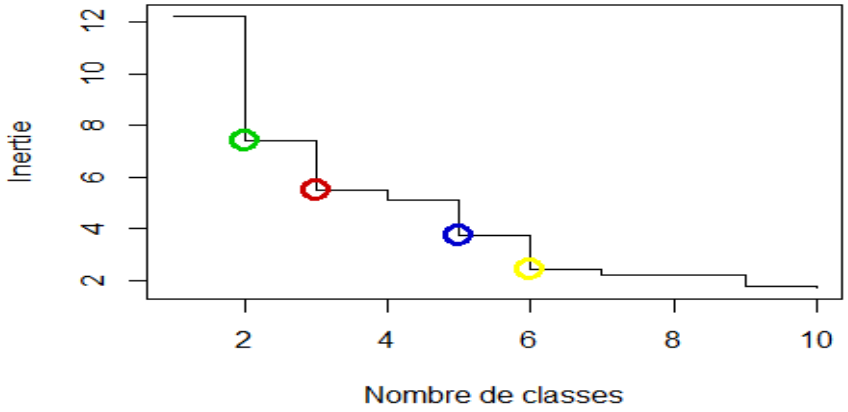
3.4.- Classification avec la méthode de Ward

C'est une méthode qui regroupe les observations sur le critère de la distance séparant les points.

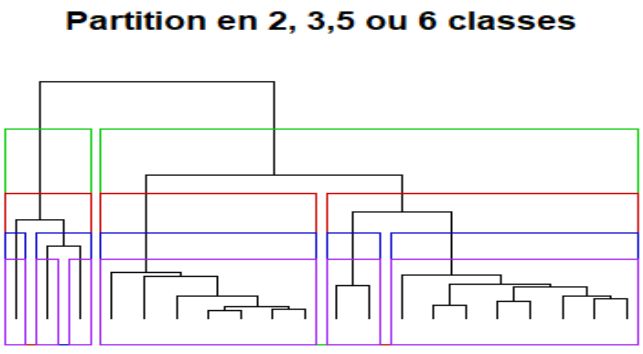


On observe qu'à la hauteur 12 du dendrogramme, il divise les observations en 2, c'est le résultat qu'on avait avec la méthode kmeans pour k=2. Ensuite à la hauteur 7 pour la partie 2, on a également une division des observations en 2. Pour la première classe on obtient le même partitionnement qu'on avait avec kmeans.

Pour trouver le nombre optimal de cluster on peut visualiser l'inertie en fonction du nombre de clusters.



On observe donc 4 sauts qui sont assez elevés.



Suite à l'analyse hiérarchique, On a observé 4 sauts (2,3,5 et 6), le saut le plus élevé est le partitionnement pour en 2 clusters, qui montre une meilleure cohésion. En utilisant l'indice de silhouette, on a que le nombre optimal de clusters est effectivement 2.

En conclusion le choix du cluster est très flou, la meilleure solution serait d'avoir une idée métier des experts du domaine ou d'explorer davantage la nature des données.

4.-Analyse des correspondances multiples

On dispose du fichier "chiens" qui détaille les caractéristiques des chiens en fonction de la race. Les variables sont les suivantes :

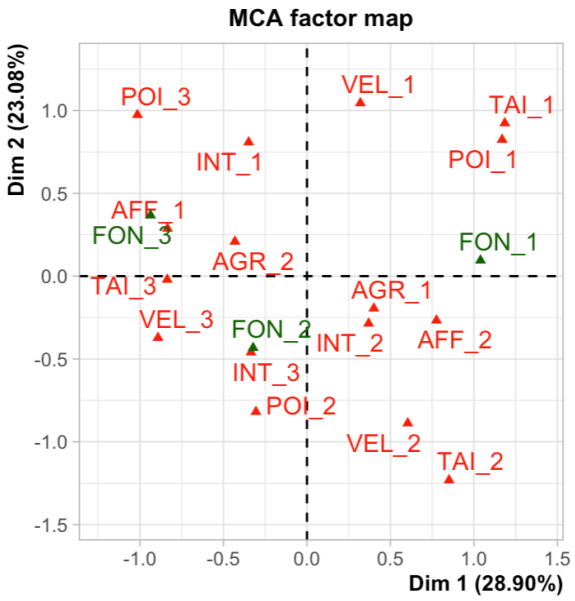
TAI (Taille) : 3 = grande, 2 = moyenne et 1 = petite.
POI (Poids) : 3 = lourd, 2 = moyen et 1 = léger.
VEL (Vélocité) : 3 = rapide, 2 = moyenne et 1 = lent.
INT (Intelligence) : 3 = grande, 2 = moyenne et 1 = faible.
AFF (Affectuosité) : 2 = grande et 1 = faible.
AGR (Agressivité) : 2 = grande et 1 = faible.
FON (Fonction) : 3 = garde, 2 = chasse et 1 = compagnie

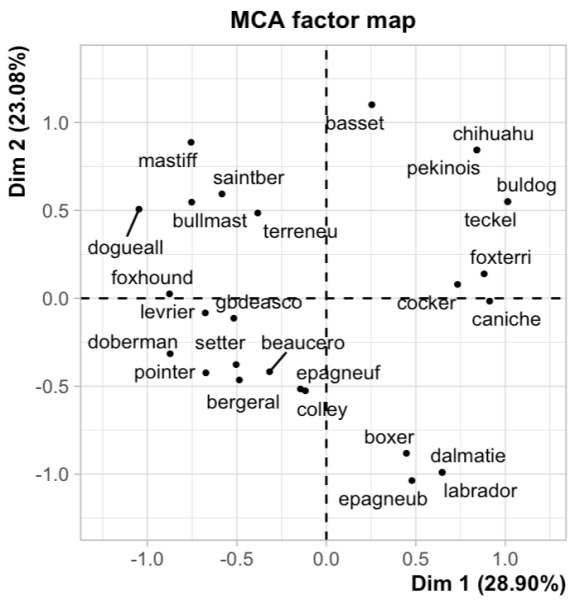
4.1.- En prenant la variable FON comme variable supplémentaire, faire une analyse des correspondances multiples de ces données

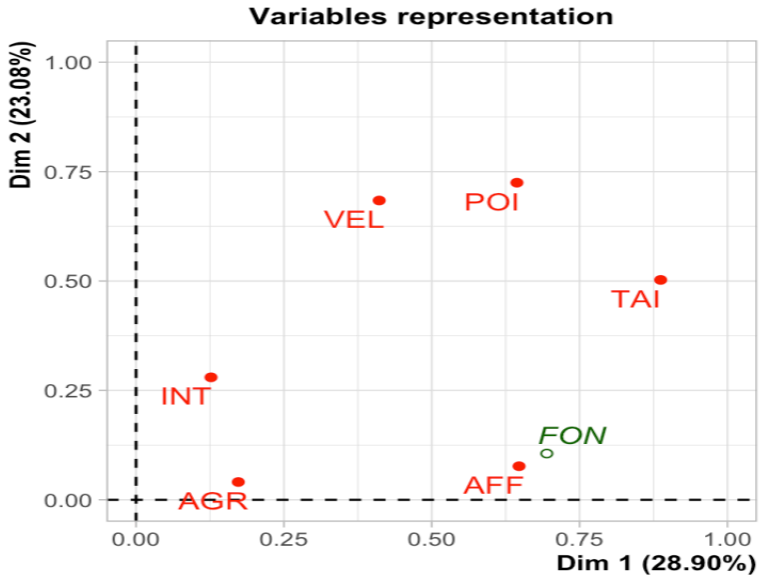
Nous avons commencé par regarder notre table de données afin de comprendre nos données.

	TAI	POI	VEL	INT	AFF	AGR	FON
beaucero	3	2	3	2	2	2	3
basset	1	1	1	1	1	2	2
bergeral	3	2	3	3	2	2	3
boxer	2	2	2	2	2	2	1
buldog	1	1	1	2	2	1	1
bullmast	3	3	1	3	1	2	3
caniche	1	1	2	3	2	1	1
chihuahu	1	1	1	1	2	1	1
cocker	2	1	1	2	2	2	1
colley	3	2	3	2	2	1	1
dalmatie	2	2	2	2	2	1	1
doberman	3	2	3	3	1	2	3
dogueall	3	3	3	1	1	2	3
epagneub	2	2	2	3	2	1	2
epagneuf	3	2	2	2	1	1	2
foxhound	3	2	3	1	1	2	2
foxterri	1	1	2	2	2	2	1
gbdeasco	3	2	2	1	1	2	2
labrador	2	2	2	2	2	1	2
levrier	3	2	3	1	1	1	2
mastiff	3	3	1	1	1	2	3
pekinois	1	1	1	1	2	1	1
pointer	3	2	3	3	1	1	2
saintber	3	3	1	2	1	2	3
setter	3	2	3	2	1	1	2
teckel	1	1	1	2	2	1	1
terreneu	3	3	1	2	1	1	3

Grâce à l'analyse des correspondances multiples, nous avons obtenus les graphiques suivants :







Le premier graphique représente le nuage des modalités.

Le taux d'inertie expliquée par le premier plan factoriel est d'environ 52%.

Puis, le second graphique représente la projection des individus dans le premier plan factoriel.

Et enfin, le troisième graphique nous montre la représentation des variables.

Ensuite, nous avons cherché à regarder les liens entre les individus et les variables.

Pour avoir un aperçu des individus et des variables, on peut faire une description des dimensions

Nous avons donc obtenu ces tables de données :

```
$`Dim 1`

Link between the variable and the categorical variable (1-way anova)
============================================
           R2      p.value
TAI 0.8870733 4.300901e-12
AFF 0.6476559 4.184726e-07
FON 0.6945841 6.587193e-07
POI 0.6440465 4.137327e-06
VEL 0.4111741 1.737173e-03
AGR 0.1729238 3.098130e-02

Link between variable and the categories of the categorical variables
============================================================
            Estimate      p.value
AFF=AFF_2  0.5588770 4.184726e-07
FON=FON_1  0.7720647 6.464827e-07
POI=POI_1  0.8462780 4.553328e-06
TAI=TAI_1  0.5448885 4.642696e-05
AGR=AGR_1  0.2887829 3.098130e-02
TAI=TAI_2  0.3131915 3.574570e-02
VEL=VEL_2  0.4116525 4.330936e-02
AGR=AGR_2 -0.2887829 3.098130e-02
POI=POI_3 -0.6694073 1.053636e-02
FON=FON_3 -0.5991170 7.740227e-04
VEL=VEL_3 -0.6263912 4.193918e-04
AFF=AFF_1 -0.5588770 4.184726e-07
TAI=TAI_3 -0.8580800 8.616471e-13

$`Dim 2`

Link between the variable and the categorical variable (1-way anova)
============================================
           R2      p.value
POI 0.7246877 1.896299e-07
VEL 0.6840074 9.911136e-07
TAI 0.5024857 2.299680e-04
INT 0.2798701 1.945048e-02

Link between variable and the categories of the categorical variables
============================================================
            Estimate      p.value
VEL=VEL_1  0.6925225 5.081985e-07
TAI=TAI_1  0.6409665 3.178556e-03
POI=POI_1  0.3088260 4.043759e-03
INT=INT_1  0.4883806 4.960456e-03
POI=POI_3  0.4015914 1.470012e-02
VEL=VEL_2 -0.5062855 1.662406e-03
TAI=TAI_2 -0.6961017 1.281728e-03
POI=POI_2 -0.7104174 2.047241e-08

$`Dim 3`

Link between the variable and the categorical variable (1-way anova)
============================================
           R2     p.value
POI 0.3422226 0.006560656
VEL 0.2914908 0.016000953
TAI 0.2910127 0.016131004
INT 0.2337099 0.040993867

Link between variable and the categories of the categorical variables
============================================================
            Estimate     p.value
POI=POI_3  0.4643851 0.001436675
TAI=TAI_2  0.4132636 0.010456295
INT=INT_2  0.2966816 0.012234739
VEL=VEL_3 -0.3497115 0.003675718
```

Cependant, l'interprétation n'étant pas évidente, nous avons affiché les contributions des variables et des individus pour une meilleure analyse.

	Dim 1	Dim 2	Dim 3		Dim 4	Dim 5
TAI_1	12.5978150	9.58661729	7.7724161	TAI_1	0.39590918	0.011470466
TAI_2	4.6420727	12.17067028	15.1042375	TAI_2	2.29739639	1.976131613
TAI_3	13.4585463	0.01019149	0.1151394	TAI_3	1.70285200	0.782823704
POI_1	14.0104164	8.72224556	3.0131477	POI_1	0.85212159	0.086298489
POI_2	1.6736860	15.06207234	2.1911472	POI_2	0.76846202	2.082406452
POI_3	6.6040417	7.60886705	21.8333953	POI_3	0.08953284	7.763356494
VEL_1	1.3119931	17.51742290	4.7224026	VEL_1	0.25282794	3.847622109
VEL_2	3.7368537	10.11705778	2.9720027	VEL_2	4.29655213	4.528255671
VEL_3	9.1804174	1.99644722	15.3351608	VEL_3	2.02848142	0.003766389
INT_1	1.2492400	8.39130827	2.8924483	INT_1	0.01841315	35.239586898
INT_2	2.2742083	1.70014447	9.2531808	INT_2	18.54990013	1.143358563
INT_3	0.8633836	2.03240794	6.3188861	INT_3	38.22931061	27.885847123
AFF_1	11.6215827	1.72364624	0.1630694	AFF_1	0.35219351	0.100786614
AFF_2	10.7914697	1.60052866	0.1514216	AFF_2	0.32703683	0.093587570
AGR_1	2.8813167	0.84758676	3.9298252	AGR_1	14.36693087	6.959671259
AGR_2	3.1029565	0.91278575	4.2321194	AGR_2	15.47207940	7.495030587

	Dim 1	Dim 2	Dim 3
beaucero	0.7737679	1.679591265	0.18076074
basset	0.4965777	11.674160199	0.63849009
bergeral	1.8193797	2.076582082	4.35652825
boxer	1.5391043	7.484976084	8.40776174
buldog	7.8970523	2.910764017	0.46889522
bullmast	4.3555519	2.879430481	4.34672445
caniche	6.4006040	0.002522556	5.83649387
chihuahu	5.4366197	6.854956241	3.87745169
cocker	4.1352521	0.060190825	7.69957277
colley	0.1058588	2.664533548	1.96907570
dalmatie	3.2216221	9.438522677	3.69230120
doberman	5.8638360	0.958117236	3.59193035
dogueall	8.4304649	2.474089271	0.47818860
epagneub	1.7574399	10.350777943	0.06732262
epagneuf	0.1614884	2.560978462	0.24085714
foxhound	5.9090133	0.006132611	2.30290971
foxterri	5.9773564	0.185906708	0.05029451
gbdeasco	2.0582234	0.123802149	0.03403454
labrador	3.2216221	9.438522677	3.69230120
levrier	3.5214957	0.066583845	6.22807925
mastiff	4.3944998	7.584703763	6.06432357
pekinois	5.4366197	6.854956241	3.87745169
pointer	3.4866395	1.729709149	8.25594928
saintber	2.6172536	3.392717496	14.03963501
setter	1.9545479	1.369226351	1.46711842
teckel	7.8970523	2.910764017	0.46889522
terreneu	1.1310567	2.266782110	7.66665317

Par la suite, afin d'évaluer la qualité de la représentation des modalités, nous avons affiché les cosinus carrés.

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
TAI_1	0.49144201	0.2987546600	0.132809435	0.005052544	1.394894e-04
TAI_2	0.16462520	0.3448030588	0.234627550	0.026653716	2.184658e-02
TAI_3	0.87503205	0.0005293413	0.003279033	0.036219310	1.586620e-02
POI_1	0.57531341	0.2861238116	0.054196308	0.011447021	1.104688e-03
POI_2	0.10044717	0.7221387844	0.057601141	0.015087719	3.895941e-02
POI_3	0.23420393	0.2155641859	0.339157541	0.001038734	8.582564e-02
VEL_1	0.06021292	0.6422447857	0.094932948	0.003795952	5.504701e-02
VEL_2	0.15344741	0.3318791146	0.053456249	0.057717964	5.796523e-02
VEL_3	0.39792110	0.0691296921	0.291151283	0.028763587	5.089122e-05
INT_1	0.05129787	0.2752677726	0.052025334	0.000247354	4.510944e-01
INT_2	0.12673870	0.0756897524	0.225873819	0.338187895	1.986299e-02
INT_3	0.03207684	0.0603213262	0.102831012	0.464645451	3.229645e-01
AFF_1	0.64765585	0.0767360421	0.003980589	0.006420928	1.750915e-03
AFF_2	0.64765585	0.0767360421	0.003980589	0.006420928	1.750915e-03
AGR_1	0.17292377	0.0406368567	0.103307716	0.282075371	1.302074e-01
AGR_2	0.17292377	0.0406368567	0.103307716	0.282075371	1.302074e-01

Pour mesurer la qualité de la représentation d'une variable, il faut calculer la somme des cosinus carrés sur les deux dimensions concernées.

Ici nous sommes sur le premier plan factoriel, on obtient donc :

TAI_1 : 0.49144201 + 0.2987546600 = 0.7901966700

TAI_2 : 0.16462520 + 0.3448030588 = 0.5094282588

TAI_3 : 0.87503205 + 0.0005293413 = 0.8755613913

POI_1 : 0.57531341 + 0.2861238116 = 0.8614372216

POI_2 : 0.10044717 + 0.7221387844 = 0.8225859544

POI_3 : 0.23420393 + 0.2155641859 = 0.4497681159

VEL_1 : 0.06021292 + 0.6422447857 = 0.7024577057

VEL_2 : 0.15344741 + 0.3318791146 = 0.4853265246

VEL_3 : 0.39792110 + 0.0691296921 = 0.4670507921

INT_1 : 0.05129787 + 0.2752677726 = 0.3265656426

INT_2 : 0.12673870 + 0.0756897524 = 0.2024284524

INT_3 : 0.03207684 + 0.0603213262 = 0.0923981662

AFF_1 : 0.64765585 + 0.0767360421 = 0.7243918921

AFF_2 : 0.64765585 + 0.0767360421 = 0.7243918921

AGR_1 : 0.17292377 + 0.0406368567 = 0.2135606267

AGR_2 : 0.17292377 + 0.0406368567 = 0.2135606267

Les variables qui sont bien représentés sur le premier plan factoriel (affichées en noir) c'est-à-dire que la somme de leur cosinus carré de la dimension 1 et 2 est supérieur à 0,5 sont : TAI_1, TAI_2, TAI_3, POI_1, POI_2, VEL_1, AFF_1 et AFF_2.

Nous avons fait de même pour notre variable supplémentaire FON :

```
          Dim 1       Dim 2       Dim 3       Dim 4
FON_1 0.63542896 0.005344696 0.002711706 0.004655675
FON_2 0.05196785 0.093498803 0.060812866 0.016917930
FON_3 0.36926611 0.056814951 0.095887149 0.042617046
            Dim 5
FON_1 7.303724e-05
FON_2 9.809804e-02
FON_3 1.104784e-01
```

FON_1 : 0.63542896+0.005344696=0.640773656

FON_2 : 0.05196785+0.093498803=0.145466653

FON_3 : 0.36926611+0.056814951=0.426081061

On observe que FON_1 est bien représenté avec une somme des cosinus carrés d'environ 0,64. A l'inverse, FON_2 et FON_3 sont mal représentés avec une somme des cosinus carrés respectives de 0,14 et 0,42.

C'est le même principe du côté des individus :

```
              Dim 1        Dim 2       Dim 3
beaucero 0.08863547 0.1536995944 0.009069781
basset   0.03380431 0.6348671357 0.019038597
bergeral 0.15372250 0.1401636585 0.161231704
boxer    0.11133075 0.4325235284 0.266393303
buldog   0.62448464 0.1838806326 0.016241584
bullmast 0.27069077 0.1429582059 0.118328181
caniche  0.38519392 0.0001212751 0.153853124
chihuahu 0.37993129 0.3826952203 0.118691149
cocker   0.27915682 0.0032460020 0.227671518
colley   0.01239617 0.2492609870 0.100999475
dalmatie 0.23628517 0.5530165596 0.118619154
doberman 0.48761169 0.0636477694 0.130832617
dogueall 0.56079391 0.1314738467 0.013933070
epagneub 0.10498339 0.4939526916 0.001761558
epagneuf 0.01753323 0.2221256292 0.011454493
foxhound 0.55831304 0.0004628928 0.095309358
foxterri 0.43627101 0.0108396312 0.001607917
gbdeasco 0.18602321 0.0089387139 0.001347381
labrador 0.23628517 0.5530165596 0.118619154
levrier  0.33881559 0.0051177295 0.262473933
mastiff  0.29999507 0.4136333336 0.181335511
pekinois 0.37993129 0.3826952203 0.118691149
pointer  0.29459212 0.1167506763 0.305546227
saintber 0.20156282 0.2087298540 0.473604996
setter   0.22389437 0.1252980645 0.073613569
teckel   0.62448464 0.1838806326 0.016241584
terreneu 0.08840069 0.1415315741 0.262465947
```

Voici la somme des cosinus carrés des individus sur la dimension 1 et 2 :

beaucero : 0.08863547 + 0.1536995944 = 0.2423350644

basset : 0.03380431 + 0.6348671357 = 0.6686714457

bergéral : 0.15372250 + 0.1401636585 = 0.2938861585

boxer : 0.11133075 + 0.4325235284 = 0.5438542784

bulldog : 0.62448464 + 0.1838806326 = 0.8083652726

bullmast : 0.27069077 + 0.1429582059 = 0.4136489759

caniche : 0.38519392 + 0.0001212751 = 0.3853151951

chihuahua : 0.37993129 + 0.3826952203 = 0.7626265103

cocker : 0.27915682 + 0.0032460020 = 0.282402822

colley : 0.01239617 + 0.2492609870 = 0.261657157

dalmatie : 0.23628517 + 0.5530165596 = 0.7893017296

doberman : 0.48761169 + 0.0636477694 = 0.5512594594

dogueallemand : 0.56079391 + 0.1314738467 = 0.6922677567

épagneubrit : 0.10498339 + 0.4939526916 = 0.5989360816

épagneufran : 0.01753323 + 0.2221256292 = 0.2396588592

foxhound : 0.55831304 + 0.0004628928 = 0.5587759328

foxterrier : 0.43627101 + 0.0108396312 = 0.4471106412

goldenretriever : 0.18602321 + 0.0089387139 = 0.1949619239

labrador : 0.23628517 + 0.5530165596 = 0.7893017296

levrier : 0.33881559 + 0.0051177295 = 0.3439333195

mastiff : 0.29999507 + 0.4136333336 = 0.7136284036

pékinois : 0.37993129 + 0.3826952203 = 0.7626265103

pointer : 0.29459212 + 0.1167506763 = 0.4113427963

saintbernard : 0.20156282 + 0.2087298540 = 0.410292674

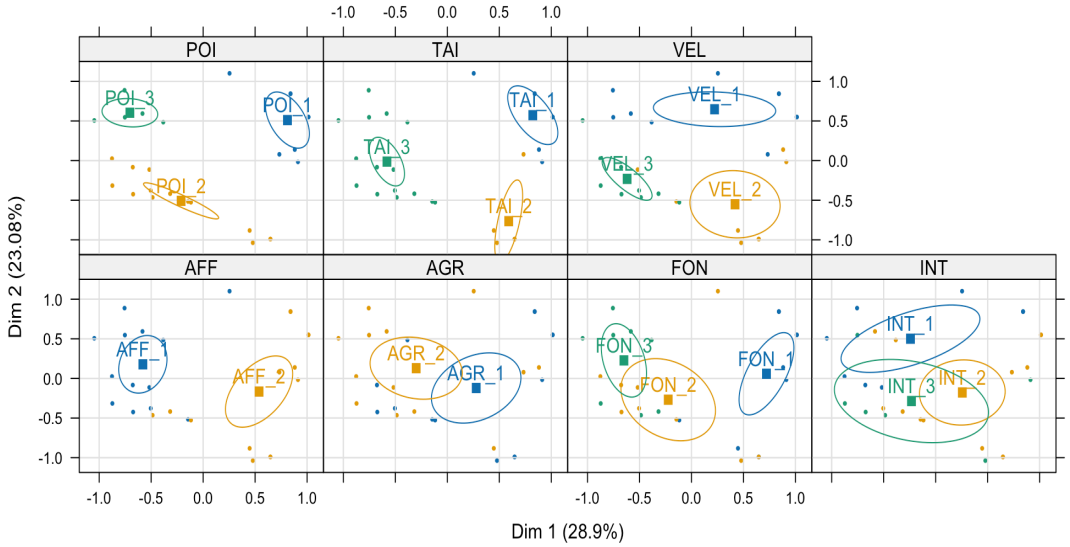
setter : 0.22389437 + 0.1252980645 = 0.3491924345

teckel : 0.62448464 + 0.1838806326 = 0.8083652726

terreneuve : 0.08840069 + 0.1415315741 = 0.2299322641

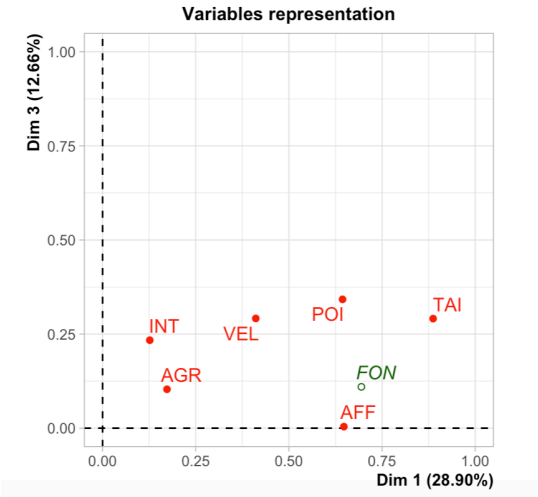
On remarque qu'il y a presque autant d'individus bien représentés (en noir) que mal représentés (en bleu). Parmi les individus les mieux représentés, on a : les teckels avec une somme des cosinus carré d'environ 0,8 suivie par les labradors et les dalmatien avec une somme des cosinus carré d'environ 0,79.

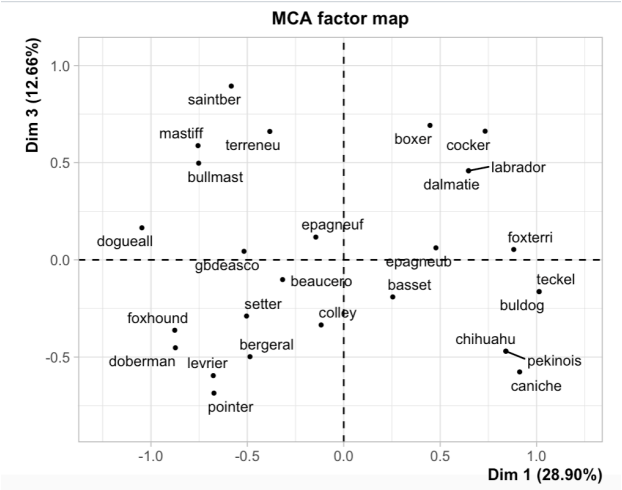
Et enfin, nous avons cherché à voir si les modalités sont similaires puis nous avons obtenu cet ensemble de graphiques représenté par des ellipses.

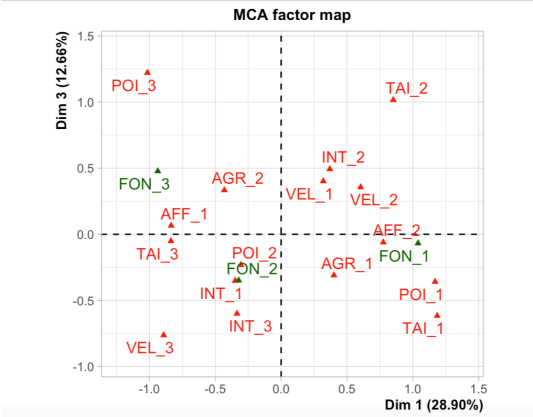


Les ellipses entourent les groupes d'individus qui ont des modalités similaires. Plus une ellipse est grande, plus la dispersion des individus est importante dans cette direction.
Si les ellipses se touchent, les individus sont confondus.

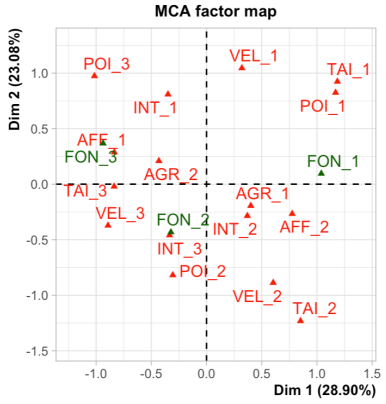
Pour mener nos investigations plus loin, nous nous sommes intéressés à d'autres dimensions. En effet, par exemple, certaines variables mal représentés ont une plus grosse contribution pour l'axe 3, on a donc générer des graphiques sur les dimension 1 et 3 :







4.2.- En déduire une description des différentes races de chiens



	Dim 1	Dim 2
TAI_1	12.5978150	9.58661729
TAI_2	4.6420727	12.17067028
TAI_3	13.4585463	0.01019149
POI_1	14.0104164	8.72224556
POI_2	1.6736860	15.06207234
POI_3	6.6040417	7.60886705
VEL_1	1.3119931	17.51742290
VEL_2	3.7368537	10.11705778
VEL_3	9.1804174	1.99644722
INT_1	1.2492400	8.39130827
INT_2	2.2742083	1.70014447
INT_3	0.8633836	2.03240794
AFF_1	11.6215827	1.72364624
AFF_2	10.7914697	1.60052866
AGR_1	2.8813167	0.84758676
AGR_2	3.1029565	0.91278575

Les contributions

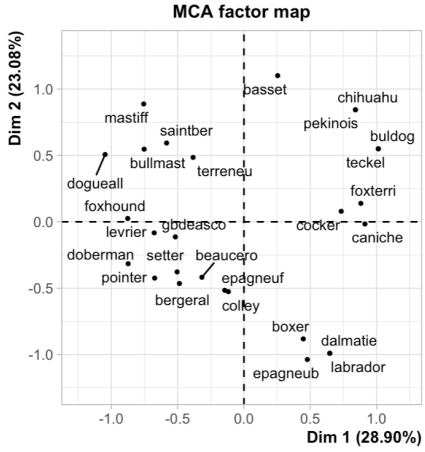
Pour l'axe 1, les variables qui contribuent le plus sont POI_1 (poids léger) avec une contribution de 14% et TAI_3 (grande taille) avec une contribution de 13%.

Ainsi, le premier axe oppose les individus légers (situés à droite) et petits des individus lourds et grands (situés à gauche).

Pour l'axe 2, les variables qui contribuent le plus sont VEL_1 (vélocité lente) avec une contribution de 17,5% et POI_2 (poids moyen) avec une contribution de 15%.

Ainsi, on pourra dire que les individus qui sont lents sont situés en haut et ceux qui sont situés plus bas sont plus rapides.

On peut remarquer que les variables INT_2 et INT_3 contribuent peu sur l'axe 1 et 2 mais beaucoup plus sur l'axe 3 et 4. En effet, notamment sur l'axe 4 où INT_3 a une contribution de 38,2% et INT_2 a une contribution de 18,5%.



	Dim 1	Dim 2
beaucero	0.7737679	1.679591265
basset	0.4965777	11.674160199
bergeral	1.8193797	2.076582082
boxer	1.5391043	7.484976084
buldog	7.8970523	2.910764017
bullmast	4.3555519	2.879430481
caniche	6.4006040	0.002522556
chihuahu	5.4366197	6.854956241
cocker	4.1352521	0.060190825
colley	0.1058588	2.664533548
dalmatie	3.2216221	9.438522677
doberman	5.8638360	0.958117236
dogueall	8.4304649	2.474089271
epagneub	1.7574399	10.350777943
epagneuf	0.1614884	2.560978462
foxhound	5.9090133	0.006132611
foxterri	5.9773564	0.185906708
gbdeasco	2.0582234	0.123802149
labrador	3.2216221	9.438522677
levrier	3.5214957	0.066583845
mastiff	4.3944998	7.584703763
pekinois	5.4366197	6.854956241
pointer	3.4866395	1.729709149
saintber	2.6172536	3.392717496
setter	1.9545479	1.369226351
teckel	7.8970523	2.910764017
terreneu	1.1310567	2.266782110

Les contributions

Si on regarde les contributions des individus, on peut voir que "dogueall", "bulldog" et "teckel" contribuent le plus à l'axe 1. Ainsi, on peut voir que "dogueall" (situé à gauche) est opposé aux chiens "bulldog" et "teckel" (situés à droite). Cela confirme ce qui a été dit au dessus car le Dogue Allemand, qui a une allure très imposante, est l'une des races de chiens les plus grandes, contrairement aux bulldog et aux teckels qui sont plutôt petits et légers.

Sur l'axe 2, les individus qui contribuent le plus sont "basset" et "epagneub". On peut voir qu'ils sont opposés sur l'axe, l'individu "basset" se situe en haut face à l'individu "epagneub" qui est lui situé en bas. Ce qui est en adéquation avec notre analyse, l'individu "basset" est donc plus lent (vélocité lente) alors que l'individu "epagneub" est plus rapide (vélocité rapide).
En effet, l'epagneul breton est un chien de chasse connu pour être rapide alors que basset est aussi un chien de chasse mais connu lui pour être plus lent dû à son corp long et ses pattes courtes.

	Dim 1	Dim 2
TAI_1	0.49144201	0.2987546600
TAI_2	0.16462520	0.3448030588
TAI_3	0.87503205	0.0005293413
POI_1	0.57531341	0.2861238116
POI_2	0.10044717	0.7221387844
POI_3	0.23420393	0.2155641859
VEL_1	0.06021292	0.6422447857
VEL_2	0.15344741	0.3318791146
VEL_3	0.39792110	0.0691296921
INT_1	0.05129787	0.2752677726
INT_2	0.12673870	0.0756897524
INT_3	0.03207684	0.0603213262
AFF_1	0.64765585	0.0767360421
AFF_2	0.64765585	0.0767360421
AGR_1	0.17292377	0.0406368567
AGR_2	0.17292377	0.0406368567

Les cosinus carrés

Pour savoir si les individus et les variables sont bien projetés sur les axe 1 et 2 on regarde leurs cosinus carré.
Les cosinus carrés sont des mesures de la qualité de la représentation des variables supplémentaires sur les axes. Un cosinus carré élevé indique une bonne représentation de la variable sur l'axe correspondant, tandis qu'un cosinus carré bas indique une mauvaise représentation.
Les variables les mieux représentées sont TAI_3 avec une somme des cosinus carré d'environ 0,88 et POI_1 avec une somme des cosinus carré d'environ 0,86.
Les variables INT_1, INT_2, INT_3, AGR_1, AGR_2, POI_3, VEL_2, VEL_3 se font écraser par la projection car la somme de leur cosinus carré est inférieure à 0,5.
Cependant ce résultat est à nuancer, nous pouvons penser que POI_3, VEL_2, VEL_3 ont une représentation fictive car leur somme des cosinus carré est très proche de 0,5.

Nous pouvons aussi remarquer que les variables de l'axe 1 (POI_1 et TAI_3) et les variables de l'axe 2 (VEL_1 et POI_2) sont bien représentées donc ces axes sont bien représentés.

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
FON_1	0.63542896	0.005344696	0.002711706	0.004655675	7.303724e-05
FON_2	0.05196785	0.093498803	0.060812866	0.016917930	9.809804e-02
FON_3	0.36926611	0.056814951	0.095887149	0.042617046	1.104784e-01

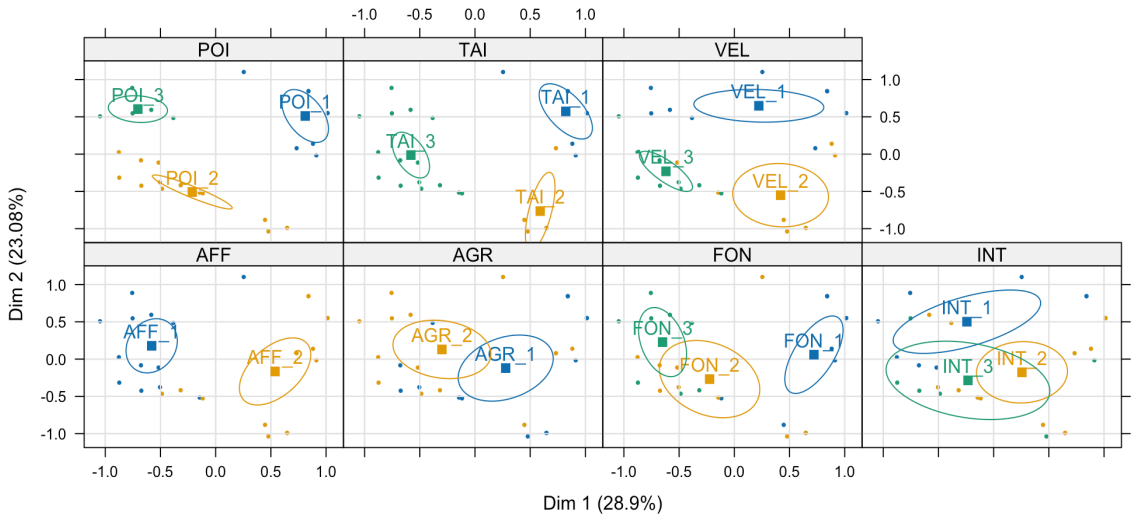
Les cosinus carrés

Penchons nous maintenant sur la variable supplémentaire “FON”. On remarque qu’une seule variable est bien représentée soit “FON_1” avec une somme des cosinus carré d’environ 0,64. Alors que “FON_2” et “FON_3” sont mal représentés avec une somme des cosinus carré respectives de 0,14 et 0,42.
Nous pouvons en déduire que “FON_2” et “FON_3” n'apportent pas beaucoup d'informations à l’analyse, elles n’ont donc pas d’influence significative sur la structure des données contrairement à “FON_1”.

	Dim 1	Dim 2			
beaucero	0.08863547	0.1536995944	epagneuf	0.01753323	0.2221256292
basset	0.03380431	0.6348671357	foxhound	0.55831304	0.0004628928
bergeral	0.15372250	0.1401636585	foxterri	0.43627101	0.0108396312
boxer	0.11133075	0.4325235284	gbdeasco	0.18602321	0.0089387139
buldog	0.62448464	0.1838806326	labrador	0.23628517	0.5530165596
bullmast	0.27069077	0.1429582059	levrier	0.33881559	0.0051177295
caniche	0.38519392	0.0001212751	mastiff	0.29999507	0.4136333336
chihuahu	0.37993129	0.3826952203	pekinois	0.37993129	0.3826952203
cocker	0.27915682	0.0032460020	pointer	0.29459212	0.1167506763
colley	0.01239617	0.2492609870	saintber	0.20156282	0.2087298540
dalmatie	0.23628517	0.5530165596	setter	0.22389437	0.1252980645
doberman	0.48761169	0.0636477694	teckel	0.62448464	0.1838806326
dogueall	0.56079391	0.1314738467	terreneu	0.08840069	0.1415315741
epagneub	0.10498339	0.4939526916			

Les cosinus carrés

Les individus les mieux représentés sont teckel avec une somme des cosinus carré d’environ 0,808 et bulldog avec une somme des cosinus carré d’environ 0,806.
On avait aussi remarqué que ces deux individus contribuaient le plus sur l’axe 1 donc ils étaient bien plutôt petits et légers.
Parmi les individus les moins bien représentés il y a les individus golden retriever avec une somme des cosinus carré d’environ 0,19 et terre-neuve avec une somme des cosinus carré d’environ 0,23. Ces individus se retrouvent au centre écrasé par la projection.



Pour visualiser la dispersion des individus et identifier des groupes similaires, nous pouvons regarder les ellipses de confiance autour des points représentant les individus dans l'espace des premières dimensions de l'ACM. Cela permet de voir si les modalités sont vraiment différentes les unes des autres ou si elles se sont fait écrasé par le graphique. Les ellipses entourent les groupes d'individus qui ont des modalités similaires. Plus une ellipse est grande, plus la dispersion des individus est importante dans cette direction.

Tout d'abord, concernant le poids, un individu lourd, ne sera pas moyen ou léger car les ellipses ne se touchent pas. Le raisonnement reste le même pour la taille, la vélocité et l'affectuosité. En effet, les ellipses ne se touchent pas donc un individu grand, ne sera pas moyen ou petit, un individu rapide ne sera pas moyen ou lent et un individu avec une grande affectuosité ne sera pas un individu avec une faible affectuosité.

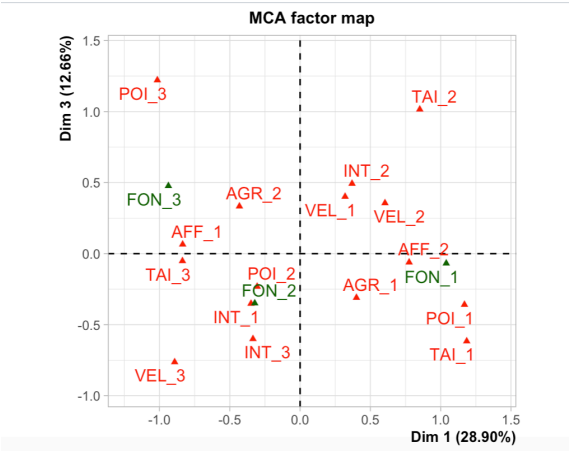
Concernant l'agressivité, on voit que les ellipses se touchent donc il y a une confusion entre les individus non-agressifs et ceux qui sont agressifs. Cela montre que cette variable s'est fait écrasé par la projection (comme nous l'avions montré plus haut avec les cosinus carré).
Pour la variable supplémentaire fonction, il y'a une confusion entre les modalités FON_2 et FON_3 mais pas entre FON_1 et FON_2 ou entre FON_1 et FON_2. Chaque modalité de la variable supplémentaire est considérée comme une variable distincte dans l'analyse et sont donc projetées de manière indépendante sur les axes de l'ACM. Ainsi, comme nous l'avions montré

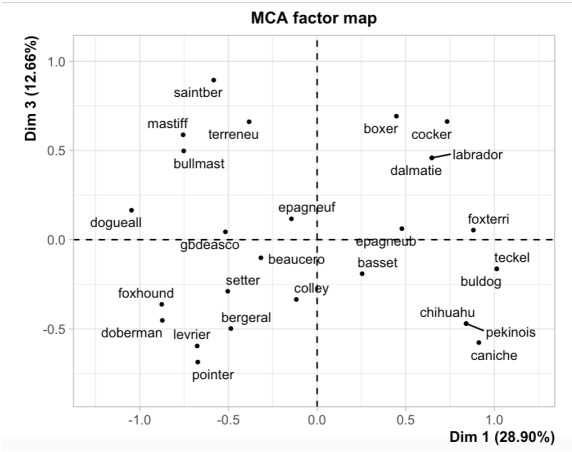
au-dessus avec la somme des cosinus carré, les variables FON_2 et FON_3 sont mal représentées sur les axes de l'analyse des correspondances multiples. Cela peut signifier que ces variables supplémentaires ne sont donc pas fortement corrélées avec les variables actives incluses dans l'ACM et qu'elles n'apportent pas beaucoup d'information à l'analyse (en termes de leur projection sur les axes de l'ACM).

Enfin, concernant l'intelligence, nous pouvons voir que les modalités se font écraser par le graphique. Les ellipses se touchent donc il y a une confusion entre les 3. Les variables se sont faites écraser par la projection.

Globalement, l'interprétation du premier plan factoriel est la suivante.
Premièrement, lorsqu'on regarde le nuage des modalités, on observe que la variable FON_1 se trouve à droite du graphique. Cette variable supplémentaire est bien représentée sur les axes de l'analyse des correspondances multiples. Ainsi, à droite on retrouve les chiens petits de taille, légers et lents, ce qui nous confirme le fait que ce sont des chiens de compagnies. On retrouve donc parmi les individus qui sont bien représentés : les bulldogs, les chihuahuas, les teckels et les pékinois comme chiens de compagnie.

On peut penser que les chiens de chasse se trouvent tout en bas du graphique et qu'ils regroupent des caractéristiques telles qu'une vélocité moyenne, une taille moyenne et un poids moyen. Parmi les individus bien représentés, on retrouve les épagneuls bretons, les labradors et les bassets. Enfin, on peut également penser que les chiens de gardes se situent à gauche du graphique et regroupent des caractéristiques tels qu'une grande taille et une faible affectuosité. Parmi les individus bien représentés, on retrouve les dogues allemands, les mastiffs et les dobermans.
Cependant, il est difficile d'identifier précisément où se trouvent les deux autres types de chiens. Les variables supplémentaires FON_2 et FON_3 sont mal représentées sur les axes de l'analyse des correspondances multiples. Cela peut indiquer qu'elles n'ont pas d'impact significatif sur la structure de nos données. De plus, un nombre élevé d'individus de classe 2 et 3 se sont fait écraser par la projection et se retrouvent donc au centre du graphique (ce qui ne permet pas de les interpréter).





Penchons nous maintenant sur la représentation du plan factoriel avec la dimension 1 et 3.
Les plus grosses contributions sur l'axe 3 sont représentées par la variable "POI_3" et "VEL_3".
Or, ce sont deux variables qui ont été écrasées par la projection sur les axes 1 et 2 donc elles étaient mal représentées.

	Dim 1	Dim 2	Dim 3
beaucero	0.08863547	0.1536995944	0.009069781
basset	0.03380431	0.6348671357	0.019038597
bergeral	0.15372250	0.1401636585	0.161231704
boxer	0.11133075	0.4325235284	0.266393303
buldog	0.62448464	0.1838806326	0.016241584
bullmast	0.27069077	0.1429582059	0.118328181
caniche	0.38519392	0.0001212751	0.153853124
chihuahu	0.37993129	0.3826952203	0.118691149
cocker	0.27915682	0.0032460020	0.227671518
colley	0.01239617	0.2492609870	0.100999475
dalmatie	0.23628517	0.5530165596	0.118619154
doberman	0.48761169	0.0636477694	0.130832617
dogueall	0.56079391	0.1314738467	0.013933070
epagneub	0.10498339	0.4939526916	0.001761558
epagneuf	0.01753323	0.2221256292	0.011454493
foxhound	0.55831304	0.0004628928	0.095309358
foxterri	0.43627101	0.0108396312	0.001607917
gbdeasco	0.18602321	0.0089387139	0.001347381
labrador	0.23628517	0.5530165596	0.118619154
levrier	0.33881559	0.0051177295	0.262473933
mastiff	0.29999507	0.4136333336	0.181335511
pekinois	0.37993129	0.3826952203	0.118691149
pointer	0.29459212	0.1167506763	0.305546227
saintber	0.20156282	0.2087298540	0.473604996
setter	0.22389437	0.1252980645	0.073613569
teckel	0.62448464	0.1838806326	0.016241584
terreneu	0.08840069	0.1415315741	0.262465947

	Dim 1	Dim 2	Dim 3
TAI_1	0.49144201	0.2987546600	0.132809435
TAI_2	0.16462520	0.3448030588	0.234627550
TAI_3	0.87503205	0.0005293413	0.003279033
POI_1	0.57531341	0.2861238116	0.054196308
POI_2	0.10044717	0.7221387844	0.057601141
POI_3	0.23420393	0.2155641859	0.339157541
VEL_1	0.06021292	0.6422447857	0.094932948
VEL_2	0.15344741	0.3318791146	0.053456249
VEL_3	0.39792110	0.0691296921	0.291151283
INT_1	0.05129787	0.2752677726	0.052025334
INT_2	0.12673870	0.0756897524	0.225873819
INT_3	0.03207684	0.0603213262	0.102831012
AFF_1	0.64765585	0.0767360421	0.003980589
AFF_2	0.64765585	0.0767360421	0.003980589
AGR_1	0.17292377	0.0406368567	0.103307716
AGR_2	0.17292377	0.0406368567	0.103307716

Les cosinus carrés

Lorsqu'on regarde les cosinus carrés de ces deux variables, on peut voir que la somme des cosinus carré dépasse 0.5, ce qui indique qu'elles sont bien représentées sur les axes 1 et 3.

	Dim 1	Dim 2	Dim 3
beaucero	0.7737679	1.679591265	0.18076074
basset	0.4965777	11.674160199	0.63849009
bergeral	1.8193797	2.076582082	4.35652825
boxer	1.5391043	7.484976084	8.40776174
buldog	7.8970523	2.910764017	0.46889522
bullmast	4.3555519	2.879430481	4.34672445
caniche	6.4006040	0.002522556	5.83649387
chihuahu	5.4366197	6.854956241	3.87745169
cocker	4.1352521	0.060190825	7.69957277
colley	0.1058588	2.664533548	1.96907570
dalmatie	3.2216221	9.438522677	3.69230120
doberman	5.8638360	0.958117236	3.59193035
dogueall	8.4304649	2.474089271	0.47818860
epagneub	1.7574399	10.350777943	0.06732262
epagneuf	0.1614884	2.560978462	0.24085714
foxhound	5.9090133	0.006132611	2.30290971
foxterri	5.9773564	0.185906708	0.05029451
gbdeasco	2.0582234	0.123802149	0.03403454
labrador	3.2216221	9.438522677	3.69230120
levrier	3.5214957	0.066583845	6.22807925
mastiff	4.3944998	7.584703763	6.06432357
pekinois	5.4366197	6.854956241	3.87745169
pointer	3.4866395	1.729709149	8.25594928
saintber	2.6172536	3.392717496	14.03963501
setter	1.9545479	1.369226351	1.46711842
teckel	7.8970523	2.910764017	0.46889522
terreneu	1.1310567	2.266782110	7.66665317

	Dim 1	Dim 2	Dim 3
TAI_1	12.5978150	9.58661729	7.7724161
TAI_2	4.6420727	12.17067028	15.1042375
TAI_3	13.4585463	0.01019149	0.1151394
POI_1	14.0104164	8.72224556	3.0131477
POI_2	1.6736860	15.06207234	2.1911472
POI_3	6.6040417	7.60886705	21.8333953
VEL_1	1.3119931	17.51742290	4.7224026
VEL_2	3.7368537	10.11705778	2.9720027
VEL_3	9.1804174	1.99644722	15.3351608
INT_1	1.2492400	8.39130827	2.8924483
INT_2	2.2742083	1.70014447	9.2531808
INT_3	0.8633836	2.03240794	6.3188861
AFF_1	11.6215827	1.72364624	0.1630694
AFF_2	10.7914697	1.60052866	0.1514216
AGR_1	2.8813167	0.84758676	3.9298252
AGR_2	3.1029565	0.91278575	4.2321194

Les contributions

Les individus qui contribuent le plus sur l'axe 3 sont saint bernard, boxer et pointer avec une contribution respective de 14%, 8,4% et 8,2%.
On remarque que "saintber" et "pointer" sont bien représentés avec une somme des cosinus carré de 0,68 et 0,6 alors qu'ils étaient mal représentés donc écrasés par la projection sur le premier plan factoriel.

Concernant les variables supplémentaires, seule FON_1 est bien représentée avec une somme des cosinus carré de 0,63. La somme des cosinus carré s'est amélioré pour FON_3 mais elle reste mal représentée par la projection.

L'interprétation de l'axe 1 reste la même, cet axe oppose les individus légers, petits et affectueux des individus plus lourds, plus grands et moins affectueux. En effet, les variables qui contribuent le plus sur cet axe soit "POI_1" et "TAI_3" restent bien représentées sur ce plan factoriel avec une somme des cosinus carré de 0,62 et 0,87.
À droite, nous retrouvons bien des petits chiens légers tels que les caniches ou les teckels par exemple.

Les variables qui contribuent le plus sur l'axe 3 sont "POI_3" et "VEL_3" avec une contribution respective de 21,8% et 15,3%. Ces deux variables sont bien représentées, avec une somme des cosinus carré de 0,57 pour "POI_3" et 0,69 pour "VEL_3".
Elles sont de même opposé graphiquement avec les individus lourds en haut et les individus rapides en bas.
Concernant les individus, le saint Bernard contribue le plus à l'axe 3, suivi par le boxer et le pointer.
Ainsi, tout en haut à gauche on retrouve le saint Bernard qui est un chien assez grand de taille donc lourd, suivi par le boxer qui également grand et lourd. Tout en bas à gauche du graphique, on retrouve le pointer qui est un chien très athlétique et réputé pour sa rapidité.

Globalement, si on essaye d'interpréter le graphique sur l'axe 1 et 3, on peut voir que les chiens situés le plus à droite ont des caractéristiques typiques des chiens de compagnies (petits, légers et affectueux) tels que les caniches ou les teckels par exemple.
On constate qu'un grand nombre de chiens de type 2 (chasse) et 3 (garde) se font écraser par la projection donc il est difficile de bien les interpréter. On pourrait penser que les chiens de gardes se trouvent en haut à gauche et les chiens de chasse en bas à gauche, mais la séparation entre ces deux groupes reste difficile. Contrairement aux chiens de compagnies, les chiens situés à gauche sont moins affectueux et plus grands de taille. Ainsi, les chiens situés en haut seront plus lourds donc correspondent bien à des chiens de garde et ceux situés en bas seront plus rapides donc correspondent bien à des chiens de chasse.
Cependant, il ne faut pas oublier que le taux d'inertie expliquée n'est pas très élevé (environ 42%) donc l'interprétation reste fragile.