National Poll on Healthy Aging (NPHA)

Auteurs du Projet

Anastasios Tsiompanidis Noah Kohrs

1- Motivation et Positionnement du Projet

L'objectif de cette analyse est de développer et de valider des algorithmes de machine learning capables de prédire le nombre de médecins consultés par une personne âgée au cours d'une année, en se basant sur un sous-ensemble filtré des données du National Poll on Healthy Aging (NPHA). Ce projet est crucial pour comprendre les besoins spécifiques des seniors en matière de santé et pour fournir des informations utiles aux prestataires de soins, aux décideurs politiques et aux défenseurs des intérêts des personnes âgées. Cette étude contribue à une meilleure gestion des ressources médicales et à l'amélioration des politiques publiques axées sur le vieillissement.

2- Analyse descriptive

```
npha <- read.csv("NPHA-doctor-visits.csv")
summary(npha)
```

```
##  Number.of.Doctors.Visited      Age    Physical.Health  Mental.Health
##  Min.   :1.000             Min.   :2   Min.   :-1.000   Min.   :-1.000
##  1st Qu.:2.000             1st Qu.:2   1st Qu.: 2.000   1st Qu.: 1.000
##  Median :2.000             Median :2   Median : 3.000   Median : 2.000
##  Mean   :2.112             Mean   :2   Mean   : 2.794   Mean   : 1.989
##  3rd Qu.:3.000             3rd Qu.:2   3rd Qu.: 3.000   3rd Qu.: 3.000
##  Max.   :3.000             Max.   :2   Max.   : 5.000   Max.   : 5.000
##  Dental.Health     Employment    Stress.Keeps.Patient.from.Sleeping
##  Min.   :-1.00   Min.   :1.000   Min.   :0.0000
##  1st Qu.: 2.00   1st Qu.:3.000   1st Qu.:0.0000
##  Median : 3.00   Median :3.000   Median :0.0000
##  Mean   : 3.01   Mean   :2.807   Mean   :0.2479
##  3rd Qu.: 4.00   3rd Qu.:3.000   3rd Qu.:0.0000
##  Max.   : 6.00   Max.   :4.000   Max.   :1.0000
##  Medication.Keeps.Patient.from.Sleeping Pain.Keeps.Patient.from.Sleeping
##  Min.   :0.00000                        Min.   :0.0000
##  1st Qu.:0.00000                        1st Qu.:0.0000
##  Median :0.00000                        Median :0.0000
##  Mean   :0.05602                        Mean   :0.2185
##  3rd Qu.:0.00000                        3rd Qu.:0.0000
##  Max.   :1.00000                        Max.   :1.0000
##  Bathroom.Needs.Keeps.Patient.from.Sleeping Unknown.Keeps.Patient.from.Sleeping
##  Min.   :0.0000                             Min.   :0.0000
##  1st Qu.:0.0000                             1st Qu.:0.0000
##  Median :1.0000                             Median :0.0000
##  Mean   :0.5042                             Mean   :0.4174
##  3rd Qu.:1.0000                             3rd Qu.:1.0000
##  Max.   :1.0000                             Max.   :1.0000
##  Trouble.Sleeping Prescription.Sleep.Medication      Race           Gender
```

```
##  Min.   :-1.000   Min.   :-1.000                Min.   :1.000   Min.   :1.00
##  1st Qu.: 2.000   1st Qu.: 3.000                1st Qu.:1.000   1st Qu.:1.00
##  Median : 3.000   Median : 3.000                Median :1.000   Median :2.00
##  Mean   : 2.408   Mean   : 2.829                Mean   :1.426   Mean   :1.55
##  3rd Qu.: 3.000   3rd Qu.: 3.000                3rd Qu.:1.000   3rd Qu.:2.00
##  Max.   : 3.000   Max.   : 3.000                Max.   :5.000   Max.   :2.00
```

Comme nos valeurs sont catégorielles représentées par des chiffres, on va les remplacer par des labels pour une meilleure compréhension.

On va d'abord définir les labels pour chaque variable catégorielle.

Note: Le dataset utilisé contient plusieurs erreurs de labellisation, ce qui nous oblige à les corriger.

```
doctor_labels <- c("0-1", "2-3", "4 or more")
age_labels <- c("50-64", "65-80")

# On a ajouté la valeur "Very Poor" nous mêmes car il n'y avait
#pas de labelling indiqué pour la valeur 6.
# Cela suit la logique et nous évite la présence de NA's
health_labels <- c("Refused", "Excellent", "Very Good", "Good", "Fair", "Poor", "Very Poor")
empl_labels <- c("Refused", "Full-time", "Part-time", "Retired", "Not working")
yes_no_labels <- c("No", "Yes")
gender_labels <- c("M", "F")
medication_labels <- c("Refused", "Use regularly", "Use occasionally", "Do not use")

# Les valeurs devraient être "No" et "Yes", mais elles sont mal labellisées dans le dataset.
# Nous supposons que ces corrections sont appropriées.
sleep_labels <- c("Refused", "No", "A bit", "Yes")
race_labels <- c("Not asked", "Refused", "White", "Black", "Other", "Hispanic", "2+ Races")
```

```
colnames(npha)
```

```
##  [1] "Number.of.Doctors.Visited"
##  [2] "Age"
##  [3] "Physical.Health"
##  [4] "Mental.Health"
##  [5] "Dental.Health"
##  [6] "Employment"
##  [7] "Stress.Keeps.Patient.from.Sleeping"
##  [8] "Medication.Keeps.Patient.from.Sleeping"
##  [9] "Pain.Keeps.Patient.from.Sleeping"
## [10] "Bathroom.Needs.Keeps.Patient.from.Sleeping"
## [11] "Unknown.Keeps.Patient.from.Sleeping"
## [12] "Trouble.Sleeping"
## [13] "Prescription.Sleep.Medication"
## [14] "Race"
## [15] "Gender"
```

```
npha$Number.of.Doctors.Visited = factor(npha$Number.of.Doctors.Visited, levels = 1:3, labels = doctor_la
npha$Age = factor(npha$Age, levels = 1:2, labels = age_labels, ordered = FALSE)
npha$Physical.Health = factor(npha$Physical.Health, levels = c(-1, 1:6), labels = health_labels, ordered
npha$Mental.Health = factor(npha$Mental.Health, levels = c(-1, 1:6), labels = health_labels, ordered = T
```

```r
npha$Dental.Health = factor(npha$Dental.Health, levels = c(-1, 1:6), labels = health_labels, ordered = T
npha$Employment = factor(npha$Employment, levels = c(-1, 1:4), labels = empl_labels, ordered = FALSE)
npha$Stress.Keeps.Patient.from.Sleeping = factor(npha$Stress.Keeps.Patient.from.Sleeping, levels = 0:1,
npha$Medication.Keeps.Patient.from.Sleeping = factor(npha$Medication.Keeps.Patient.from.Sleeping, levels
npha$Pain.Keeps.Patient.from.Sleeping = factor(npha$Pain.Keeps.Patient.from.Sleeping, levels = 0:1, labe
npha$Bathroom.Needs.Keeps.Patient.from.Sleeping = factor(npha$Bathroom.Needs.Keeps.Patient.from.Sleeping
npha$Unknown.Keeps.Patient.from.Sleeping = factor(npha$Unknown.Keeps.Patient.from.Sleeping, levels = 0:1
npha$Trouble.Sleeping = factor(npha$Trouble.Sleeping, levels = c(-1, 1:3), labels = sleep_labels, ordere
npha$Prescription.Sleep.Medication = factor(npha$Prescription.Sleep.Medication, levels = c(-1, 1:3), lab
npha$Race = factor(npha$Race, levels = 0:6, labels = race_labels, ordered = FALSE)
npha$Gender = factor(npha$Gender, levels = 1:2, labels = gender_labels, ordered = FALSE)
```

On obtient:

```r
summary(npha)
```

```
##  Number.of.Doctors.Visited    Age       Physical.Health   Mental.Health
##  0-1      :131             50-64:  0   Refused  :  1    Refused  : 10
##  2-3      :372             65-80:714   Excellent: 36    Excellent:219
##  4 or more:211                         Very Good:239    Very Good:282
##                                        Good     :291    Good     :167
##                                        Fair     :126    Fair     : 34
##                                        Poor     : 21    Poor     :  2
##                                        Very Poor:  0    Very Poor:  0
##    Dental.Health       Employment  Stress.Keeps.Patient.from.Sleeping
##  Refused  :  4   Refused    :  0   No :537
##  Excellent: 66   Full-time  : 50   Yes:177
##  Very Good:215   Part-time  : 55
##  Good     :208   Retired    :592
##  Fair     :127   Not working: 17
##  Poor     : 39
##  Very Poor: 55
##  Medication.Keeps.Patient.from.Sleeping Pain.Keeps.Patient.from.Sleeping
##  No :674                                No :558
##  Yes: 40                                Yes:156
##
##
##
##
##
##  Bathroom.Needs.Keeps.Patient.from.Sleeping Unknown.Keeps.Patient.from.Sleeping
##  No :354                                    No :416
##  Yes:360                                    Yes:298
##
##
##
##
##
##  Trouble.Sleeping  Prescription.Sleep.Medication        Race     Gender
##  Refused:  2       Refused         :  3           Not asked:  0   M:321
##  No     : 62       Use regularly   : 38           Refused  :578   F:393
##  A bit  :291       Use occasionally: 34           White    : 52
##  Yes    :359       Do not use      :639           Black    : 20
```

```
##                                                 Other    : 44
##                                                 Hispanic : 20
##                                                 2+ Races :  0
```

On observe que l'age des patients est toujours entre 65 et 80 ans, il s'agit donc d'une variable constante sur notre jeu de données. Nous allons donc l'écarter de la suite de l'analyse car cela ne nous fournit aucune information utile.

On fait alors:

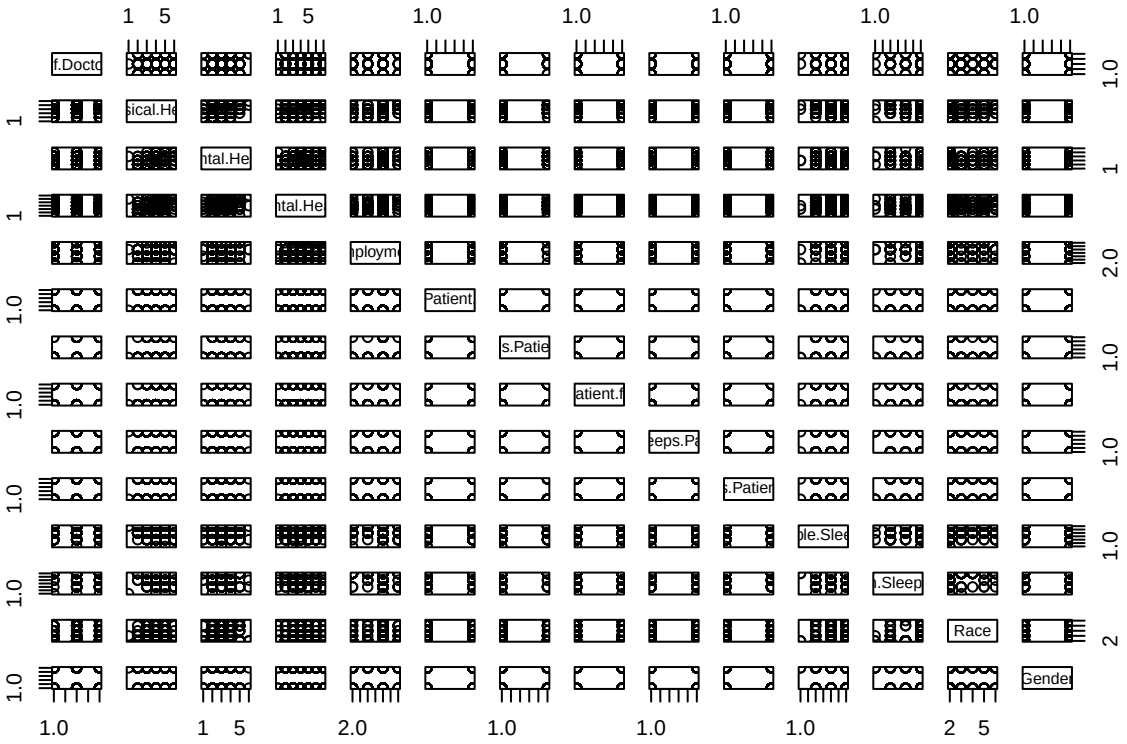
```
npha <- npha[, c(1, 3:ncol(npha))]
# On vérifie que tout s'est bien passé
colnames(npha)
```

```
##  [1] "Number.of.Doctors.Visited"
##  [2] "Physical.Health"
##  [3] "Mental.Health"
##  [4] "Dental.Health"
##  [5] "Employment"
##  [6] "Stress.Keeps.Patient.from.Sleeping"
##  [7] "Medication.Keeps.Patient.from.Sleeping"
##  [8] "Pain.Keeps.Patient.from.Sleeping"
##  [9] "Bathroom.Needs.Keeps.Patient.from.Sleeping"
## [10] "Unknown.Keeps.Patient.from.Sleeping"
## [11] "Trouble.Sleeping"
## [12] "Prescription.Sleep.Medication"
## [13] "Race"
## [14] "Gender"
```

On observe également que la variable Dental.Health contient des valeurs inconnues qui ne sont associées à aucun label. Nous allons donc les remplacer par des valeurs manquantes.

Essayons d'avoir une vue d'ensemble de nos données.

```
plot(npha)
```

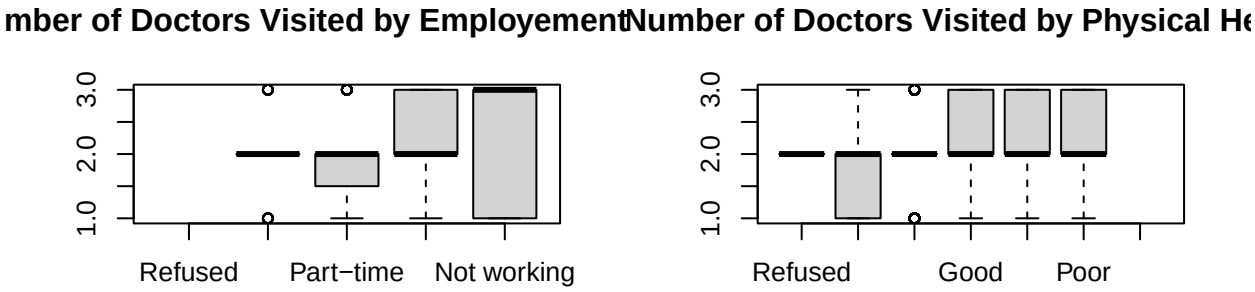


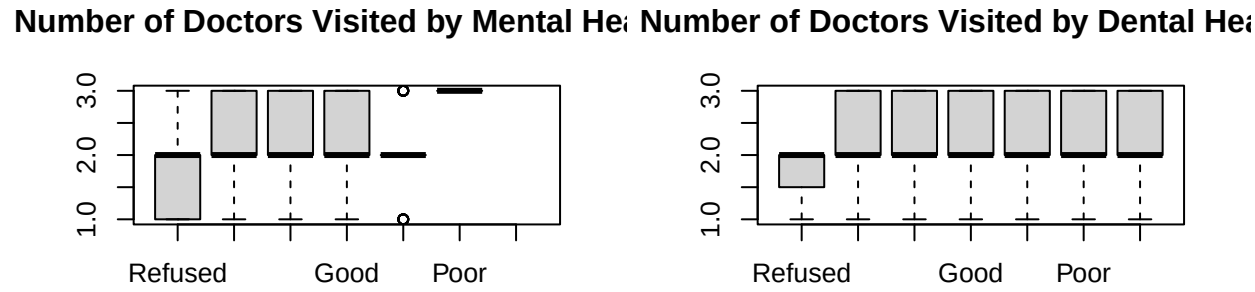
Comme attendu, c'est indigeste en vu du nombre de variables présentes dans le jeu de données.

On va donc essayer de voir les relations entre les variables et le nombre de visites chez le médecin.

```
par(mfrow = c(2, 2))
# boxplot(split(npha$Number.of.Doctors.Visited, npha$Age), main = "Number of Doctors Visited by Age Gro
# Ceci même si en théorie est intéressant, ne donne pas de résultats utiles car on a seulement 1 variab

boxplot(split(npha$Number.of.Doctors.Visited, npha$Employment), main = "Number of Doctors Visited by Emp
boxplot(split(npha$Number.of.Doctors.Visited, npha$Physical.Health), main = "Number of Doctors Visited
boxplot(split(npha$Number.of.Doctors.Visited, npha$Mental.Health), main = "Number of Doctors Visited by
boxplot(split(npha$Number.of.Doctors.Visited, npha$Dental.Health), main = "Number of Doctors Visited by
```





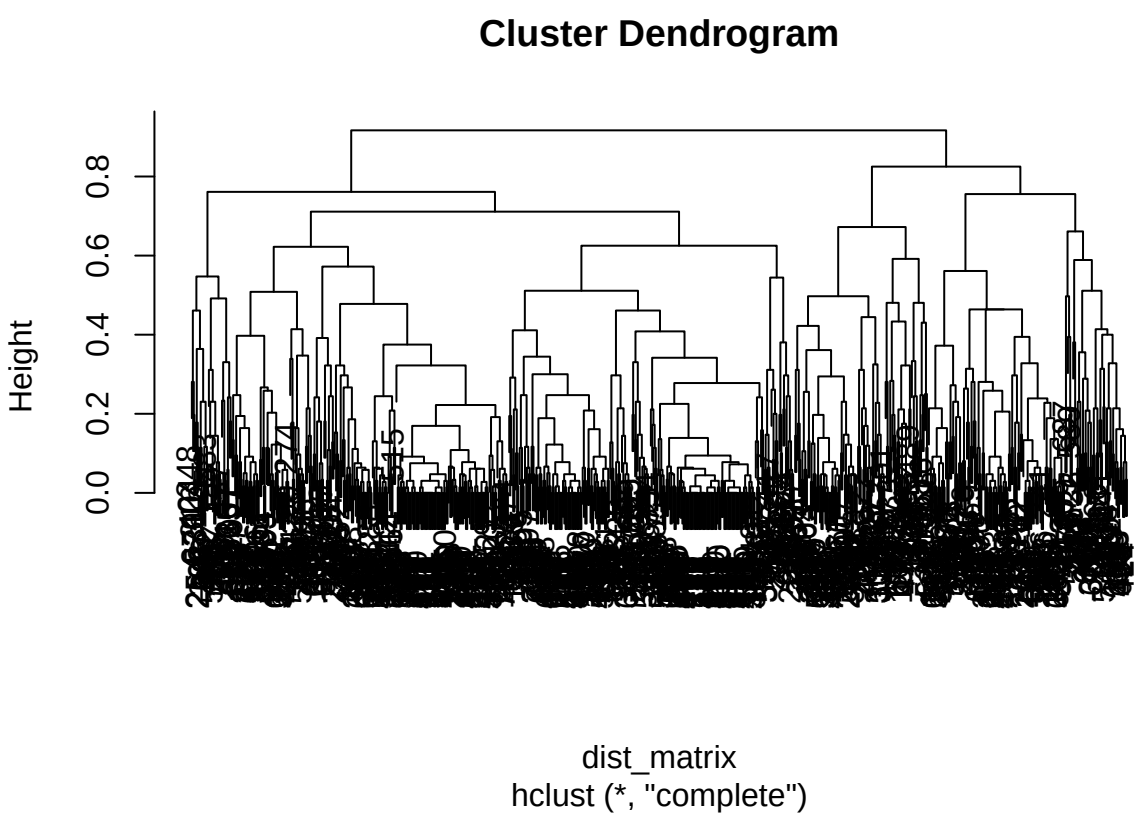
3- Classification non supervisée :

```
par(mfrow = c(1, 1))
library(cluster)
### Suppression de la variable cible (Number.of.Doctors.Visited)
npha_sans_visites <- npha[, -1]
summary(npha_sans_visites)
```

```
##   Physical.Health   Mental.Health   Dental.Health       Employment
##  Refused  :  1    Refused  : 10   Refused  :  4   Refused    :  0
##  Excellent: 36    Excellent:219   Excellent: 66   Full-time  : 50
##  Very Good:239    Very Good:282   Very Good:215   Part-time  : 55
##  Good     :291    Good     :167   Good     :208   Retired    :592
##  Fair     :126    Fair     : 34   Fair     :127   Not working: 17
##  Poor     : 21    Poor     :  2   Poor     : 39
##  Very Poor:  0    Very Poor:  0   Very Poor: 55
##  Stress.Keeps.Patient.from.Sleeping Medication.Keeps.Patient.from.Sleeping
##  No :537                            No :674
##  Yes:177                            Yes: 40
##
##
##
##
```

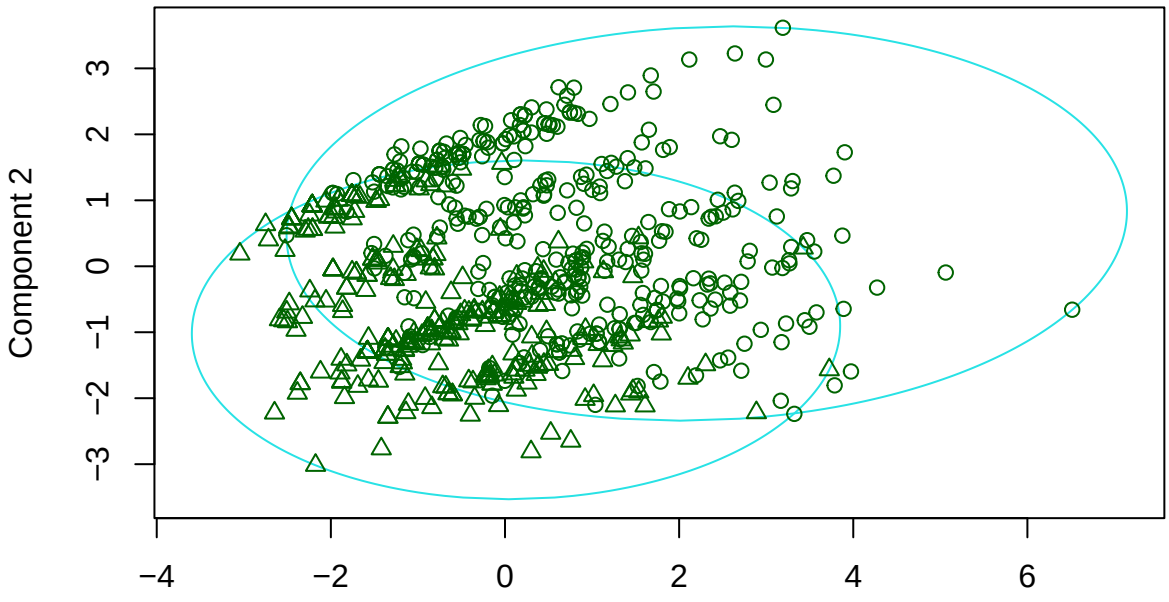
```
##
##  Pain.Keeps.Patient.from.Sleeping Bathroom.Needs.Keeps.Patient.from.Sleeping
##  No :558                          No :354
##  Yes:156                          Yes:360
##
##
##
##
##
##  Unknown.Keeps.Patient.from.Sleeping Trouble.Sleeping
##  No :416                             Refused:  2
##  Yes:298                             No     : 62
##                                      A bit  :291
##                                      Yes    :359
##
##
##
##   Prescription.Sleep.Medication        Race     Gender
##  Refused         :  3          Not asked:  0   M:321
##  Use regularly   : 38          Refused  :578   F:393
##  Use occasionally: 34          White    : 52
##  Do not use      :639          Black    : 20
##                                Other    : 44
##                                Hispanic : 20
##                                2+ Races :  0
```

```r
dist_matrix <- daisy(npha_sans_visites[, -ncol(npha_sans_visites)])
hclust_result <- hclust(dist_matrix)
plot(hclust_result)
```

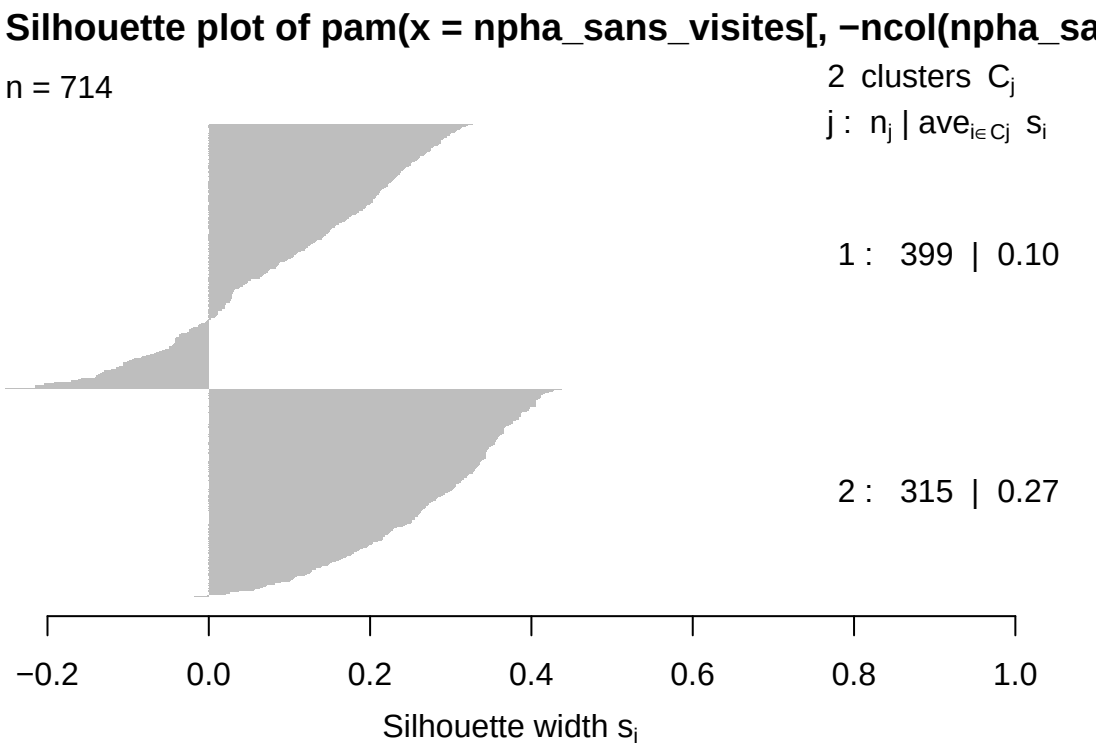



```
pam_result <- pam(npha_sans_visites[, -ncol(npha_sans_visites)], k = 2)
plot(pam_result)
```

clusplot(pam(x = npha_sans_visites[, −ncol(npha_sans_visites)], k =



These two components explain 31.54 % of the point variability.

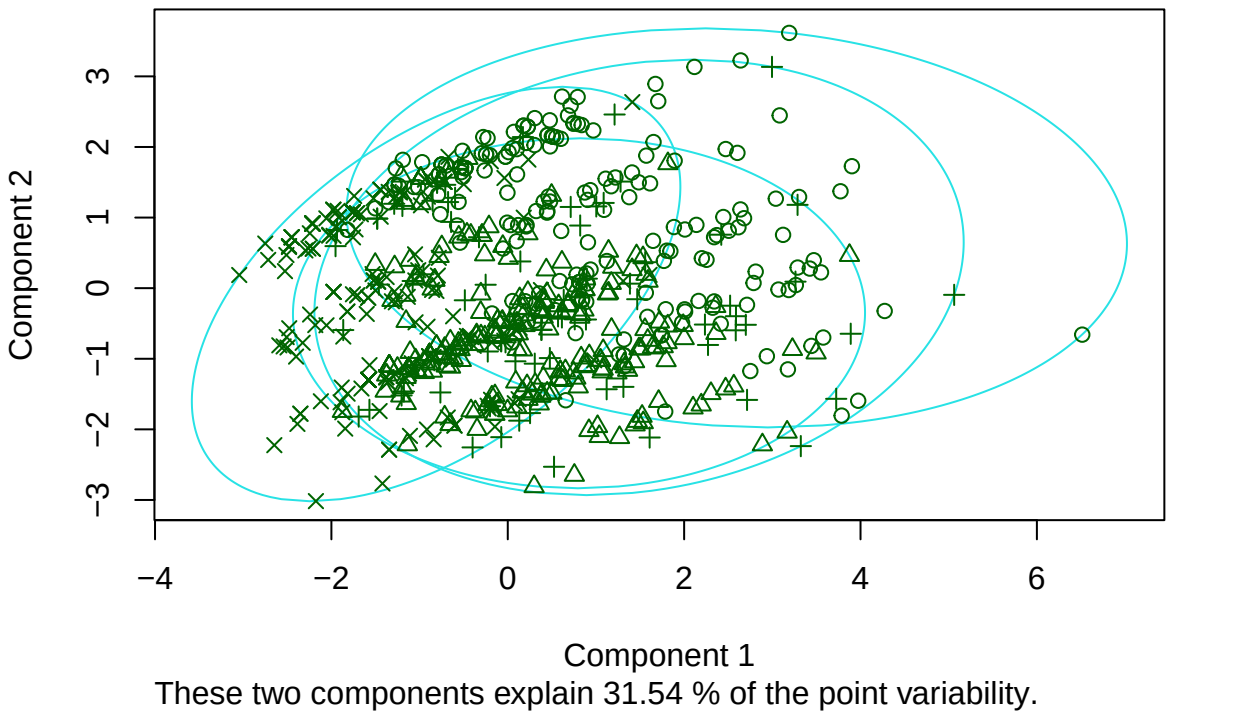


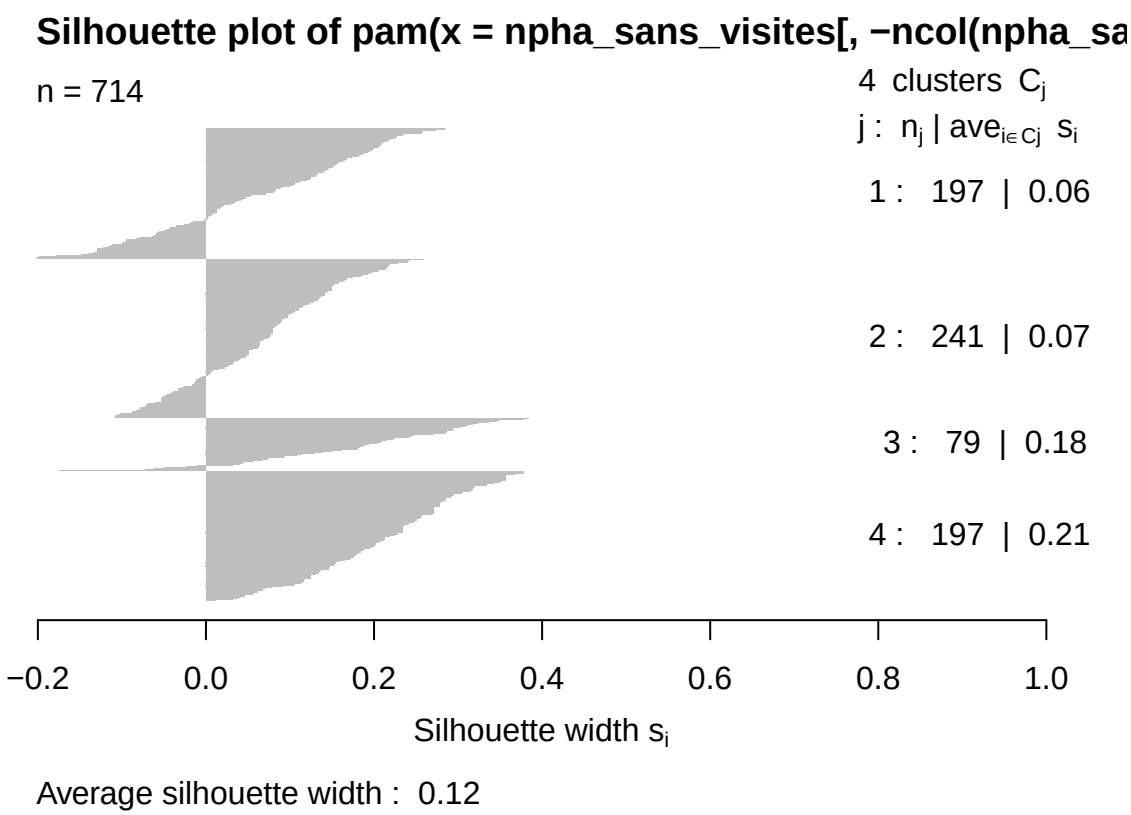
Interprétation du graphique de clustering PAM Le graphique de clustering PAM met en évidence deux groupes principaux parmi les observations. Cependant, la séparation entre ces clusters n'est pas nette, indiquant une certaine hétérogénéité au sein des groupes. L'explication de la variance à hauteur de 31,54 % suggère que les deux premières composantes principales ne capturent qu'une partie limitée des informations contenues dans les données. Cette faible variance implique que d'autres dimensions pourraient être nécessaires pour mieux différencier les groupes. De plus, la dispersion des points montre que certains individus sont proches de la frontière entre les clusters, suggérant que les variables choisies ne permettent pas de segmenter clairement la population analysée.

Interprétation du Silhouette Plot

Le Silhouette Plot révèle une cohésion interne relativement faible des clusters, avec une valeur moyenne de 0,17. Ce score indique que de nombreuses observations se situent à la limite de leur groupe, ce qui traduit une séparation imparfaite entre les clusters. En particulier, le premier cluster présente une silhouette moyenne plus basse, ce qui signifie que ses individus sont plus dispersés et donc moins homogènes. À l'inverse, le second cluster semble mieux défini, bien que sa cohésion reste modérée. Globalement, ces résultats suggèrent que le choix du nombre de clusters pourrait être optimisé ou que certaines variables devraient être réévaluées pour améliorer la qualité de la classification.

```
#Nous testons avec k = 4 car la faible valeur de silhouette moyenne obtenue avec k = 2 suggère
#que la séparation des groupes n'est pas optimale. En augmentant le nombre de clusters, nous
#espérons mieux capturer les différentes catégories. Tester k = 4 permet d'évaluer si une s
#egmentation plus fine améliore la cohérence interne des groupes et réduit le chevauchement
#entre les observations.
pam_result <- pam(npha_sans_visites[, -ncol(npha_sans_visites)], k = 4)
plot(pam_result)
```


clusplot(pam(x = npha_sans_visites[, −ncol(npha_sans_visites)], k =
Component 2
3
2
1
0
−1
−2
−3
−4
−2
0
2
4
6
Component 1
These two components explain 31.54 % of the point variability.



4- Classification supervisée :

```
library(rpart)
# Indices pour chaque catégorie
# Ici aucun class est assez rare pour faire un max d'une classe

indices_1 <- which(npha$Number.of.Doctors.Visited == "0-1")
indices_2 <- which(npha$Number.of.Doctors.Visited == "2-3")
indices_3 <- which(npha$Number.of.Doctors.Visited == "4 or more")

sample_1 <- sample(indices_1, round(50, digits = 0))
sample_2 <- sample(indices_2, round(50, digits = 0))
sample_3 <- sample(indices_3, round(50, digits = 0))

sub <- c(sample_1, sample_2, sample_3)
length(sample_1)
```

```
## [1] 50
```

```
length(indices_1)
```

```
## [1] 131
```

```r
length(sample_2)
```

```
## [1] 50
```

```r
length(sample_3)
```

```
## [1] 50
```

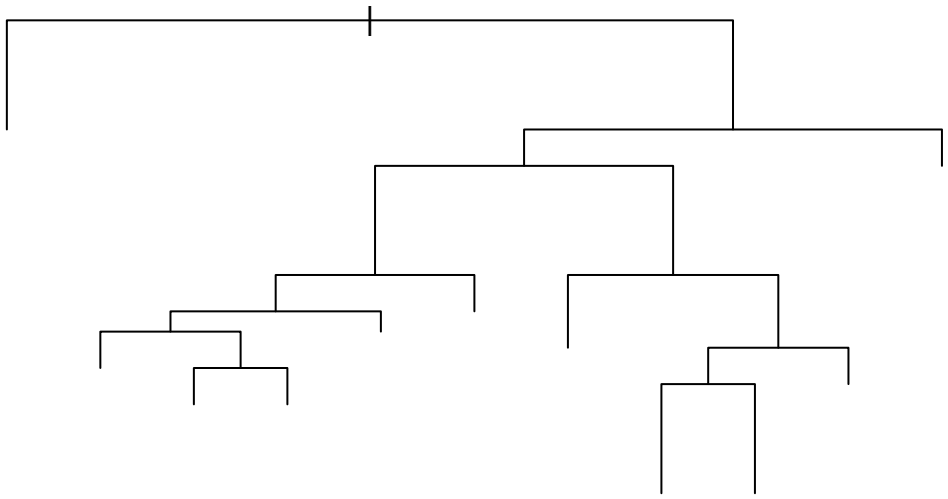
```r
all(sub %in% 1:nrow(npha))  # Doit envoyer TRUE
```

```
## [1] TRUE
```

```r
fit <- rpart(npha$Number.of.Doctors.Visited ~ ., data = npha, subset = sub)
fit
```

```
## n= 150
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##   1) root 150 100 0-1 (0.33333333 0.33333333 0.33333333)
##     2) Race=White,Hispanic 9   3 2-3 (0.33333333 0.66666667 0.00000000) *
##     3) Race=Refused,Black,Other 141  91 4 or more (0.33333333 0.31205674 0.35460993)
##       6) Prescription.Sleep.Medication=Refused,Use occasionally,Do not use 134  87 0-1 (0.35074627 0
##        12) Physical.Health=Refused,Excellent,Very Good 56  33 2-3 (0.32142857 0.41071429 0.26785714)
##          24) Employment=Part-time,Retired 49  27 2-3 (0.30612245 0.44897959 0.24489796)
##            48) Stress.Keeps.Patient.from.Sleeping=No 36  23 0-1 (0.36111111 0.36111111 0.27777778)
##              96) Dental.Health=Good,Fair,Poor,Very Poor 9   4 2-3 (0.33333333 0.55555556 0.11111111)
##              97) Dental.Health=Refused,Excellent,Very Good 27  17 0-1 (0.37037037 0.29629630 0.33333
##               194) Mental.Health=Refused,Excellent 18  10 0-1 (0.44444444 0.27777778 0.27777778) *
##               195) Mental.Health=Very Good,Good,Fair,Poor,Very Poor 9   5 4 or more (0.22222222 0.333
##            49) Stress.Keeps.Patient.from.Sleeping=Yes 13   4 2-3 (0.15384615 0.69230769 0.15384615) *
##          25) Employment=Full-time,Not working 7   4 0-1 (0.42857143 0.14285714 0.42857143) *
##        13) Physical.Health=Good,Fair,Poor,Very Poor 78  48 4 or more (0.37179487 0.24358974 0.384615
##          26) Employment=Part-time,Not working 9   3 0-1 (0.66666667 0.11111111 0.22222222) *
##          27) Employment=Full-time,Retired 69  41 4 or more (0.33333333 0.26086957 0.40579710)
##            54) Stress.Keeps.Patient.from.Sleeping=No 55  34 0-1 (0.38181818 0.27272727 0.34545455)
##             108) Dental.Health=Good,Fair,Poor,Very Poor 42  22 0-1 (0.47619048 0.23809524 0.28571429)
##             109) Dental.Health=Refused,Excellent,Very Good 13   6 4 or more (0.07692308 0.38461538 0
##            55) Stress.Keeps.Patient.from.Sleeping=Yes 14   5 4 or more (0.14285714 0.21428571 0.64285
##       7) Prescription.Sleep.Medication=Use regularly 7   2 4 or more (0.00000000 0.28571429 0.714285
```

```r
plot(fit)
```



```r
res <- table(predict(fit, npha[-sub, ], type = "class"), npha[-sub, "Number.of.Doctors.Visited"])
res
```

```
##
##             0-1 2-3 4 or more
##   0-1        42 131        63
##   2-3        22  88        36
##   4 or more  17 103        62
```

```r
err <- (1 - sum(diag(res)) / sum(res)) * 100
cat("Le taux d'erreur est de :", err, "%\n")
```

```
## Le taux d'erreur est de : 65.95745 %
```