BIG DATA MINING

PROJET FINAL

DARTMOUTH

Encadrant:
Julien JACQUES
Auteurs:
Long NGUYEN PHUOC - Master 2 SISE
Agnès BAUMER- Master 2 SISE

TABLE DES MATIERES

SECTION 1 : NETTOYAGE DES DONNÉES

Le jeu de données est complexe et volumineux. Nous avons décidé de traiter seulement les données jugées exploitables. La méthodologie globale peut se résumer en 4 étapes:

- Étape 1 : Importation des données;
- Étape 2 : Transformation des données (liste des transformations détaillées dans le tableau 1);
- Étape 3 : Jointure avec la liste des étudiants pour faire apparaitres des lignes manquantes;
- Étape 4 : Traitement les valeurs manquantes.

ÉTAPE 1 : IMPORTATION DES DONNÉES

Pour les fichiers simple, nous avons importé directement en dataframe.

```
df_class <- read.csv("~/R/Projet Big Data Mining/dataset/education-used-done/class.csv",header = FALSE)
```

Pour les dossiers de plusieurs fichiers qui traitent le même sujet, nous les avons importé tout ensemble dans un même dataframe grâce au library “vroom”.

```
library(vroom)
#Préparer le dossier des fichiers dinning
list_of_files <- list.files(path = "~/R/Projet Big Data Mining/dataset/dinning-used", recursive = TRUE,
                            pattern = "\\.txt$",
                            full.names = TRUE)
#Mettre tout le contenu dans un dataframe avec une colonne de nom de fichier
df_dinning <- vroom(list_of_files, delim = ",",col_names = FALSE, id = "FileName")
```

ÉTAPE 2 : TRANSFORMATION DES DONNÉES

Les transformations sont listées dans le tableau suivant selon leur dossier initial.

Dossier	**Variable(s)**	**Transformation**
dinning	Breakfast/Lunch/Snack/Supper	Créer 4 nouvelles variables: somme des breakfast/lunch/snack/supper pour chaque étudiant
education	class	Créer 1 nouvelle variable: somme des classes suivis par étudiant
education	deadlines	Créer 1 nouvelle variable: somme des deadlines par étudiant

education	grades: gpa.all/gpa.13s/cs.65	Pas de transformation
education	piazza: online/views/contributions/questions/notes/answers	Pas de transformation
sensing	activity	Créer 4 nouvelles variables: somme des activity0 / activity1 / activity2 / activity3 pour chaque étudiant
sensing	audio	Créer 3 nouvelles variables: somme des audio0/audio1/audio2 pour chaque étudiant
sensing	dark	Créer 1 nouvelle variable: durée totale par étudiant qui est la différence entre deux variables initiales
sensing	phonelock	Créer 1 nouvelle variable: durée totale par étudiant qui est la différence entre deux variables initiales
sensing	gps	Créer 2 variables : “moving” et “stationary” qui sont le nombre total de chaque posture détectée par le GPS par étudiant.
sensing	conversation	Créer 1 nouvelle variable: durée totale par étudiant qui est la différence entre deux variables initiales
survey	BigFive	Pas de transformation
survey	FlourishingScale	Transformer les variables quantitatives en factor
survey	LonelinessScale	Pas de transformation
survey	panas	Transformer les variables quantitatives en factor
survey	PerceivedStressScale	Pas de transformation
survey	PHQ-9	Pas de transformation
survey	psqi	Standardiser les réponses et transformer en factor pour les 4 variables : - During the past month, what time have you usually gone to bed at night? - During the past month, how long (in minutes) has it usually taken you to fall asleep each night? - When have you usually gotten up in the morning? - During the past month, how many hours of actual sleep did you get at night? (This may be different than the number of hours you spent in bed.)
survey	vr_12	Pas de transformation
EMA	tous les dossiers	Supprimer les données jugées non-exploitables: ● CancelledClasses: toutes les variables

		• Dartmounth_now: la variable “location” • Dimensions: toutes les variables • Dimensions_protestors: la variable “location” • GreenKey1: toutes les variables • GreenKey2: toutes les variables • Mood2: la variable “null” • QR_Code: toutes les variables Pour chaque étudiant, synthétiser toutes les réponses pour une même question par leur mode.

Tableau 1 - Les transformations de données

ÉTAPE 3 : JOINTURE AVEC LA LISTE DES ÉTUDIANTS

Certaines données ne sont pas exhaustives. Pour détecter les étudiants manquants dans chaque tableau, nous avons fait une jointure avec la liste complète d’étudiants.

```
library(sqldf)
grades <- sqldf("SELECT *
                FROM etudiants
                LEFT JOIN df_grades
                ON etudiants.V1 = df_grades.uid")
```

ÉTAPE 4 : TRAITEMENT DES VALEURS MANQUANTES

Nous avons utilisé 3 solutions pour traiter des valeurs manquantes. Pour des variables quantitatives, le library “missForest” semble être le meilleur. Pour des variables du dossier EMA, le library “mice” fait bien son travail. Enfin, pour le dossier « survey » qui comporte des variables catégorielles, aucun des packages marchait. Nous avons rempli les valeurs manquantes de chaque variable par leur mode

```
#Créer une fonction mode
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
#Créer une fonction qui remplit les NA
fillmode <- function(x){
  return(ifelse(x=="" | is.na(x),getmode(x),x))
}

survey_imputed <- data.frame(apply(df_survey_3,2,fillmode))
```

RÉSULTAT : 235 VARIABLES

Nous avons obtenu au final 235 variables. Pour le clustering, nous avons utilisé deux options :

- Option 1 : 26 variables quantitatives
- Option 2 : 235 variables mixtes

Pour la prédiction, nous avons constaté que seul le dossier “survey” comporte des données d’apprentissage (“pre”) et de test (“post”). Par conséquent, nous ne pouvons utiliser seulement 202 variables indépendantes pour prédire la variable “During.the.past.month..how.would.you. Rate.your.sleep.quality.overall.”. De plus, dans les données d’apprentissage, il manque les réponses des étudiants suivants : u25, u41, u54. Dans les données de test, il manque les réponses des étudiants : u8, u12, u13, u22.

SECTION 2 : CLUSTERING

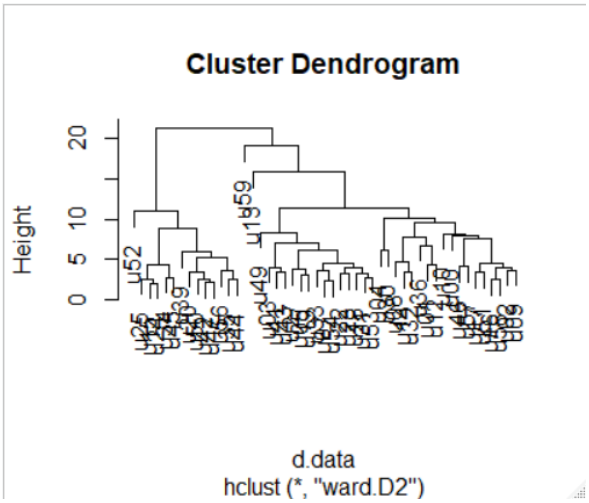
Une fois toutes les données nettoyées, on peut passer à la classification. Nos modèles se résume par le tableau suivant :

Variables	**CAH**	**K-mean**
26 quantitatives	Modèle 1	Modèle 2
26 quantitatives avec ACP	Modèle 3	Modèle 4
235 variables mixtes avec AFDM (ade4)	Modèle 5	Modèle 6
235 variables mixtes avec AFDM (FactomineR)	Modèle 7	Modèle 8

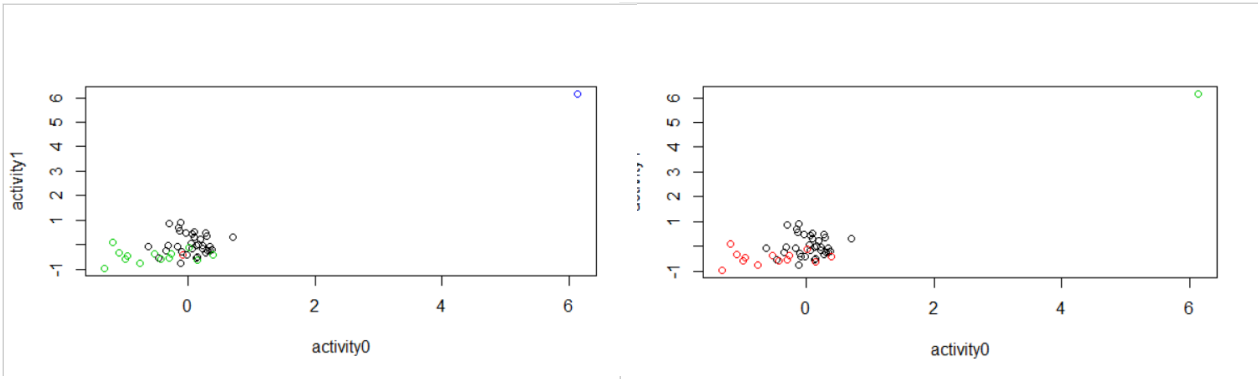
Tableau 2 - Les modèles de clustering

OPTION 1 : AVEC 26 VARIABLES QUANTITATIVES

MODELE 1 : CAH sans ACP

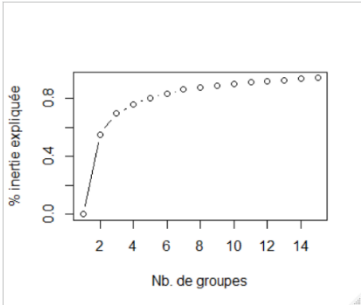


Selon le dendrogramme, nous décidons de faire 3 et 4 clusters.

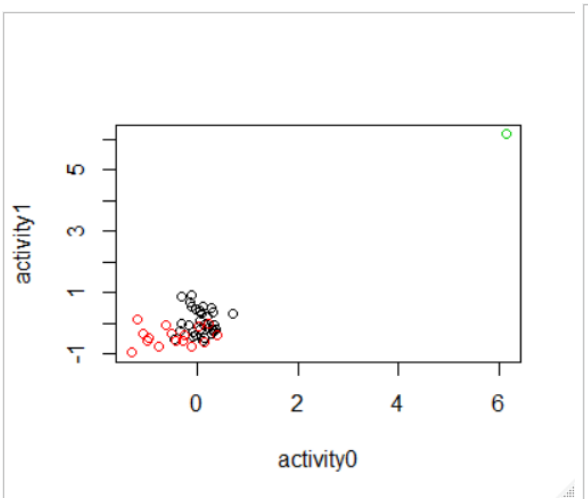


k = 4 k = 3

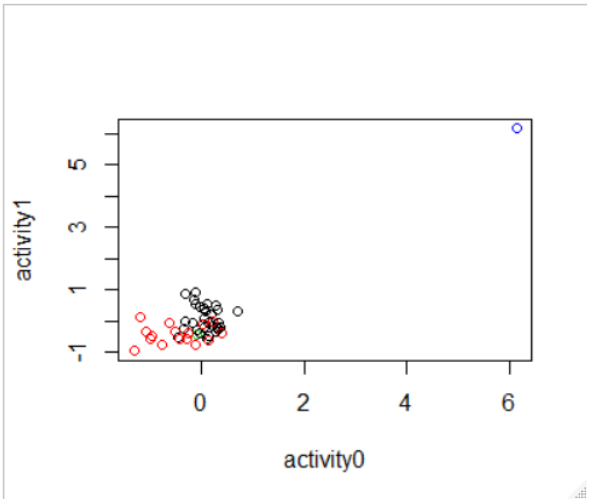
MODELE 2 : K- MEANS sans ACP



Selon le graphe des inerties expliquées, nous décidons de faire 3 et 4 clusters.

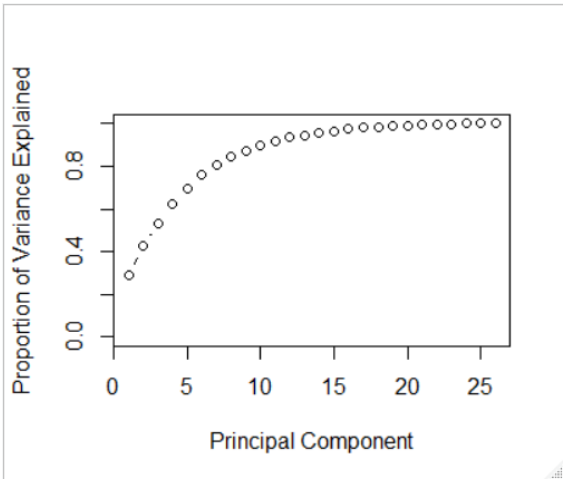


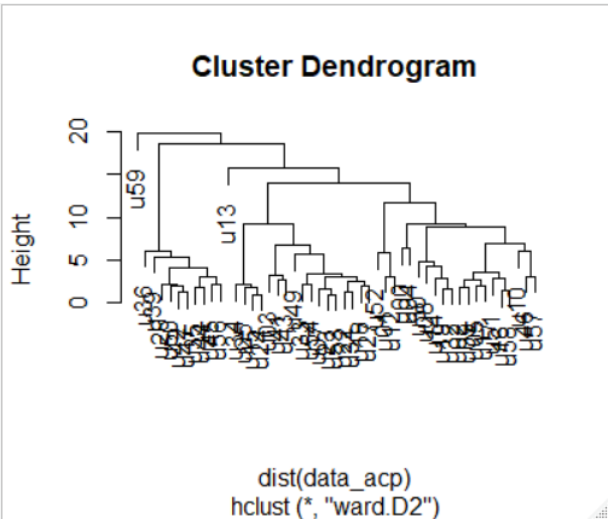
k = 3



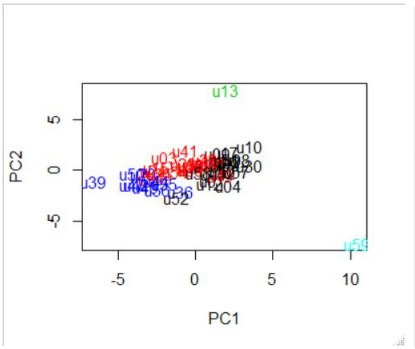
k = 4

MODELE 3 : CAH avec ACP

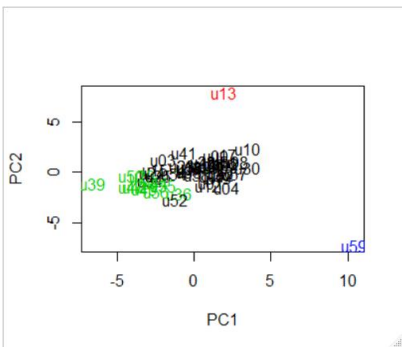




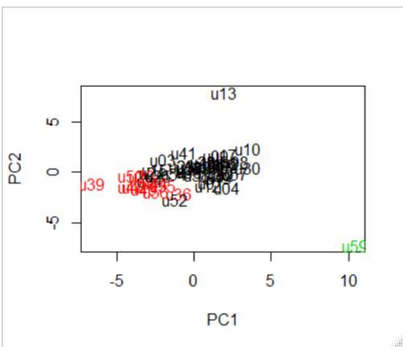
Selon les proportions de la variance expliquée, nous prenons 8 composants principaux.



k = 5

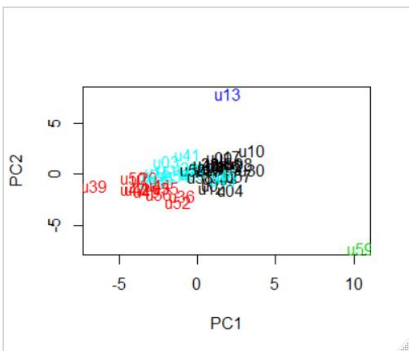


k = 4

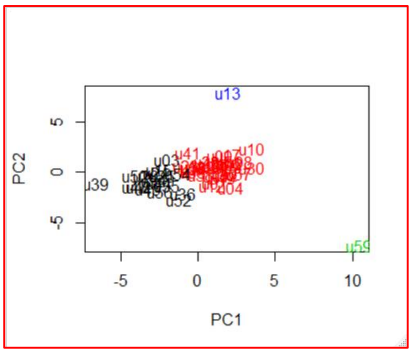


k = 3

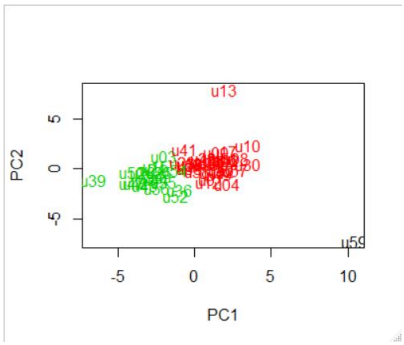
MODELE 4 : K- MEANS avec ACP



k = 5



k = 4

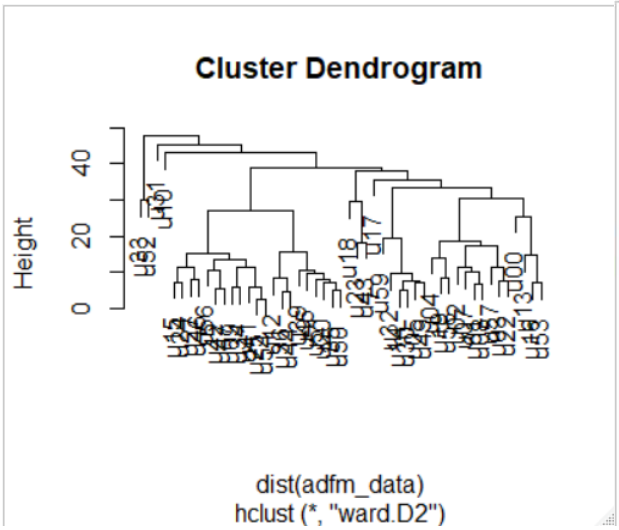


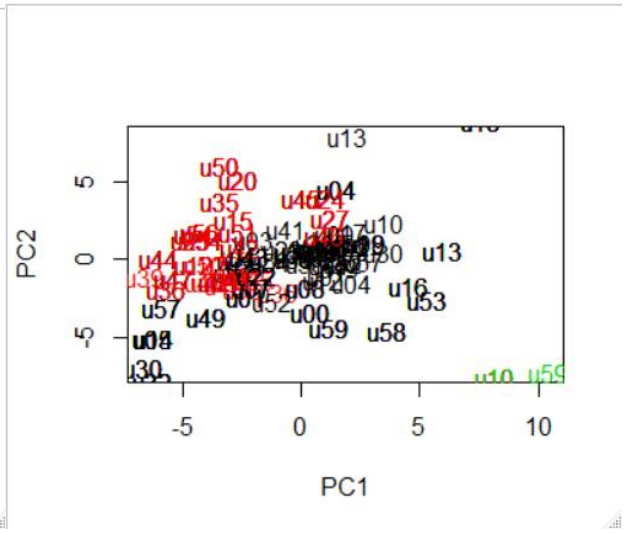
k = 3

OPTION 2 : AVEC 235 VARIABLES MIXTES RÉDUITS PAR AFDM

MODELE 5 : CAH AVEC 235 VARIABLES RÉDUITS PAR ADE4

Nous avons choisi de garder 10 composants principaux après la réduction de dimensions sur les variables mixtes avec le package « ade4 ».

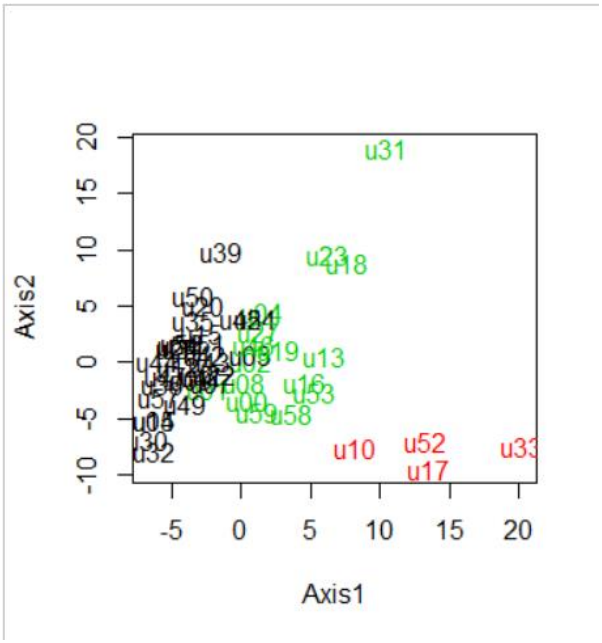




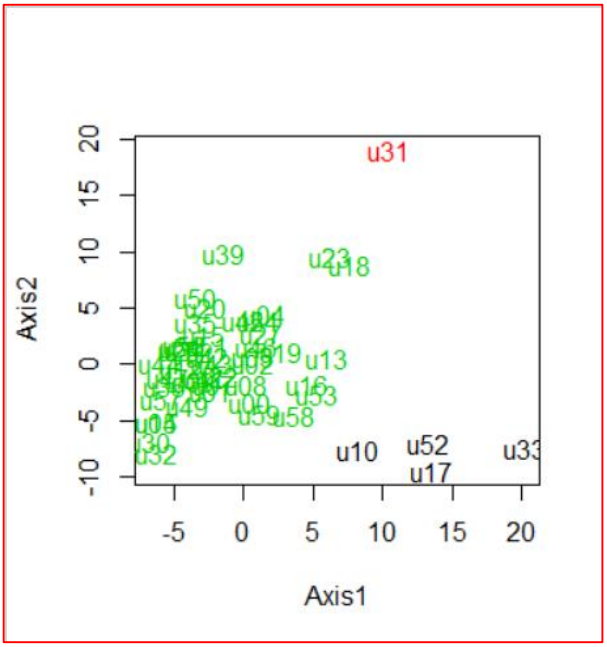
k = 5

MODELE 6 : K-MEANS AVEC 235 VARIABLES RÉDUITS PAR ADE4

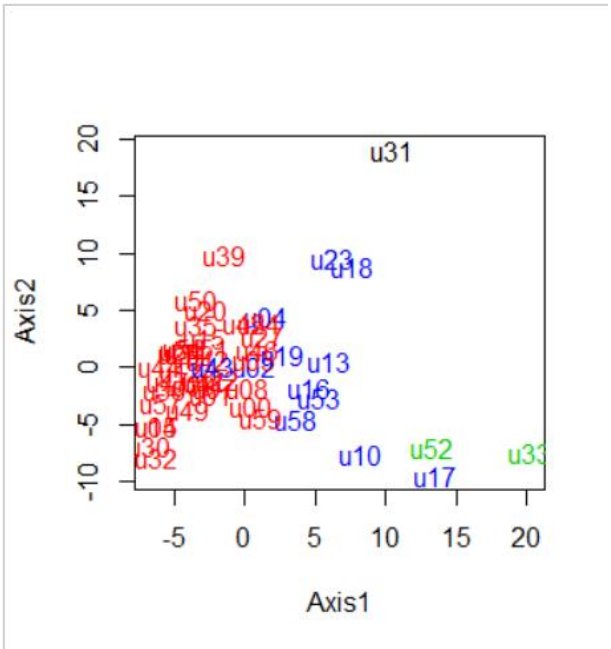
Nous avons choisi de garder 10 et 15 composants principaux après la réduction de dimensions sur les variables mixtes avec le package « ade4 ».



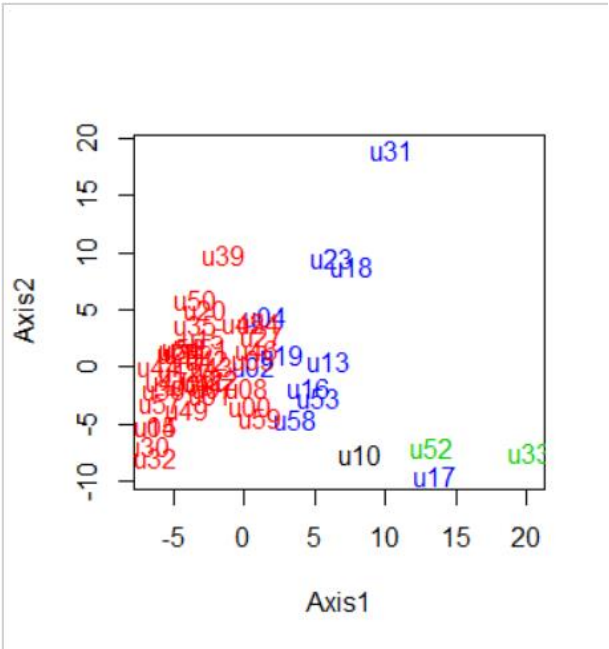
k = 3 / composants principaux = 10



k = 3 / composants principaux = 15

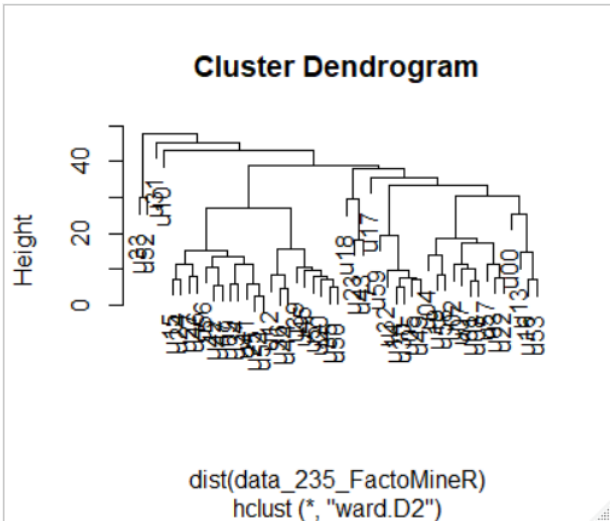


k = 4 / composants principaux = 10

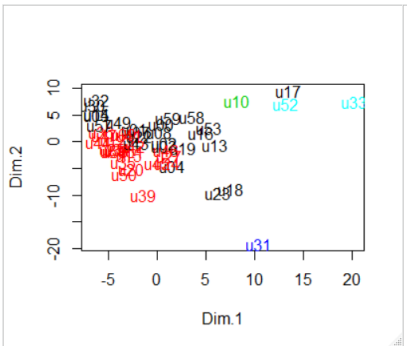


k = 4 / composants principaux = 15

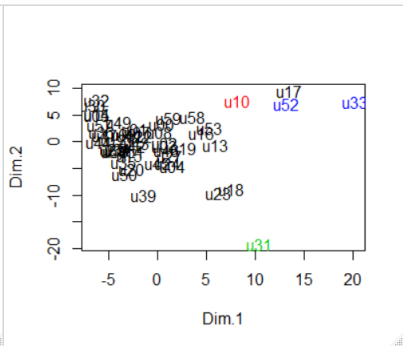
MODELE 7 : CAH AVEC 235 VARIABLES RÉDUITS PAR FACTOMINER



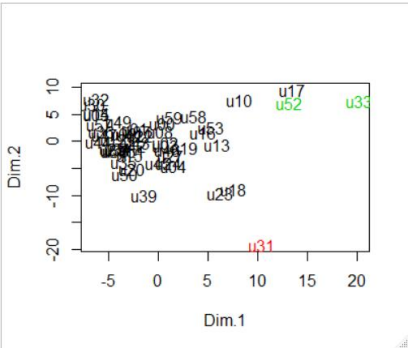
Nous avons choisi de garder 10 composants principaux après la réduction de dimensions sur les variables mixtes avec le package « FactoMineR ».



k = 5



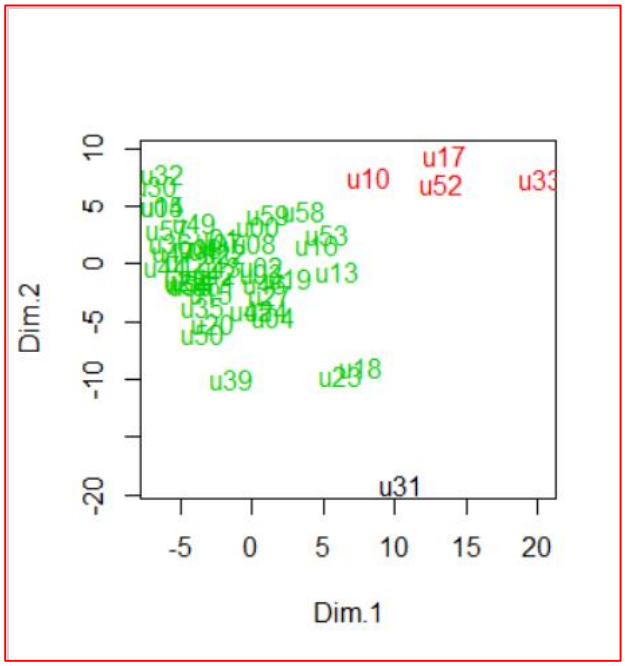
k = 4



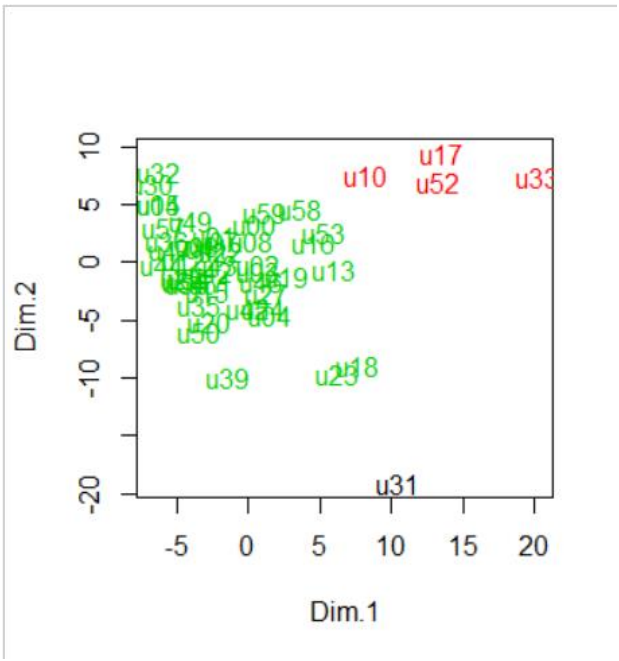
k = 3

MODELE 8 : K- MEANS AVEC 235 VARIABLES RÉDUITS PAR FACTOMINER

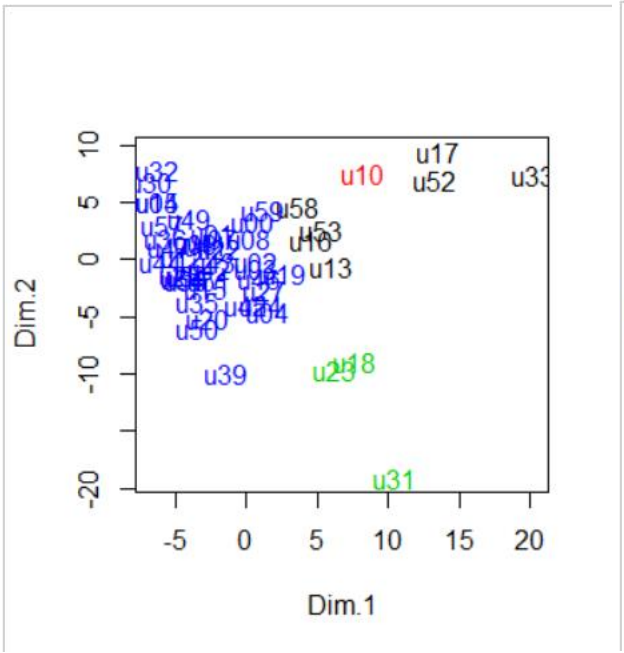
Nous avons choisi de garder 10 et 15 composants principaux après la réduction de dimensions sur les variables mixtes avec le package « FactoMineR ».



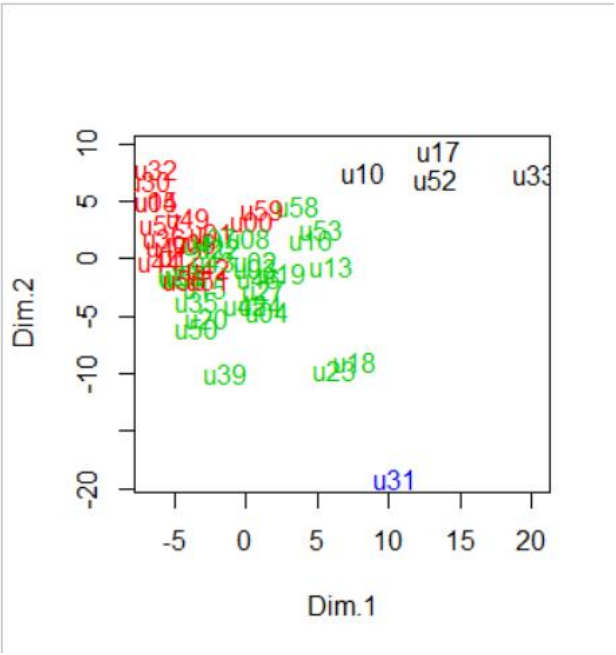
k = 3 / composants principaux = 10



k = 3 / composants principaux = 15



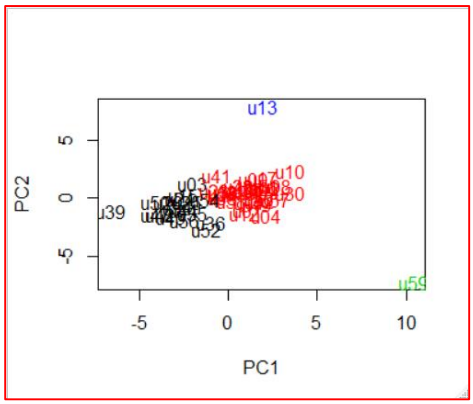
k = 4 / composants principaux = 10



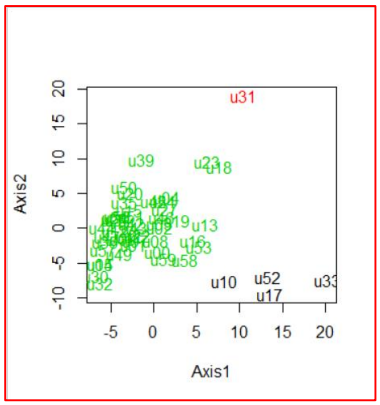
k = 4 / composants principaux = 15

INTERPRETATION

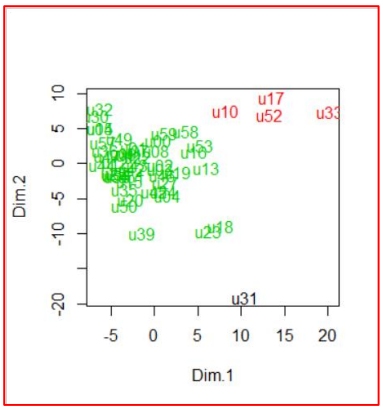
Les meilleurs modèles, selon nous, sont les suivants :



Modèle 4



Modèle 6



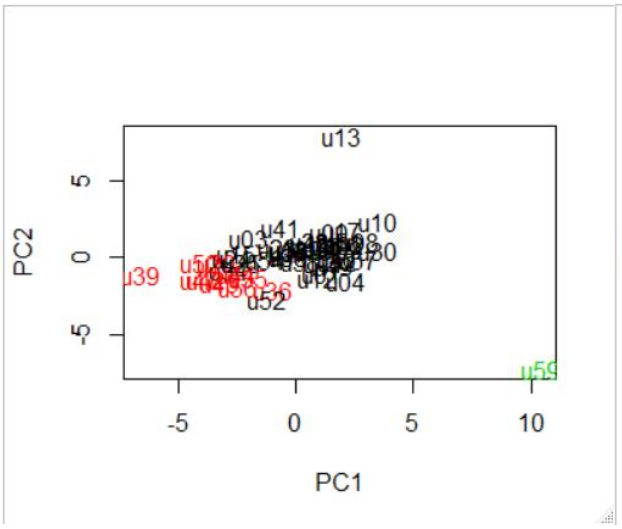
Modèle 8

Nous pouvons observer que les techniques de réduction de dimension comme ACP et AFDM améliorent nettement le visuel et donnent de meilleurs résultats en clustering.

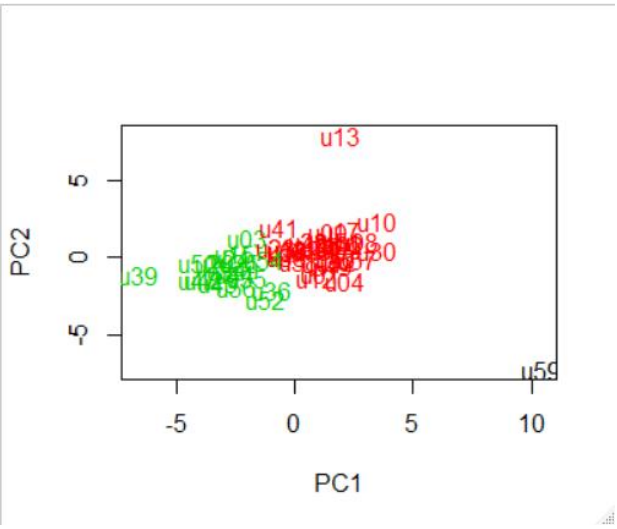
Le clustering sur 26 variables quantitatives (modèle 4) est très clair et fait sortir 2 valeurs outliers : u13 et u59.

Les coordonnées des observations sur l'axe 2 ont des signes opposés selon « ade4 » ou « FactoMineR ».

Sur les 26 variables quantitatives, le résultat des méthodes CAH et Kmeans est similiaire.



CAH avec k = 3



K-MEANS avec k = 3

Nous avons fait des choix forts et arbitraires en enlevant des variables jugées inexploitables et en créant des variables calculées à partir des variables initiales. Une piste d'améliorations, par conséquent, est d'injecter de nouvelles variables au fur à mesure tout en observant le résultat rendu.

SECTION 3 : PRÉDICTION

Puisque seul le dossier « survey » dispose des données d'apprentissage (« pre ») et de test (« post »), nous avons construit notre modèle sur les 202 variables explicatives du dossier. Nous avons choisi de faire modèle d'arbre de décision avec le library « rpart ».
Le temps de calcul sur 2020 variables est rapide. Nous avons rencontré un problème de niveaux manquants sur les variables catégorielles des données d'apprentissage par rapport aux données test. L'astuce est de fusionner les données test et apprentissage pour mettre à niveau et puis diviser à nouveau en deux jeux de données initiaux.

```
#Mise à niveau (levels) des variables trains
totalData <- rbind(train, test)
for (i in 1:length(names(totalData))) {
  levels(train[, i]) <- levels(totalData[, i])
}
```

La matrice de confusion :

	Fairly bad	Fairly good	Very bad	Very good
Fairly bad	1	5	0	0
Fairly good	7	19	0	0
Very bad	1	1	0	0
Very good	0	5	0	0

Toutes les métriques grâce au library « mltest »

	balanced.accuracy	DOR	F0.5	F1	F2	FDR	FNR	FOR	FPR	geometric.mean	Jaccard
Fairly bad	0.4351852	0.4750000	0.1190476	0.1333333	0.1515152	0.8888889	0.8333333	0.20833333	0.2962963	0.3424674	0.07142857
Fairly good	0.4070513	0.2467532	0.6506849	0.6785714	0.7089552	0.3666667	0.2692308	0.87500000	0.9166667	0.2467741	0.51351351
Very bad	0.5000000	NaN	NaN	NaN	NaN	NaN	1.0000000	0.09090909	0.0000000	0.0000000	0.00000000
Very good	0.5000000	NaN	NaN	NaN	NaN	NaN	1.0000000	0.20000000	0.0000000	0.0000000	0.00000000

	L	lambda	MCC	MK	NPV	OP	precision	recall	specificity	Youden
Fairly bad	0.5625000	1.184211	-0.1122626	-0.09722222	0.7916667	-0.1042008	0.1111111	0.1666667	0.70370370	-0.1296296
Fairly good	0.7972028	3.230769	-0.2119557	-0.24166667	0.1250000	-0.2824551	0.6333333	0.7307692	0.08333333	-0.1858974
Very bad	NaN	1.000000	NaN	NaN	0.9090909	-0.4871795	NaN	0.0000000	1.00000000	0.0000000
Very good	NaN	1.000000	NaN	NaN	0.8000000	-0.4871795	NaN	0.0000000	1.00000000	0.0000000

CONCLUSION

Nous avons pu constater que dans un contexte Big Data, la fouille de données est très complexe. Le premier défi est d'appliquer les algorithmes de façon « aveugle ». C'est à dire de ne pas prendre en compte les limites des modèles. En effet, certains modèles ne sont pas faits pour être appliquer sur les jeux de données avec plus de colonnes que de lignes.
Le deuxième risque est la consolidation de données de sources hétérogènes. Cette consolidation est très délicate car les données ne sont pas collectées de la même manière ni au même rythme pour toutes les sources. Il faut donc s'assurer de la cohérence des données collectées.