Projet ACP

Introduction :

Le projet est une initiation à l'Analyse en Composante Principale et à la découverte de ses tenants et aboutissant

Il est constitué de 4 exercices.

Le but est d'appréhender l'ACP et apprendre à projeter les données de meilleur façon pour moins de perte.

*** Exercice 1 ***

Creation des donnees

```
# On prend la moyenne de chaque intervalle
CA <- c(0.125, 0.375, 0.75, 1.75, 3.75, 7.50)

# Labels pour la droite des abscisses
names(CA) <- c("[0,0.25[","[0.25,0.5[","[0.5,1[","[1,2.5]","[2.5,5[","[5,10[")

# Effectif des entreprises pour chaque intervalle
nb_entreprises <- c(137, 106, 112, 154, 100, 33)
```

1) Calcul du chiffre d'affaire moyen et de l'ecart type de la serie

```
meanCA <- mean(CA*nb_entreprises)
meanCA
```

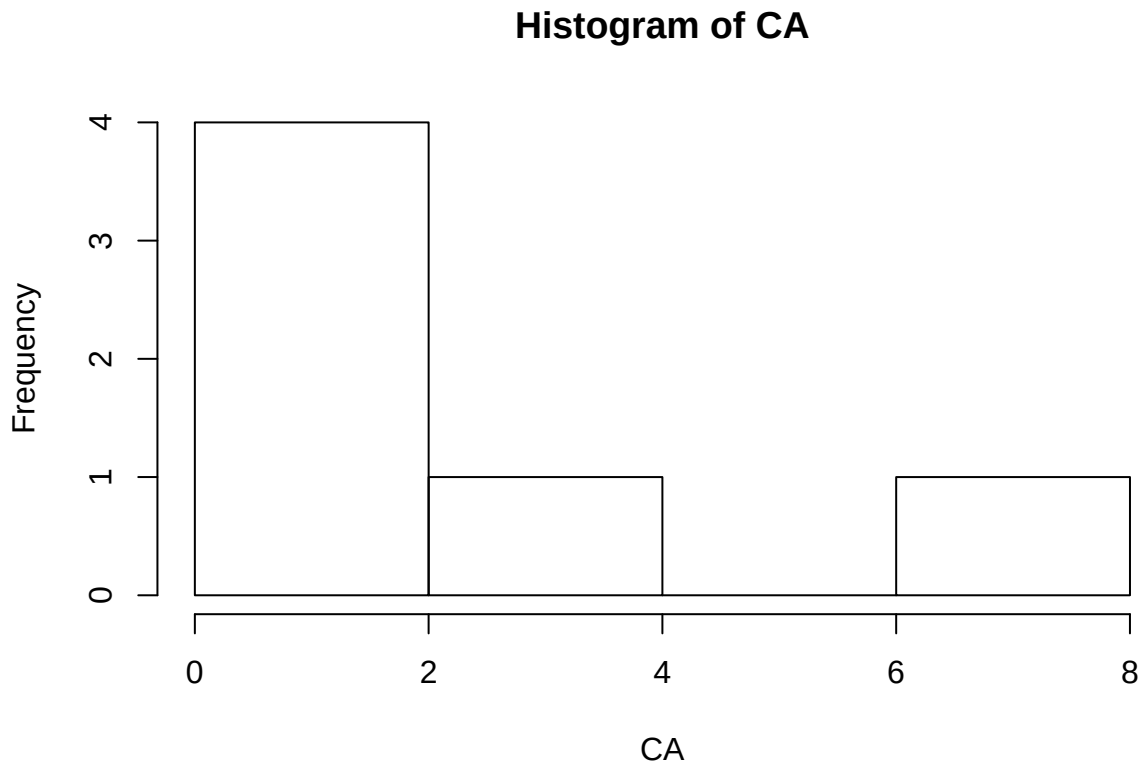
```
## [1] 172.1458
```

```r
sum_entreprises <- sum(nb_entreprises)
ecart_type <- sd(CA)*sqrt((sum_entreprises-1)/sum_entreprises)
ecart_type
```

```
## [1] 2.835042
```

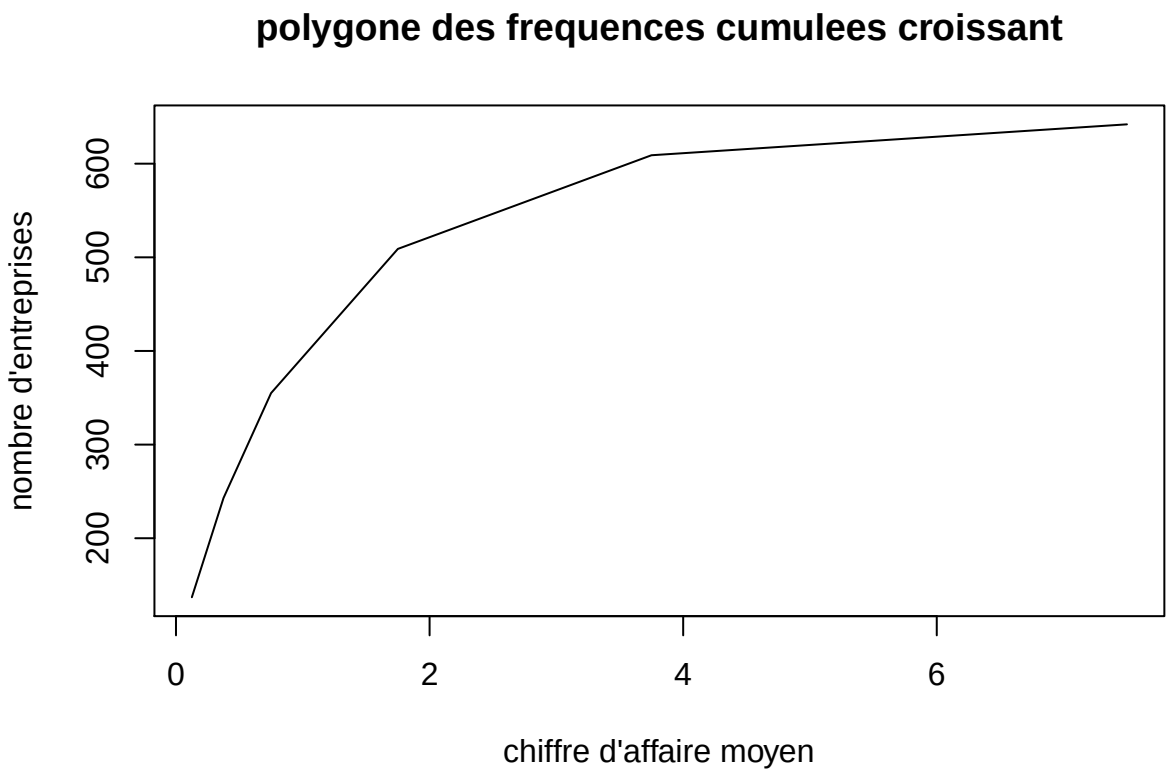
2) Construction de l'histogramme des frequences

```r
hist(CA)
```

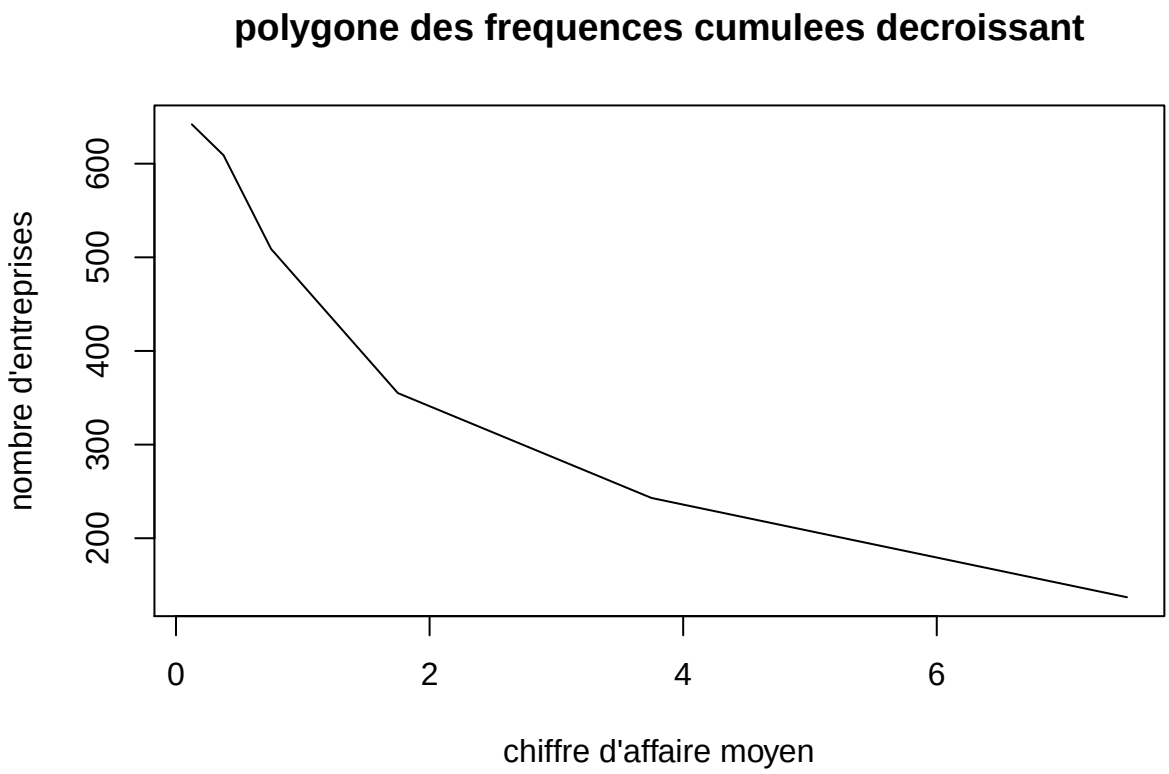


3) Construction du polygones des fréquences cummulées

```r
# Polygones des frequences cumulees croissant
fc = cumsum(nb_entreprises)
plot(CA,fc,type="l", main="polygone des frequences cumulees croissant",xlab="chiffre d'affaire moyen", y
```




```r
# Polygones des frequences cumulees decroissant
nb_entreprises2 <- c(sum_entreprises, -nb_entreprises[6], -nb_entreprises[5], -nb_entreprises[4], -nb_e
fc2 = cumsum(nb_entreprises2)
plot(CA,fc2,type="l", main="polygone des frequences cumulees decroissant",xlab="chiffre d'affaire moyen
```

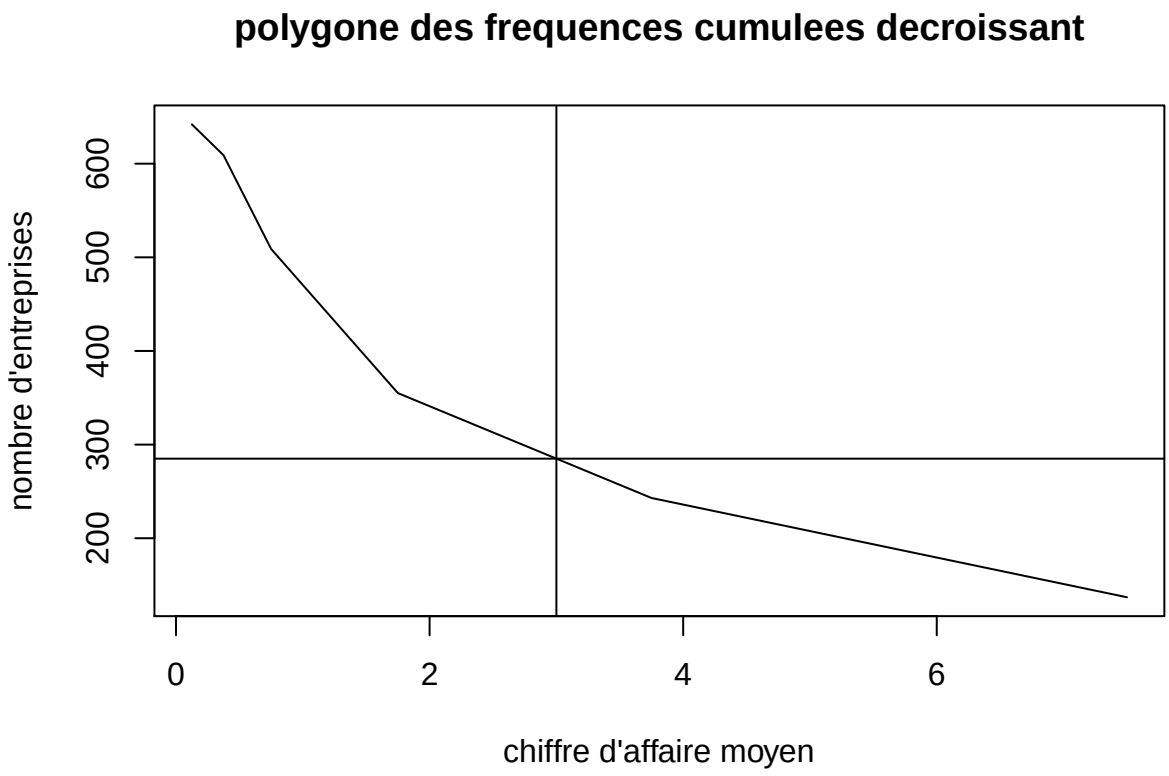


4) Calcul de la médiane et calcul de la proportion d'entreprise dont le chiffre d'affaire supèrieur à 3 millions

```
# Calcul de la mediane
median(CA)
```

```
## [1] 1.25
```

```
# Determination de manière empirique de la proportion d'entreprise dont CA > 3M
plot(CA,fc2,type="l", main="polygone des frequences cumulees decroissant",xlab="chiffre d'affaire moyen
abline(v=3)
abline(h=285)
```




```
proportion_entreprise = 285/sum_entreprises

print(paste("La proportion d'entreprise dont le chiffre d'affaire est supèrieur à 3 millions d'euros est
```

```
## [1] "La proportion d'entreprise dont le chiffre d'affaire est supèrieur à 3 millions d'euros est de p
```

*** Exercice 2 ***

1-2) Visualisation et import des donnees

Creation des donnees

```
data <- read.table("C:/Users/admin/Documents/Cours/Mathématiques avancées pour le Big Data avec R/Exerci
```

3) Calcul de la moyenne et de la variance des 8 variables

```
meantab <- c(mean(data$x1), mean(data$x2) ,mean(data$x3) ,mean(data$x4),mean(data$y1) ,mean(data$y2), me
for(i in 1:8) print(paste("moyenne de la variable ",i,": ",meantab[i]))
```

```
## [1] "moyenne de la variable  1 :  9"
## [1] "moyenne de la variable  2 :  9"
## [1] "moyenne de la variable  3 :  9"
## [1] "moyenne de la variable  4 :  9"
## [1] "moyenne de la variable  5 :  7.50090909090909"
## [1] "moyenne de la variable  6 :  7.50090909090909"
## [1] "moyenne de la variable  7 :  7.5"
## [1] "moyenne de la variable  8 :  7.50090909090909"
```

```
cat("\n")
```

```
vartab <- c(var(data$x1), var(data$x2) ,var(data$x3) ,var(data$x4),var(data$y1) ,var(data$y2), var(data$
for(i in 1:8) print(paste("variance de la variable ",i,": ",vartab[i]))
```

```
## [1] "variance de la variable  1 :  11"
## [1] "variance de la variable  2 :  11"
## [1] "variance de la variable  3 :  11"
## [1] "variance de la variable  4 :  11"
## [1] "variance de la variable  5 :  4.12726909090909"
## [1] "variance de la variable  6 :  4.12762909090909"
## [1] "variance de la variable  7 :  4.12262"
## [1] "variance de la variable  8 :  4.12324909090909"
```

4) Calcul des covariances et des coefficients de corrélation des couples

```r
covtab <- c(cov(data$x1,data$y1),cov(data$x2,data$y2),cov(data$x3,data$y3),cov(data$x4,data$y4))
print("covariances des corrélation des couples : ")
```

```
## [1] "covariances des corrélation des couples : "
```

```r
covtab
```

```
## [1] 5.501 5.500 5.497 5.499
```

```r
cortab <- c(cor(data$x1,data$y1),cor(data$x2,data$y2),cor(data$x3,data$y3),cor(data$x4,data$y4))
print("coefficient des corrélation des couples : ")
```

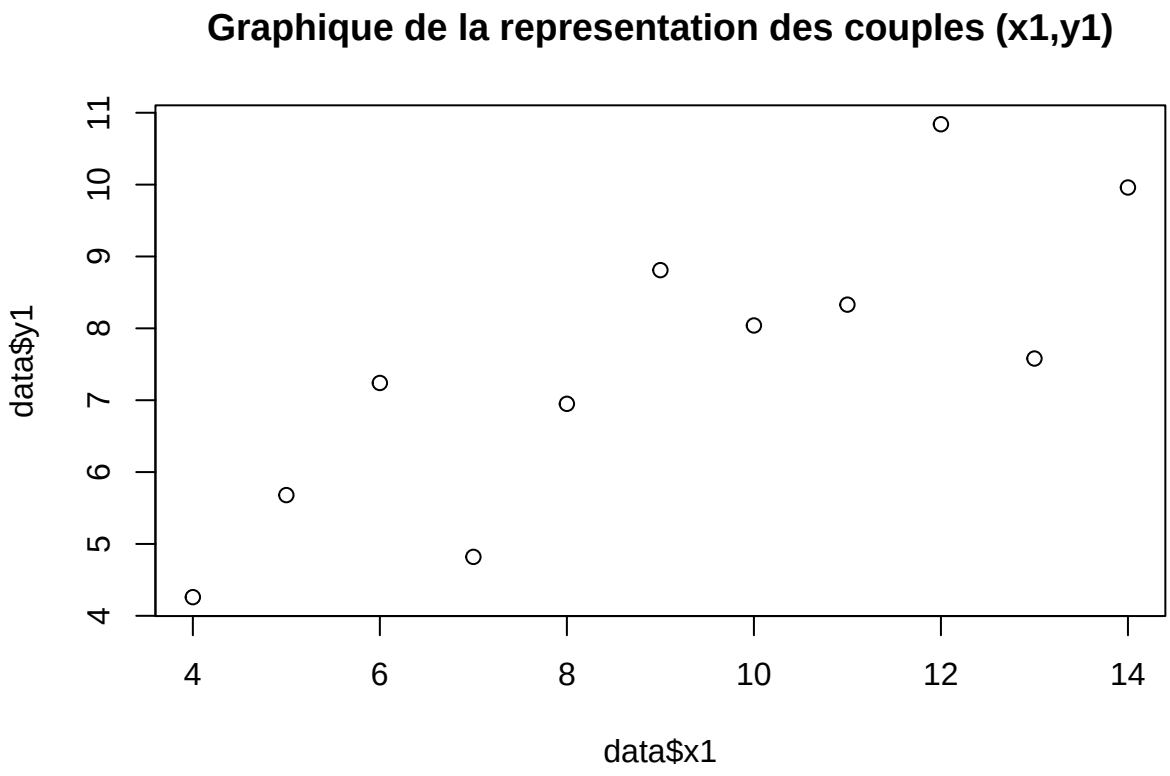
```
## [1] "coefficient des corrélation des couples : "
```

```r
cortab
```

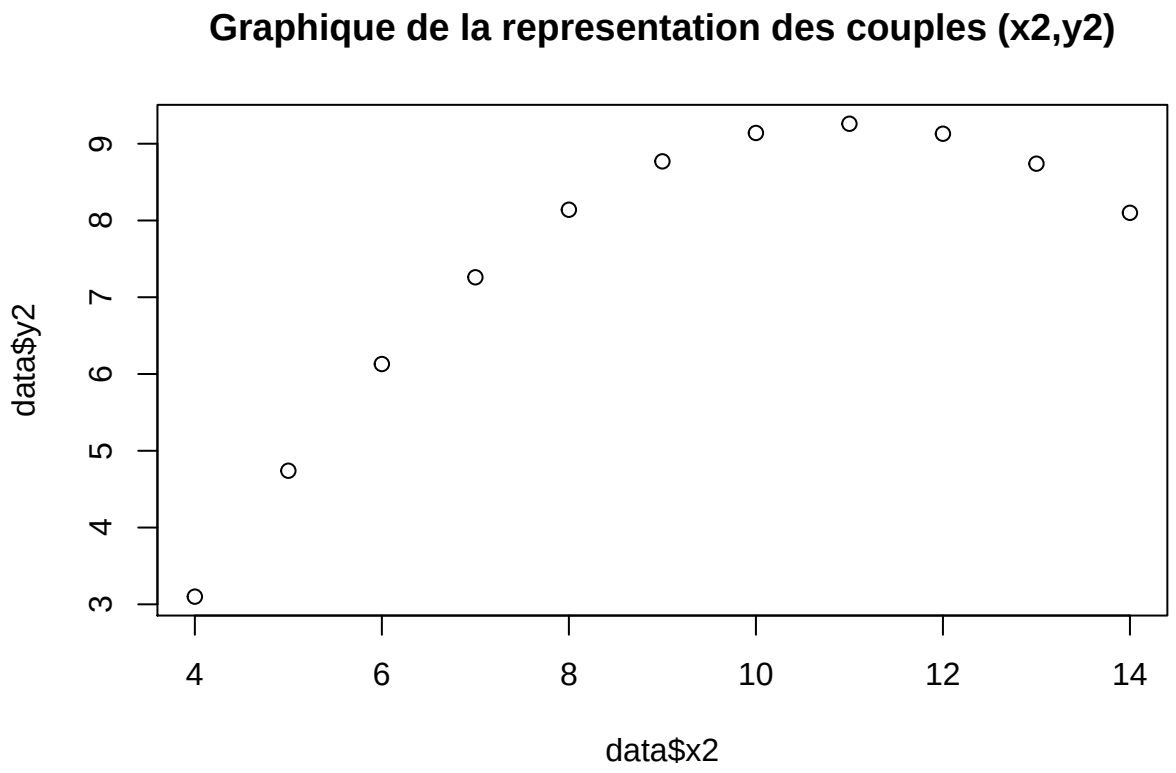
```
## [1] 0.8164205 0.8162365 0.8162867 0.8165214
```

5) Graphique des représentations de couples de valeurs

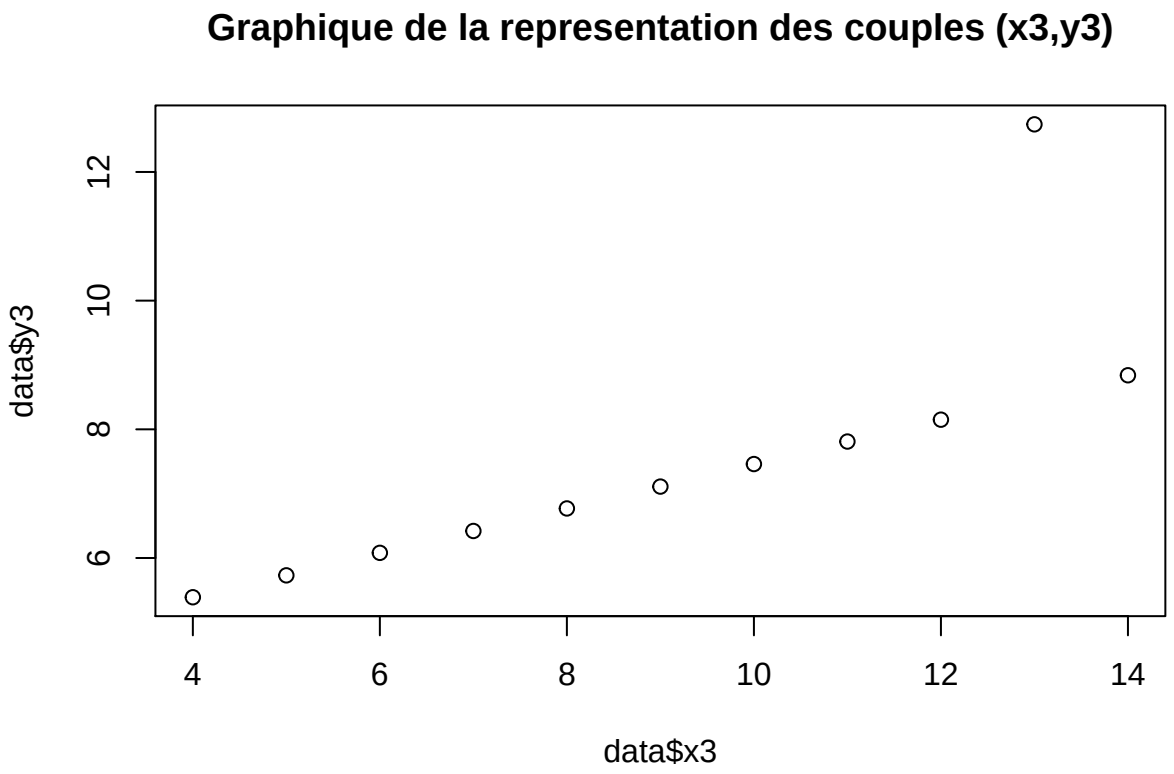
```r
plot(data$x1,data$y1)
title(main = "Graphique de la representation des couples (x1,y1)")
```



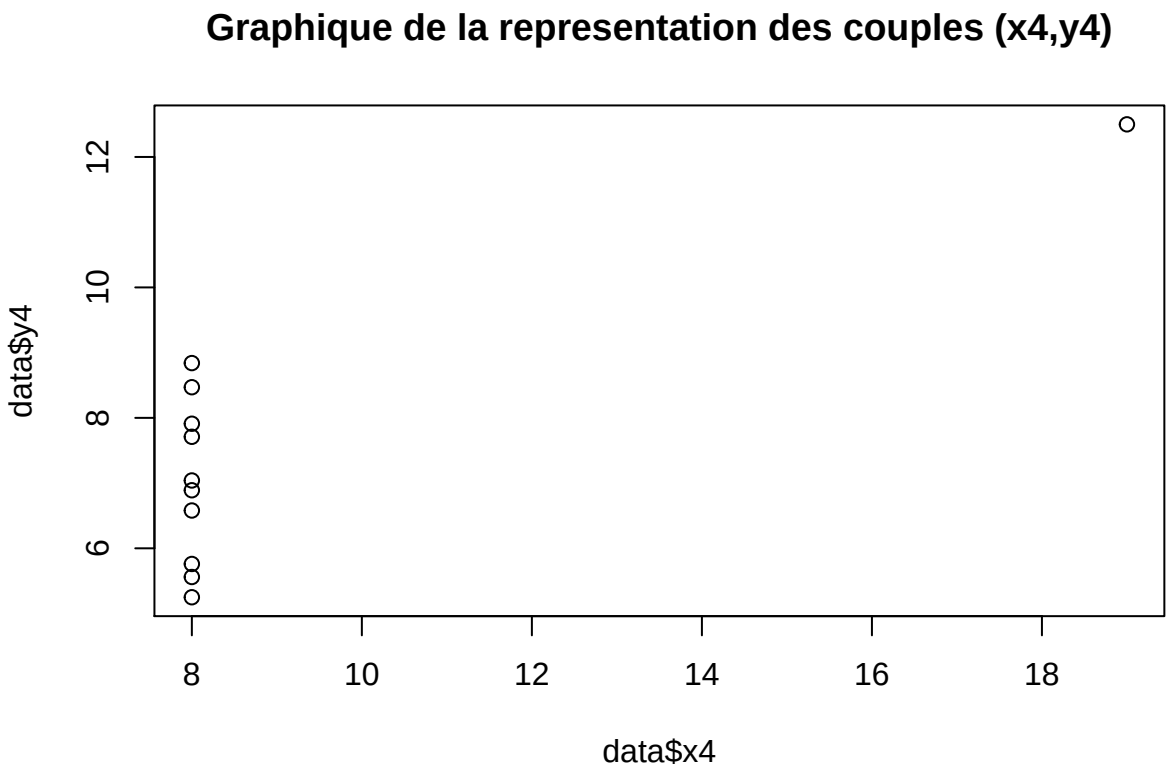
```r
plot(data$x2,data$y2)
title(main = "Graphique de la representation des couples (x2,y2)")
```




```r
plot(data$x3,data$y3)
title(main = "Graphique de la representation des couples (x3,y3)")
```




```
plot(data$x4,data$y4)
title(main = "Graphique de la representation des couples (x4,y4)")
```



6) Résultats questions 3, 4 et 5 pour le couple (x1,y1) après que les variables ont été centrés réduits

```
# Centrage et réduction des variables x1 et y1
centered.data.x1 <- scale(data$x1, scale = FALSE)
centered.data.y1 <- scale(data$y1, scale = FALSE)

print(paste("Moyenne des valeurs centrée réduite de x1 :",mean(centered.data.x1)))
```

```
## [1] "Moyenne des valeurs centrée réduite de x1 : 0"
```

```
print(paste("Moyenne des valeurs centrée réduite de y1 :",mean(centered.data.y1)))
```

```
## [1] "Moyenne des valeurs centrée réduite de y1 : 0"
```

```
var.scaled.x1 <- var(centered.data.x1)
var.scaled.y1 <- var(centered.data.y1)

print(paste("Variance des valeurs centrée réduite de x1 :",var.scaled.x1))
```

```
## [1] "Variance des valeurs centrée réduite de x1 : 11"
```

```r
print(paste("Variance des valeurs centrée réduite de y1 :",var.scaled.y1))
```

```
## [1] "Variance des valeurs centrée réduite de y1 : 4.12726909090909"
```

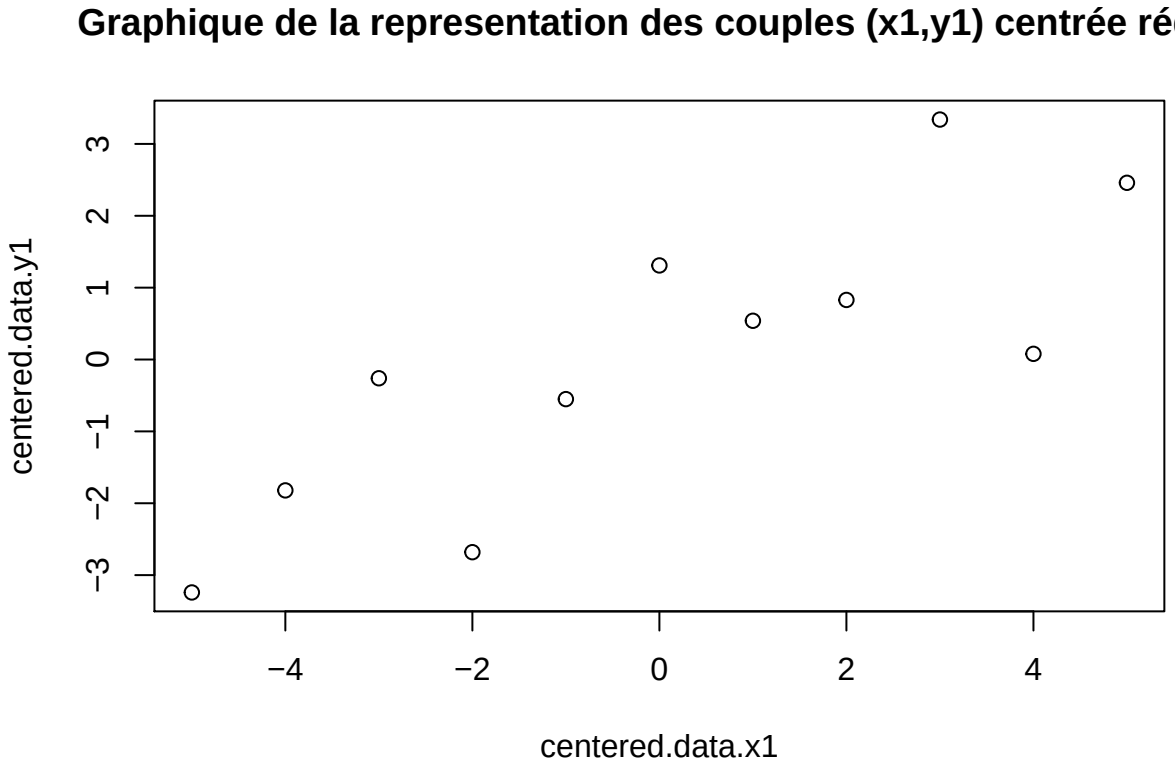
```r
cor.scaled <- cor(centered.data.x1,centered.data.y1 )
cov.scaled <- cov(centered.data.x1,centered.data.y1 )

print(paste("Coefficients des corrélation du couples (x1,y1) centrée réduit :",cor.scaled))
```

```
## [1] "Coefficients des corrélation du couples (x1,y1) centrée réduit : 0.81642051634484"
```

```r
print(paste("Covariances des corrélation du couples (x1,y1) centrée réduit :",cov.scaled))
```

```
## [1] "Covariances des corrélation du couples (x1,y1) centrée réduit : 5.501"
```

```r
plot(centered.data.x1,centered.data.y1)
title(main = "Graphique de la representation des couples (x1,y1) centrée réduit")
```



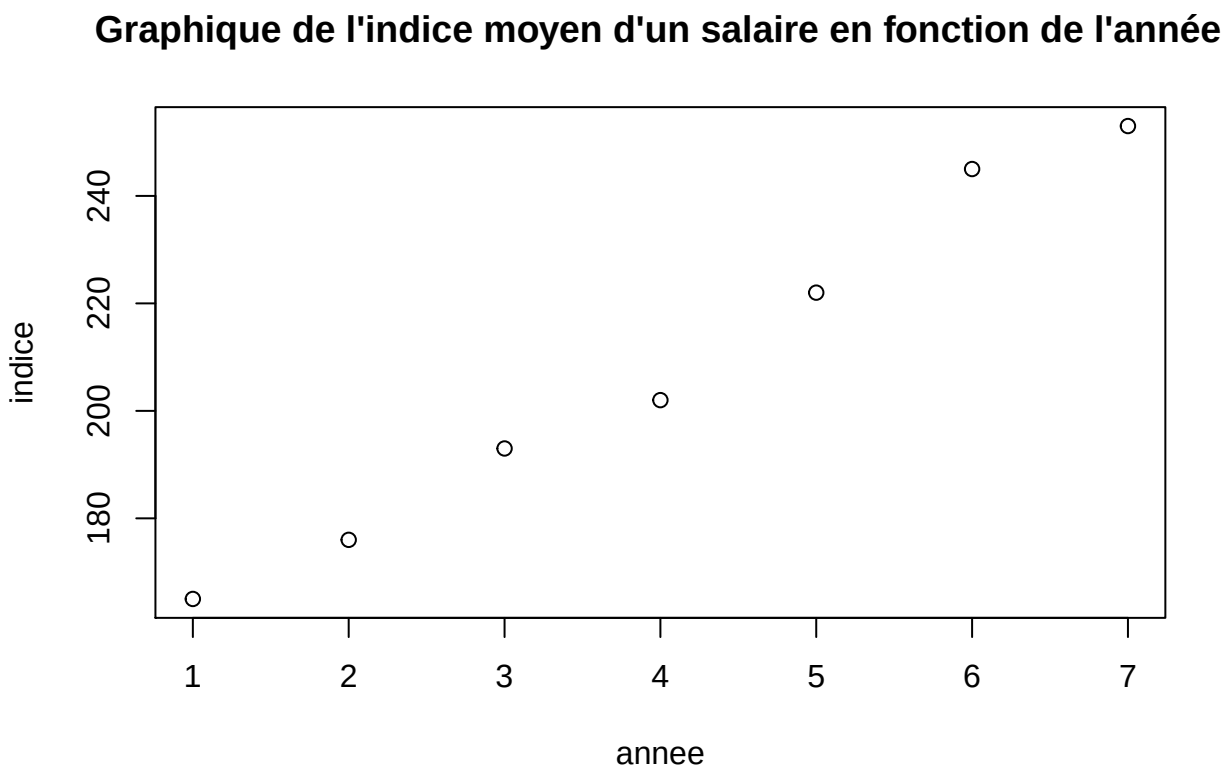
*** Exercice 3 ***

Creation des donnees

```
annee <- c(1:7)
indice <- c(165,176,193,202,222,245,253)
```

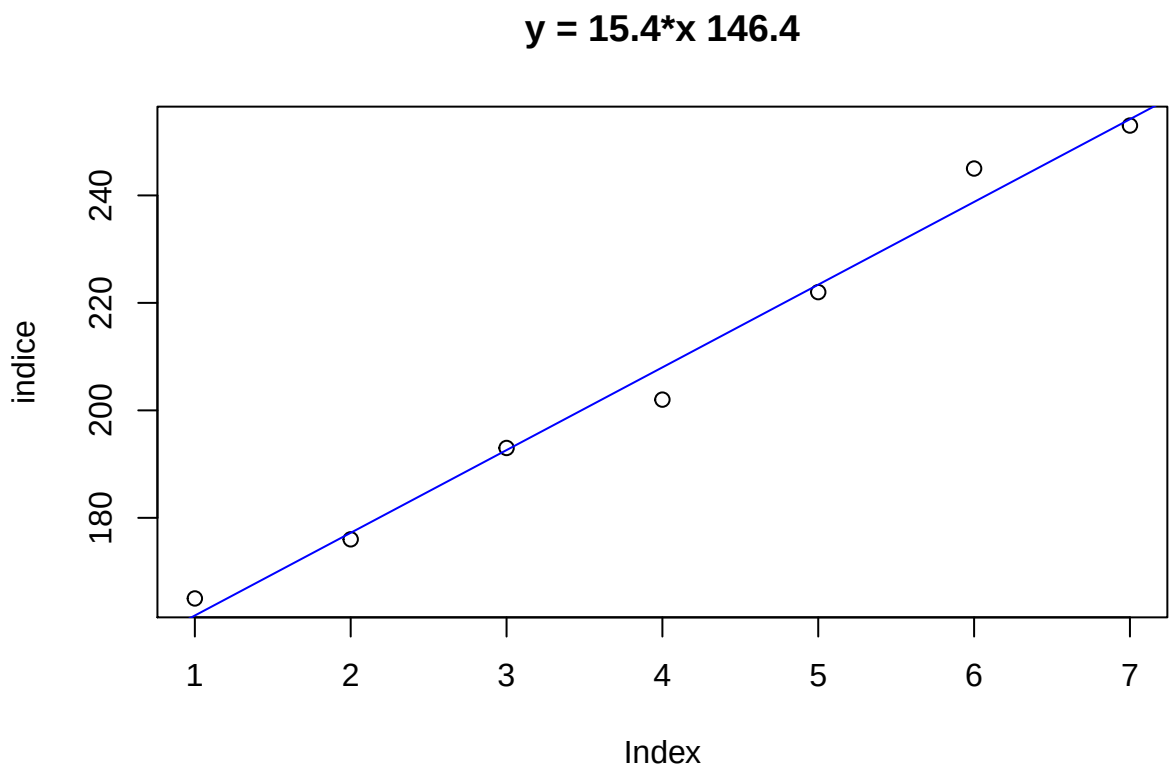
1) Nuage de points representant la série statistique

```
plot(annee,indice)
title(main = "Graphique de l'indice moyen d'un salaire en fonction de l'année")
```



2) Calcul de l'équation de la droite

```r
donnees <- data.frame(annee, indice)
reg<-lm(indice ~ annee, data=donnees)
coeff=coefficients(reg)

# Equation de la droite de regression :
equation_droite = paste0("y = ", round(coeff[2],1), "*x ", round(coeff[1],1))

# Graphe
plot(indice, main=equation_droite)
abline(reg, col="blue")
```




```r
# Cette formule nous renvois les coefficient a et b de l'equation de la droite (ax +b)
lm(indice ~ annee)
```

```
##
## Call:
## lm(formula = indice ~ annee)
##
## Coefficients:
## (Intercept)        annee
##      146.43        15.39
```

```r
a <- 15.39
b <- 146.43
```

```r
print("L'équation de la droite est donc y = 15.4x + 146.4")
```

```
## [1] "L'équation de la droite est donc y = 15.4x + 146.4"
```

3) Calcul de l'indice à l'année 9

```r
# on remplace x par 9 dans l'équation de la droite de regréssion linéaire pour trouver l'indice de l'an

indice_9 <- (a*9)+b
print(paste("L'indice moyen de salaire à l'année 9 sera donc de : ",indice_9))
```

```
## [1] "L'indice moyen de salaire à l'année 9 sera donc de :  284.94"
```

---

	popu	entr	rece	sean	comm	etab	salle	faut	artes	multi	depart	reg
D1	0.515	0.769	4.065	26	16	20	35	6288	12	0	Ain	Rhone-Alpes
D2	0.536	0.731	3.942	28	14	15	38	7403	8	0	Aisne	Picardie
D3	0.345	0.499	2.789	17	7	11	28	3956	4	0	Allier	Auvergne
D4	0.14	0.453	2.262	17	13	15	23	3480	7	0	Alpes de Hautes Provence	Provence-Alpes-Coted'Azur
D5	0.121	0.522	2.908	21	19	23	35	6053	5	0	Hautes Alpes	Provence-Alpes-Coted'Azur
D6	1.011	3.52	21.731	111	23	42	94	16764	8	1	Alpes Maritimes	Provence-Alpes-Coted'Azur
D7	0.286	0.401	1.909	12	20	23	34	5814	9	0	Ardeche	Rhone-Alpes
D8	0.29	0.371	1.985	13	6	7	15	3214	4	0	Ardennes	Champagne-Ardenne
D9	0.137	0.201	0.926	6	9	10	11	2839	6	0	Ariege	Midi-Pyrenees
D10	0.292	0.573	3.491	20	4	5	19	4077	2	1	Aube	Champagne-Ardenne
D11	0.31	0.674	3.435	29	11	15	37	7625	1	1	Aude	Languedoc-Roussillon
D12	0.264	0.433	2.213	15	13	16	27	4527	9	0	Aveyron	Midi-Pyrenees

Figure 1: Data

Exercice 4

PARTIE 1 : Première approche

popu : population du département (en millions)
entr : nombre d'entrées réalisées (en millions) rece : recettes (en millions d'euros)
sean : nombre de séances (en milliers) comm : nombre de communes équipées de salles de cinéma
etab : nombre de cinémas en activité salle : nombre de salles en activité
faut : nombre de fauteuils disponibles artes : nombre de salles d'art et d'essai
multi : nombre de multiplexes (au moins 8 salles)

	popu	entr	rece	sean	comm	etab	salle	faut	artes	multi
popu	1.0000000	0.7127322	0.6780419	0.7705001	0.6059985	0.7697961	0.8501478	0.8696339	0.7178729	0.7980112
entr	0.7127322	1.0000000	0.9978181	0.9874124	0.1932529	0.7607484	0.9253889	0.9098303	0.6959383	0.6292376
rece	0.6780419	0.9978181	1.0000000	0.9793410	0.1484393	0.7363797	0.9066167	0.8887391	0.6666153	0.5929633
sean	0.7705001	0.9874124	0.9793410	1.0000000	0.2699183	0.7970704	0.9617397	0.9446397	0.7190334	0.7027220
comm	0.6059985	0.1932529	0.1484393	0.2699183	1.0000000	0.7488693	0.4912274	0.5328285	0.6429825	0.5181113
etab	0.7697961	0.7607484	0.7363797	0.7970704	0.7488693	1.0000000	0.9051412	0.9093102	0.8455280	0.6735925
salle	0.8501478	0.9253889	0.9066167	0.9617397	0.4912274	0.9051412	1.0000000	0.9905868	0.7868377	0.7930314
faut	0.8696339	0.9098303	0.8887391	0.9446397	0.5328285	0.9093102	0.9905868	1.0000000	0.7953758	0.8136436
artes	0.7178729	0.6959383	0.6666153	0.7190334	0.6429825	0.8455280	0.7868377	0.7953758	1.0000000	0.5545097
multi	0.7980112	0.6292376	0.5929633	0.7027220	0.5181113	0.6735925	0.7930314	0.8136436	0.5545097	1.0000000

Figure 2: Matrice de corrélation

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	5.469266669	78.1323810	78.13238
Dim.2	0.866839890	12.3834270	90.51581
Dim.3	0.454304583	6.4900655	97.00587
Dim.4	0.171356862	2.4479552	99.45383
Dim.5	0.022267163	0.3181023	99.77193
Dim.6	0.009890233	0.1412890	99.91322
Dim.7	0.006074600	0.0867800	100.00000

Figure 3: Valeurs propres

	Dim1	Dim2	Dim3	Dim4	Dim5	Dim6	Dim7
sean	-0.3824447	-0.4538122	0.16909278	-0.1152524	-0.23258445	0.66731180	0.326365536
comm	-0.2809884	0.7920782	-0.17717926	-0.1833598	-0.40674963	0.25135136	-0.004639216
etab	-0.4076755	0.1888646	0.17892869	-0.4412893	0.73273032	-0.05770124	0.176510446
salle	-0.4160009	-0.2218443	0.02844936	-0.1844426	-0.08921790	-0.04063688	-0.856322809
faut	-0.4191567	-0.1738895	-0.02028038	-0.1419550	-0.40884050	-0.69352829	0.354084058
artes	-0.3703693	0.2026177	0.51267945	0.7431998	0.06562412	-0.04317252	-0.020399213
multi	-0.3502280	-0.1024225	-0.80245797	0.3896712	0.25327570	0.06075609	0.056812641

Figure 4: Vecteurs propres

1) Calcul de la matrice de corrélation

Creation des donnees

```
data <- read.table("C:/Users/admin/Documents/Cours/Mathématiques avancées pour le Big Data avec R/Exerci
```

```
DF1 <- data.frame(data[0],data[1:10])
print("Matrice de corrélation :")
```

```
## [1] "Matrice de corrélation :"
```

```
cor(DF1)
```

```
##            popu      entr      rece      sean      comm      etab
## popu  1.0000000 0.7127322 0.6780419 0.7705001 0.6059985 0.7697961
## entr  0.7127322 1.0000000 0.9978181 0.9874124 0.1932529 0.7607484
## rece  0.6780419 0.9978181 1.0000000 0.9793410 0.1484393 0.7363797
## sean  0.7705001 0.9874124 0.9793410 1.0000000 0.2699183 0.7970704
## comm  0.6059985 0.1932529 0.1484393 0.2699183 1.0000000 0.7488693
## etab  0.7697961 0.7607484 0.7363797 0.7970704 0.7488693 1.0000000
## salle 0.8501478 0.9253889 0.9066167 0.9617397 0.4912274 0.9051412
## faut  0.8696339 0.9098303 0.8887391 0.9446397 0.5328285 0.9093102
## artes 0.7178729 0.6959383 0.6666153 0.7190334 0.6429825 0.8455280
## multi 0.7980112 0.6292376 0.5929633 0.7027220 0.5181113 0.6735925
##           salle      faut     artes     multi
## popu  0.8501478 0.8696339 0.7178729 0.7980112
## entr  0.9253889 0.9098303 0.6959383 0.6292376
## rece  0.9066167 0.8887391 0.6666153 0.5929633
## sean  0.9617397 0.9446397 0.7190334 0.7027220
## comm  0.4912274 0.5328285 0.6429825 0.5181113
## etab  0.9051412 0.9093102 0.8455280 0.6735925
## salle 1.0000000 0.9905868 0.7868377 0.7930314
## faut  0.9905868 1.0000000 0.7953758 0.8136436
## artes 0.7868377 0.7953758 1.0000000 0.5545097
## multi 0.7930314 0.8136436 0.5545097 1.0000000
```

2) Corrélation entre les variables

Tout d'abord, il faut noter qu'on considère une forte corrélation entre deux variables à partir de 90%.

On trouve que "SEAN" qui représente le nombre de séance (en milliers) est fortement corrélé avec "ENTR" et SALLE. "ENTR" représentant le nombre d'entrées réalisé et salle, le nombre de salles en activité.

Mais la plus grande corrélation est entre "ENTR" et "RECE" , elle est maximale.

Ici, on peut aisément dire que plus le nombre d'entrées d'un film augmente et plus la recette de celui ci augmente.

A l'inverse, "COMM" est très peu corrélé avec RECE On peut donc dire que le nombre de communes équipé de cinema n'influence pas réellement la recette du département.

3) Comparaison du departement 75 avec les autres

Paris a une seule commune, un seul département, c'est pourquoi on peut observer que ses valeurs sont plus élevées que les autres. De plus, c'est le deuxième département le plus peuplé (2 125 000 habitants)

PARTIE 2 : ACP Version 1

L'analyse de composante principale ou ACP est une méthode d'analyse des données qui consiste à transformer des variables corrélées en nouvelles variables décorrélé les unes des autres. On les appelles les composantes principales (ou axes principaux) .

C'est une méthode statistique d'exploration de données multivariées (données contenant plusieurs variables). Elle permet de supporter l'analyse et la visualisation d'un jeu de données contenant des individus décrits pas plusieurs variables quantitatives.

Le but de l'ACP est d'identifier les variables corrélées réduire le nombre de variables et rendre l'information moins redondante plus facile à interpréter. En effet il est très compliqué de visualiser des données dans un espace multidimensionnelle.

L'ACP est utilisé dans plusieurs domaines de nos jours tel que la biologie, le traitement d'images et le Big data dans la gestion et l'interprétation des données.

Le but de cet exercice sera de trouver les axes principaux d'analyse de cette donnée multivariée puis projeter et interpréter les données.

4) Cercle de corrélation et plan de composante principale

On ne laisse que les variables que l'on souhaite étudier

```
#import des bibliothèques nécéssaires
library("factoextra")
```

```
## Warning: package 'factoextra' was built under R version 3.4.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.4.3
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

```
library("FactoMineR")
```

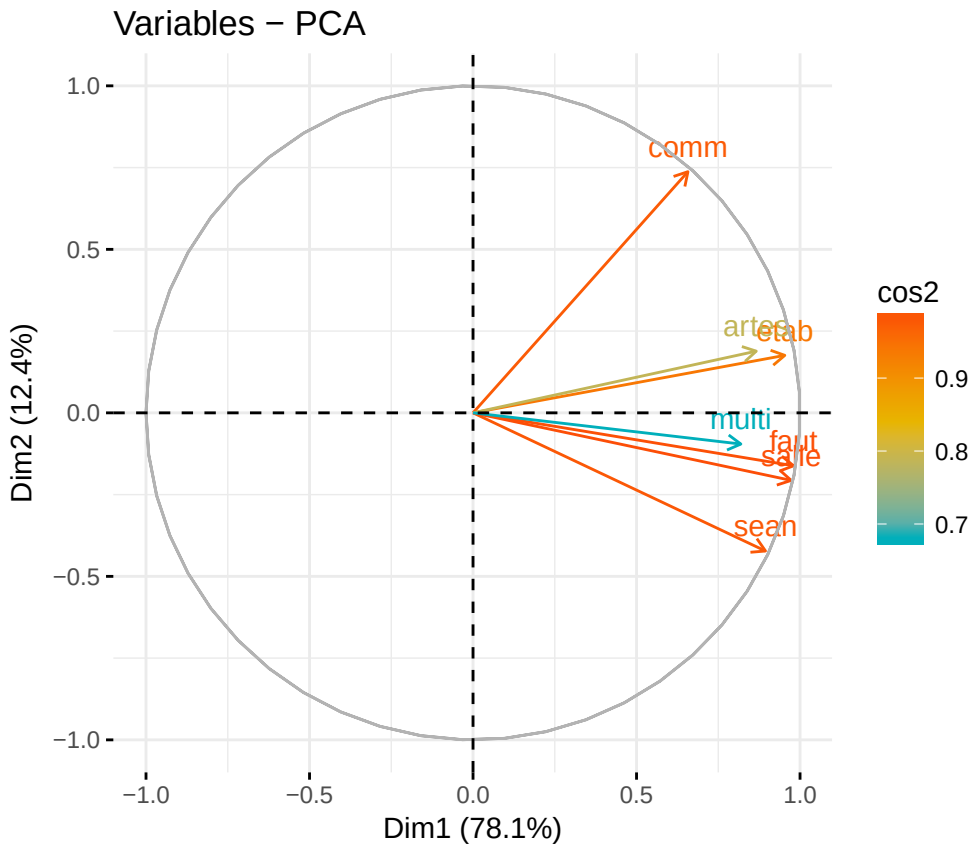
```
## Warning: package 'FactoMineR' was built under R version 3.4.3
```

```
data <- read.table("C:/Users/admin/Documents/Cours/Mathématiques avancées pour le Big Data avec R/Exerci
DF2 <- data.frame(data[0],data[4:10])

DF2.scaled <- scale(DF2, center = TRUE, scale = TRUE)

cor_df2 <- cor(DF2.scaled)
round(cor_df2,2)
```

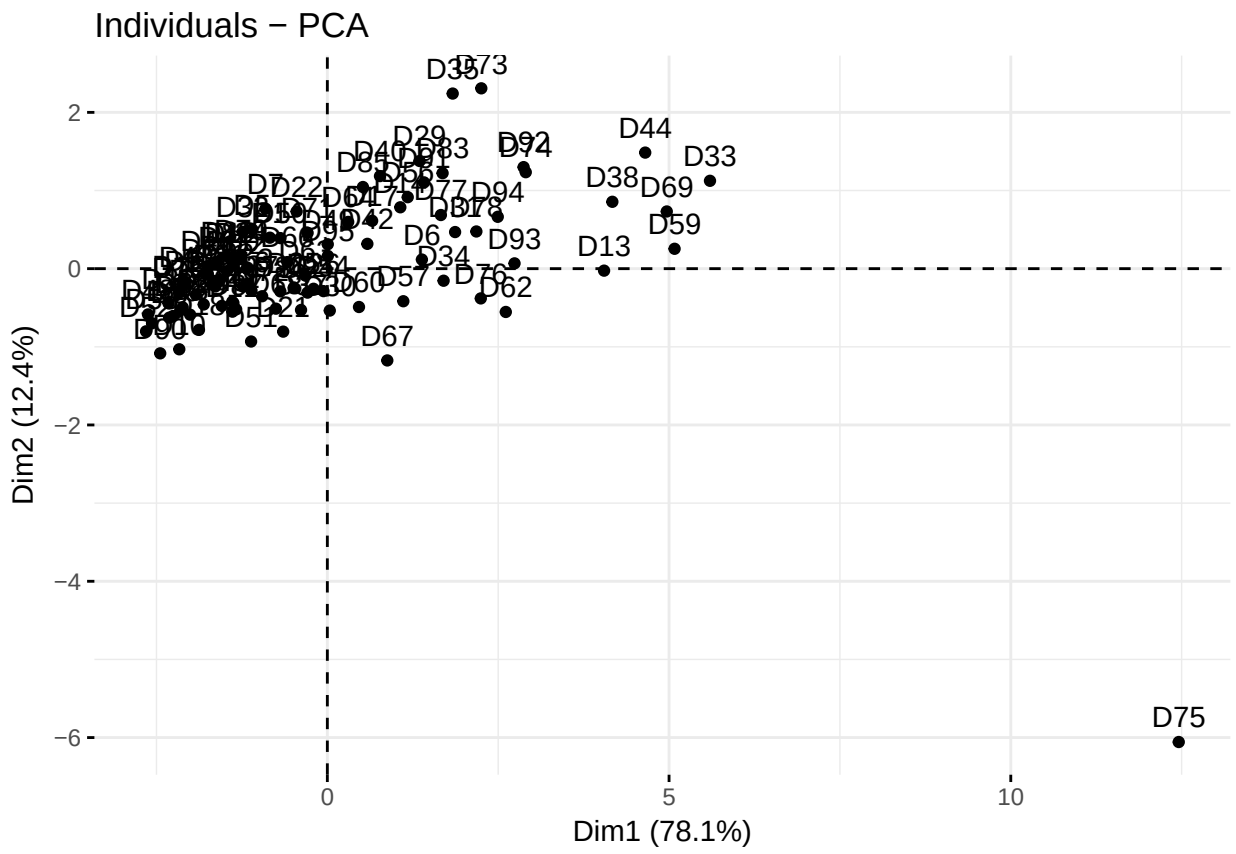
```
##       sean comm etab salle faut artes multi
## sean  1.00 0.27 0.80  0.96 0.94  0.72  0.70
## comm  0.27 1.00 0.75  0.49 0.53  0.64  0.52
## etab  0.80 0.75 1.00  0.91 0.91  0.85  0.67
```

```
## salle 0.96 0.49 0.91  1.00 0.99  0.79  0.79
## faut  0.94 0.53 0.91  0.99 1.00  0.80  0.81
## artes 0.72 0.64 0.85  0.79 0.80  1.00  0.55
## multi 0.70 0.52 0.67  0.79 0.81  0.55  1.00
```

```
res.pca <- PCA(DF2,  graph = FALSE)

fviz_pca_var(res.pca, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"))
```




```
fviz_pca_ind(res.pca)
```



5) Interprétation rapide des deux première composante principale du cercle de corrélation

Par rapport à la première composante principale F1: On remarque que les variables sont toute corrélées positivement à F1. On observe aussi que la variable “FAUT” représentant le nombre de fauteuil disponible et “SALLE” représentant le nombre de salles en activité sont les plus corrélés à F1. Donc, on peut supposer que la première composante principale représente le nombre de place vendu dans le département.

Par rapport à la seconde composante principale F2: On observe que la variable “COMM” qui représente le nombre de commune équipé de salle de cinéma est la plus corrélé. Donc, on peut supposer que la deuxième composante principale représente le nombre de communes par département

On peut noter que la composante principale F2 est compliqué à analyser car les vecteurs des variables sont éloignés de F2. L’interprétation de F1 est plus fidèle car la qualité de représentation est bonne pour toutes les variables et les coefficients de corrélation sont relativement élevé.

6) Observation et particularité de paris

On observe que la tendance est presque la même dans toutes les régions sauf paris. On remarque que la totalité des départements hormis Paris sont proche de l'origine de l'axe sauf Paris qui est très loin. D'après la question 2, nous avons déduis que le nombre de communes qui ont des salles de cinéma par département est très peu corrélé avec la recette du département. Donc on peut supposer que pour tous les départements, le nombre de commune est corrélé avec la recette du département sauf pour Paris. Ce qui explique la particularité de la ville de Paris.

7) Normalisation de l'ACP et intérêt

Tous les départements n'ont pas le même nombre de population. Diviser chaque donnée par sa population nous permettra d'avoir des données normées. Cela permettra d'avoir une analyse concentrée sur la tendance de la population selon les départements.

PARTIE 2 : ACP Version 2

8) Observation sur l'ACP version 2 et choix des axes principaux

Lorsque certains individus (département) ont des poids trop important, l'ACP normée permet d'avoir une comparaison plus juste et plus focalisée sur les variations de comportement pour chaque individu.

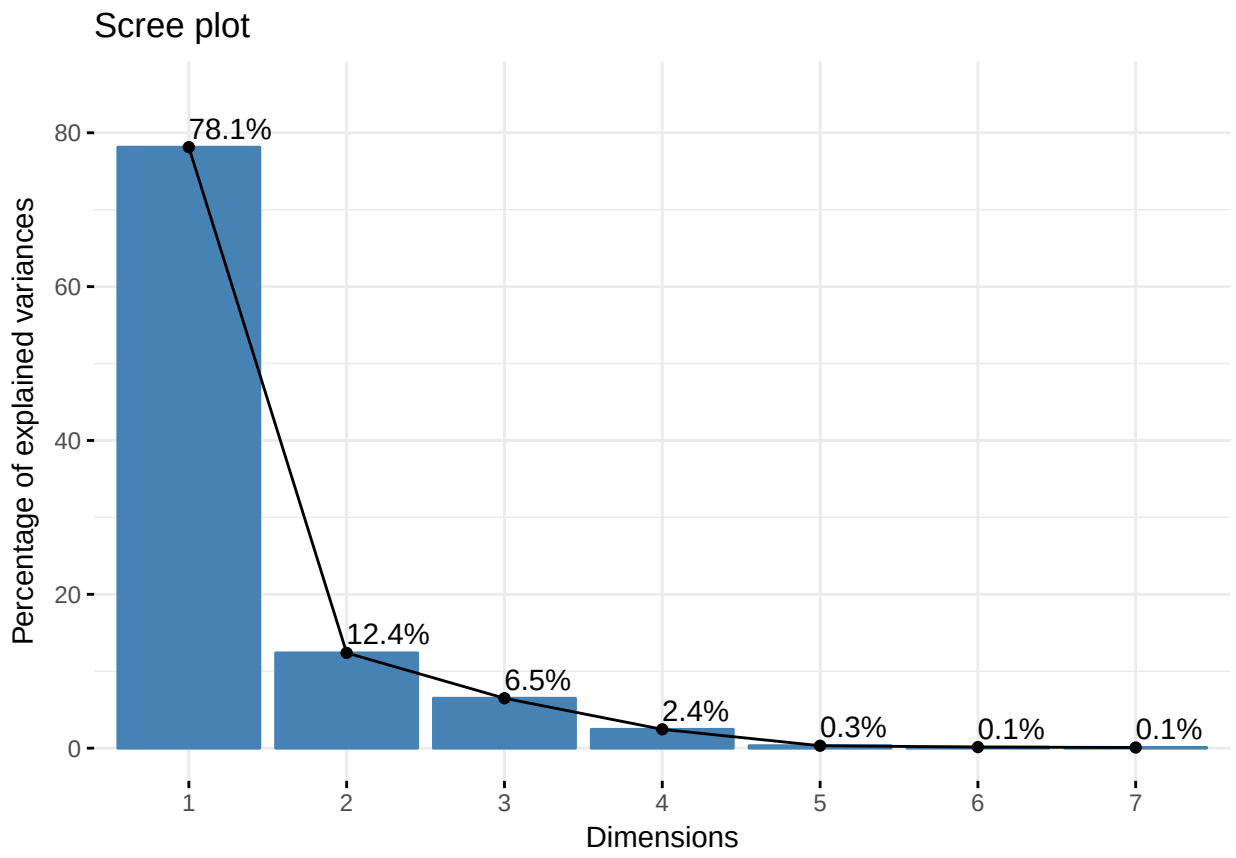
De plus, ici on retient les 4 premiers axes principaux sur les sept pour réduire le nombre de dimension et nous permettre de faire une analyse plus précise. De plus, les 4 dimensions nous permet déjà d'obtenir plus de 99% des contributions.

Oui, la situation semble meilleur car le poids des départements et les coordonnées sur les axes de Paris est moins disparate que pour l'ACP non normée.

9) Choix des axes et critères choisies en fonction des variables

Les valeurs propres peuvent être utilisées pour déterminer le nombre d'axes principaux à conserver après l'ACP. Il n'existe pas de méthode objective bien acceptée pour décider du nombre d'axes principaux qui suffisent. Pour savoir quels axes nous retenons, nous décidons de se référer au « critère de Kaiser ». Celui-ci nous dit que pour des variables centrées-réduites normée, nos choix d'axes doivent se porter sur celles qui ont des valeurs propres supérieures à 1. Nous allons donc retenir l'axe 1 et 2 qui totalisent plus de 82% (3.71+2.04/7) des contributions, ce qui déjà pas mal. D'après le cours, l'effet de taille est vérifié lorsque toutes les variables ont le même signe de corrélation avec la première composante principale. Dans notre exemple, on peut voir que toutes les valeurs de corrélations des premières composantes sont négatives, on peut donc dire qu'il y'a un « effet de taille ». Les variables sont toutes du même côté de l'axe.

```
library("factoextra")
library("FactoMineR")


fviz_eig(res.pca, addlabels = TRUE, ylim = c(0, 85))
```




```
DF1$sean <- DF1$sean*1000

DF3<-DF2/(DF1$popu)

DF3.scaled <- scale(DF3, center = TRUE, scale = TRUE)

res.cor <- cor(DF3.scaled)
round(res.cor, 2)
```

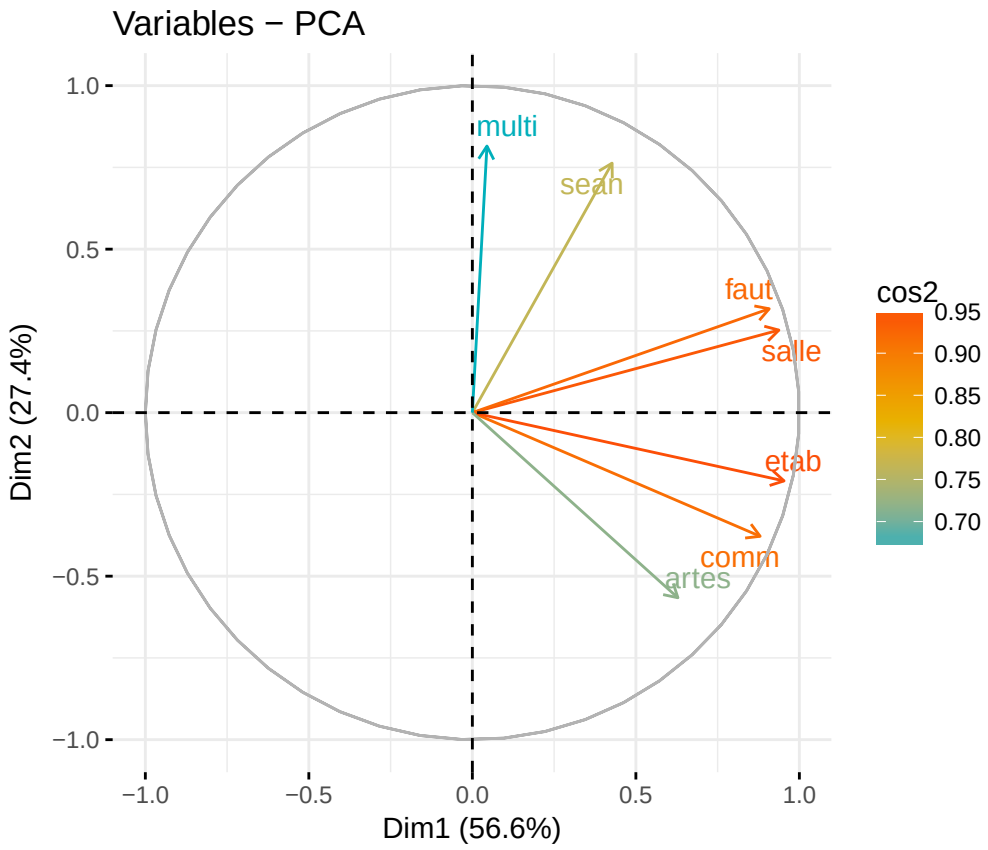
```
##        sean  comm  etab salle faut artes multi
## sean   1.00  0.01  0.21  0.60 0.58 -0.08  0.45
## comm   0.01  1.00  0.97  0.70 0.66  0.69 -0.18
## etab   0.21  0.97  1.00  0.83 0.78  0.64 -0.10
## salle  0.60  0.70  0.83  1.00 0.94  0.41  0.18
## faut   0.58  0.66  0.78  0.94 1.00  0.34  0.27
## artes -0.08  0.69  0.64  0.41 0.34  1.00 -0.34
## multi  0.45 -0.18 -0.10  0.18 0.27 -0.34  1.00
```

```
res.eig<-eigen(res.cor)
rn <- rownames(res.cor)
rownames(res.eig$vectors) <- rn
colnames(res.eig$vectors) <- c("Dim1", "Dim2", "Dim3", "Dim4", "Dim5", "Dim6", "Dim7")
res.eig$vectors
```

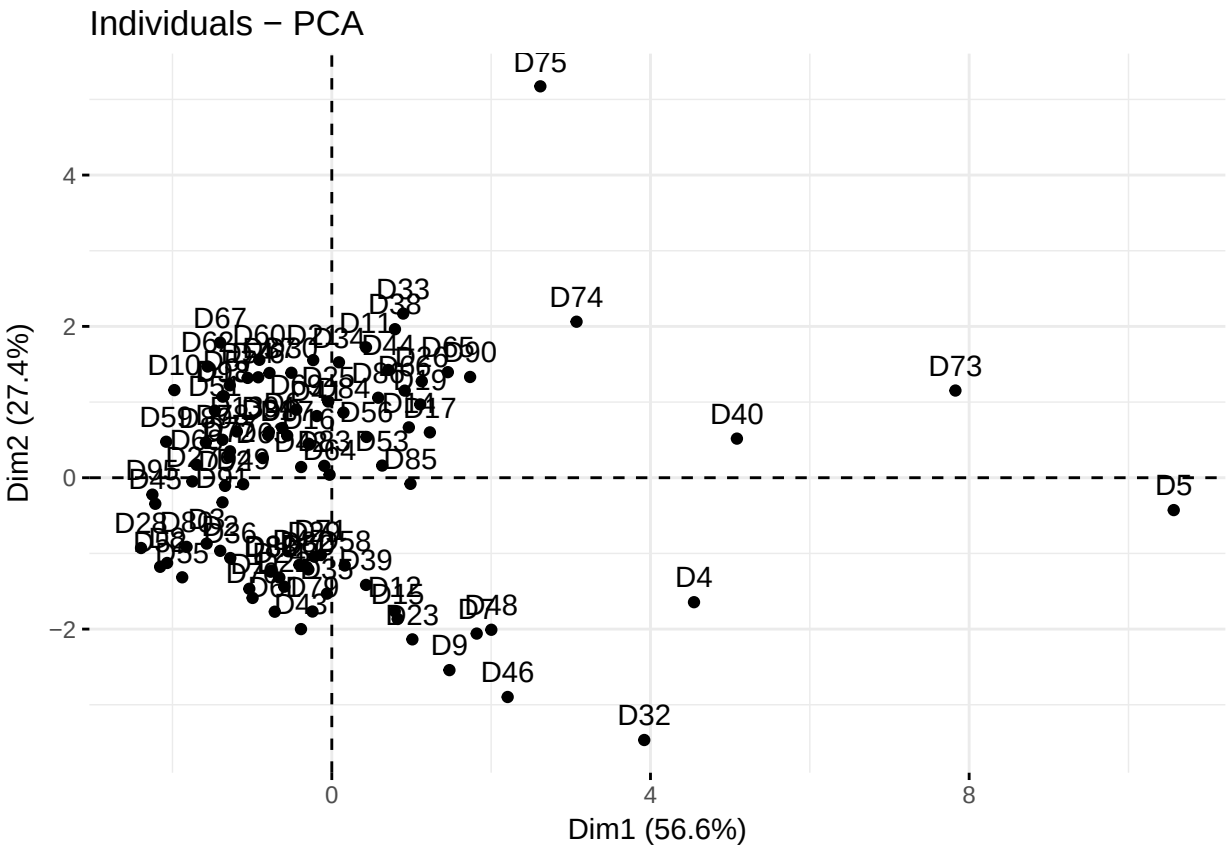
```
##              Dim1       Dim2        Dim3       Dim4        Dim5
## sean  -0.21457008  0.5508859 -0.57007884  0.2950323  0.45520217
## comm  -0.44193370 -0.2729324  0.26783483 -0.2026953  0.42070483
```

```
## etab  -0.47863357 -0.1506656  0.11719123 -0.2045266  0.37217065
## salle -0.47072457  0.1822540 -0.14278273 -0.1365608 -0.29147699
## faut  -0.45588753  0.2293742 -0.01862538 -0.1810587 -0.59922011
## artes -0.31567014 -0.4086986  0.02117525  0.8390957 -0.16774118
## multi -0.02239434  0.5886161  0.75389564  0.2729959  0.07160137
##               Dim6        Dim7
## sean   0.152390930 -0.08954556
## comm   0.118385714 -0.65297313
## etab   0.030438500  0.74377911
## salle -0.781377663 -0.10322885
## faut   0.589096758 -0.01803217
## artes  0.003558687  0.02526118
## multi -0.065089308  0.02794347
```

```
res.pca2 <- PCA(DF3.scaled,  graph = FALSE)

fviz_pca_var(res.pca2, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE)
```




```
fviz_pca_ind(res.pca2)
```



10) Choix des axes et critères utilisées en fonction des départements

Les départements qui déterminent les axes que l'on retient sont ceux pour lesquelles les valeurs sur un ou plusieurs axes sont les plus élevé en valeur absolue. Le critère que l'on utilise se base sur les valeurs extrêmes sur les axes. Donc, on voit que pour l'axe 1 les départements 5 avec -13.55, 73 avec -10.33 et pour l'axe 2 les départements 32 avec 5.55, 46 avec 4.45 sont ceux qui permettent de choisir les axes sur lesquels projeter nos données. On suppose que les départements surreprésentés sont ceux avec le poids le plus élevé, donc on peut dire que Paris est toujours surreprésenté.

11) Tentative d'interprétation des axes

Nous voyons dans la matrice de corrélation par rapport à la composante que pour l'axe 1, ce sont les variables etab, "SALLE" et "FAUT" qui semblent le plus corrélé. Or, "ETAB" représente le nombre de cinéma en

	Comp1	Comp2	Comp3	Comp4
sean	-0.50	-0.75	-0.41	0.04
comm	-0.68	0.64	0.30	-0.12
etab	-0.91	0.34	0.08	-0.17
salle	-0.93	-0.30	-0.09	-0.09
faut	-0.92	-0.33	-0.01	-0.08
artes	-0.64	0.54	-0.21	0.50
multi	-0.22	-0.68	0.66	0.24

Figure 5: Matrice de corrélation

activité, salle, le nombre de salle en activité et “FAUT” le nombre de fauteuil disponible.

Ainsi, on peut supposer que F1 représente l’activité cinématographique de la population d’un département.

Nous voyons au contraire que pour la composante F2, les variables les plus corrélé positivement sont sean, “COMM” et “MULTI” . Elles représentent respectivement le nombre de séance (en millions), le nombre de communes disposant de salle de cinéma et le nombre de multiplexe dans le département.

Ainsi, on peut supposer que l’axe 2 représente le nombre de projections de films par département.

12) Qualité de représentation et critères utilisées

Les axes retenus sont les axes de la composante principale 1 et 2, on doit donc chercher les valeurs qui sont mal représenté dans la colonne Axis 1 : 2. C’est le plan d’inertie où les données seront projetées. Nous voyons clairement que c’est le département de Mayenne, Belfort et Val-de-Marne qui sont les moins bien représenté. Le critère utilisé est la qualité de représentation.

Conclusion

L’ACP ou l’analyse de composante principale nous a aidé à résumer et interpréter les data set contenant plusieurs variables avec plus ou moins de précisions.

Elle nous a permis de réduire les dimensions d’un data set contenant plusieurs variables puis visualisér graphiquement, en perdant le moins possible d’information.

Dans l’étude de composante principal, nous avons vu que l’information contenue dans les données peut être interprété par un ensemble de multiple variation de valeurs.

Toutefois, il faut noter que l'ACP reste une méthode de projection, et que la perte d'information induite par la projection peut entraîner des interprétations erronées. De plus, l'ACP est un outil de statistique exploratoire et il n'est donc pas possible d'émettre des postulats ou de tester des hypothèses.