# Enquette Budget Temps

## Langage R et Analyse de donnée

**Etienne Sharpin**
**Jose A. Henriquez Roa**

May 27, 2021

# Contents

# 1 Description du jeu de données

Ce jeu de données représente une enquête mettant en valeur le temps passé dans multiples activités au cours d'une journée (Budget/Temps).

La population étudiée provient de plusieurs pays différents et porte sur des hommes et des femmes de différentes situations.

# 2 Chargement des données

```
path = "./Enquete_Budget_Temps.xlsx"
library("readxl")
df01 <- read_excel(path)

summary(df01)
```

On charge les données depuis le fichier excel.

# 3 Normalisation des distributions

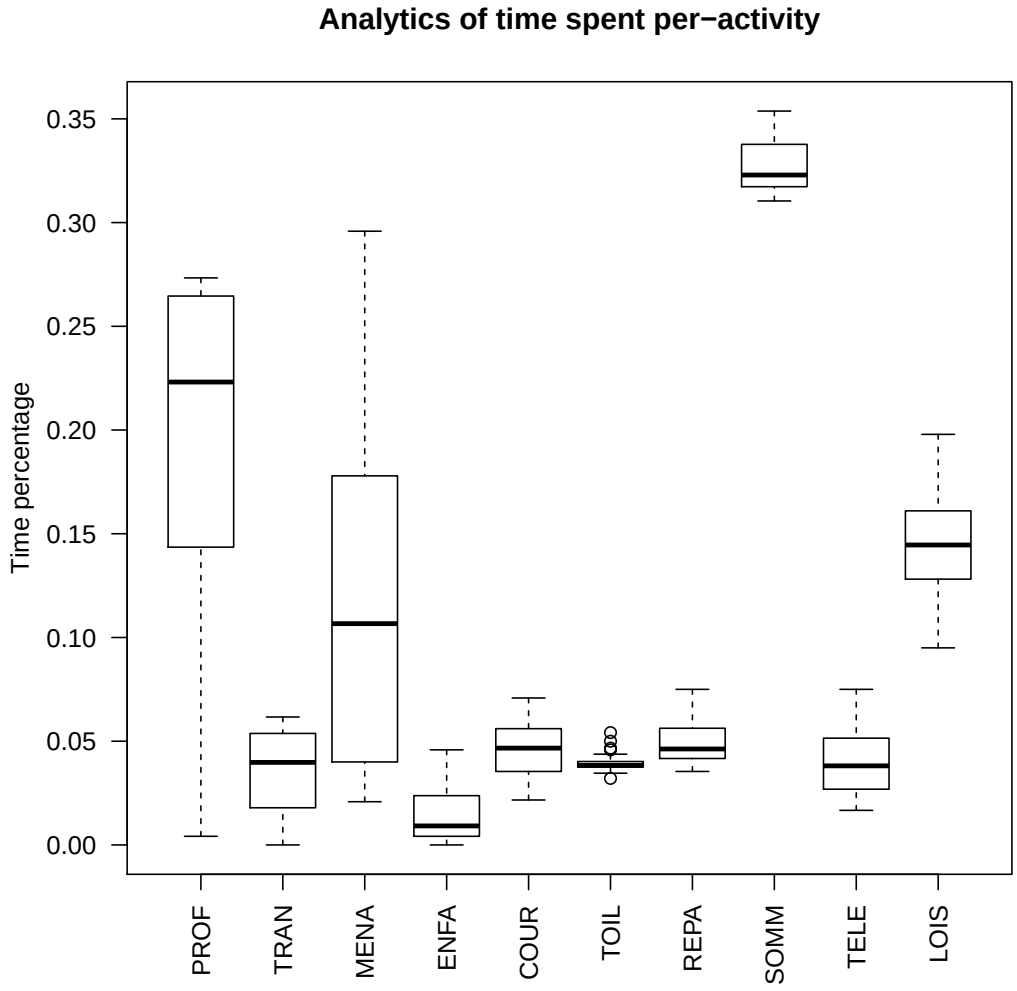
```
df02 <- df01
df02[, 2:11] <- df01[, 2:11] / 2400
```

On normalise les données en divisant chacune des valeurs numériques par 2400 ce qui nous permet une meilleur distribution afin d'obtenir de meilleurs résultats lors des différentes analyses statistiques que nous allons effectuer par la suite.

# 4 Analyse des boîtes à moustaches

```r
df03 <- df02[2:11]

pdf("plot/box-plot.pdf")
boxplot(df03, use.cols=FALSE, las=2, ylab="Time percentage", main="Analytics of time spent per-activity")
dev.off()
```



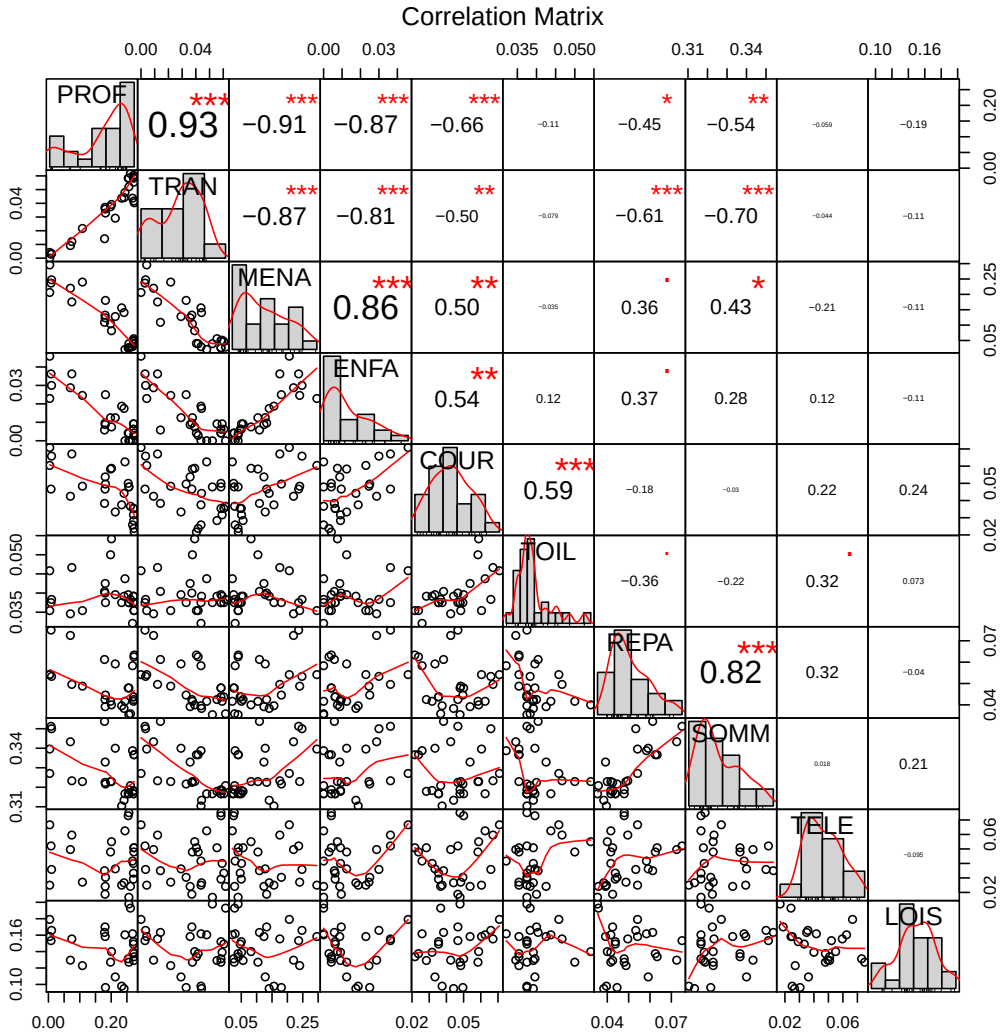

Le diagramme en boîte à moustache est très utile notamment pour observer la dispersion des valeurs sur certaines variables. Ainsi, nous pouvons voir que certaines activités comme aller aux toilettes qui sont du ressort de la nécessité, sont une activité ayant très très peu de variance. Effectivement, chaque individus quelque soit sa classe a les mêmes besoins. On peut également l'observer au niveau du sommeil ou des repas, où l'écart est faible avec des valeurs relativements centrées.

Les valeures de certaines activitées sont bien plus dispersées, comme c'est le cas pour le travail ou le ménage et possède des temps bien plus conséquents que les autres activitées car étant considérées comme essentielles d'y consacrer du temps de vie.

# 5 Matrice de corrélation

```
pdf("plot/correlation-matirx.pdf")
library("PerformanceAnalytics")
chart.Correlation(df03, histogram=TRUE, title="hello")

mtext("Correlation Matrix", side=3, line=3)
dev.off()
```




Cette matrice de corrélation met en évidence bon nombres de facteurs, typiquement, avec une corrélation casi linéaire positive de 0.93 entre le temps consacré au travail et celui passé dans les transports. En effet pour aller travailler il faut se déplacer.

L'activité professionnelle étant très chronovore, elle est également en corrélation négative avec le ménage ou les enfants, en effet lorsque l'on est sur son lieu de travail, on est pas chez soi . Elle l'est d'ailleurs avec la plupart des autres activitées comme les activitées primaires même si elle influe également sur le temps de sommeil, en effet le travail pose des contraintes de sommeil plus stricts pouvant conduire les gens à moins dormir.

# 6 Carte Radar

Pour comparer le temps alloué aux différentes tâches entre les individus de différents groupes, nous avons choisi la méthode du graphique radar. Le code suivant suivant montre comment nous avons tracé ces graphiques pour les 28 groupes, présentés dans l'annexe. Un graphique radar contenant tous les groupes est présenté à la fin de cette section.

```
library(fmsb)

pdf("plot/radar-chart-all.pdf")
df04 <- rbind(rep(max(df03),28), rep(min(df03),28), df03)
radarchart(df04, title="Per-Instance comparison")
dev.off()

for (i in 1:28) {
    pdf(paste("plot/radar-chart/radar-chart-", tolower(df01$ID[i]), ".pdf", sep=""))
    df05 <- rbind(rep(max(df03),28), rep(min(df03),28), df03[i,])
    radarchart(df05, title=df01$ID[i])
    dev.off()
}
```

For this we have used the library *fmsb* which includes the function *radarchart(...)* from which the plots were drawn. Figure 1 shows a comparison of time distributions between all groups present in the dataset. The individual radar charts can be found in the annex 11.

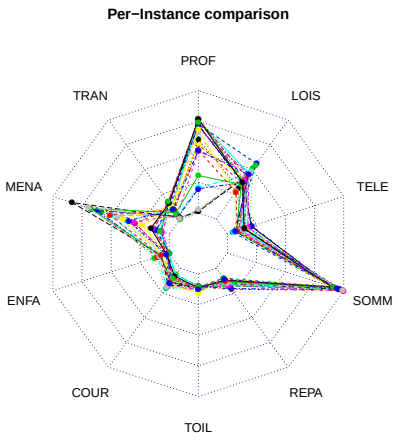



Figure 1: Comparaison entre tous les groupes de profils

# 7 Graphique à barres

Étant donné que les graphiques radar qui intègrent des valeurs de référence ont tendance à avoir une allure plutôt encombrée, nous avons également choisi de visualiser la distribution du temps à l'aide de diagrammes à barres, dans lesquels nous avons inclus les pourcentages de temps sur l'axe des ordonnées.

```
for (i in 1:28) {
    pdf(paste("plot/bar-plot/bar-plot-", tolower(df01$ID[i]), ".pdf", sep=""))
    barplot(t(as.matrix(df03[i,])), beside=TRUE, main=df01$ID[i],
            ylab="Time percentage", names.arg=colnames(df03), las=2)
    dev.off()
}
```

Les graphiques générés sont dans l'annexe 11

# 8 ACP

La ligne suivante effectue l'ACP que nous utilisons dans la section suivante de clustering pour la visualisation des résultats.

```
  pca = prcomp(df03)
```

Les pourcentages de la variance expliquée pour chaque composante principale sont obtenus par la commande suivante.

```
  summary(pca)$importance[2,]
```

qui produit le résultat suivant:

```
PC1: 0.8804 PC2: 0.07166 PC3: 0.02654 PC4: 0.01521 PC5: 0.0032 PC6: 0.00158
PC7: 0.00072 PC8: 0.00043 PC9: 0.00026 PC10: 0
```

Comme on peut le voir un peu plus que 95% de la variance est expliquée par les deux premières composantes principales. Ce qui signifie que la projection bidimensionnelle des données sera d'assez bonne qualité. Le graphique suivant est une représentation graphique de la sortie ci-dessus.

```
pdf("plot/principal-component-explained-variance.pdf")
barplot(summary(pca)$importance[2,],ylab="Explained Variance Proportion",
        ylim=c(0,1), main="Principal Component Explained Variance")
dev.off()
```

Le graphique résultant peut être vu dans la figure 2.

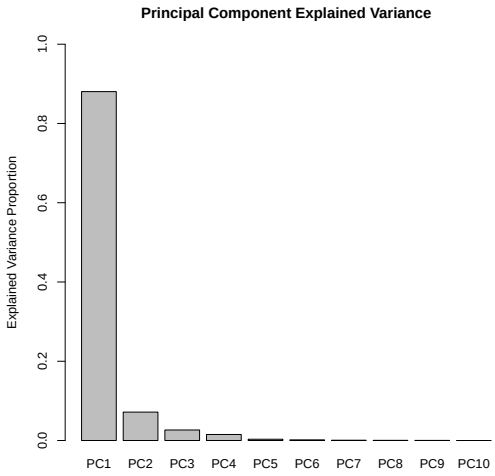



Figure 2: Principal component explained variance

L'extrait de code suivant montre la projection de tous les attributs dans les deux premières composantes principales.

```
pdf("plot/pca-attribute-projection.pdf")
library("factoextra")
fviz_pca_var(pca, col.var = "cos2", col.ind = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"))
dev.off()
```

Comme le montre la figure 3. Dans celle-ci, nous voyons que les deux attributs les mieux projetés sont de loin les attributs PROF et MENA, correspondant respectivement aux attributs désignant le temps passé à travailler et de nettoyage.

Puis, la visualisation de la projection de l'instance a été réalisée à travers le code suivant.

```
pdf("plot/pca-instance-projection.pdf")
library("factoextra")
fviz_pca_ind(pca, col.var = "cos2", col.ind = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"))
dev.off()
```

Le graphique résultant est dans la figure 3.

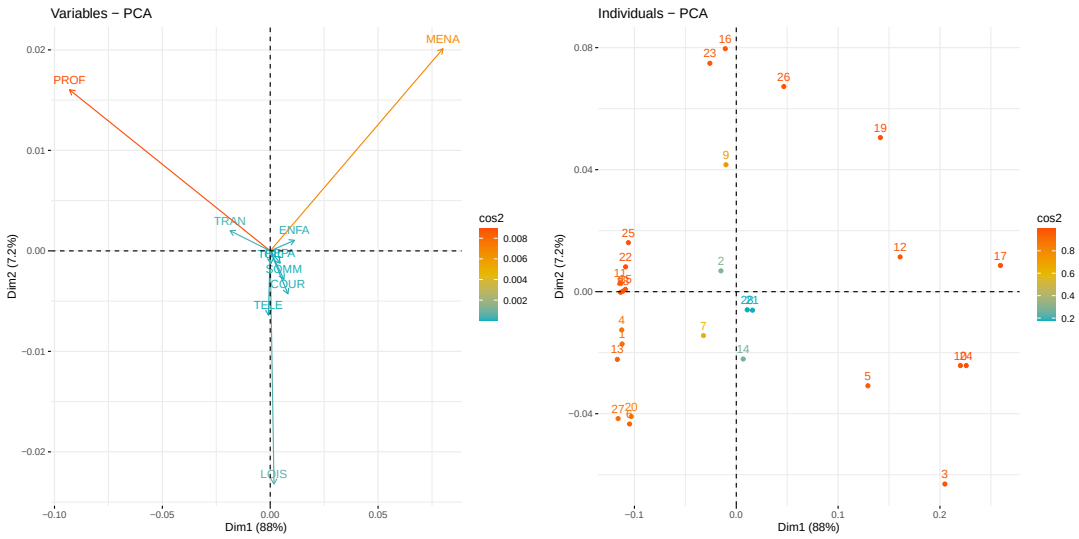



Figure 3: projection d'attributs (à gauche) et projection d'instances (à droite)

D'après les pourcentages de variance expliquée, nous constatons que la plupart des instances ont une bonne qualité de projection.

# 9 Clustering

Pour analyser les similitudes entre les profils, nous avons regroupé les instances à l'aide de l'algorithme d'apprentissage automatique KMeans. Cet algorithme nécessite d'abord de définir manuellement le nombre de clusters dans lesquels les instances seront regroupées. Ceci est fait dans le code suivant.

```
pdf("plot/kmeans-inertia.pdf")
fviz_nbclust(df03, kmeans, method="wss")
dev.off()
pdf("plot/kmeans-silhouette.pdf")
fviz_nbclust(df03, kmeans, method="silhouette")
dev.off()
pdf("plot/kmeans-gap-stat.pdf")
fviz_nbclust(df03, kmeans, method="gap_stat")
dev.off()
```

Les trois graphiques sont présentés dans la figure 4. Dans ceux-ci, la courbe de score Silhouette et la statistique d'écart montrent que deux sont une bonne quantité de clusters. Alors que la courbe du coude montre également que trois est une bonne option. Ainsi, en prenant tous les résultats en compte, nous avons choisi deux clusters pour les suivants analytique.

Enfin, le clustering lui-même a été effectué avec l'extrait de code suivant.

```
pdf("plot/clusters.pdf")
km = kmeans(df03, centers=2, nstart=25)
fviz_cluster(km, data=df03)
dev.off()
```

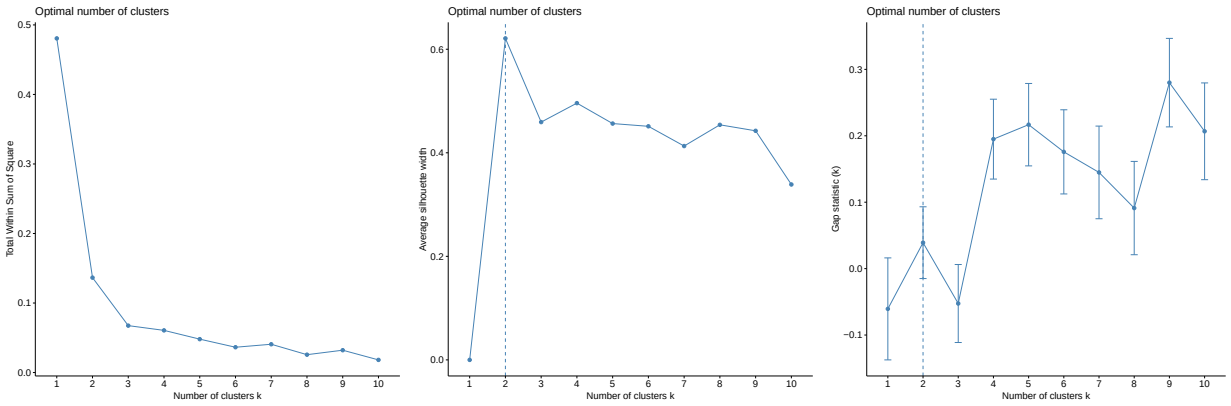



Figure 4: Courbe du coude (droite), scores de la silhouette (centre), statistiques de l'écart (droite)

Le résultat est dans la figure 5. Pour le nombre d'instances par cluster, nous avons.

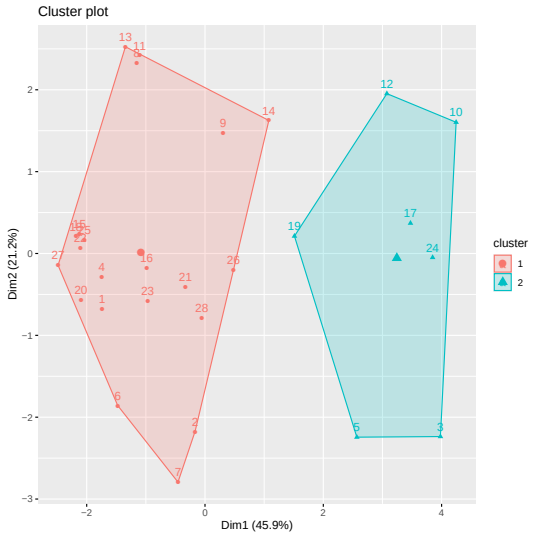



Figure 5: Clustering

```
print(paste("cluster01:", nrow(df01[km$cluster==1,])))
print(paste("cluster02:", nrow(df01[km$cluster==2,])))
```

ce qui donne le résultat suivant:

```
[1] "cluster01:  21"
[1] "cluster02:  7"
```

Nous constatons qu'il n'y a pas un nombre uniforme d'instances par groupe, ce qui signifie que l'algorithme ML a travaillé sur un certain type d'intuition obtenue à partir des données lors du regroupement.

Ensuite, pour voir à quel point les instances d'un même cluster sont réellement similaires, nous échantillonnons deux instances de clusters différents puis deux instances du même cluster et nous

les comparons. La figure suivante montre d'abord la comparaison entre deux instances de clusters différents, puis la comparaison de deux instances du même cluster.

```
pdf("plot/fne-hme-comparison.pdf")
instanceIndices = c(24,25)
df04 <- rbind(rep(max(df03),28), rep(min(df03),28), df03[instanceIndices,])
title = paste(df01$ID[instanceIndices[1]],"-",df01$ID[instanceIndices[2]],
              "Comparison", sep=" ")
radarchart(df04, title=title)
dev.off()
pdf("plot/hme-fay-comparison.pdf")
instanceIndices = c(25,16)
df04 <- rbind(rep(max(df03),28), rep(min(df03),28), df03[instanceIndices,])
title = paste(df01$ID[instanceIndices[1]],"-",df01$ID[instanceIndices[2]],
              "Comparison", sep=" ")
radarchart(df04, title=title)
dev.off()
```

Le résultat est dans la figure 6.


Figure 6: Comparaison par cluster

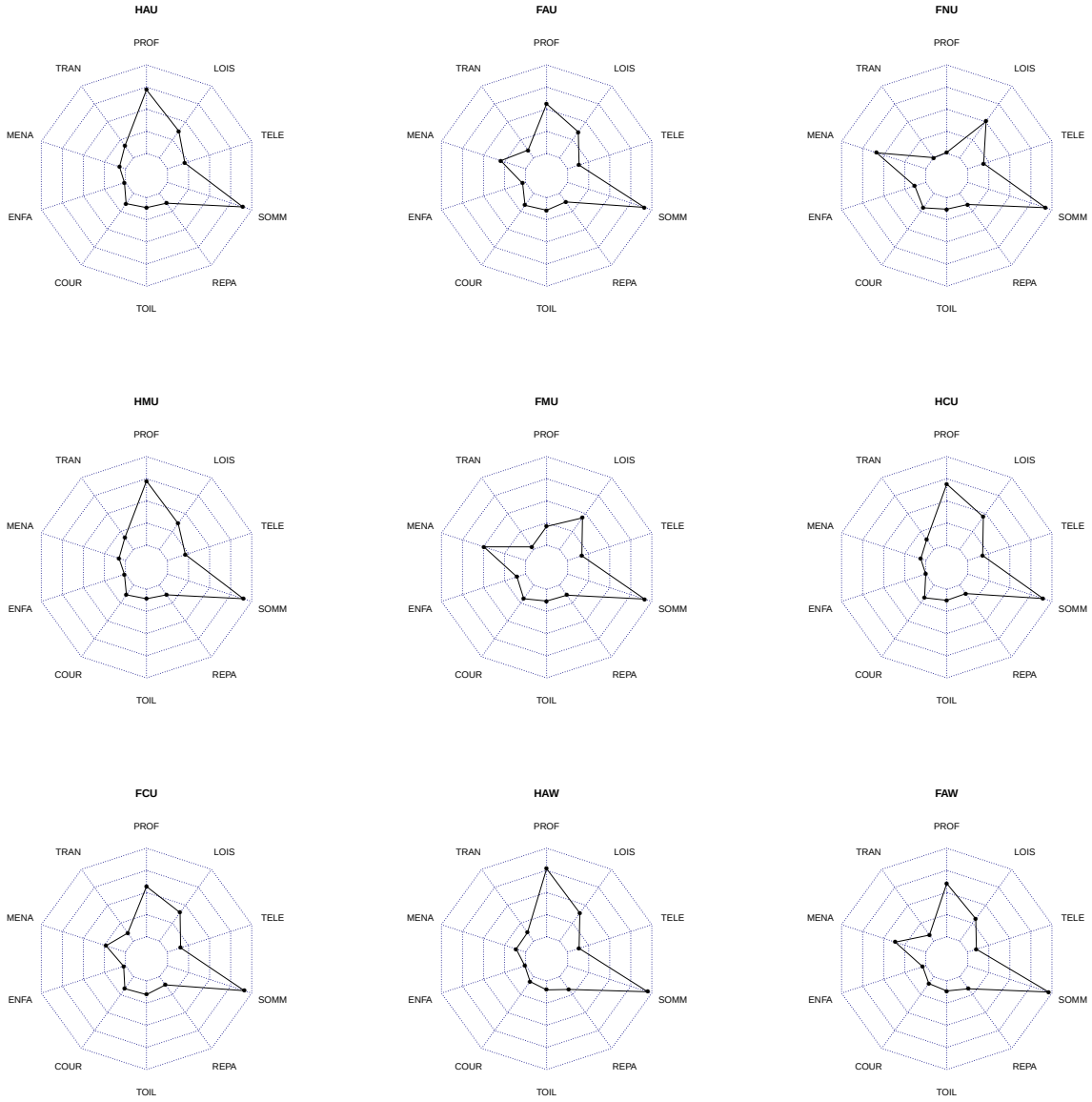
Dans le graphique de gauche, nous voyons qu'entre ces deux cas, il y a une différence considérable entre le temps passé à nettoyer et à travailler. Alors que le cas le plus à droite est assez similaire tout du long.

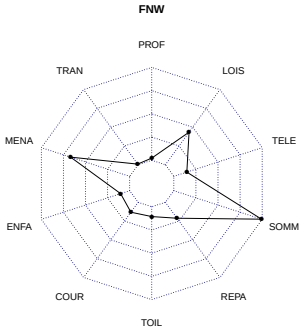
## 10 Critiques

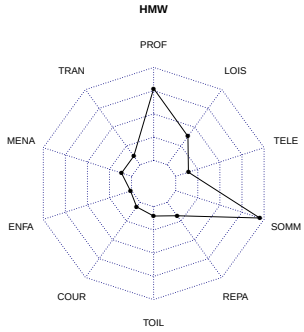
La principale critique à laquelle je peux penser est liée au fait que les données ont déjà été traitées et que les individus ont déjà été regroupés comme indiqué dans la description des données. Ce qui nous laisse avec la seule option de faire une analyse secondaire, contenant tous les possibles biais de l'analyse précédente.

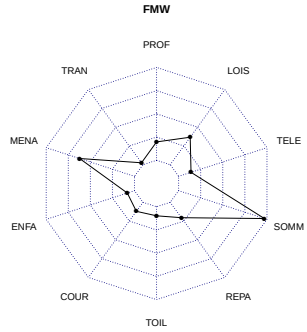
# 11 Annex

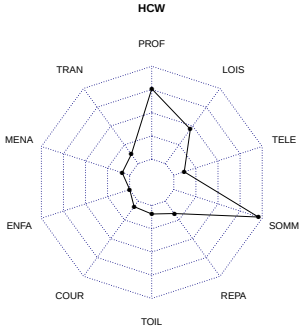
## 11.1 Carte Radar

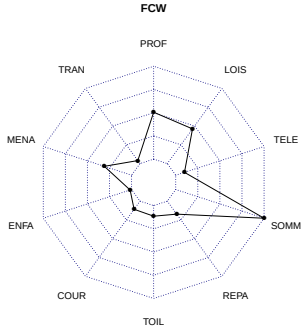


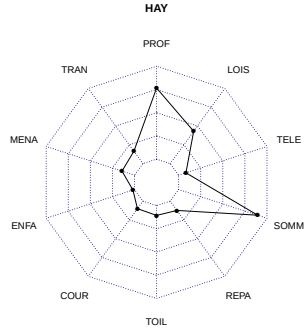

FNW
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

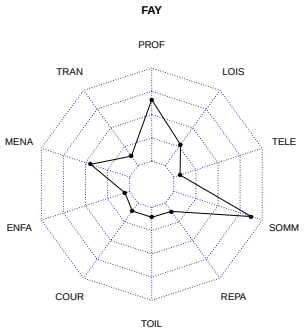

HMW
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

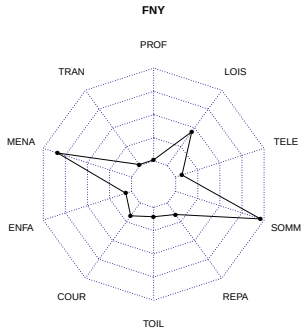

FMW
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

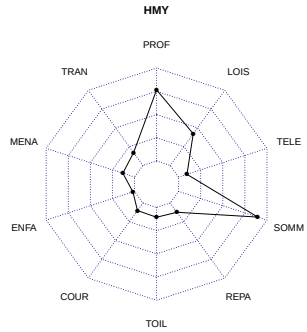

HCW
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

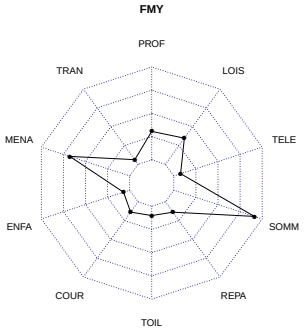

FCW
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

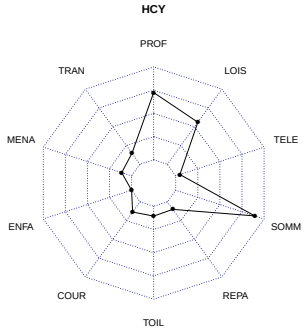

HAY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

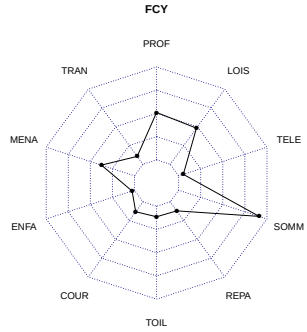

FAY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN


FNY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

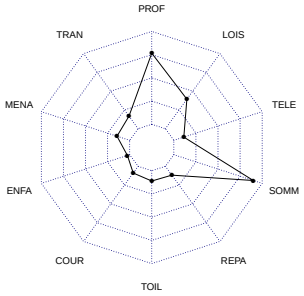

HMY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN


FMY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

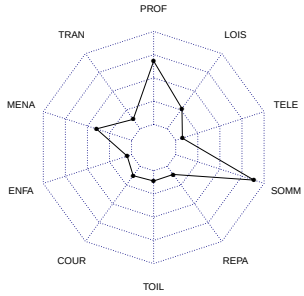

HCY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN


FCY
PROF
LOIS
TELE
SOMM
REPA
TOIL
COUR
ENFA
MENA
TRAN

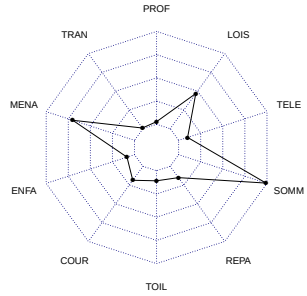
**HAE**



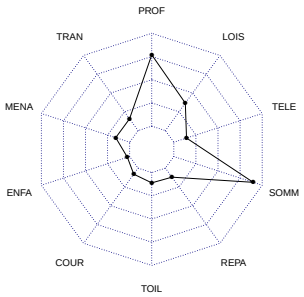

**FAE**



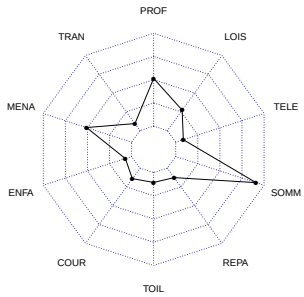

**FNE**



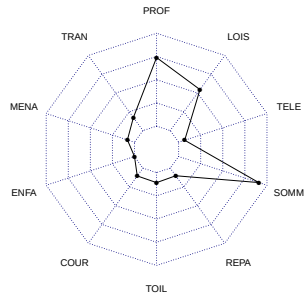

**HME**



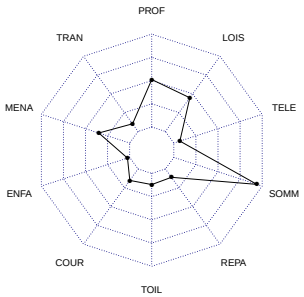

**FME**



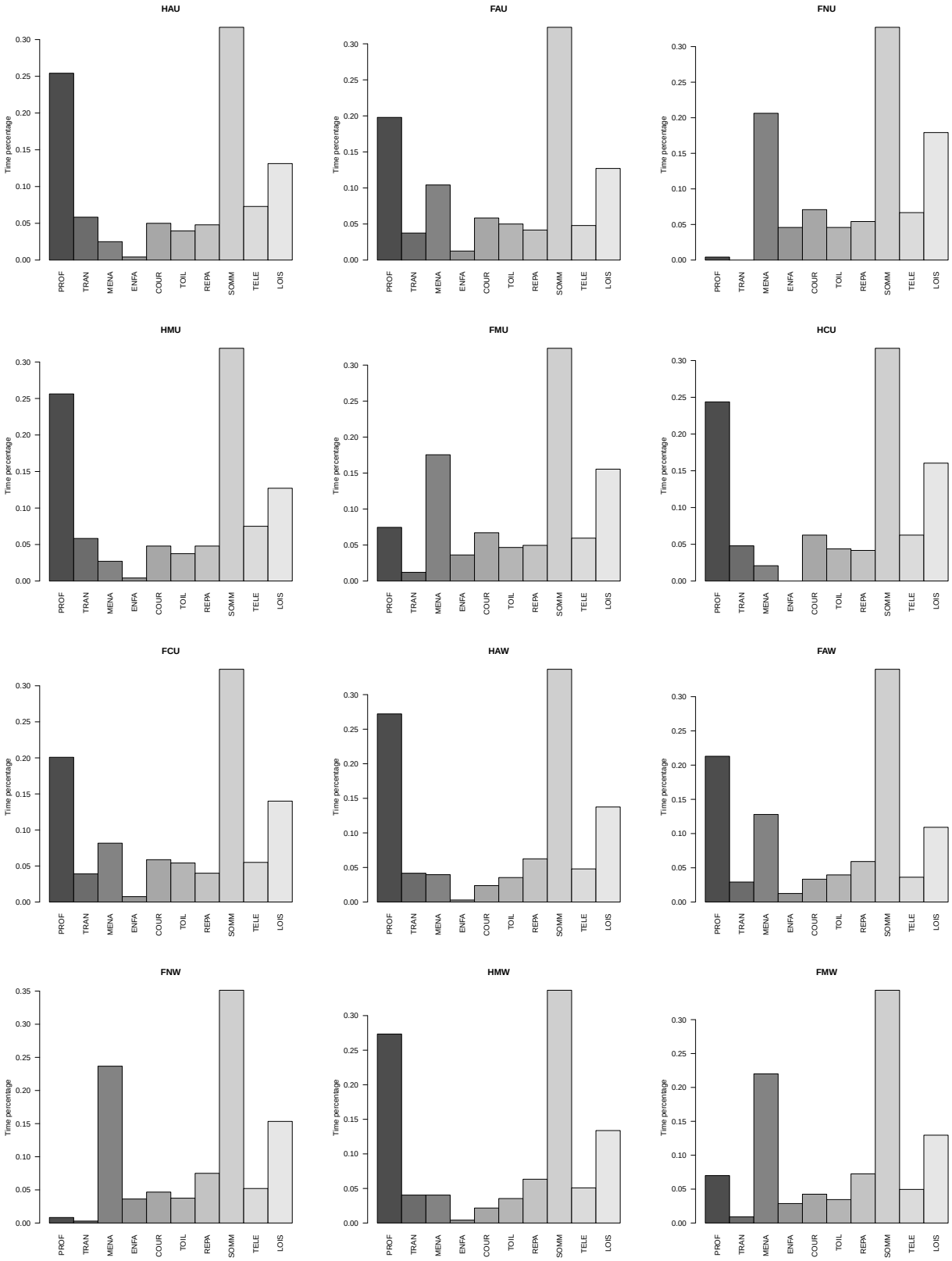

**HCE**

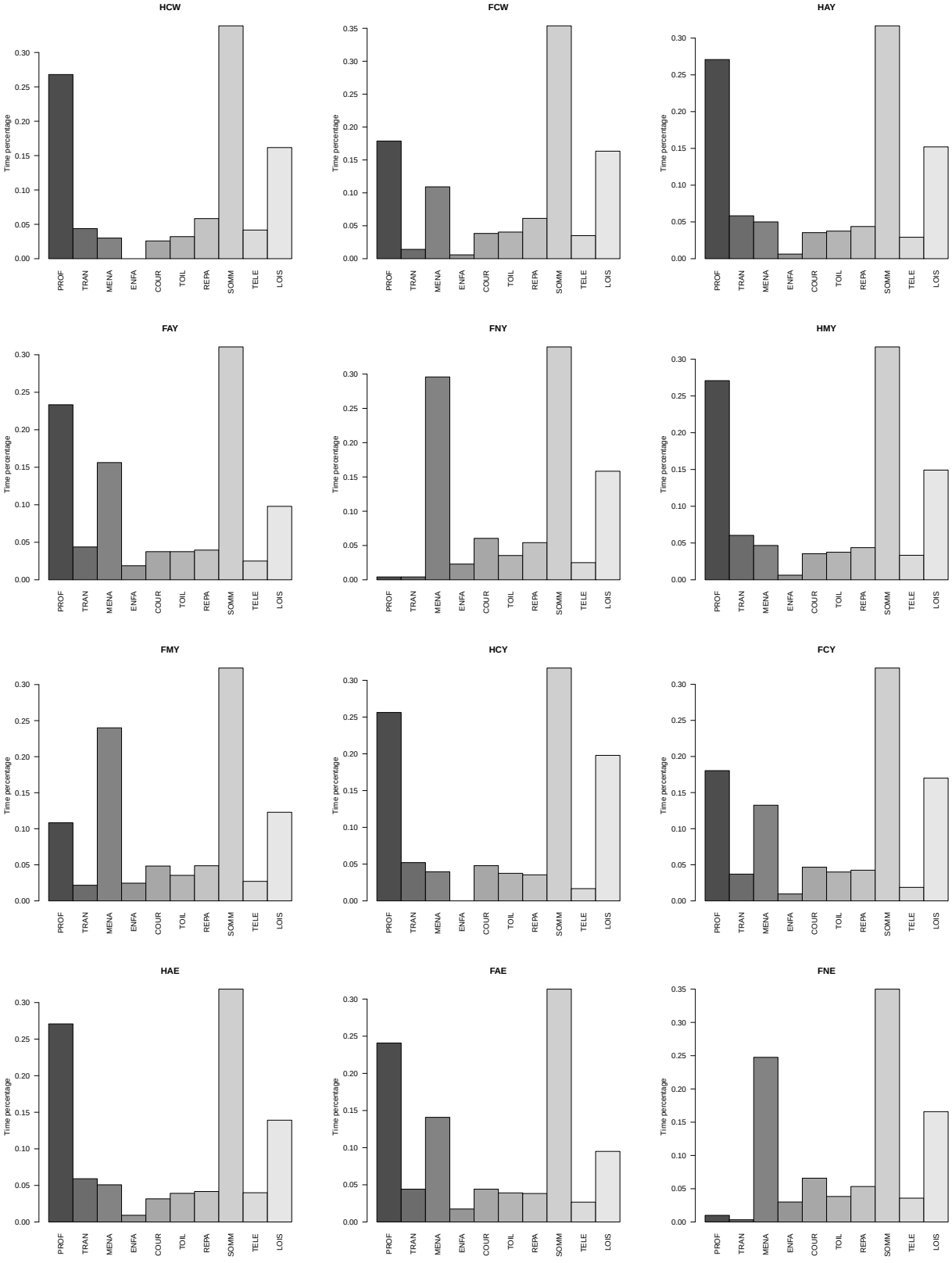


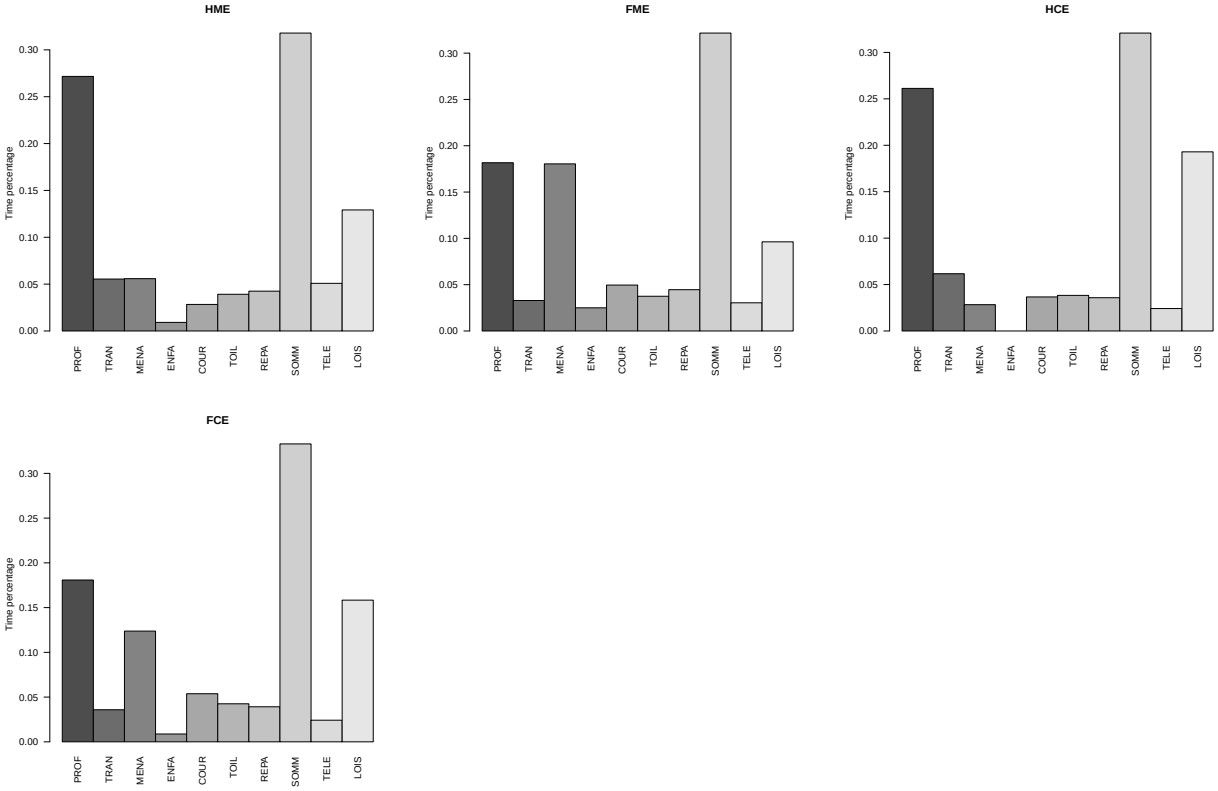

**FCE**



## 11.2 Graphique à barres




HCW
FCW
HAY
FAY
FNY
HMY
FMY
HCY
FCY
HAE
FAE
FNE
Time percentage
PROF
TRAN
MENA
ENFA
COUR
TOIL
REPA
SOMM
TELE
LOIS


HME
FME
HCE
FCE
Time percentage
0.00
0.05
0.10
0.15
0.20
0.25
0.30
PROF
TRAN
MENA
ENFA
COUR
TOIL
REPA
SOMM
TELE
LOIS