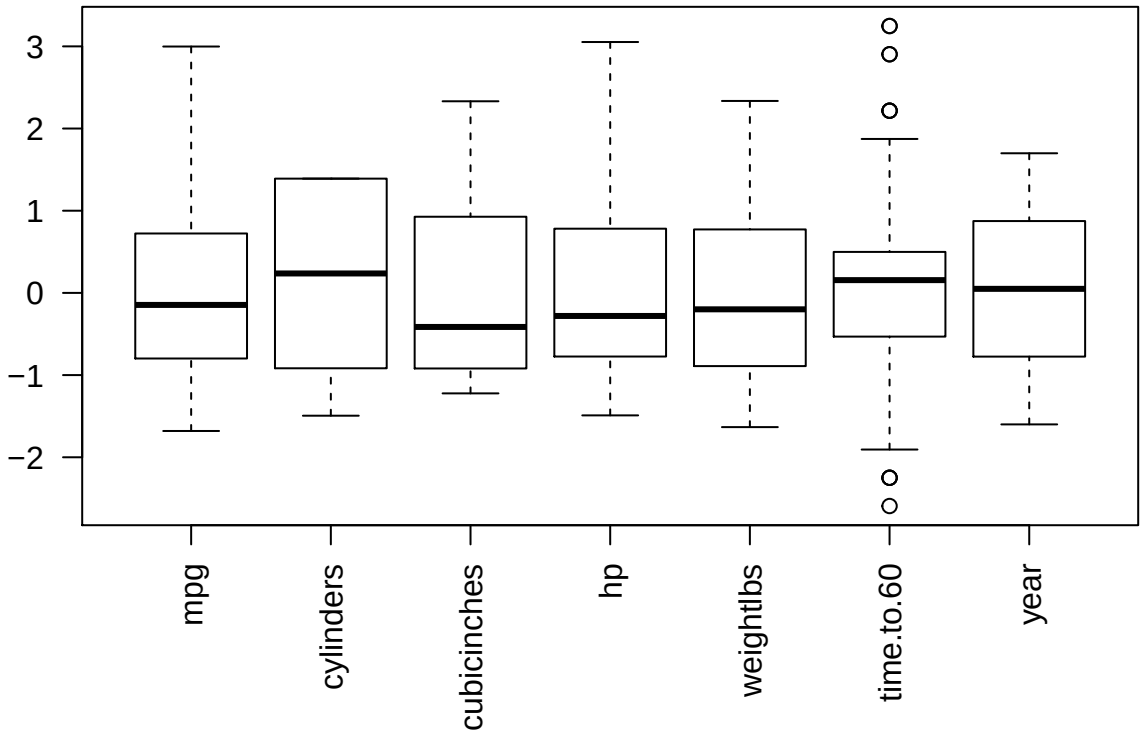
Figures

Yann Trividic

```
par(mar=c(7,3,2,2))
boxplot(cars.scaled[-8],
        las=2,
        main="Fig. 1 : Diagrammes en boîte des variables quantitatives de cars")
```

Fig. 1 : Diagrammes en boîte des variables quantitatives de cars

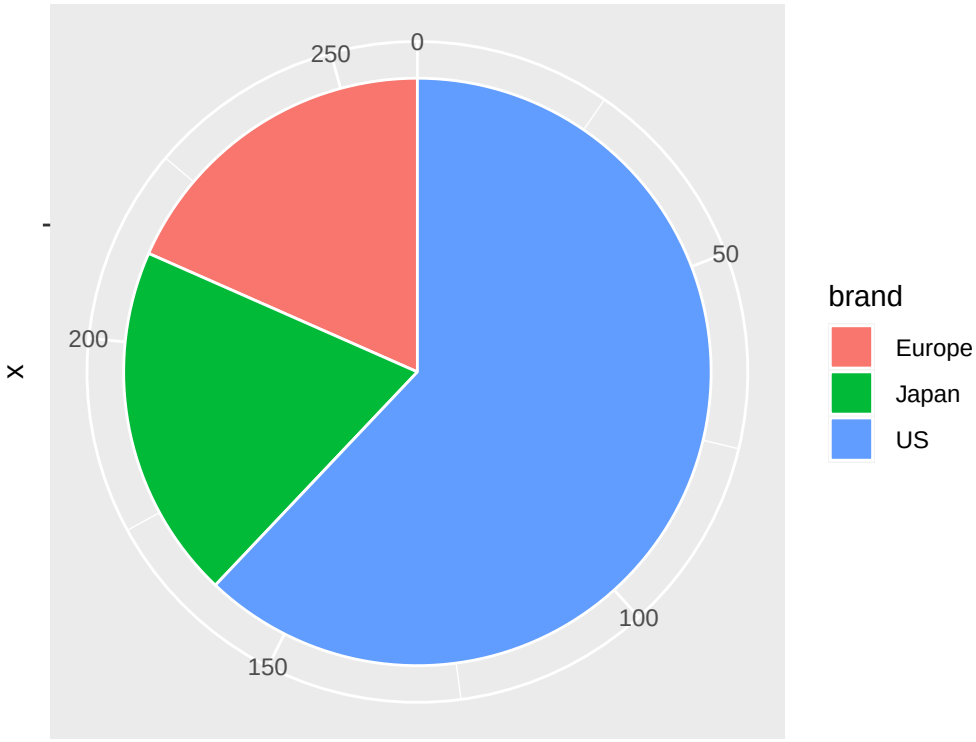



```
library(ggplot2, warn.conflicts = FALSE)
library(dplyr, warn.conflicts = FALSE)

contingency_table <- table(cars$brand)
df <- data.frame(brand = names(contingency_table),
                 Occurrences = as.vector(contingency_table))

ggplot(df, aes(x = "", y = Occurrences, fill = brand)) +
    ggtitle("Fig. 2 : Diagramme circulaire de brand") +
    geom_bar(width = 1, stat = "identity", color = "white") +
    coord_polar("y", start = 0)
```

Fig. 2 : Diagramme circulaire de brand

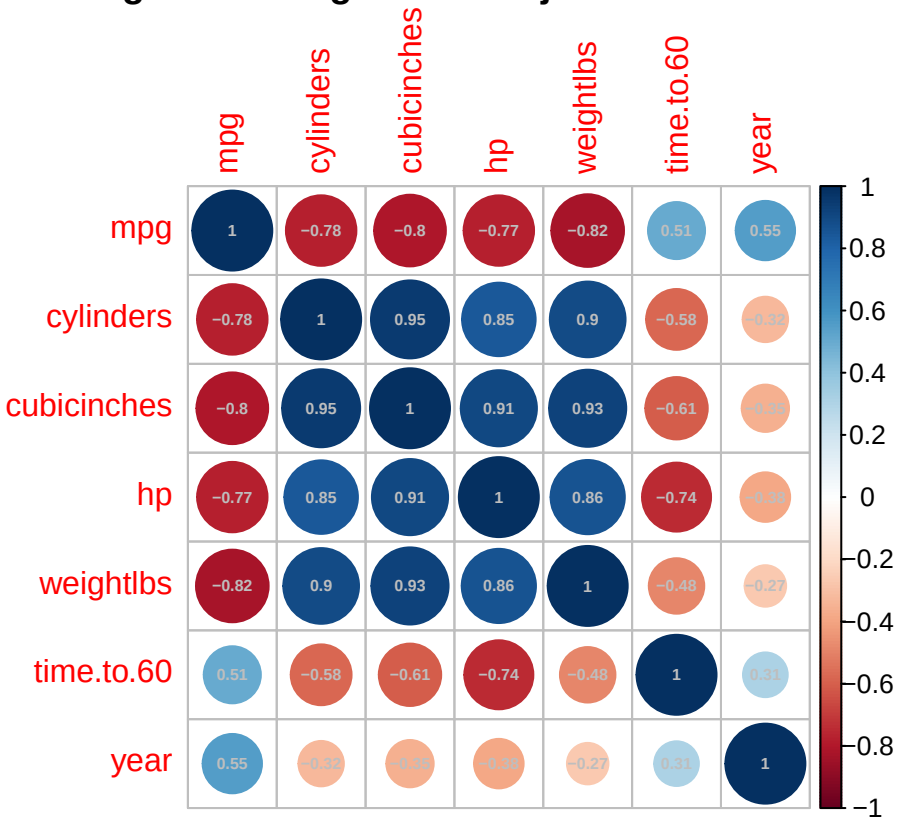



```
library(corrplot)
```

```
## corrplot 0.88 loaded
```

```
corrplot(cars.correlations,addCoef.col = "grey",
         number.cex = 0.5,
         title="Fig 3. Correlogramme du jeu de donnees",
         mar=c(0,0,1,0))
```

Fig 3. Correlogramme du jeu de donnees

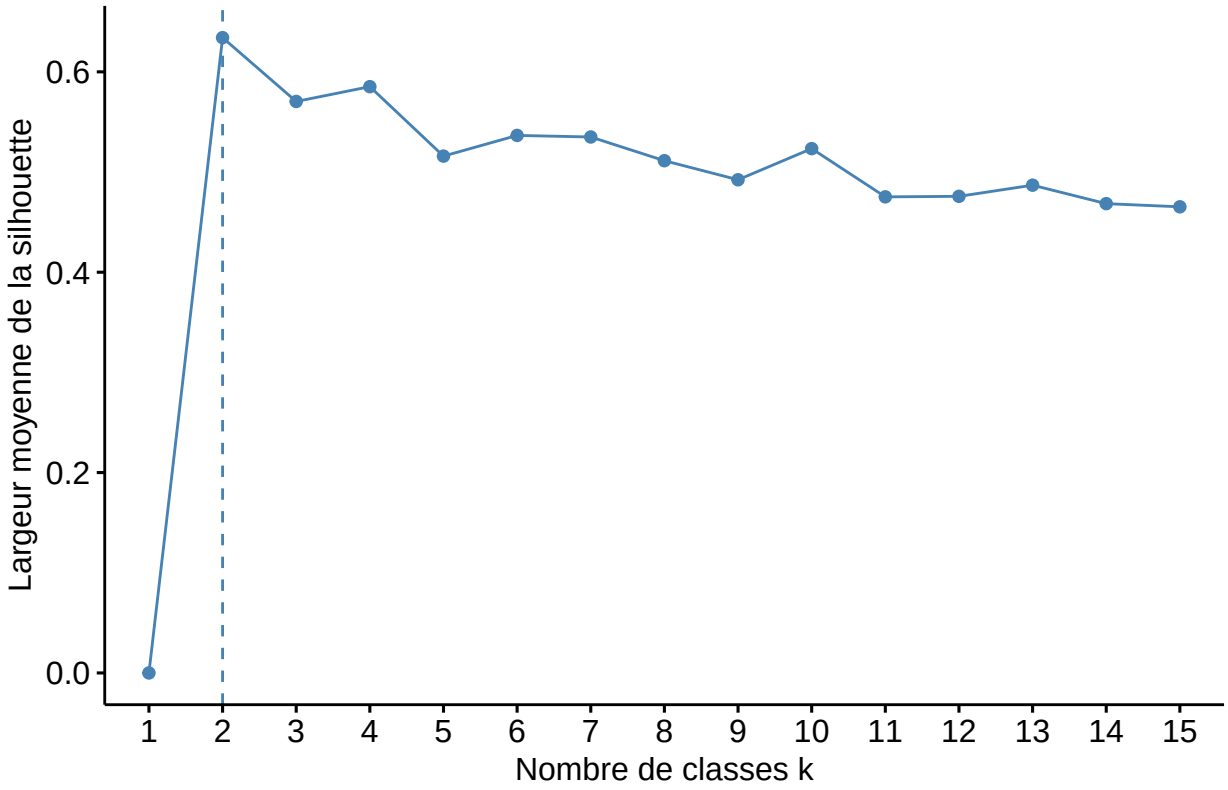



```
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

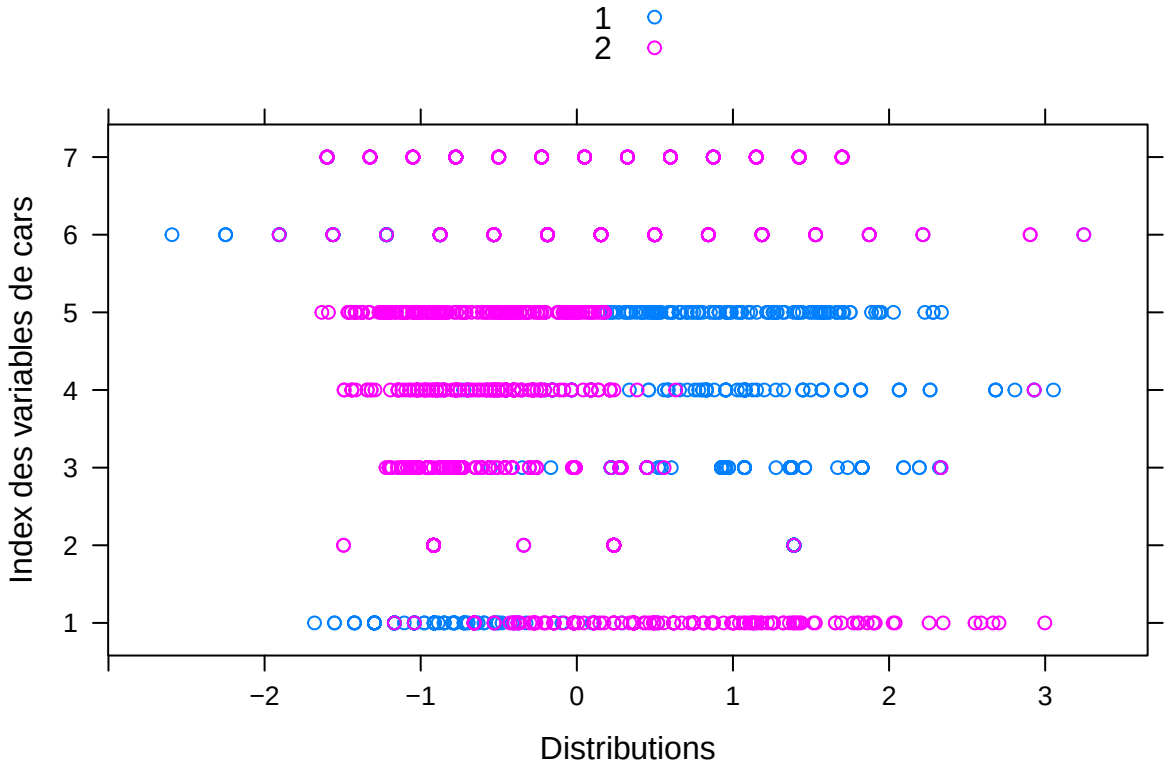
```
fviz_nbclust(cars[-8],
             FUNcluster = kmeans,
             method = c("silhouette"),
             k.max = 15,
             nboot = 100) +
    labs(title="Fig. 4 : Nombre de classes optimal") +
    xlab("Nombre de classes k") +
    ylab("Largeur moyenne de la silhouette")
```

Fig. 4 : Nombre de classes optimal



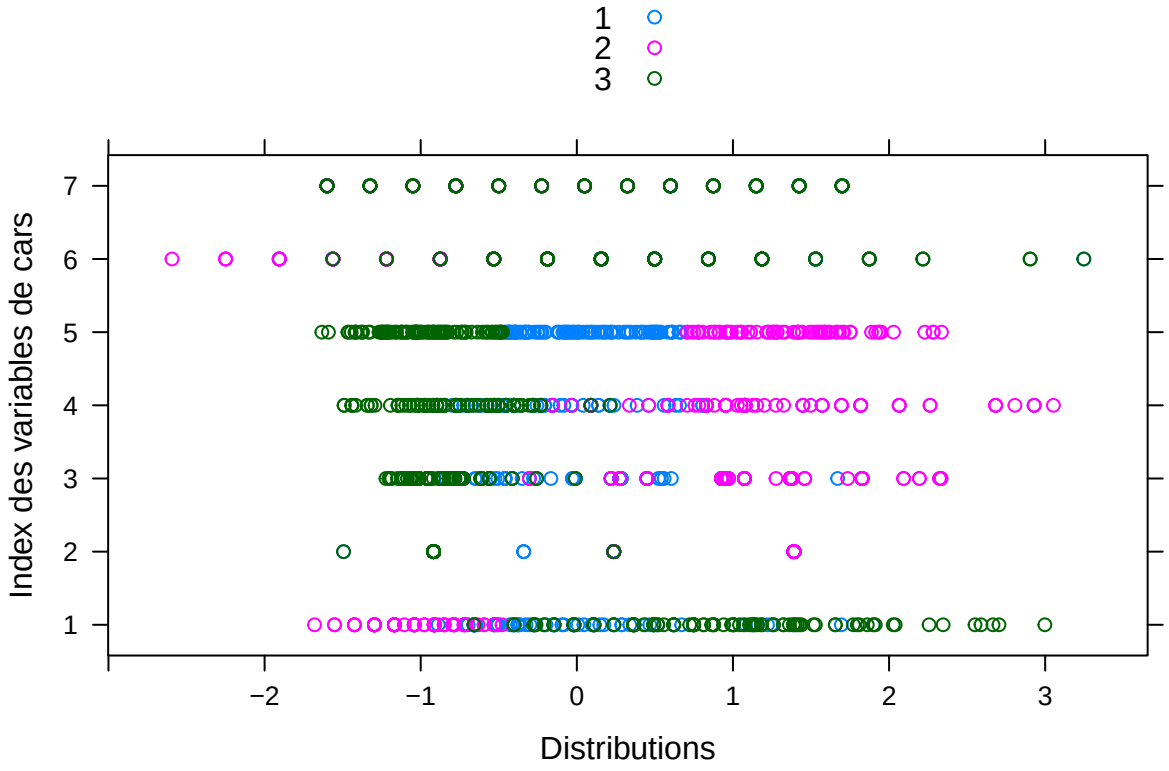

```
stripplot_clusters(cars.scaled[-8], cars.kmeans.cluster2$cluster, 5)
```

Fig. 5 : Distribution des variables de cars colorées par 2 classes



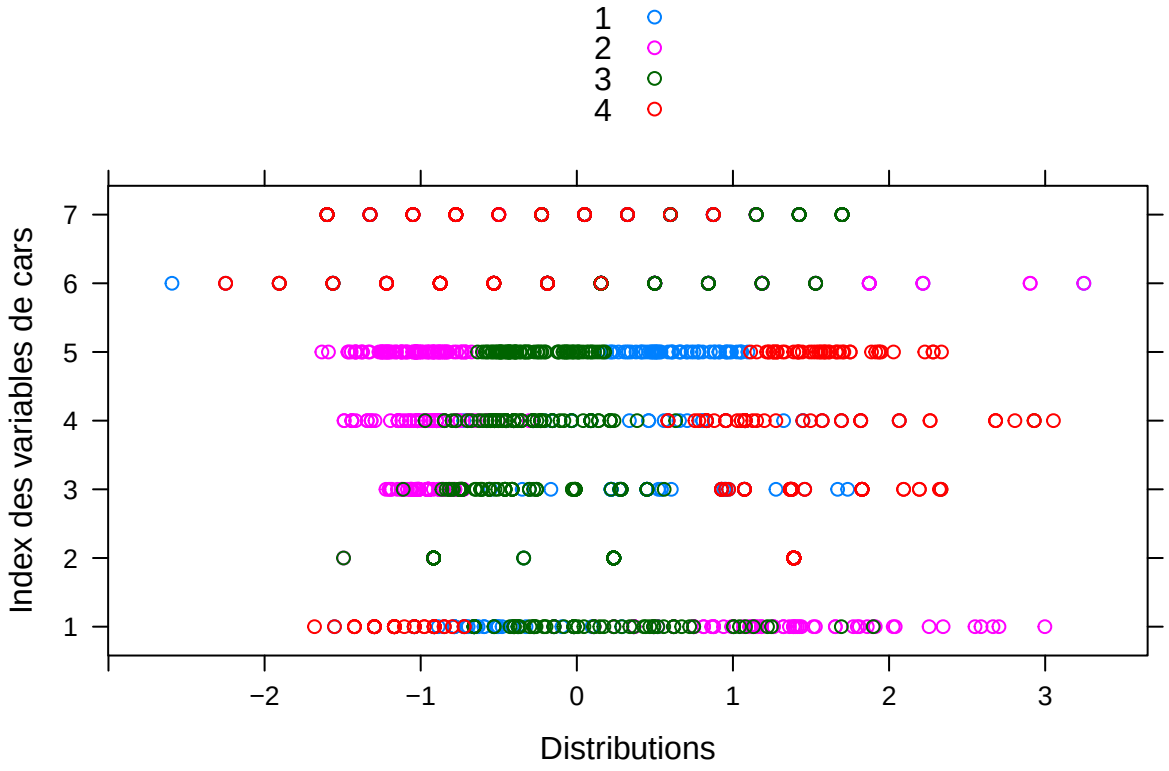

```
stripplot_clusters(cars.scaled[-8], cars.kmeans.cluster3$cluster, 6)
```

Fig. 6 : Distribution des variables de cars colorées par 3 classes



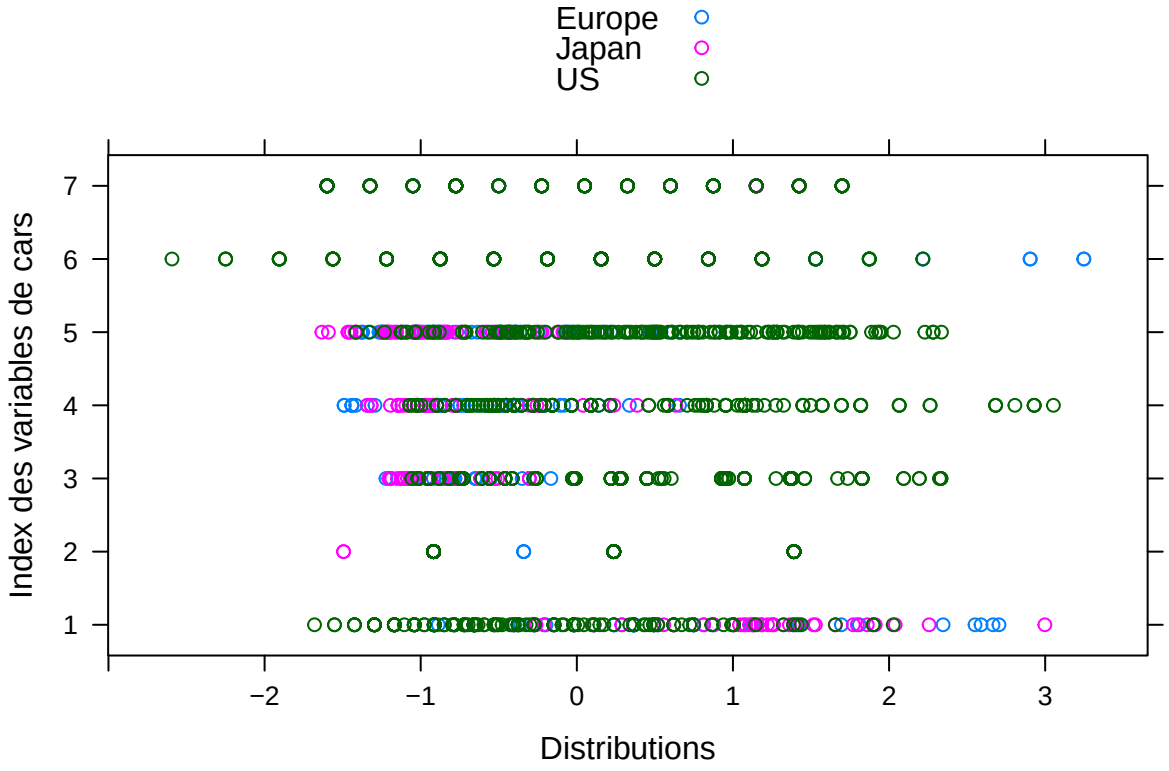

```
stripplot_clusters(cars.scaled[-8], cars.kmeans.cluster4$cluster, 7)
```

Fig. 7 : Distribution des variables de cars colorées par 4 classes



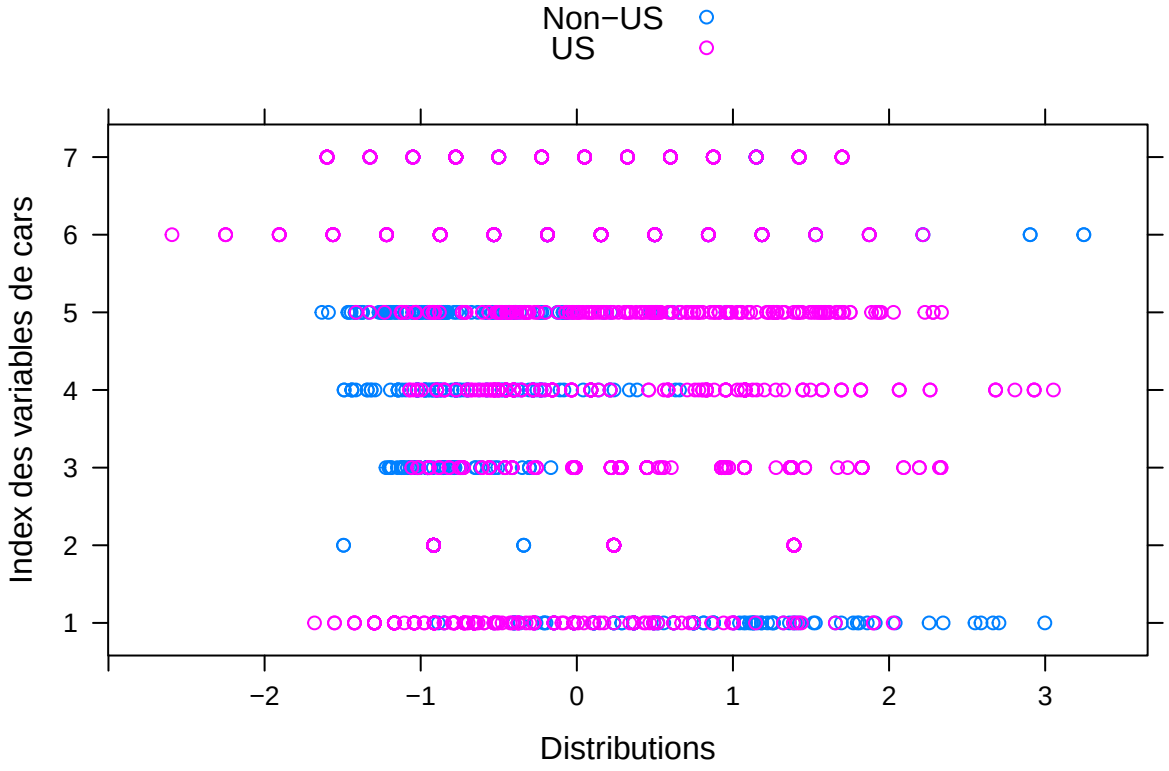

```
stripplot_clusters(cars.scaled[-8], cars$brand, 8)
```

Fig. 8 : Distribution des variables de cars colorées par 3 classes



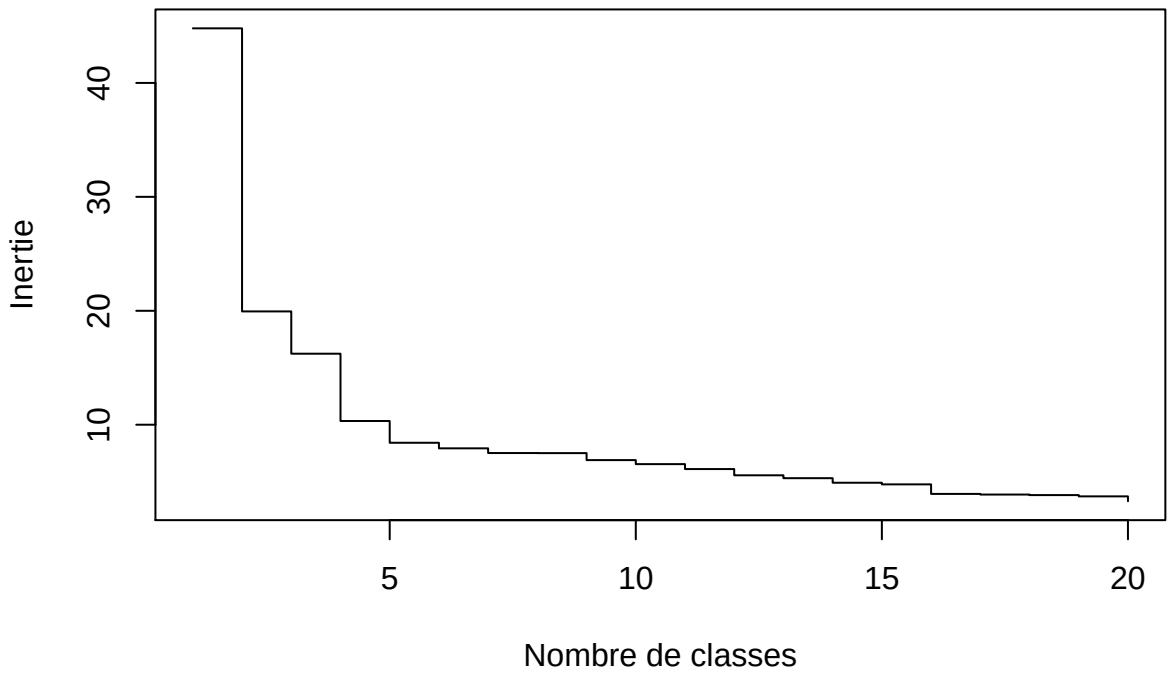

```
stripplot_clusters(cars.scaled[-8], cars.combined.brand, 9)
```

Fig. 9 : Distribution des variables de cars colorées par 2 classes



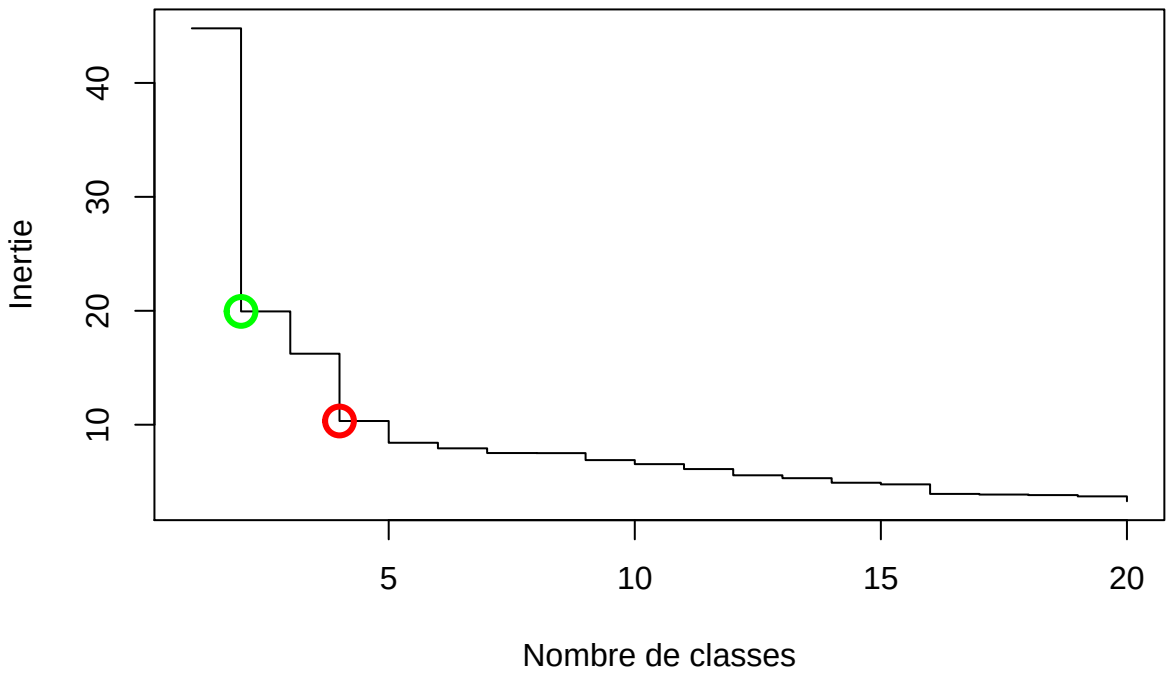

```
plot(cars.hc.ward.inertia[1:20], type = "s",
     xlab = "Nombre de classes",
     ylab = "Inertie",
     main = "Fig. 10 : Inertie du dendrogramme en fonction du nombre de classes")
```

Fig. 10 : Inertie du dendrogramme en fonction du nombre de classe



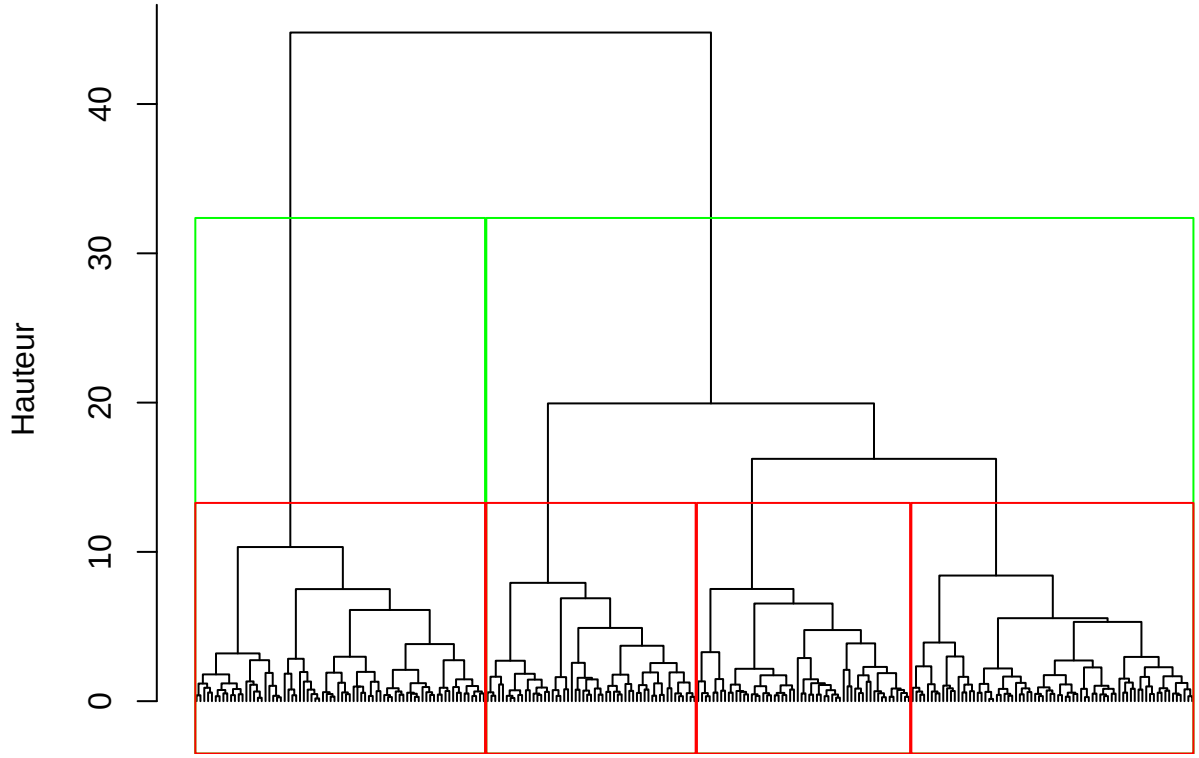

```
plot(cars.hc.ward.inertia[1:20], type = "s",
     xlab = "Nombre de classes",
     ylab = "Inertie",
     main = "Fig 11. Inertie du dendrogramme avec plus grands gains entoures")
points(c(2, 4),
       cars.hc.ward.inertia[c(2, 4)],
       col = c("green", "red"),
       cex = 2, lwd = 3)
```

Fig 11. Inertie du dendrogramme avec plus grands gains entoures



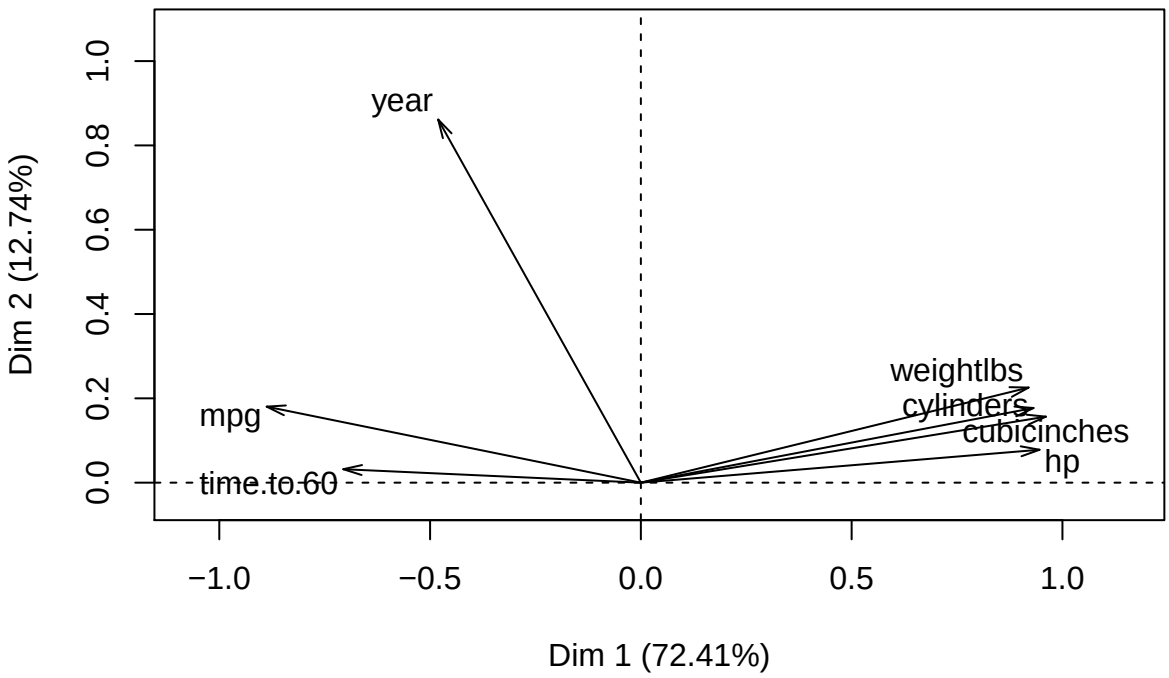

```
par(mar=c(1,4.5,2,0))
plot(cars.hc.ward, labels = FALSE,
     main = "Fig. 12 : Partitions en 2 ou 4 classes",
     ylab = "Hauteur",
     hang = -1)
rect.hclust(cars.hc.ward, 2, border = "green")
rect.hclust(cars.hc.ward, 4, border = "red")
```

Fig. 12 : Partitions en 2 ou 4 classes



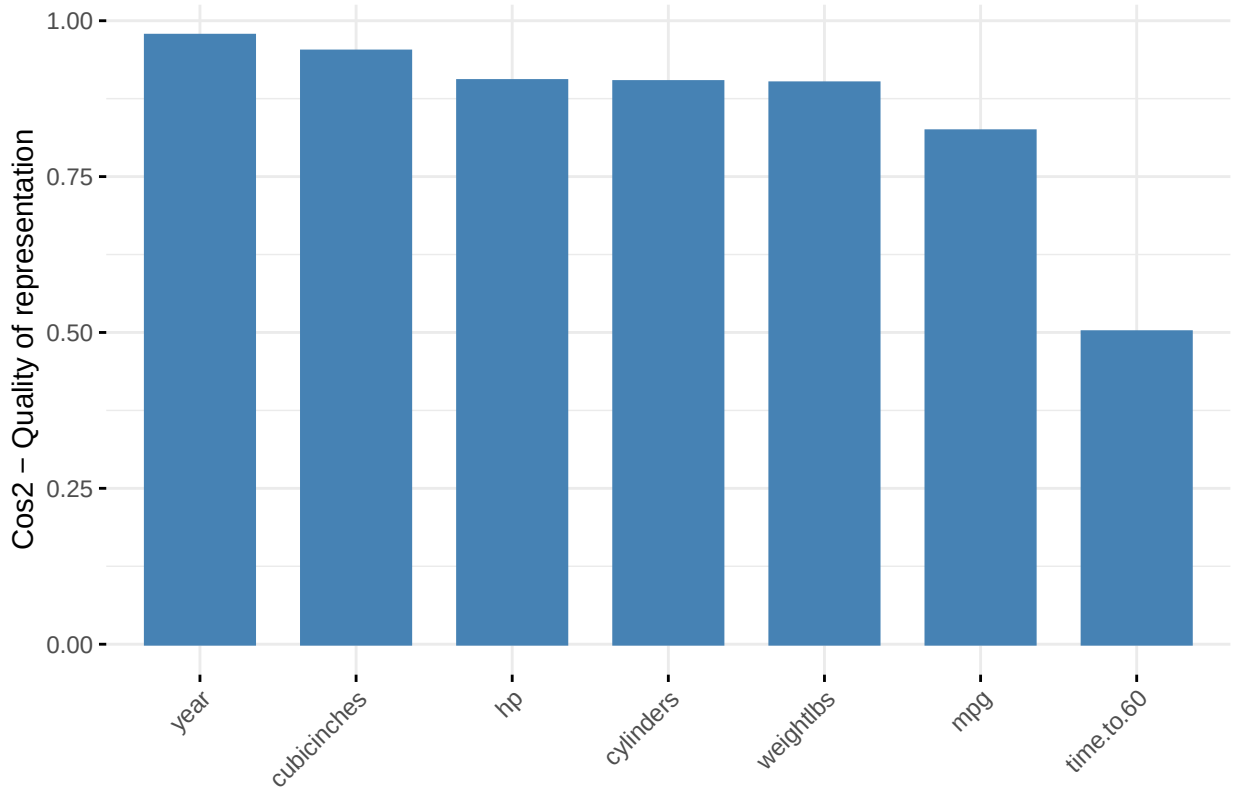

```
library(FactoMineR)
plot(cars.pca, choix="var", axes=1:2, title="Fig. 13 : Variables representees sur le premier plan facto
```

Fig. 13 : Variables representees sur le premier plan factoriel



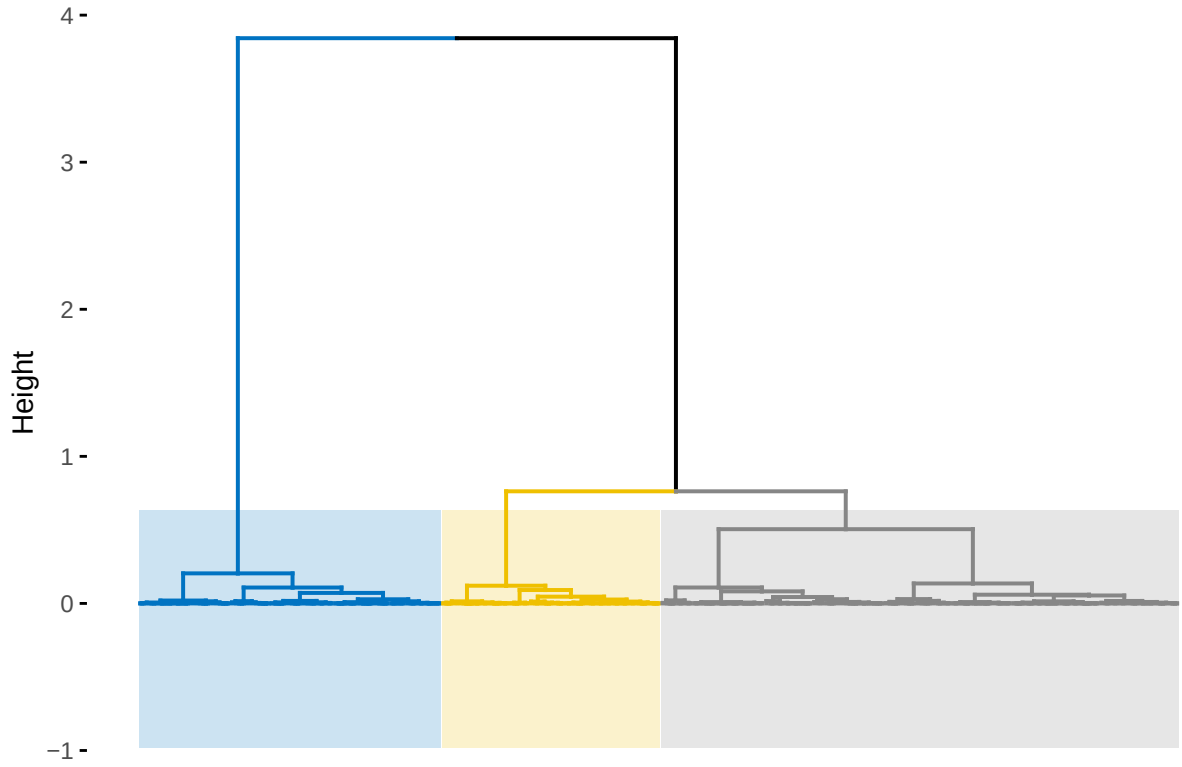

```
fviz_cos2(cars.pca, "var", axes = 1:2, title = "Fig. 14 : Cos2 des variables du premier plan factoriel")
```

Fig. 14 : Cos2 des variables du premier plan factoriel




```
fviz_dend(cars.hcpc,
          cex = 0.7,                     # Taille du text
          palette = "jco",               # Palette de couleur ?ggpubr::ggpar
          rect = TRUE, rect_fill = TRUE, # Rectangle autour des groupes
          rect_border = "jco",           # Couleur du rectangle
          #labels_track_height = 0.8      # Augment l'espace pour le texte en 3D
          show_labels = FALSE,            #Retire les labels
          main = "Fig. 15 : Dendrogramme de la HCPC"
          )
```

Fig. 15 : Dendrogramme de la HCPC




```
fviz_cluster(cars.hcpc,
             repel = TRUE,            # Evite le chevauchement des textes
             show.clust.cent = TRUE, # Montre le centre des clusters
             palette = "jco",         # Palette de couleurs, voir ?ggpubr::ggpar
             ggtheme = theme_minimal(),
             main = "Fig. 16 : Classes trouvees par l'HCPC projetees sur le premier plan factoriel"
             )
```

Fig. 16 : Classes trouvees par l'HCPC projetees sur le premier plan factoriel