

Projet d'apprentissage non supervisé

Marie Guibert - Clémence Chesnais

2023-03-25

Environnement de travail

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.4
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(stargazer)
```

```
##
## Please cite as:
##
## Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
## R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
library(gridExtra)
```

```
##
## Attachement du package : 'gridExtra'
##
## L'objet suivant est masqué depuis 'package:dplyr':
##
##      combine
```

Question 1

Importation des données

```
data <- read.csv("Pays_donnees.csv",sep="," ,dec=".",stringsAsFactors = T)
str(data)
```

```
## 'data.frame':    167 obs. of  10 variables:
## $ pays          : Factor w/ 167 levels "Afghanistan",...: 1 2 3 4 5 6 7 8 9 10 ...
## $ enfant_mort   : num  90.2 16.6 27.3 119 10.3 14.5 18.1 4.8 4.3 39.2 ...
## $ exports       : num  10 28 38.4 62.3 45.5 18.9 20.8 19.8 51.3 54.3 ...
## $ sante         : num  7.58 6.55 4.17 2.85 6.03 8.1 4.4 8.73 11 5.88 ...
## $ imports       : num  44.9 48.6 31.4 42.9 58.9 16 45.3 20.9 47.8 20.7 ...
## $ revenu        : int  1610 9930 12900 5900 19100 18700 6700 41400 43200 16000 ...
## $ inflation     : num  9.44 4.49 16.1 22.4 1.44 20.9 7.77 1.16 0.873 13.8 ...
## $ esper_vie     : num  56.2 76.3 76.5 60.1 76.8 75.8 73.3 82 80.5 69.1 ...
## $ fert          : num  5.82 1.65 2.89 6.16 2.13 2.37 1.69 1.93 1.44 1.92 ...
## $ pib_h         : int  553 4090 4460 3530 12200 10300 3220 51900 46900 5840 ...
```

```
# summary(data)
```

Dans ce jeu de données, nous pouvons observer 10 variables dont 9 numériques et 1 facteur comprenant les différents pays (individus). Nous avons choisi de transformer la variable pays en facteur pour simplifier nos traitement des données.

Afin de pouvoir analyser ces données, nous allons réaliser des statistiques descriptives de base.

Statistiques descriptives

```
sum(is.na(data))
```

```
## [1] 0
```

Le jeu de données ne présente pas de valeur manquante, nous n'avons pas besoin de faire de modification de ce point de vue.

Résumé des données :

```
stargazer(data,type="text",title="Résumé des données",out="resume_donnees.txt")
```

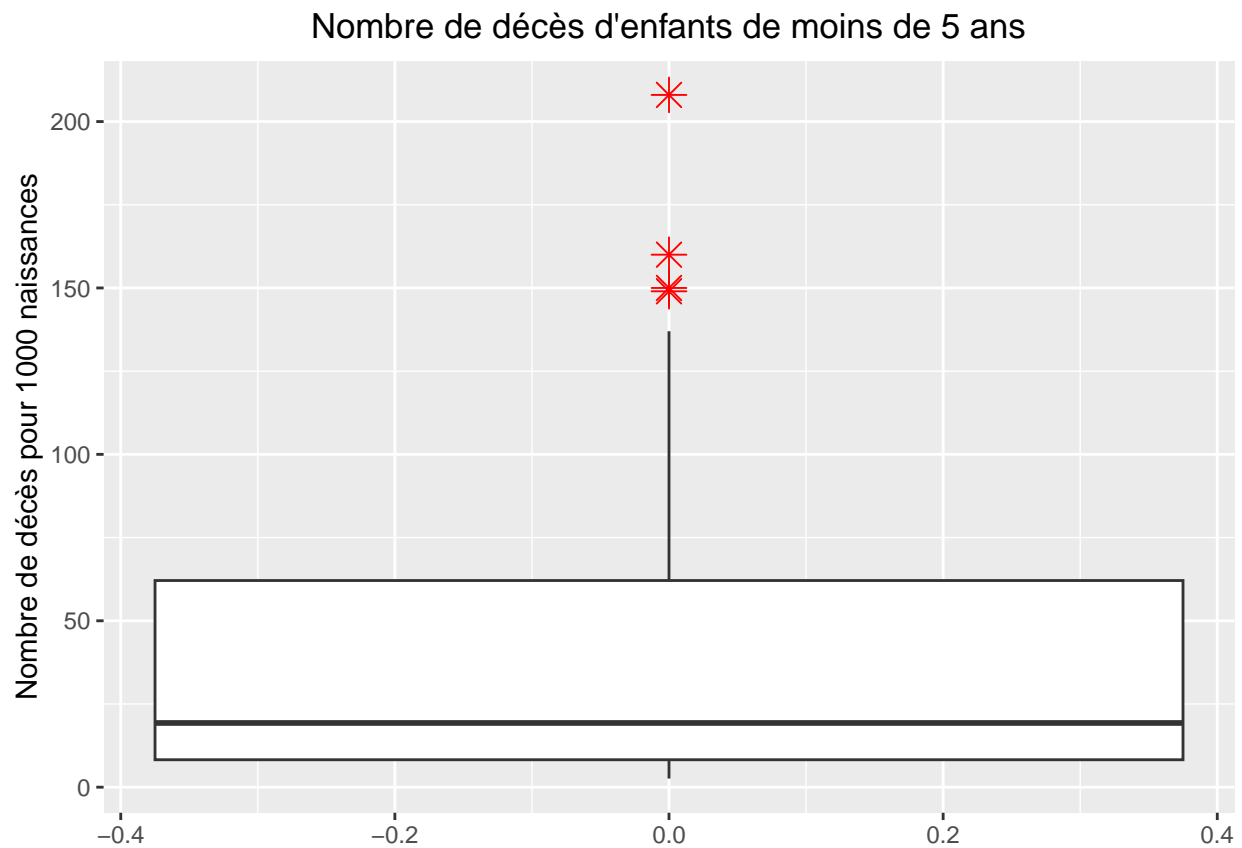
```
##
## Résumé des données
## =====
## Statistic      N      Mean      St. Dev.    Min      Max
## -----
## enfant_mort 167    38.270    40.329    2.600    208.000
## exports     167    41.109    27.412    0.109    200.000
## sante        167     6.816     2.747     1.810     17.900
## imports     167    46.890    24.210    0.066    174.000
## revenu      167  17,144.690 19,278.070   609    125,000
## inflation   167     7.782    10.571   -4.210    104.000
## esper_vie   167    70.556     8.893    32.100    82.800
## fert        167     2.948     1.514     1.150     7.490
## pib_h       167  12,964.160 18,328.710   231    105,000
## -----
```

Ce résumé statistique nous permet d'avoir une vue d'ensemble sur les données.

Notre jeu de données est composé de 167 pays très hétérogènes. En effet, nous pouvons observer une assez grande différence entre le minimum et le maximum de chaque variable, ce qui prouve la diversité de notre échantillon.

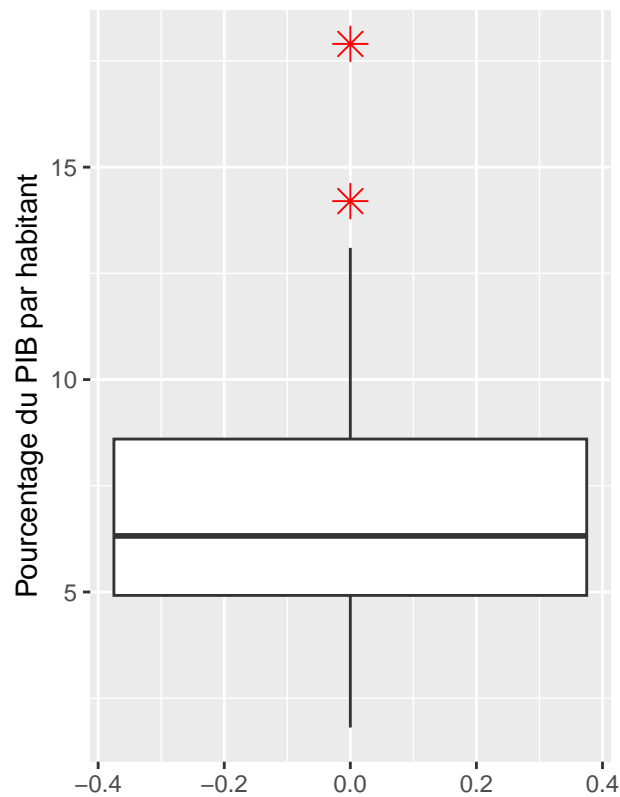
Graphiques :

```
ggplot(data=data, aes(y=enfant_mort)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,outlier.size=4)+  
  labs(title="Nombre de décès d'enfants de moins de 5 ans",y="Nombre de décès pour 1000 naissances")+  
  theme(plot.title = element_text(hjust=0.5))
```

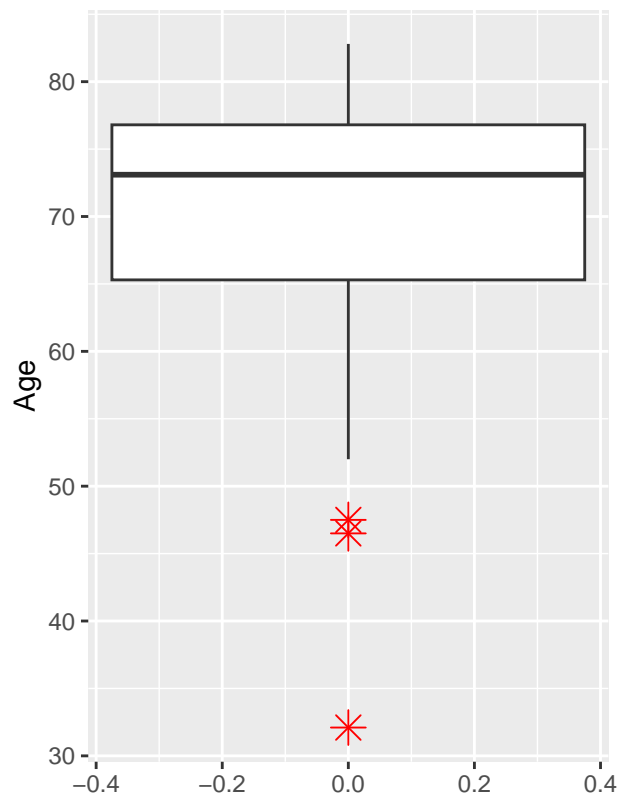


```
sante <- ggplot(data=data, aes(y=sante)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,outlier.size=4)+  
  labs(title="Dépenses totales de santé par habitant",y="Pourcentage du PIB par habitant")+  
  theme(plot.title = element_text(hjust=0.5))  
  
esperance <- ggplot(data=data, aes(y=esper_vie)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,outlier.size=4)+  
  labs(title="Espérance de vie",y="Age")+  
  theme(plot.title = element_text(hjust=0.5))  
  
grid.arrange(sante,esperance,ncol=2)
```

Dépenses totales de santé par habita



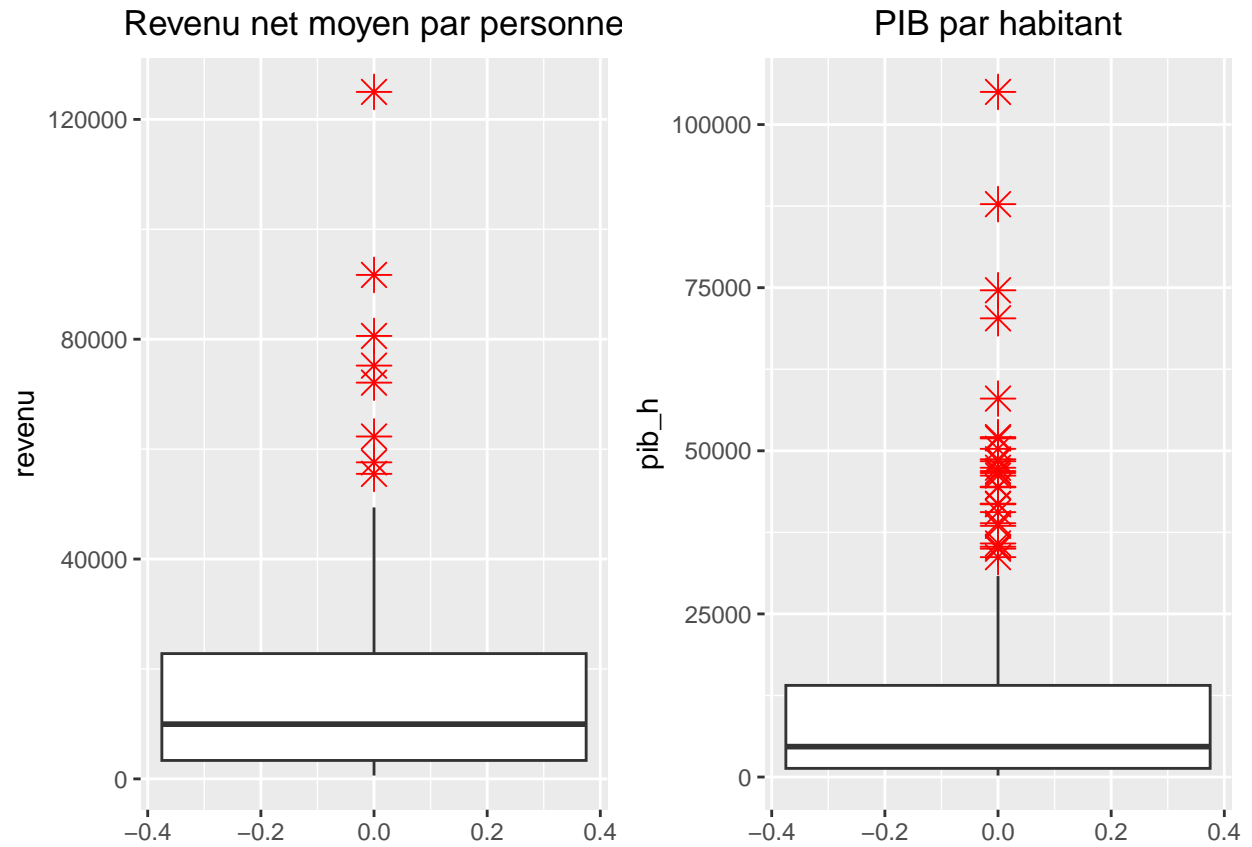
Espérance de vie



```
revenu_net <- ggplot(data=data, aes(y=revenu)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,outlier.size=4)+
  labs(title="Revenu net moyen par personne")+
  theme(plot.title = element_text(hjust=0.5))

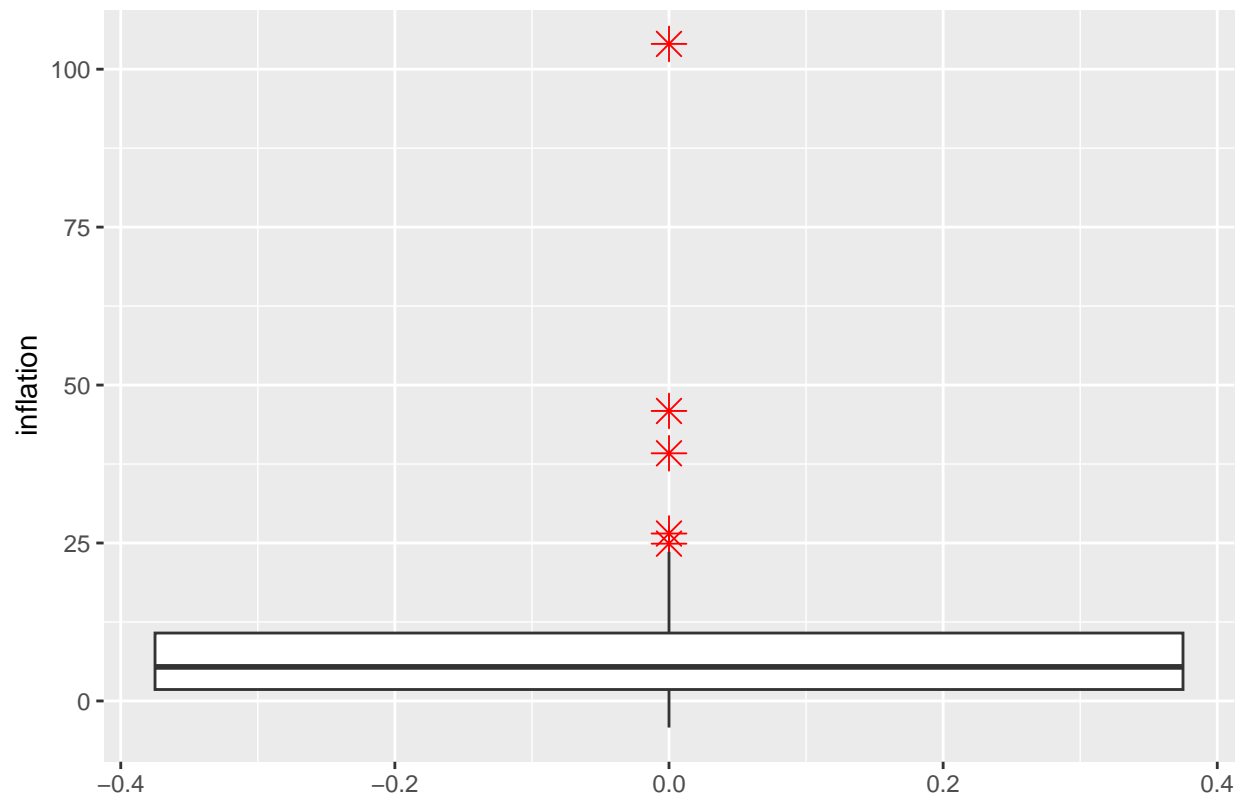
pib_hab <- ggplot(data=data, aes(y=pib_h)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,outlier.size=4)+
  labs(title="PIB par habitant")+
  theme(plot.title = element_text(hjust=0.5))

grid.arrange(revenu_net,pib_hab,ncol=2)
```

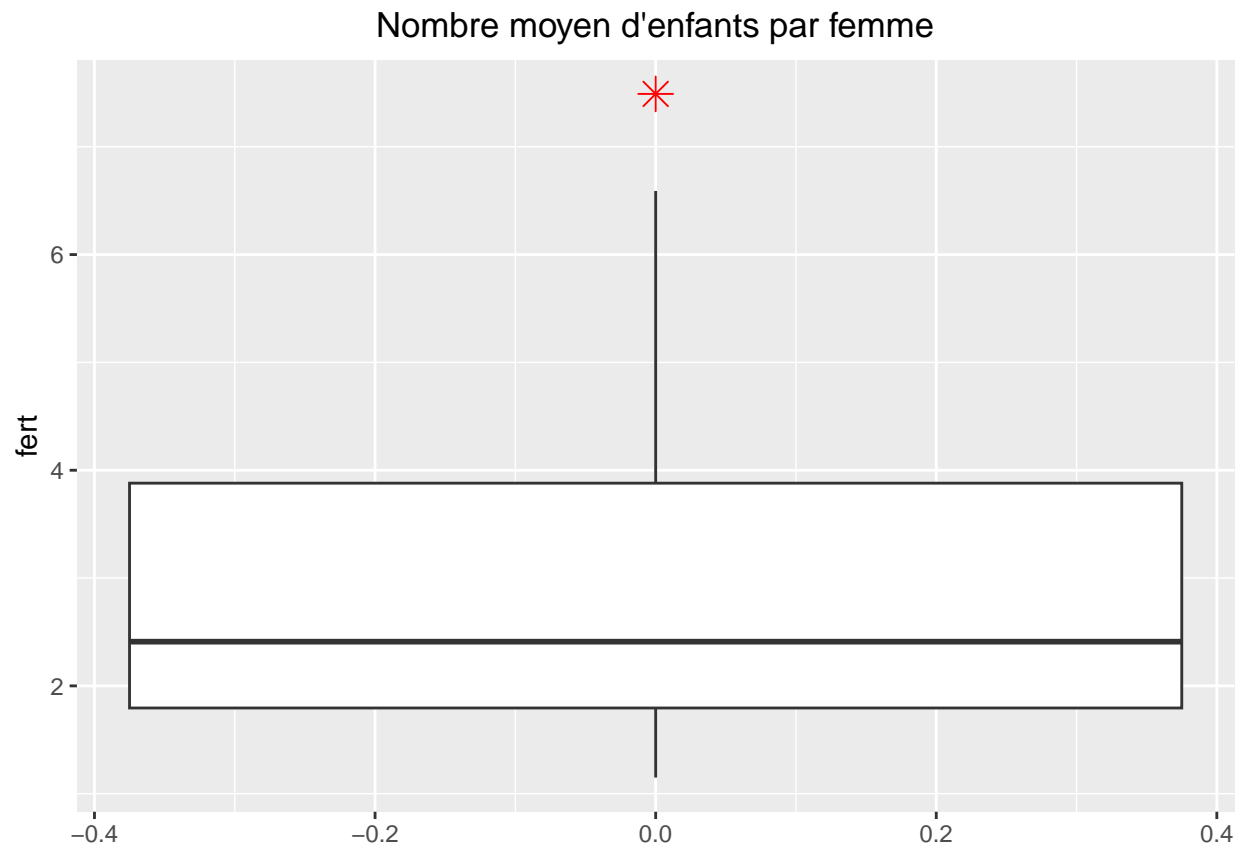


```
ggplot(data=data, aes(y=inflation)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8, outlier.size=4)+
  labs(title="Mesure du taux de croissance annuel du PIB total")+
  theme(plot.title = element_text(hjust=0.5))
```

Mesure du taux de croissance annuel du PIB total



```
ggplot(data=data, aes(y=fert)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8, outlier.size=4)+  
  labs(title="Nombre moyen d'enfants par femme")+  
  theme(plot.title = element_text(hjust=0.5))
```



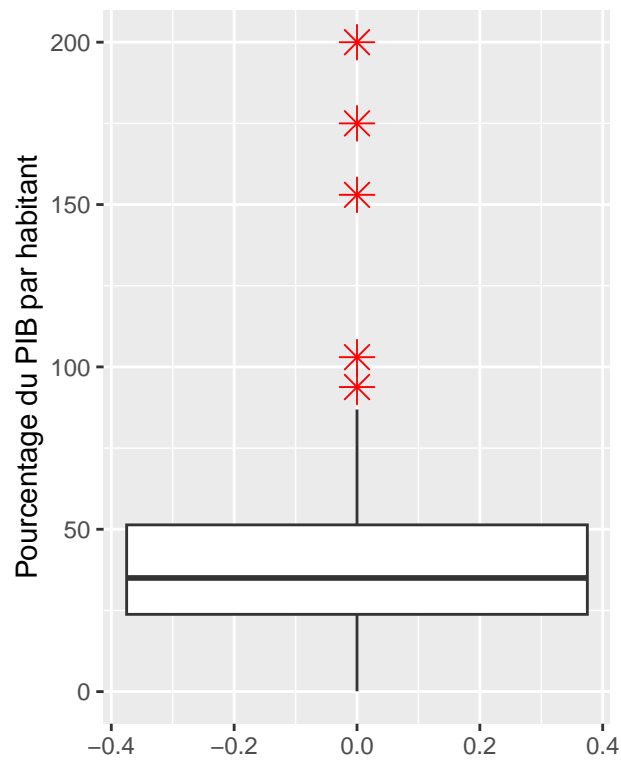
Imports et Exports :

```
imports <- ggplot(data=data, aes(y=imports)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8, outlier.size=4)+
  labs(title="Importations de biens et services \npar habitant", y="Pourcentage du PIB par habitant")+
  theme(plot.title = element_text(hjust=0.5))

exports <- ggplot(data=data, aes(y=exports)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8, outlier.size=4)+
  labs(title="Exportations de biens et services \npar habitant", y="Pourcentage du PIB par habitant")+
  theme(plot.title = element_text(hjust=0.5))

grid.arrange(exports, imports, ncol=2)
```

Exportations de biens et services
par habitant



Importations de biens et services
par habitant

