

Simulations partie 2

Jawad Boulahfa

02/09/2020

```
rm(list = ls())
```

Installation du package nnls2d

```
#devtools::install_github("Jawad-Boulahfa/nnls2d")
```

Chargement des packages

```
# Pour pcls2 et les fonctions de simulation
library(nnls2d, quietly = TRUE)

# Pour le calcul parallèle
library(foreach, quietly = TRUE)
library(iterators, quietly = TRUE)
library(parallel, quietly = TRUE)
library(doParallel, quietly = TRUE)

# Pour ggplot2 et la manipulation des dataframes
library(tidyverse, quietly = TRUE)

# Pour tracer plusieurs graphiques en même temps
library(gridExtra, quietly = TRUE)
```

Initialisation

Valeurs de n et de σ

```
liste_n <- 10^5
liste_sigma <- 1
```

Initialisation du calcul parallèle

```
nb_one_simu <- 5000 # val max pour shapiro
cores <- detectCores()
```

Choix du nombre de classes pour les histogrammes

```
classes_hist <- 50
```

Choix du α pour les IC

```
alpha <- 0.05
```

Choix de la valeur de beta

```
beta = c(1, 0)
```

Simulations

```
simulations <- nnls2d::simulations(cores = cores,  
                                   liste_n = liste_n,  
                                   liste_sigma = liste_sigma,  
                                   beta = beta, nb_one_simu = nb_one_simu,  
                                   classes_hist = classes_hist, alpha = alpha)
```

On change les noms pour plus de lisibilité.

```
liste_final_df <- simulations$liste_final_df  
  
liste_plot_nnls_1 <- simulations$liste_plot_nnls_1  
liste_plot_nnls_2 <- simulations$liste_plot_nnls_2  
  
liste_plot_lm_1 <- simulations$liste_plot_lm_1  
liste_plot_lm_2 <- simulations$liste_plot_lm_2  
  
liste_comparaison_1 <- simulations$liste_comparaison_1  
liste_comparaison_2 <- simulations$liste_comparaison_2  
  
liste_comparaison_1_without_0 <-  
  simulations$liste_comparaison_1_without_0  
liste_comparaison_2_without_0 <-  
  simulations$liste_comparaison_2_without_0  
  
liste_plot_comparaison_1 <- simulations$liste_plot_comparaison_1  
liste_plot_comparaison_2 <- simulations$liste_plot_comparaison_2  
  
liste_plot_comparaison_1_without_0 =  
  simulations$liste_plot_comparaison_1_without_0  
liste_plot_comparaison_2_without_0 =  
  simulations$liste_plot_comparaison_2_without_0  
  
liste_resultats_df <- simulations$liste_resultats_df  
liste_comments_resultats_df <- simulations$liste_comments_resultats_df  
  
liste_resultats_theoriques_df <-  
  simulations$liste_resultats_theoriques_df  
liste_comments_resultats_theoriques_df <-  
  simulations$liste_comments_resultats_theoriques_df  
  
liste_IC_df <- simulations$liste_IC_df  
liste_comments_IC_df <- simulations$liste_comments_IC_df  
  
liste_rapports_1 <- simulations$liste_rapports_1
```

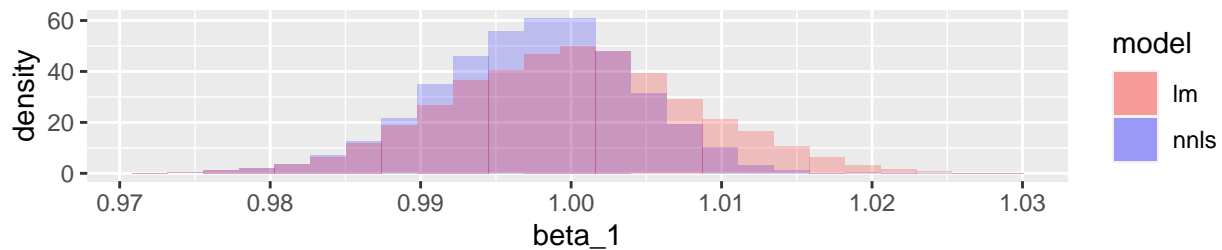
```
liste_rapports_2 <- simulations$liste_rapports_2

liste_comments_rapports_1 <- simulations$liste_comments_rapports_1
liste_comments_rapports_2 <- simulations$liste_comments_rapports_2
```

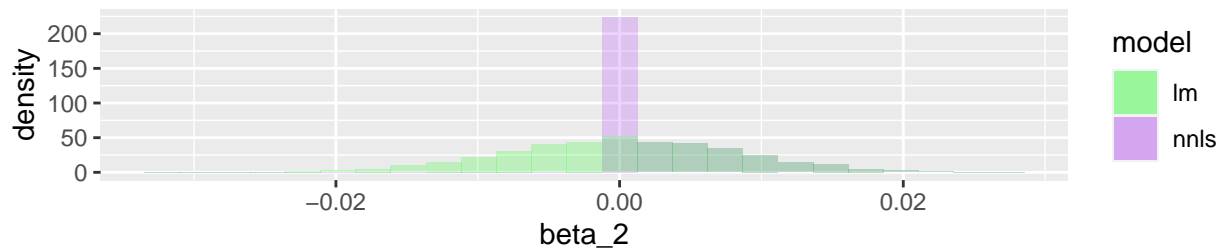
Histogrammes comparatifs

```
grid.arrange(liste_plot_comparison_1[[1]],
             liste_plot_comparison_2[[1]],
             nrow=2, ncol=1)
```

Comparaison des distributions de β_{nnls_1} et de β_{lm_1}
 nombre de répétitions = 5000, $n = 1e+05$, $\sigma = 1$
 nombre de classes = 25
 $\beta_1 = 1$, $\beta_2 = 0$



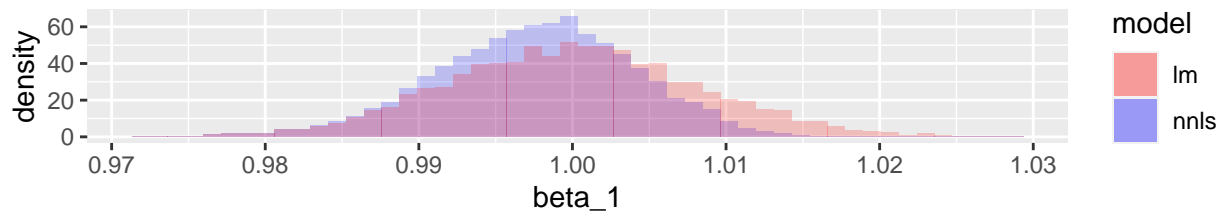
Comparaison des distributions de β_{nnls_2} et de β_{lm_2}
 nombre de répétitions = 5000, $n = 1e+05$, $\sigma = 1$
 nombre de classes = 25
 $\beta_1 = 1$, $\beta_2 = 0$



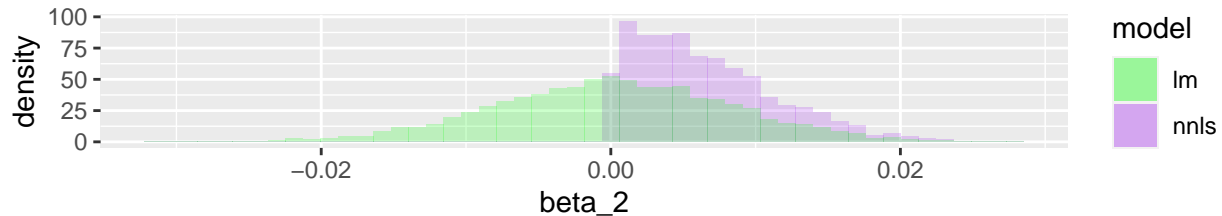
```
#ggsave(filename = "comparison_first_simu.png", plot =
#         grid.arrange(liste_plot_comparison_1[[1]],
#                       liste_plot_comparison_2[[1]], nrow=2, ncol=1),
#         width = 20, height = 20, units = "cm", device = "png")
```

```
grid.arrange(liste_plot_comparison_1_without_0[[1]],
             liste_plot_comparison_2_without_0[[1]],
             nrow=2, ncol=1)
```

Comparaison des distributions de
 beta_nnl1 (valeurs > 0) et de beta_lm_1
 nombre de répétitions = 5000, n = 1e+05, sigma = 1
 nombre de classes = 50
 beta_1 = 1, beta_2 = 0



Comparaison des distributions de
 beta_nnl2 (valeurs > 0) et de beta_lm_2
 nombre de répétitions = 5000, n = 1e+05, sigma = 1
 nombre de classes = 50
 beta_1 = 1, beta_2 = 0



Tests de normalité

```
shapiro_1 <-  
  shapiro.test(liste_final_df[[1]]$beta_nnl1)  
pvalue_1 <- shapiro_1$p.value
```

```
print(shapiro_1)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data:  liste_final_df[[1]]$beta_nnl1  
## W = 0.99657, p-value = 2.429e-09
```

```
shapiro_2 <-  
  shapiro.test(liste_final_df[[1]]$beta_nnl2)  
pvalue_2 <- shapiro_2$p.value
```

```
print(shapiro_2)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data:  liste_final_df[[1]]$beta_nnl2  
## W = 0.74097, p-value < 2.2e-16
```

Biais, variance, erreur quadratique moyenne

```
cat(liste_comments_resultats_df[[1]], sep = "\n")
```

```
## Simulation no.1  
## Nombre de répétitions = 5000  
## n = 1e+05  
## sigma = 1
```

```
print(liste_resultats_df[[1]])
```

```
##           biais      variance      EQM  
## nnls_1 -2.450063e-03 4.259807e-05 4.860088e-05  
## lm_1    -2.328867e-05 6.715960e-05 6.716014e-05  
## nnls_2  3.332792e-03 2.301496e-05 3.412246e-05  
## lm_2     9.734100e-05 6.825561e-05 6.826509e-05
```

Sauvegarde des résultats

```
save.image(file = "simulations_part_2.RData")
```

Chargement des résultats

```
rm(list = ls())
```

```
#load(file = "simulations_part_2.RData")
```

```
#rm(list = ls())
```