DM Statistique bayésienne

Rudy Detain

21/03/2019

Table des matières

1 Analyse exploratoire

```
df <- read_csv("xid-1430229_1.csv")
```

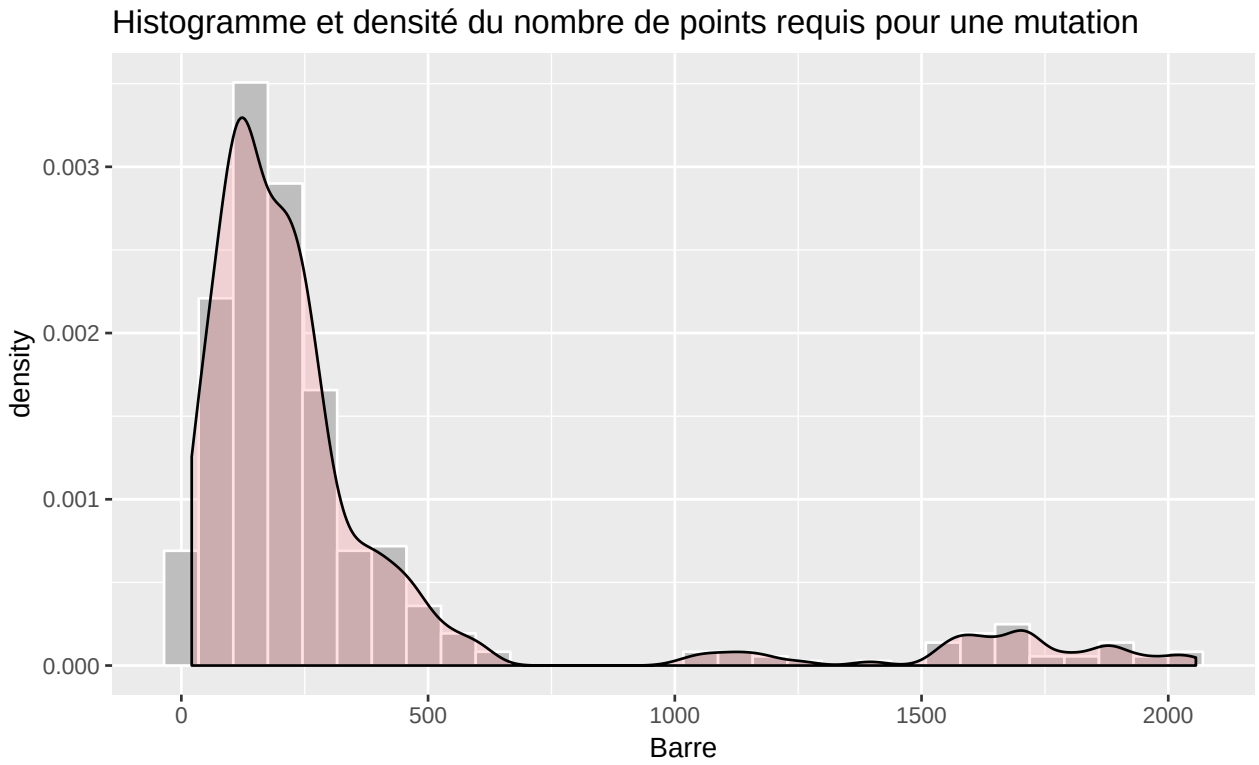
Nous formatons convenablement les covariables de type *factor*.

```
df$code_etablissement = as.factor(df$code_etablissement)
df$ville = as.factor(df$ville)
df$etablissement = as.factor(df$etablissement)
df$commune = as.factor(df$commune)
df$Matiere = as.factor(df$Matiere)
```

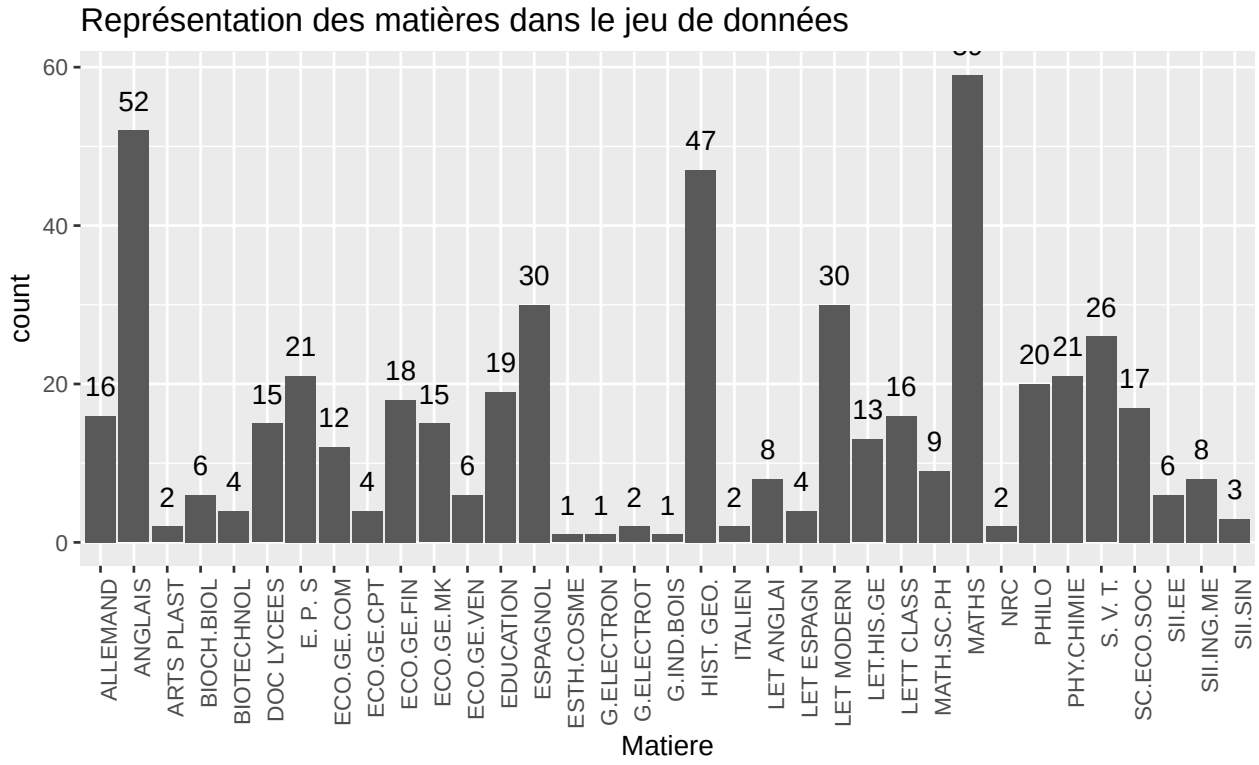
Le jeu de données contient 516 couples établissements / matière différents.

```
## Observations: 516
## Variables: 23
## $ code_etablissement                <fct> 0780422K, 0780422K, 0780422...
## $ ville                             <fct> LES MUREAUX, LES MUREAUX, L...
## $ etablissement                     <fct> LYCEE FRANCOIS VILLON, LYCE...
## $ commune                           <fct> 78440, 78440, 78440, 78383,...
## $ Matiere                           <fct> PHY.CHIMIE, MATHS, DOC LYCE...
## $ Barre                             <dbl> 118.0, 93.0, 38.0, 199.0, 4...
## $ effectif_presents_serie_l         <dbl> 25, 25, 25, 34, 34, 34, 21,...
## $ effectif_presents_serie_es        <dbl> 54, 54, 54, 47, 47, 47, 47,...
## $ effectif_presents_serie_s         <dbl> 97, 97, 97, 47, 47, 47, 81,...
## $ taux_brut_de_reussite_serie_l     <dbl> 56, 56, 56, 79, 79, 79, 86,...
## $ taux_brut_de_reussite_serie_es    <dbl> 85, 85, 85, 98, 98, 98, 96,...
## $ taux_brut_de_reussite_serie_s     <dbl> 80, 80, 80, 85, 85, 85, 90,...
## $ taux_reussite_attendu_serie_l     <dbl> 72, 72, 72, 87, 87, 87, 90,...
## $ taux_reussite_attendu_serie_es    <dbl> 86, 86, 86, 93, 93, 93, 95,...
## $ taux_reussite_attendu_serie_s     <dbl> 75, 75, 75, 91, 91, 91, 93,...
## $ effectif_de_seconde               <dbl> 304, 304, 304, 194, 194, 19...
## $ effectif_de_premiere              <dbl> 222, 222, 222, 168, 168, 16...
## $ taux_acces_brut_seconde_bac       <dbl> 61, 61, 61, 80, 80, 80, 77,...
## $ taux_acces_attendu_seconde_bac    <dbl> 64, 64, 64, 69, 69, 69, 73,...
## $ taux_acces_brut_premiere_bac      <dbl> 84, 84, 84, 92, 92, 92, 88,...
## $ taux_acces_attendu_premiere_bac   <dbl> 81, 81, 81, 87, 87, 87, 87,...
## $ taux_brut_de_reussite_total_series <dbl> 81, 81, 81, 88, 88, 88, 92,...
## $ taux_reussite_attendu_total_series <dbl> 79, 79, 79, 89, 89, 89, 92,...
```

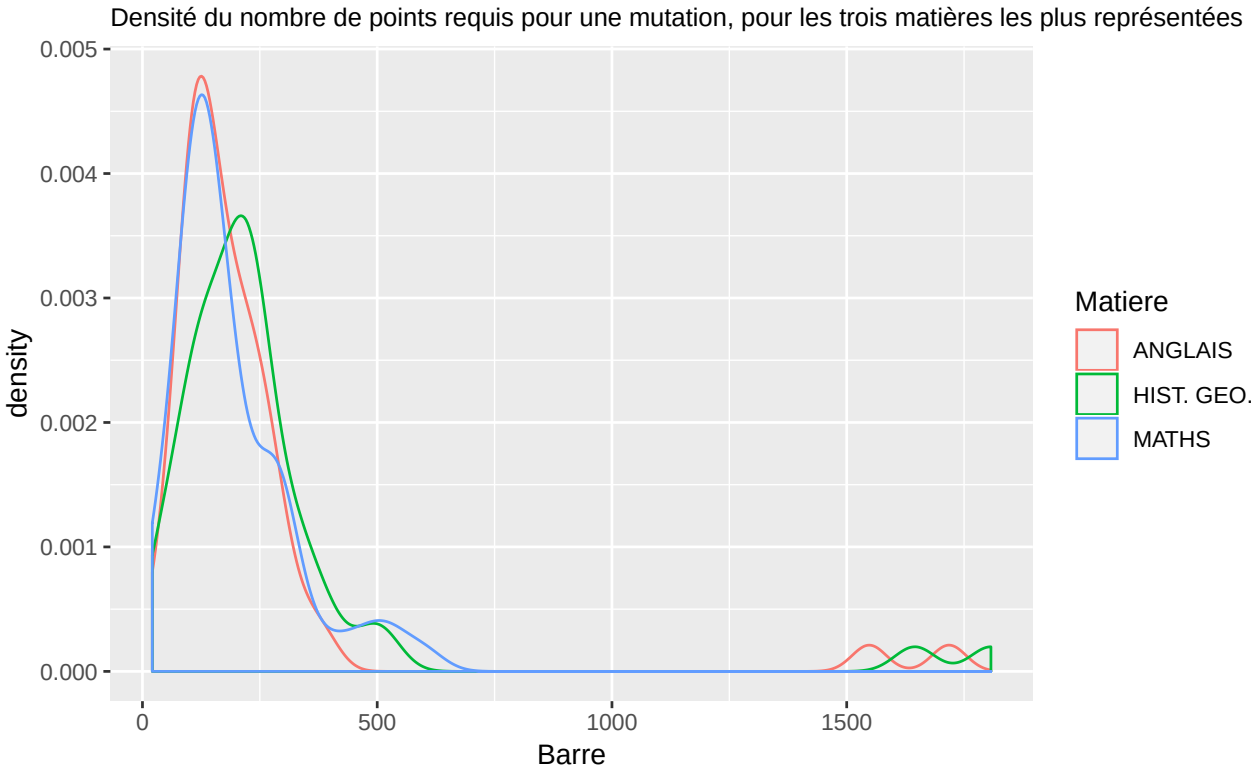
L'histogramme du nombre de points requis pour une mutation nous montre qu'une attention particulière devra être portée à la queue de distribution (représentative des couples établissement/matière difficiles à obtenir).



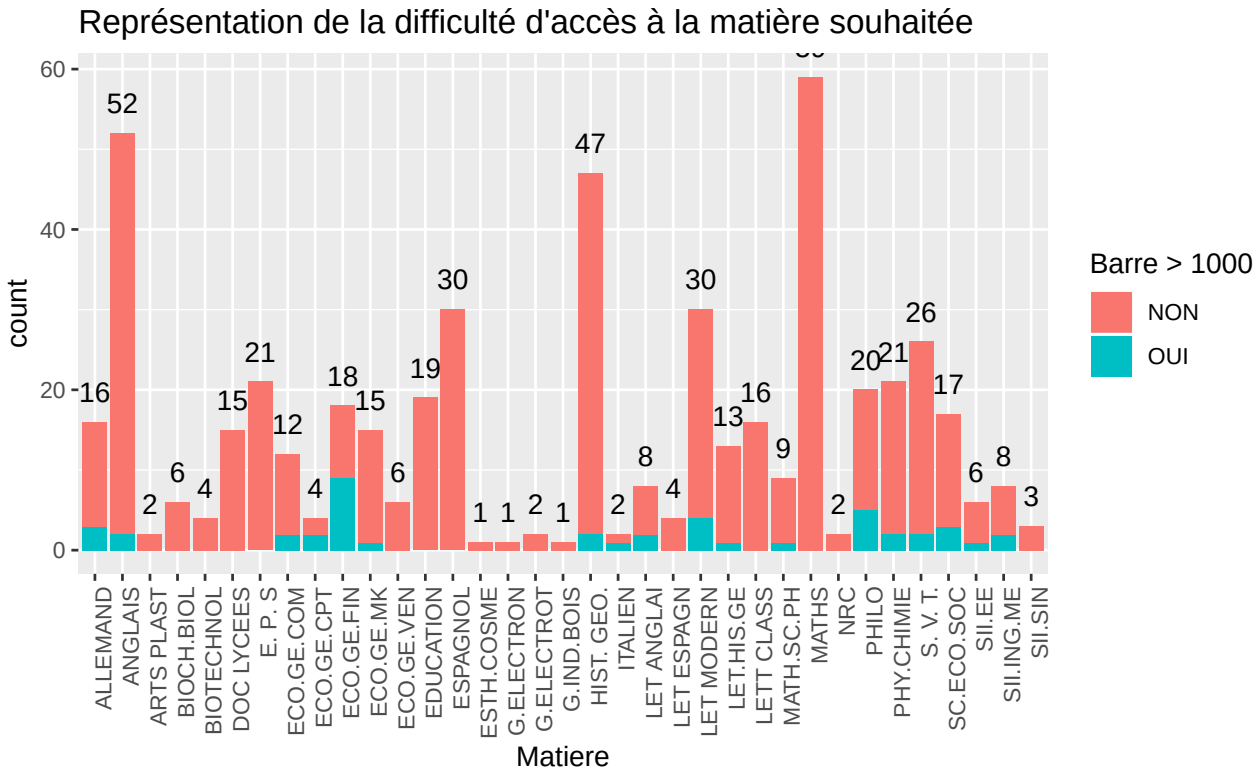
Le graphique ci-dessous nous donne un aperçu de la participation des matières dans le jeu de données.



La densité des trois matières les mieux représentées est assez similaire à celle tracée pour l'ensemble du jeu de données. La matière *MATHS* est toutefois moins concernée par les valeurs extrêmes.



Le graphique ci-dessous nous donne un aperçu de la participation des matières pouvant être difficiles à obtenir dans le jeu de données.

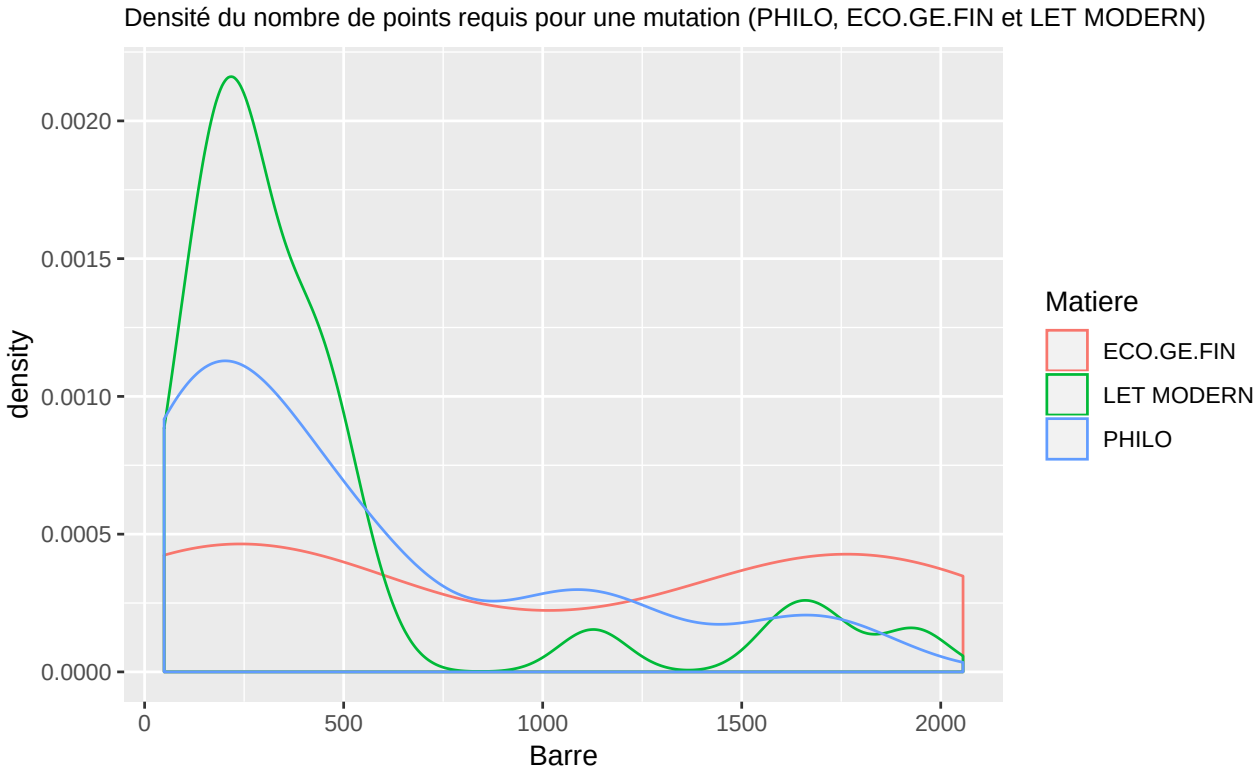


Nous affichons ci-dessous, par matière, la proportion de couples établissement / matière dont le nombre de points requis est supérieur à 1000.

```r
table = table(df1[df1$Barre > 1000, ]$Matiere)/table(df1$Matiere)
table = sort(round(table, 2), decreasing = T)
table
```

```
##
## ECO.GE.CPT ECO.GE.FIN    ITALIEN LET ANGLAI      PHILO SII.ING.ME
##       0.50       0.50       0.50       0.25       0.25       0.25
##   ALLEMAND SC.ECO.SOC ECO.GE.COM     SII.EE LET MODERN MATH.SC.PH
##       0.19       0.18       0.17       0.17       0.13       0.11
## PHY.CHIMIE LET.HIS.GE   S. V. T.  ECO.GE.MK    ANGLAIS HIST. GEO.
##       0.10       0.08       0.08       0.07       0.04       0.04
## ARTS PLAST BIOCH.BIOL BIOTECHNOL DOC LYCEES    E. P. S ECO.GE.VEN
##       0.00       0.00       0.00       0.00       0.00       0.00
##  EDUCATION   ESPAGNOL ESTH.COSME G.ELECTRON G.ELECTROT G.IND.BOIS
##       0.00       0.00       0.00       0.00       0.00       0.00
## LET ESPAGN LETT CLASS      MATHS        NRC    SII.SIN
##       0.00       0.00       0.00       0.00       0.00
```

Nous observons la densité de 3 matières bien représentées dans le jeu de données, et dont la proportion calculée ci-dessus est relativement importante. Nous remarquons que celles-ci sont très différentes. Seule *LET MODERN* a une distribution relativement proche de celle du jeu de données.



2 Régression linéaire

2.1 Question 1

Nous réalisons tout d'abord une régression linéaire fréquentiste. Nous obtenons des *NA* pour les estimations de certains coefficients.

```
summary(lm(Barre ~ ., data = df))
```

Les covariables *commune* et *ville* nous donnent la même information. Nous retenons la covariable *commune*.

```
length(unique(df$ville))
```

```
## [1] 97
```

```
length(unique(df$commune))
```

```
## [1] 97
```

De plus, nous remarquons que certains lycées (4 au total) possèdent le même nom.

```
w1 = unique(df[, c(1, 3)])
unique(w1[duplicated(w1$etablissement), ][2])
```

```
## # A tibble: 4 x 1
##   etablissement
##   <fct>
## 1 LYCEE DESCARTES
## 2 LYCEE JACQUES PREVERT
## 3 LYCEE RENE CASSIN
## 4 LYCEE LEONARD DE VINCI
```

Nous supprimons donc également la covariable *etablissement*, la covariable *code_etablissement* étant suffisante pour identifier chaque lycée de manière unique.

Nous remarquons que les coefficients des covariables quantitatives ne sont pas bien estimés (voir Annexe 1). Ces dernières représentent en effet les caractéristiques intrasèques des établissements, et sont donc toutes corrélées à la covariable *code_etablissement*. Nous choisissons de supprimer cette dernière.

Nous choisissons de normaliser les données. Cette opération est d'ailleurs conseillée dans le chapitre 3 de l'ouvrage **Bayesian Essential for R** (Jean-Michel Marin • Christian P. Robert), le processus inférentiel étant conditionné par la matrice de design *X*. Nous créons cette matrice à l'aide de la fonction *model.matrix* afin de transformer les covariables de type factor en covariables muettes.

```
X = model.matrix(Barre ~ ., data = df2)  #l'intercept est rajouté automatiquement.
X1 = scale(X[, c(2:ncol(X))])
X2 = cbind(X[, 1], X1)
y = as.matrix(df2[, c(2)])
```

Nous allons maintenant procéder à une inférence bayésienne en utilisant la loi a priori de *Zellner*. Nous réutilisons dans un premier temps le code vu en TP4 (voir Annexe 2). Nous fixons le paramètre g égal à n, de sorte que le poids de cette loi soit le même que celui d'une seule observation.

Nous pourrions comparer nos résultats avec ceux de la fonction *BayesReg* du package *bayess*. Cependant, une erreur apparaît lors de l'exécution du code. Ceci est probablement dû à la très grande valeur de *s2* qui fait tendre le log10bf relatif à chaque covariable vers une forme indéterminée.

```
BayesReg(y, X1)
```

```
## Error in if (bayesfactor[i] < 0) evid[i + 1] = "      ": valeur manquante là où TRUE / FALSE est requ
```

Nous pouvons toutefois utiliser une partie du code de cette fonction pour vérifier le calcul de l'espérance des coefficients (a posteriori), ce code ayant pour mérite de ne pas directement faire appel à la fonction *lm* (code en Annexe 3).

```
postmean_bayesreg = bayesreg_modified(y, X1)
```

Les valeurs des coefficients obtenus par les deux méthodes (voir Annexe 4) sont très proches de celles obtenues par l'estimation du maximum de vraissemblance. Ceci est cohérent puisque nous avons fixé $g = n$, ce qui donne une importance faible à la loi a priori. L'impossibilité de calculer le log10bf nous empêche de comparer la significativité des covariables du modèle bayésien à celle du modèle fréquentiste.

Les résultats du modèle fréquentiste sont commentés en Annexe 5.

2.2 Question 2

2.2.1 Caractéristiques des établissements

Nous prenons l'hypothèse que la matière n'influe pas sur le nombre de points nécessaires pour obtenir une mutation dans l'académie. Nous supprimons donc la covariable *Matiere* du jeu de données initial et allons utiliser la méthode d'échantillonnage de Gibbs afin de déterminer quelles covariables inclure dans notre modèle final (voir Annexe 6).

```
df3 = df2[, -c(1)]
X_et = as.matrix(df3[, c(2):ncol(df3)])
X_et = cbind(1, X_et)
y_et = as.matrix(df3[, c(1)])
n_et = length(y_et)
```

Les probabilités de conservation de chaque gamma sont relativement très faibles.

Les deux variables les plus significatives sont :

- taux accès attendu premiere bac
- taux accès attendu seconde bac

```
## # A tibble: 17 x 2
##    meangamma_et row.names
##           <dbl> <chr>
##  1       0.0411 effectif_presents_serie_l
##  2       0.0461 effectif_presents_serie_es
##  3       0.0507 effectif_presents_serie_s
##  4       0.0524 taux_brut_de_reussite_serie_l
##  5       0.0819 taux_brut_de_reussite_serie_es
##  6       0.0747 taux_brut_de_reussite_serie_s
##  7       0.124  taux_reussite_attendu_serie_l
##  8       0.115  taux_reussite_attendu_serie_es
##  9       0.107  taux_reussite_attendu_serie_s
## 10       0.0449 effectif_de_seconde
## 11       0.0481 effectif_de_premiere
## 12       0.0902 taux_acces_brut_seconde_bac
## 13       0.196  taux_acces_attendu_seconde_bac
## 14       0.0787 taux_acces_brut_premiere_bac
## 15       0.333  taux_acces_attendu_premiere_bac
## 16       0.120  taux_brut_de_reussite_total_series
## 17       0.115  taux_reussite_attendu_total_series
```

Les graphs d'autocorrélation (voir Annexe 7) ne présentent pas d'anomalie (décroissance rapide).

Le modèle semble bien converger comme le montrent les graphiques en Annexe 8.

Le meilleur modèle est celui retenant la covariable *taux accès attendu premiere bac*. Sa probabilité a posteriori est cependant relativement faible, ce qui donne une incertitude sur la qualité de prédiction de ce modèle. Les modèles suivants ont tous une probabilité a posteriori inférieure à 10%.

```
##         probtop20
##  [1,] 0.131684211 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
##  [2,] 0.084631579 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
##  [3,] 0.049473684 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
##  [4,] 0.043473684 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
##  [5,] 0.041052632 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
##  [6,] 0.032000000 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0
##  [7,] 0.031894737 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
##  [8,] 0.026842105 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
##  [9,] 0.023157895 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
## [10,] 0.020947368 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
## [11,] 0.018000000 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
## [12,] 0.013578947 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1
## [13,] 0.008000000 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0
## [14,] 0.006631579 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0
## [15,] 0.006631579 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0
## [16,] 0.006526316 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1
## [17,] 0.006315789 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0
## [18,] 0.006210526 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
## [19,] 0.006210526 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0
## [20,] 0.006000000 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0
```

De même, le modèle fréquentiste possède une mauvaise qualité de prédiction illustrée par une faible valeur du R2 (voir Annexe 9).

Au vu de ces résultats, un modèle de type régression linéaire, basé uniquement sur les caractéristiques des établissements, semble peu performant.

2.2.2 Matières

Suite aux précédents résultats, nous faisons désormais l'hypothèse que les caractéristiques des établissements n'influencent pas le nombre de points requis pour une mutation. Nous retenons donc uniquement la variable *Matiere*.

```
df4 = df2[, c(1, 2)]
X_ma = model.matrix(Barre ~ ., data = df4)
y_ma = as.matrix(df4[, c(2)])
n_ma = length(y_ma)
```

Plusieurs matières apparaîssent comme significatives. Nous remarquons des probabilités a posteriori anormalement élevées pour deux matières : *MatiereECO.GE.FIN* et *MatiereECO.GE.CPT*. Ceci est probablement dû à leur faible représentativité dans le jeu de données. Le code utilisé est disponible est Annexe 10.

```
## # A tibble: 34 x 2
##    meangamma_ma row.names
##           <dbl> <chr>
##  1       0.0743 MatiereANGLAIS
##  2       0.0441 MatiereARTS PLAST
##  3       0.0623 MatiereBIOCH.BIOL
##  4       0.0798 MatiereBIOTECHNOL
```

```
##  5       0.147  MatiereDOC LYCEES
##  6       0.0423 MatiereE. P. S
##  7       0.0576 MatiereECO.GE.COM
##  8       0.918  MatiereECO.GE.CPT
##  9       1      MatiereECO.GE.FIN
## 10       0.0424 MatiereECO.GE.MK
## # ... with 24 more rows
```

La convergence est bien est atteinte hormis pour la variable *MatiereECO.GE.FIN*, probablement pour la raison évoquée ci-dessus (voir Annexe 11).

Lors de la sélection de modèle, nous remarquons que :

- les probabilités a posteriori sont très faibles pour l'ensemble des modèles retenus,
- les 2 premiers modèles ont une probabilité a posteriori identiques,
- *ECO GE CPT* et *ECO GE GIN* sont retenues dans tous les modèles.

Pour ce scénario (choix de la variable matière), nous aurions tendance à privilégier le deuxième modèle qui inclut le plus de variables à savoir :

- ECO GE CPT et ECO GE GIN
- ITALIEN
- LETTRES MODERN
- PHILO
- SII.ING.ME

```
##         probtop20
##  [1,] 0.012421053 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1
##  [2,] 0.012421053 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1
##  [3,] 0.010105263 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
##  [4,] 0.009684211 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1
##  [5,] 0.008631579 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1
##  [6,] 0.008210526 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
##  [7,] 0.008000000 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
##  [8,] 0.007368421 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
##  [9,] 0.005578947 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1
## [10,] 0.005368421 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [11,] 0.005052632 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [12,] 0.004947368 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1
## [13,] 0.004842105 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
## [14,] 0.004736842 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
## [15,] 0.004210526 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
## [16,] 0.003894737 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
## [17,] 0.003578947 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
## [18,] 0.003157895 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1
## [19,] 0.002947368 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1
## [20,] 0.002736842 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
##
##  [1,] 0 0 0 0 1 0
##  [2,] 0 0 0 0 1 0
##  [3,] 0 0 0 0 1 0
##  [4,] 0 0 0 0 0 0
##  [5,] 0 0 0 0 0 0
##  [6,] 0 0 0 0 1 0
##  [7,] 0 0 0 0 0 0
##  [8,] 0 0 0 0 0 0
##  [9,] 0 0 0 1 1 0
```

```
## [10,] 0 0 0 0 1 0
## [11,] 0 0 0 0 0 0
## [12,] 0 0 0 1 1 0
## [13,] 0 0 0 1 1 0
## [14,] 0 0 0 0 0 0
## [15,] 0 0 0 1 1 0
## [16,] 0 0 0 0 1 0
## [17,] 0 0 0 0 0 0
## [18,] 0 0 0 0 1 0
## [19,] 0 0 0 0 0 0
## [20,] 0 0 0 0 0 0
```

Les résultats sont cependant très différents de ceux obtenus par une sélection d'un modèle de type *lm* via la fonction *step* (voir Annexe 12). Aucune des variables du modèle précédent, hormis *ECO GE FIN* ne sont ici retenues. Toutefois, le *R2* est bien plus important que celui obtenu dans la partie précédente (0.49).

L'ensemble des résultats nous montre qu'une inférence bayésienne ne tenant compte que des caractéristiques des établissements OU des matières désirées reste limitée dans le but d'expliquer le nombre de points requis pour une mutation.

Cela ne semble pas incohérent car intuitivement, nous pouvons imaginer que l'accessibilité de certaines matières varie en fonction de l'établissement souhaité. C'est ce que nous allons étudier dans la question suivante.

2.3 Question 3

Nous préparons deux jeux de données *df_maths* et *df_anglais*. Les covariables retenues sont les mêmes que celles retenues dans la partie *Caractéristiques des établissements* de la question précédente.

```
df_ang = df2[df2$Matiere == "ANGLAIS", ]
df_mat = df2[df2$Matiere == "MATHS", ]
```

```
df_ang = df_ang[, -c(1)]
X_ang = as.matrix(df_ang[, c(2):ncol(df_ang)])
X_ang = cbind(1, X_ang)
y_ang = as.matrix(df_ang[, c(1)])
n_ang = length(y_ang)

df_mat = df_mat[, -c(1)]
X_mat = as.matrix(df_mat[, c(2):ncol(df_mat)])
X_mat = cbind(1, X_mat)
y_mat = as.matrix(df_mat[, c(1)])
n_mat = length(y_mat)
```

Nous utilisons la méthode d'échantillonnage de Gibbs sur les deux jeux de données. Nous calculons la proportion du temps où les gammas sont conservés pour mesurer leur significativité (voir Annexe 13). Nous remarquons que :

- les probabilités pour les deux modèles sont relativement différentes,
- les probabilités sont beaucoup plus importantes que celles obtenues dans la partie *Caractéristiques des établissements* de la question précédente,
- la majorité des probabilités sont égales ou très proche de 1.

```
## # A tibble: 17 x 3
##    meangamma_ang meangamma_mat row.names
##            <dbl>         <dbl> <chr>
##  1         1.000         0.198 effectif_presents_serie_l
```

```
##  2         0.602         0.31  effectif_presents_serie_es
##  3         0.169         1     effectif_presents_serie_s
##  4         0.997         1     taux_brut_de_reussite_serie_l
##  5         0.953         1     taux_brut_de_reussite_serie_es
##  6         0.159         0.238 taux_brut_de_reussite_serie_s
##  7         0.155         0.61  taux_reussite_attendu_serie_l
##  8         1             0.821 taux_reussite_attendu_serie_es
##  9         1             0.148 taux_reussite_attendu_serie_s
## 10         0.922         1.000 effectif_de_seconde
## 11         0.311         1     effectif_de_premiere
## 12         0.418         1     taux_acces_brut_seconde_bac
## 13         1.000         0.993 taux_acces_attendu_seconde_bac
## 14         1             0.988 taux_acces_brut_premiere_bac
## 15         1             0.402 taux_acces_attendu_premiere_bac
## 16         1             0.435 taux_brut_de_reussite_total_series
## 17         1             0.641 taux_reussite_attendu_total_series
```

Pour chacune des deux analyses, la convergence n'est pas atteinte pour une bonne partie des covariables (voir Annexe 14).

Nous observons que les deux meilleurs modèles retenus ne conservent pas les mêmes covariables. Malgré la fragilité de ces modèles sans doute liée au faible nombre d'observations des jeux de données, nous pouvons tout de même affirmer au vu de ces résultats, que la prédiction du nombre de points requis par les caractéristiques des établissements sera différente selon la matière considérée (*Anglais* ou *Maths*).

```
##                                  bestmodel_maths bestmodel_anglais
## effectif_presents_serie_l                   TRUE              TRUE
## effectif_presents_serie_es                 FALSE              TRUE
## effectif_presents_serie_s                   TRUE             FALSE
## taux_brut_de_reussite_serie_l               TRUE              TRUE
## taux_brut_de_reussite_serie_es              TRUE              TRUE
## taux_brut_de_reussite_serie_s              FALSE             FALSE
## taux_reussite_attendu_serie_l               TRUE             FALSE
## taux_reussite_attendu_serie_es              TRUE              TRUE
## taux_reussite_attendu_serie_s              FALSE              TRUE
## effectif_de_seconde                         TRUE              TRUE
## effectif_de_premiere                        TRUE             FALSE
## taux_acces_brut_seconde_bac                 TRUE             FALSE
## taux_acces_attendu_seconde_bac              TRUE              TRUE
## taux_acces_brut_premiere_bac                TRUE              TRUE
## taux_acces_attendu_premiere_bac             TRUE              TRUE
## taux_brut_de_reussite_total_series         FALSE              TRUE
## taux_reussite_attendu_total_series          TRUE              TRUE
```

3 Loi de Pareto

3.1 Question 4

Nous allons utiliser le package *VGAM*.

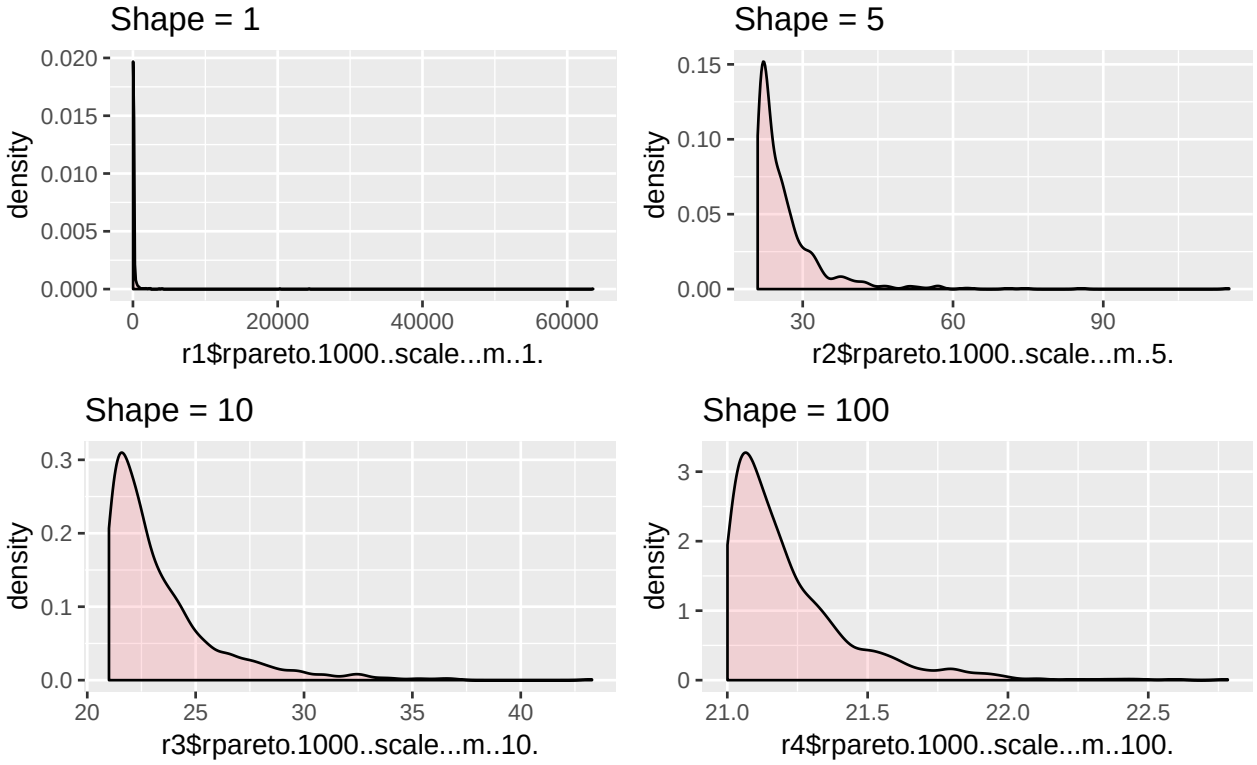
Nous faisons l'hypothèse que le nombre de points requis pour une admission suit une loi de Pareto de paramètres m et *alpha*.

Nous fixons m = 21 qui est le minimum de la covariable *Barre*. *alpha* est inconnu.

Nous allons tout d'abord générer des réalisations d'une loi de Pareto afin d'étudier l'influence du paramètre *alpha*.

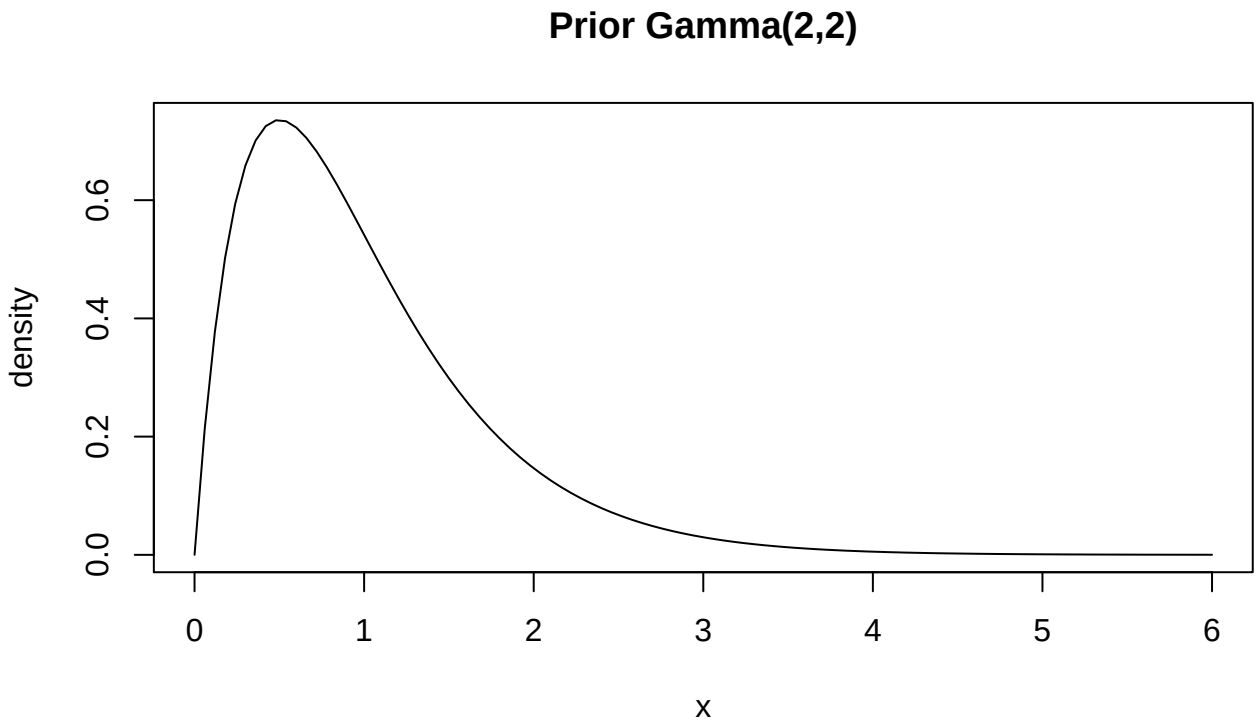
Nous remarquons que :

— cette loi est plus proche de nos données que la loi gaussienne,
— le paramètre *alpha* influe sur la hauteur de la distribution (plus *alpha* est grand, plus la hauteur de la distribution est importante).



3.2 Question 5

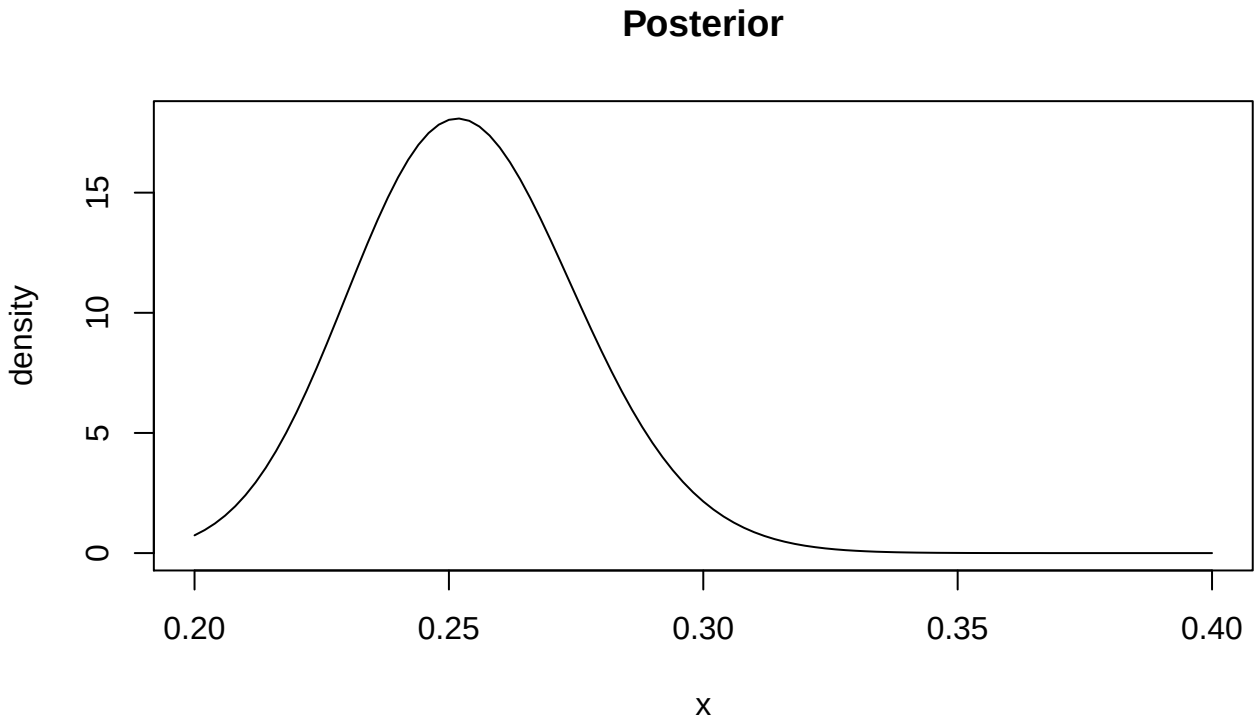
La distribution de Pareto est reliée à la distribution exponentielle. Nous choisissons donc une loi a priori Gamma (2,2) pour le paramètre *alpha*.



3.3 Question 6

Nous en déduisons la loi a posteriori du paramètre *alpha* (après réécriture de la densité de la loi de Pareto).

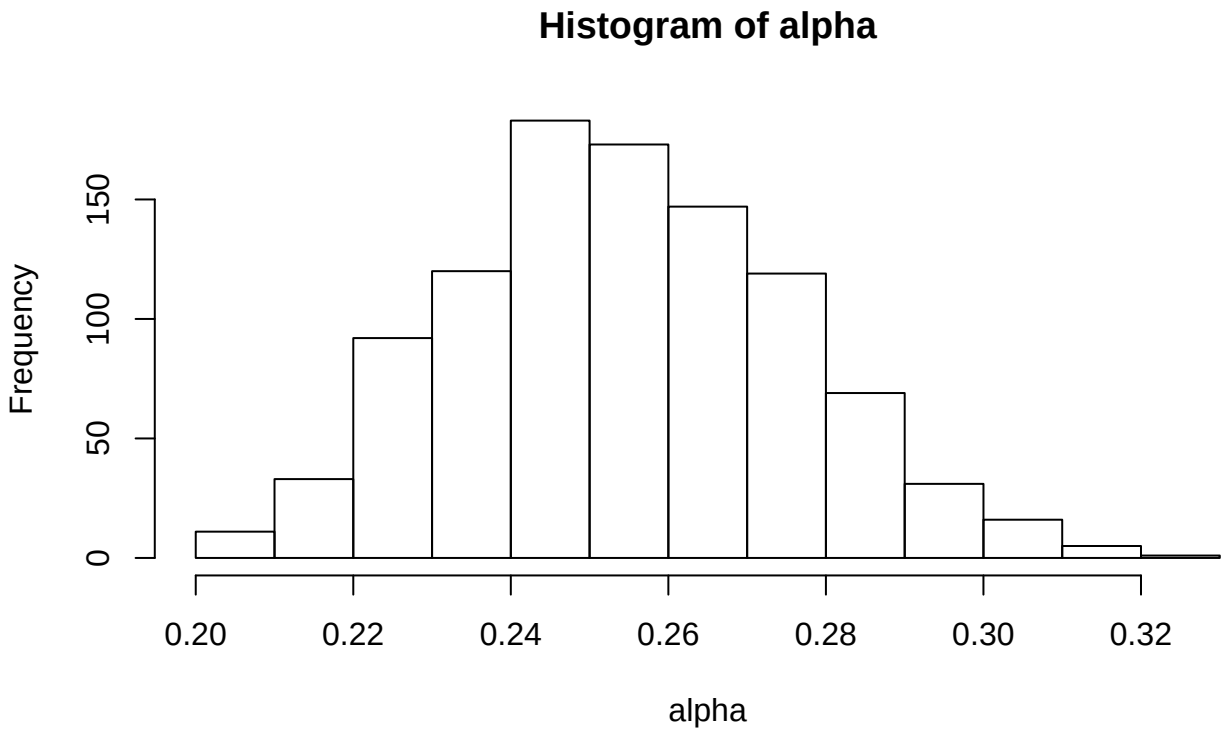
```
curve(dgamma(x, 2 + sum(log(df$Barre)/m), 2 + n), xlim = c(0.2, 0.4), main = "Posterior", ylab = "densi
```



3.4 Question 7

Nous tirons 1000 réalisations de la loi a posteriori.

```
niter = 1000
alpha = rgamma(niter, 2 + sum(log(df$Barre)/m), 2 + n)
hist(alpha)
```

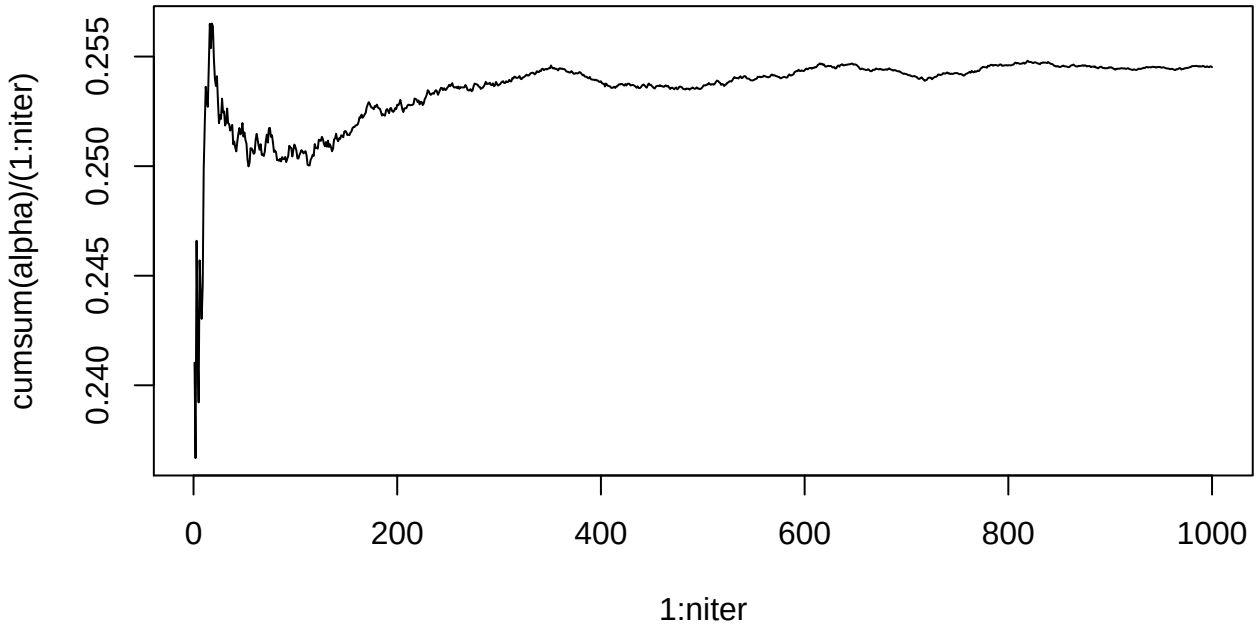


L'intervalle de crédibilité à 95% est donnée ci-dessous.

```
##      2.5%     97.5%
## 0.2148393 0.2979285
```

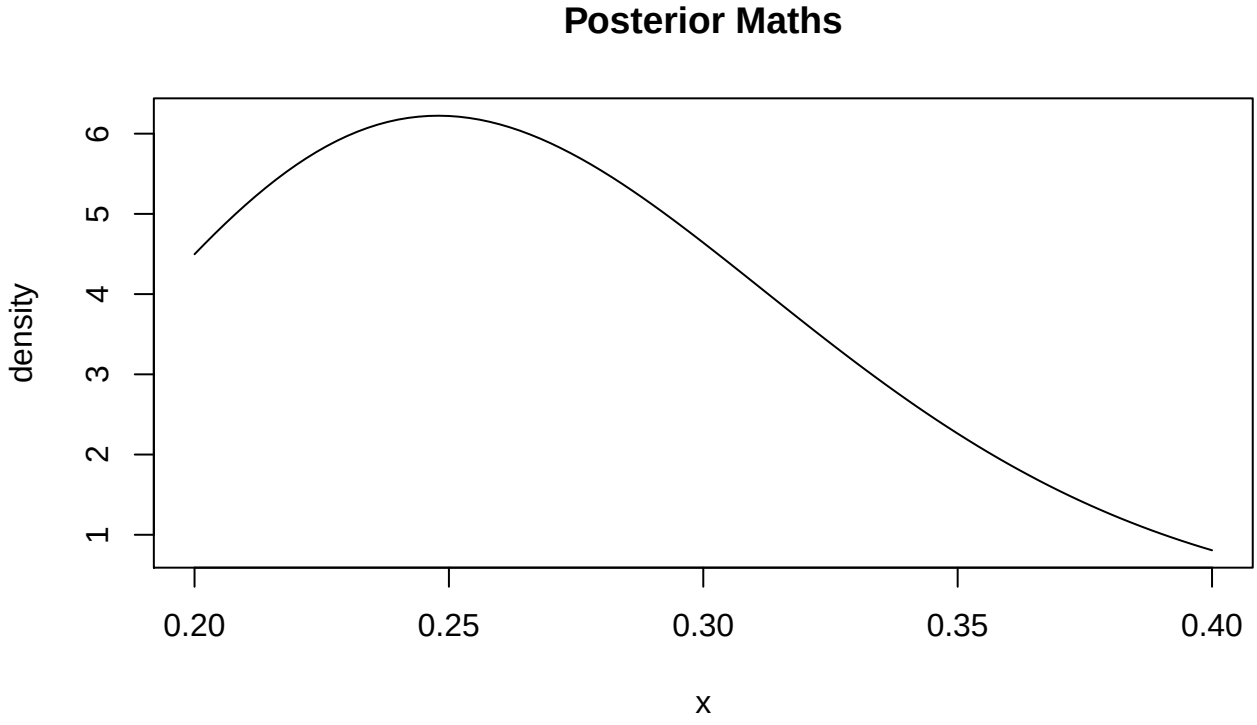
La convergence de l'estimateur vers la moyenne est bien atteinte.

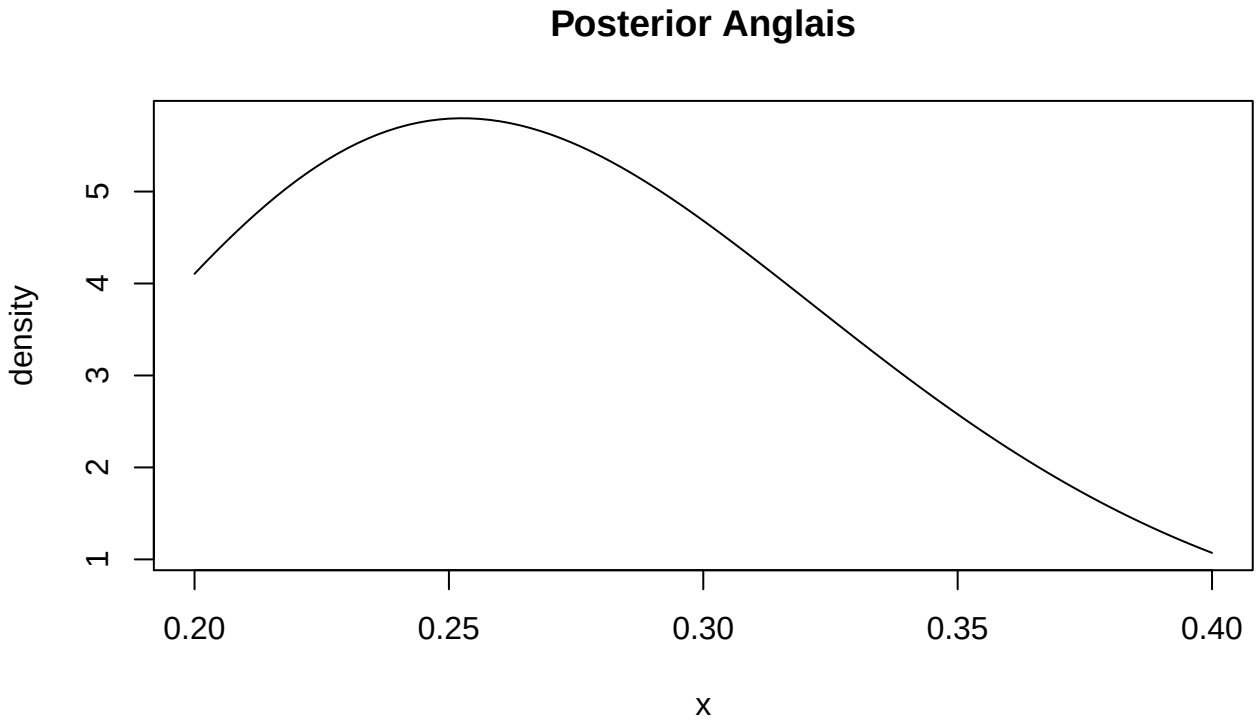
```
plot(1:niter, cumsum(alpha)/(1:niter), type = "l")
```



3.5 Question 8

Nous observons la loi a posteriori pour les deux jeux de données *df_mat* et *df_ang*.





Les résultats obtenus concernant le paramètre *alpha* sont relativement proches pour les deux matières étudiées. Ce qui nous incite finalement à penser que dans ce cas précis, la matière n'influe pratiquement pas à elle seule sur le nombre de points requis pour une mutation. Ces résultats sont ainsi différents de ceux obtenus dans le cadre d'une régression qui tient compte des caractéristiques des établissements.

```
##              mean         sd quantile 2.5% quantile 97.5%
## Maths   0.2648582 0.06593820     0.1519980      0.4160059
## Anglais 0.2714308 0.07061047     0.1518592      0.4290856
```

4 Annexes

4.1 Annexe 1

```
summary(lm(Barre ~ ., data = df2))
```

```
##
## Call:
## lm(formula = Barre ~ ., data = df2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -925.77 -161.56  -63.28   54.97 1466.67
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      -90.73748  547.87706  -0.166 0.868531
## MatiereANGLAIS                  -360.98473  112.67970  -3.204 0.001450
## MatiereARTS PLAST               -360.18334  300.85106  -1.197 0.231834
## MatiereBIOCH.BIOL               -484.96726  188.87743  -2.568 0.010552
## MatiereBIOTECHNOL               -475.05593  220.56842  -2.154 0.031772
## MatiereDOC LYCEES               -494.42336  140.96926  -3.507 0.000497
## MatiereE. P. S                  -311.60003  131.31991  -2.373 0.018059
## MatiereECO.GE.COM               -182.19980  150.84796  -1.208 0.227725
## MatiereECO.GE.CPT                425.09203  219.94565   1.933 0.053880
## MatiereECO.GE.FIN                365.68423  134.98738   2.709 0.006998
## MatiereECO.GE.MK                -296.94042  141.45669  -2.099 0.036342
## MatiereECO.GE.VEN               -431.60571  190.34790  -2.267 0.023821
## MatiereEDUCATION                -443.21331  133.96005  -3.309 0.001011
## MatiereESPAGNOL                 -423.00722  121.37224  -3.485 0.000538
## MatiereESTH.COSME               -473.78911  403.88348  -1.173 0.241364
## MatiereG.ELECTRON               -270.31613  407.89879  -0.663 0.507849
## MatiereG.ELECTROT               -306.53690  298.38398  -1.027 0.304803
## MatiereG.IND.BOIS               -623.95416  403.81442  -1.545 0.122992
## MatiereHIST. GEO.               -327.52268  113.92779  -2.875 0.004228
## MatiereITALIEN                   474.16224  295.29331   1.606 0.109013
## MatiereLET ANGLAI               -157.99756  172.95003  -0.914 0.361431
## MatiereLET ESPAGN               -471.81430  220.48041  -2.140 0.032881
## MatiereLET MODERN               -141.10189  121.87932  -1.158 0.247575
## MatiereLET.HIS.GE               -371.82799  148.63855  -2.502 0.012708
## MatiereLETT CLASS               -373.74494  140.21515  -2.666 0.007955
## MatiereMATH.SC.PH               -281.19226  165.87042  -1.695 0.090698
## MatiereMATHS                    -419.52968  110.58136  -3.794 0.000168
## MatiereNRC                      -564.46178  295.78973  -1.908 0.056967
## MatierePHILO                     -94.58707  131.78642  -0.718 0.473285
## MatierePHY.CHIMIE               -291.39934  130.73389  -2.229 0.026295
## MatiereS. V. T.                 -278.51659  125.51157  -2.219 0.026966
## MatiereSC.ECO.SOC               -253.57611  137.51884  -1.844 0.065830
## MatiereSII.EE                     26.60948  189.46860   0.140 0.888371
## MatiereSII.ING.ME                 19.74647  170.35644   0.116 0.907772
## MatiereSII.SIN                  -442.81552  246.90563  -1.793 0.073550
## effectif_presents_serie_l          0.92968    1.55022   0.600 0.548992
## effectif_presents_serie_es         0.06583    1.17317   0.056 0.955273
```

```
## effectif_presents_serie_s             0.77489    0.95696   0.810 0.418500
## taux_brut_de_reussite_serie_l         2.21139    2.42754   0.911 0.362788
## taux_brut_de_reussite_serie_es        6.06538    4.00649   1.514 0.130735
## taux_brut_de_reussite_serie_s         9.68923    6.03674   1.605 0.109164
## taux_reussite_attendu_serie_l       -12.05843    6.45609  -1.868 0.062425
## taux_reussite_attendu_serie_es        0.92899    7.83174   0.119 0.905629
## taux_reussite_attendu_serie_s        -5.95794    9.17520  -0.649 0.516432
## effectif_de_seconde                   0.16048    0.58618   0.274 0.784381
## effectif_de_premiere                 -0.80185    0.67556  -1.187 0.235858
## taux_acces_brut_seconde_bac           8.31490    5.34155   1.557 0.120236
## taux_acces_attendu_seconde_bac       -3.56988    8.74493  -0.408 0.683298
## taux_acces_brut_premiere_bac        -15.87927   10.14187  -1.566 0.118097
## taux_acces_attendu_premiere_bac      33.07093   18.58140   1.780 0.075765
## taux_brut_de_reussite_total_series  -10.25657   12.09227  -0.848 0.396768
## taux_reussite_attendu_total_series   -2.30704   20.94181  -0.110 0.912327
##
## (Intercept)
## MatiereANGLAIS                     **
## MatiereARTS PLAST
## MatiereBIOCH.BIOL                  *
## MatiereBIOTECHNOL                  *
## MatiereDOC LYCEES                  ***
## MatiereE. P. S                     *
## MatiereECO.GE.COM
## MatiereECO.GE.CPT                  .
## MatiereECO.GE.FIN                  **
## MatiereECO.GE.MK                   *
## MatiereECO.GE.VEN                  *
## MatiereEDUCATION                   **
## MatiereESPAGNOL                    ***
## MatiereESTH.COSME
## MatiereG.ELECTRON
## MatiereG.ELECTROT
## MatiereG.IND.BOIS
## MatiereHIST. GEO.                  **
## MatiereITALIEN
## MatiereLET ANGLAI
## MatiereLET ESPAGN                  *
## MatiereLET MODERN
## MatiereLET.HIS.GE                  *
## MatiereLETT CLASS                  **
## MatiereMATH.SC.PH                  .
## MatiereMATHS                       ***
## MatiereNRC                         .
## MatierePHILO
## MatierePHY.CHIMIE                  *
## MatiereS. V. T.                    *
## MatiereSC.ECO.SOC                  .
## MatiereSII.EE
## MatiereSII.ING.ME
## MatiereSII.SIN                     .
## effectif_presents_serie_l
## effectif_presents_serie_es
## effectif_presents_serie_s
```

```
## taux_brut_de_reussite_serie_l
## taux_brut_de_reussite_serie_es
## taux_brut_de_reussite_serie_s
## taux_reussite_attendu_serie_l      .
## taux_reussite_attendu_serie_es
## taux_reussite_attendu_serie_s
## effectif_de_seconde
## effectif_de_premiere
## taux_acces_brut_seconde_bac
## taux_acces_attendu_seconde_bac
## taux_acces_brut_premiere_bac
## taux_acces_attendu_premiere_bac    .
## taux_brut_de_reussite_total_series
## taux_reussite_attendu_total_series
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 388.1 on 464 degrees of freedom
## Multiple R-squared:  0.2452, Adjusted R-squared:  0.1623
## F-statistic: 2.956 on 51 and 464 DF,  p-value: 7.802e-10
```

4.2 Annexe 2

```
betahat = (lm(y ~ X2 - 1))$coefficients
residuals = (lm(y ~ X2 - 1))$residuals
s2 = t(residuals) %*% residuals
n = length(y)
g = n
postmean_TD4 = betahat * g/(g + 1)  # espérance de beta a posteriori
a = n/2
b = s2/2 + 1/(2 * g + 2) * ((t(betahat) %*% t(X2)) %*% (X2 %*% betahat))
b/(a - 1)  # espérance de sigma2
```

```
##          [,1]
## [1,] 136273.2
```

4.3 Annexe 3

```
bayesreg_modified = function(y, X, g = length(y), betatilde = rep(0, dim(X)[2]), prt = TRUE) {
    X = as.matrix(X)
    g = length(y)
    p = dim(X)[2]
    if (det(t(X) %*% X) <= 1e-07)
        stop("The design matriX1 has a rank lower than the number of eX1planatory variables!\nCalculati
            call. = FALSE)
    U = solve(t(X) %*% X) %*% t(X)
    alphaml = mean(y)
    betaml = U %*% y
    s2 = t(y - alphaml - X %*% betaml) %*% (y - alphaml - X %*% betaml)
    kappa = as.numeric(s2 + t(betatilde - betaml) %*% t(X) %*% X %*% (betatilde - betaml)/(g +
        1))
    malphabayes = alphaml
    mbetabayes = g/(g + 1) * (betaml + betatilde/g)
    msigma2bayes = kappa/(n - 3)
    valphabayes = kappa/(n * (n - 3))
    vbetabayes = diag(kappa * g/((g + 1) * (n - 3)) * solve(t(X) %*% X))
    vsigma2bayes = 2 * kappa^2/((n - 3) * (n - 4))
    postmean = c(malphabayes, mbetabayes)
    postsqrt = sqrt(c(valphabayes, vbetabayes))
    return(postmean)
}
```

4.4 Annexe 4

```
pmVSpm2VSlm = data.frame(postmean_bayesreg, postmean_TD4, betahat)
pmVSpm2VSlm
```

```
##                                    postmean_bayesreg postmean_TD4
## X2                                        321.915504   321.292843
## X2MatiereANGLAIS                         -108.562589  -108.562589
## X2MatiereARTS PLAST                       -22.358922   -22.358922
## X2MatiereBIOCH.BIOL                       -51.940225   -51.940225
## X2MatiereBIOTECHNOL                       -41.623674   -41.623674
## X2MatiereDOC LYCEES                       -82.983963   -82.983963
## X2MatiereE. P. S                          -61.509249   -61.509249
## X2MatiereECO.GE.COM                       -27.433705   -27.433705
## X2MatiereECO.GE.CPT                        37.245913    37.245913
## X2MatiereECO.GE.FIN                        67.032875    67.032875
## X2MatiereECO.GE.MK                        -49.838448   -49.838448
## X2MatiereECO.GE.VEN                       -46.225177   -46.225177
## X2MatiereEDUCATION                        -83.387025   -83.387025
## X2MatiereESPAGNOL                         -98.891107   -98.891107
## X2MatiereESTH.COSME                       -20.817059   -20.817059
## X2MatiereG.ELECTRON                       -11.876987   -11.876987
## X2MatiereG.ELECTROT                       -19.028739   -19.028739
## X2MatiereG.IND.BOIS                       -27.414920   -27.414920
## X2MatiereHIST. GEO.                       -94.147204   -94.147204
## X2MatiereITALIEN                           29.434334    29.434334
## X2MatiereLET ANGLAI                       -19.501046   -19.501046
## X2MatiereLET ESPAGN                       -41.339647   -41.339647
## X2MatiereLET MODERN                       -32.986959   -32.986959
## X2MatiereLET.HIS.GE                       -58.214133   -58.214133
## X2MatiereLETT CLASS                       -64.721863   -64.721863
## X2MatiereMATH.SC.PH                       -36.775560   -36.775560
## X2MatiereMATHS                           -133.375939  -133.375939
## X2MatiereNRC                              -35.039814   -35.039814
## X2MatierePHILO                            -18.239725   -18.239725
## X2MatierePHY.CHIMIE                       -57.521673   -57.521673
## X2MatiereS. V. T.                         -60.864840   -60.864840
## X2MatiereSC.ECO.SOC                       -45.218252   -45.218252
## X2MatiereSII.EE                             2.849888     2.849888
## X2MatiereSII.ING.ME                         2.437233     2.437233
## X2MatiereSII.SIN                          -33.633560   -33.633560
## X2effectif_presents_serie_l                19.530761    19.530761
## X2effectif_presents_serie_es                2.259850     2.259850
## X2effectif_presents_serie_s                44.847901    44.847901
## X2taux_brut_de_reussite_serie_l            25.536107    25.536107
## X2taux_brut_de_reussite_serie_es           59.704490    59.704490
## X2taux_brut_de_reussite_serie_s            88.042032    88.042032
## X2taux_reussite_attendu_serie_l           -89.360650   -89.360650
## X2taux_reussite_attendu_serie_es            7.864117     7.864117
## X2taux_reussite_attendu_serie_s           -55.859936   -55.859936
## X2effectif_de_seconde                      21.720917    21.720917
## X2effectif_de_premiere                   -101.228687  -101.228687
## X2taux_acces_brut_seconde_bac              75.405059    75.405059
## X2taux_acces_attendu_seconde_bac          -25.739243   -25.739243
```

```
## X2taux_acces_brut_premiere_bac             -109.058562  -109.058562
## X2taux_acces_attendu_premiere_bac           197.625487   197.625487
## X2taux_brut_de_reussite_total_series        -75.693606   -75.693606
## X2taux_reussite_attendu_total_series        -17.768477   -17.768477
##                                            betahat
## X2                                       321.915504
## X2MatiereANGLAIS                        -108.772982
## X2MatiereARTS PLAST                      -22.402253
## X2MatiereBIOCH.BIOL                      -52.040884
## X2MatiereBIOTECHNOL                      -41.704340
## X2MatiereDOC LYCEES                      -83.144785
## X2MatiereE. P. S                         -61.628453
## X2MatiereECO.GE.COM                      -27.486871
## X2MatiereECO.GE.CPT                       37.318095
## X2MatiereECO.GE.FIN                       67.162784
## X2MatiereECO.GE.MK                       -49.935034
## X2MatiereECO.GE.VEN                      -46.314761
## X2MatiereEDUCATION                       -83.548628
## X2MatiereESPAGNOL                        -99.082756
## X2MatiereESTH.COSME                      -20.857402
## X2MatiereG.ELECTRON                      -11.900004
## X2MatiereG.ELECTROT                      -19.065616
## X2MatiereG.IND.BOIS                      -27.468050
## X2MatiereHIST. GEO.                      -94.329660
## X2MatiereITALIEN                          29.491377
## X2MatiereLET ANGLAI                      -19.538839
## X2MatiereLET ESPAGN                      -41.419763
## X2MatiereLET MODERN                      -33.050887
## X2MatiereLET.HIS.GE                      -58.326951
## X2MatiereLETT CLASS                      -64.847293
## X2MatiereMATH.SC.PH                      -36.846831
## X2MatiereMATHS                          -133.634419
## X2MatiereNRC                             -35.107720
## X2MatierePHILO                           -18.275074
## X2MatierePHY.CHIMIE                      -57.633149
## X2MatiereS. V. T.                        -60.982795
## X2MatiereSC.ECO.SOC                      -45.305884
## X2MatiereSII.EE                            2.855411
## X2MatiereSII.ING.ME                        2.441957
## X2MatiereSII.SIN                         -33.698741
## X2effectif_presents_serie_l               19.568612
## X2effectif_presents_serie_es               2.264229
## X2effectif_presents_serie_s               44.934815
## X2taux_brut_de_reussite_serie_l           25.585596
## X2taux_brut_de_reussite_serie_es          59.820196
## X2taux_brut_de_reussite_serie_s           88.212657
## X2taux_reussite_attendu_serie_l          -89.533830
## X2taux_reussite_attendu_serie_es           7.879357
## X2taux_reussite_attendu_serie_s          -55.968191
## X2effectif_de_seconde                     21.763012
## X2effectif_de_premiere                  -101.424867
## X2taux_acces_brut_seconde_bac             75.551192
## X2taux_acces_attendu_seconde_bac         -25.789126
## X2taux_acces_brut_premiere_bac          -109.269915
```

```
## X2taux_acces_attendu_premiere_bac     198.008482
## X2taux_brut_de_reussite_total_series  -75.840299
## X2taux_reussite_attendu_total_series  -17.802912
```

4.5 Annexe 5

```
summary(lm(y ~ X2 - 1))
```

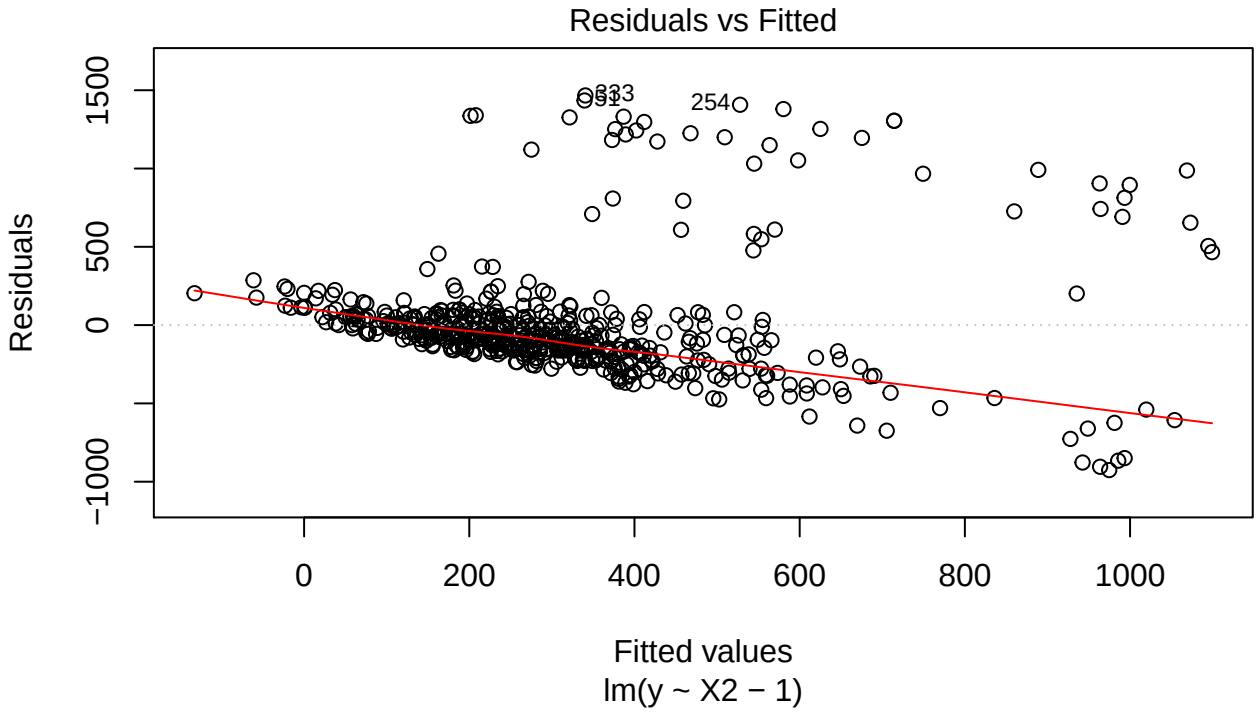
```
##
## Call:
## lm(formula = y ~ X2 - 1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -925.77 -161.56  -63.28   54.97 1466.67
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## X2                                  321.916     17.086  18.841  < 2e-16
## X2MatiereANGLAIS                   -108.773     33.953  -3.204 0.001450
## X2MatiereARTS PLAST                 -22.402     18.712  -1.197 0.231834
## X2MatiereBIOCH.BIOL                 -52.041     20.268  -2.568 0.010552
## X2MatiereBIOTECHNOL                 -41.704     19.363  -2.154 0.031772
## X2MatiereDOC LYCEES                 -83.145     23.706  -3.507 0.000497
## X2MatiereE. P. S                    -61.628     25.973  -2.373 0.018059
## X2MatiereECO.GE.COM                 -27.487     22.757  -1.208 0.227725
## X2MatiereECO.GE.CPT                  37.318     19.309   1.933 0.053880
## X2MatiereECO.GE.FIN                  67.163     24.792   2.709 0.006998
## X2MatiereECO.GE.MK                  -49.935     23.788  -2.099 0.036342
## X2MatiereECO.GE.VEN                 -46.315     20.426  -2.267 0.023821
## X2MatiereEDUCATION                  -83.549     25.252  -3.309 0.001011
## X2MatiereESPAGNOL                   -99.083     28.430  -3.485 0.000538
## X2MatiereESTH.COSME                 -20.857     17.780  -1.173 0.241364
## X2MatiereG.ELECTRON                 -11.900     17.957  -0.663 0.507849
## X2MatiereG.ELECTROT                 -19.066     18.559  -1.027 0.304803
## X2MatiereG.IND.BOIS                 -27.468     17.777  -1.545 0.122992
## X2MatiereHIST. GEO.                 -94.330     32.812  -2.875 0.004228
## X2MatiereITALIEN                     29.491     18.366   1.606 0.109013
## X2MatiereLET ANGLAI                 -19.539     21.388  -0.914 0.361431
## X2MatiereLET ESPAGN                 -41.420     19.356  -2.140 0.032881
## X2MatiereLET MODERN                 -33.051     28.548  -1.158 0.247575
## X2MatiereLET.HIS.GE                 -58.327     23.316  -2.502 0.012708
## X2MatiereLETT CLASS                 -64.847     24.328  -2.666 0.007955
## X2MatiereMATH.SC.PH                 -36.847     21.735  -1.695 0.090698
## X2MatiereMATHS                     -133.634     35.224  -3.794 0.000168
## X2MatiereNRC                        -35.108     18.397  -1.908 0.056967
## X2MatierePHILO                      -18.275     25.462  -0.718 0.473285
## X2MatierePHY.CHIMIE                 -57.633     25.857  -2.229 0.026295
## X2MatiereS. V. T.                   -60.983     27.481  -2.219 0.026966
## X2MatiereSC.ECO.SOC                 -45.306     24.570  -1.844 0.065830
## X2MatiereSII.EE                       2.855     20.332   0.140 0.888371
## X2MatiereSII.ING.ME                   2.442     21.067   0.116 0.907772
## X2MatiereSII.SIN                    -33.699     18.790  -1.793 0.073550
## X2effectif_presents_serie_l          19.569     32.630   0.600 0.548992
## X2effectif_presents_serie_es          2.264     40.349   0.056 0.955273
## X2effectif_presents_serie_s          44.935     55.492   0.810 0.418500
## X2taux_brut_de_reussite_serie_l      25.586     28.086   0.911 0.362788
## X2taux_brut_de_reussite_serie_es     59.820     39.514   1.514 0.130735
```

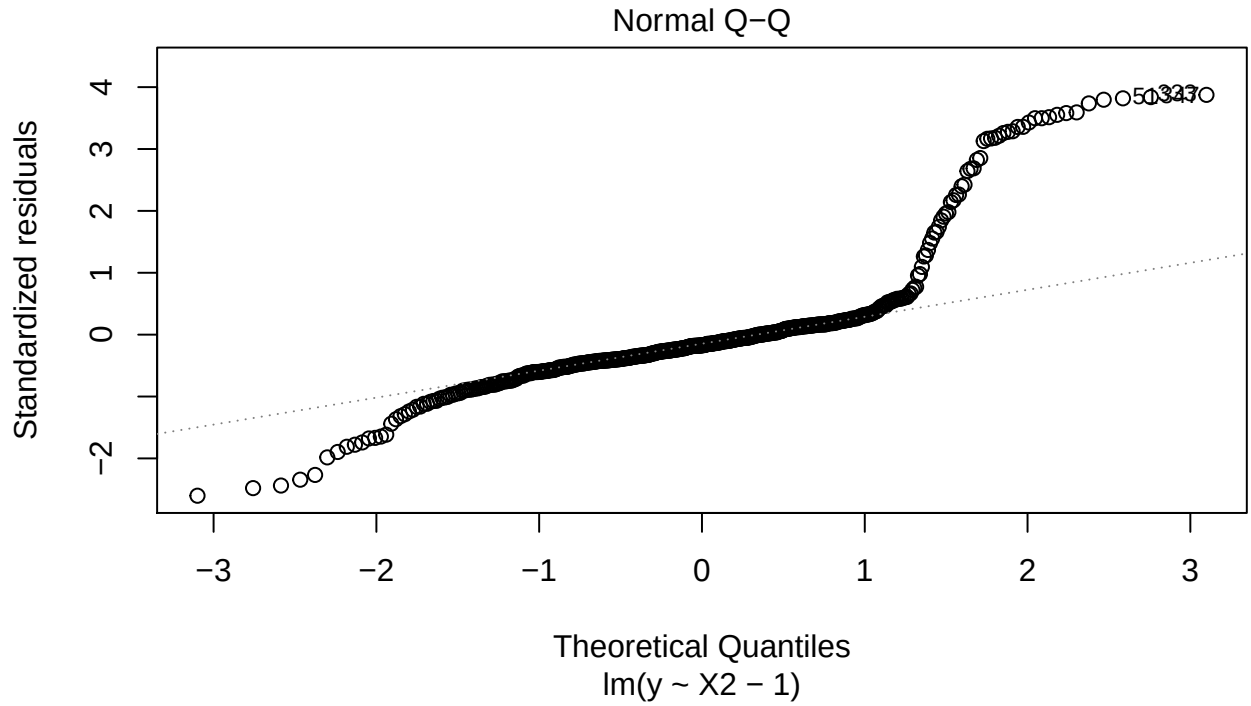
```
## X2taux_brut_de_reussite_serie_s        88.213     54.960   1.605 0.109164
## X2taux_reussite_attendu_serie_l       -89.534     47.936  -1.868 0.062425
## X2taux_reussite_attendu_serie_es        7.879     66.426   0.119 0.905629
## X2taux_reussite_attendu_serie_s       -55.968     86.191  -0.649 0.516432
## X2effectif_de_seconde                  21.763     79.493   0.274 0.784381
## X2effectif_de_premiere               -101.425     85.451  -1.187 0.235858
## X2taux_acces_brut_seconde_bac          75.551     48.535   1.557 0.120236
## X2taux_acces_attendu_seconde_bac      -25.789     63.174  -0.408 0.683298
## X2taux_acces_brut_premiere_bac       -109.270     69.789  -1.566 0.118097
## X2taux_acces_attendu_premiere_bac     198.008    111.254   1.780 0.075765
## X2taux_brut_de_reussite_total_series  -75.840     89.414  -0.848 0.396768
## X2taux_reussite_attendu_total_series  -17.803    161.603  -0.110 0.912327
##
## X2                                   ***
## X2MatiereANGLAIS                     **
## X2MatiereARTS PLAST
## X2MatiereBIOCH.BIOL                  *
## X2MatiereBIOTECHNOL                  *
## X2MatiereDOC LYCEES                  ***
## X2MatiereE. P. S                     *
## X2MatiereECO.GE.COM
## X2MatiereECO.GE.CPT                  .
## X2MatiereECO.GE.FIN                  **
## X2MatiereECO.GE.MK                   *
## X2MatiereECO.GE.VEN                  *
## X2MatiereEDUCATION                   **
## X2MatiereESPAGNOL                    ***
## X2MatiereESTH.COSME
## X2MatiereG.ELECTRON
## X2MatiereG.ELECTROT
## X2MatiereG.IND.BOIS
## X2MatiereHIST. GEO.                  **
## X2MatiereITALIEN
## X2MatiereLET ANGLAI
## X2MatiereLET ESPAGN                  *
## X2MatiereLET MODERN
## X2MatiereLET.HIS.GE                  *
## X2MatiereLETT CLASS                  **
## X2MatiereMATH.SC.PH                  .
## X2MatiereMATHS                       ***
## X2MatiereNRC                         .
## X2MatierePHILO
## X2MatierePHY.CHIMIE                  *
## X2MatiereS. V. T.                    *
## X2MatiereSC.ECO.SOC                  .
## X2MatiereSII.EE
## X2MatiereSII.ING.ME
## X2MatiereSII.SIN                     .
## X2effectif_presents_serie_l
## X2effectif_presents_serie_es
## X2effectif_presents_serie_s
## X2taux_brut_de_reussite_serie_l
## X2taux_brut_de_reussite_serie_es
## X2taux_brut_de_reussite_serie_s
```

```
## X2taux_reussite_attendu_serie_l      .
## X2taux_reussite_attendu_serie_es
## X2taux_reussite_attendu_serie_s
## X2effectif_de_seconde
## X2effectif_de_premiere
## X2taux_acces_brut_seconde_bac
## X2taux_acces_attendu_seconde_bac
## X2taux_acces_brut_premiere_bac
## X2taux_acces_attendu_premiere_bac    .
## X2taux_brut_de_reussite_total_series
## X2taux_reussite_attendu_total_series
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 388.1 on 464 degrees of freedom
## Multiple R-squared:  0.5215, Adjusted R-squared:  0.4679
## F-statistic: 9.726 on 52 and 464 DF,  p-value: < 2.2e-16
```

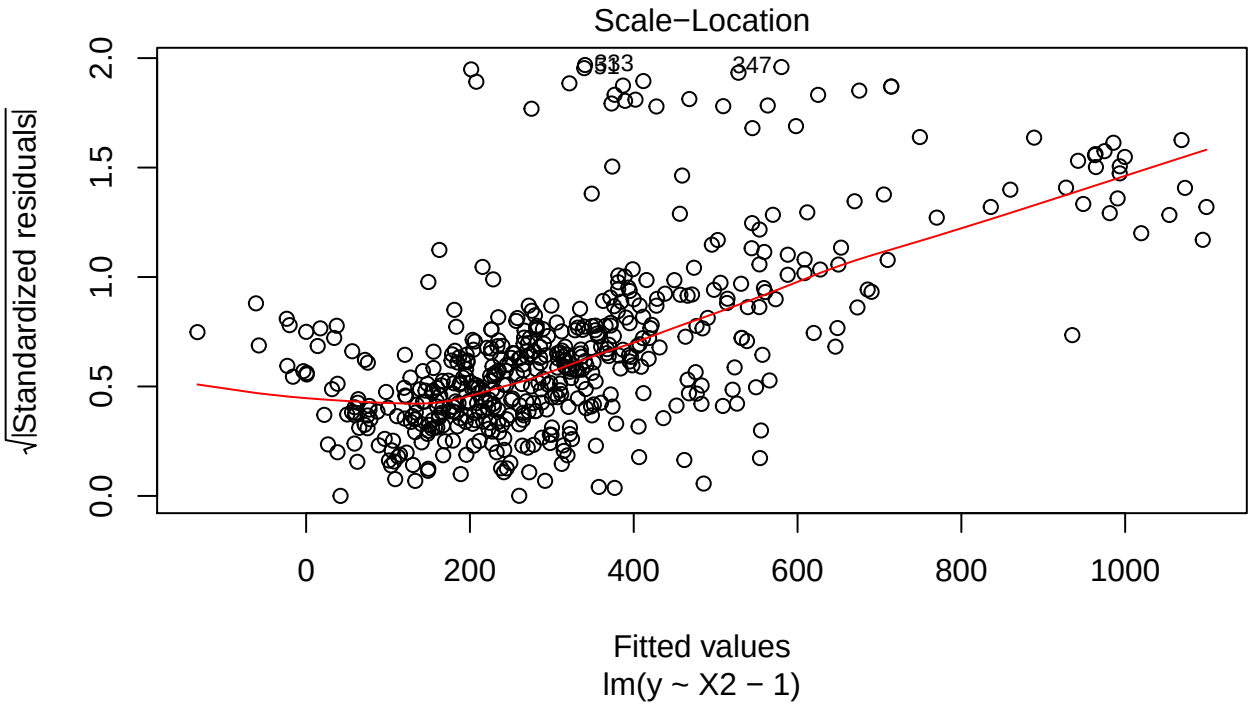
```
plot(lm(y ~ X2 - 1))
```

```
## Warning: not plotting observations with leverage one:
##   68
```

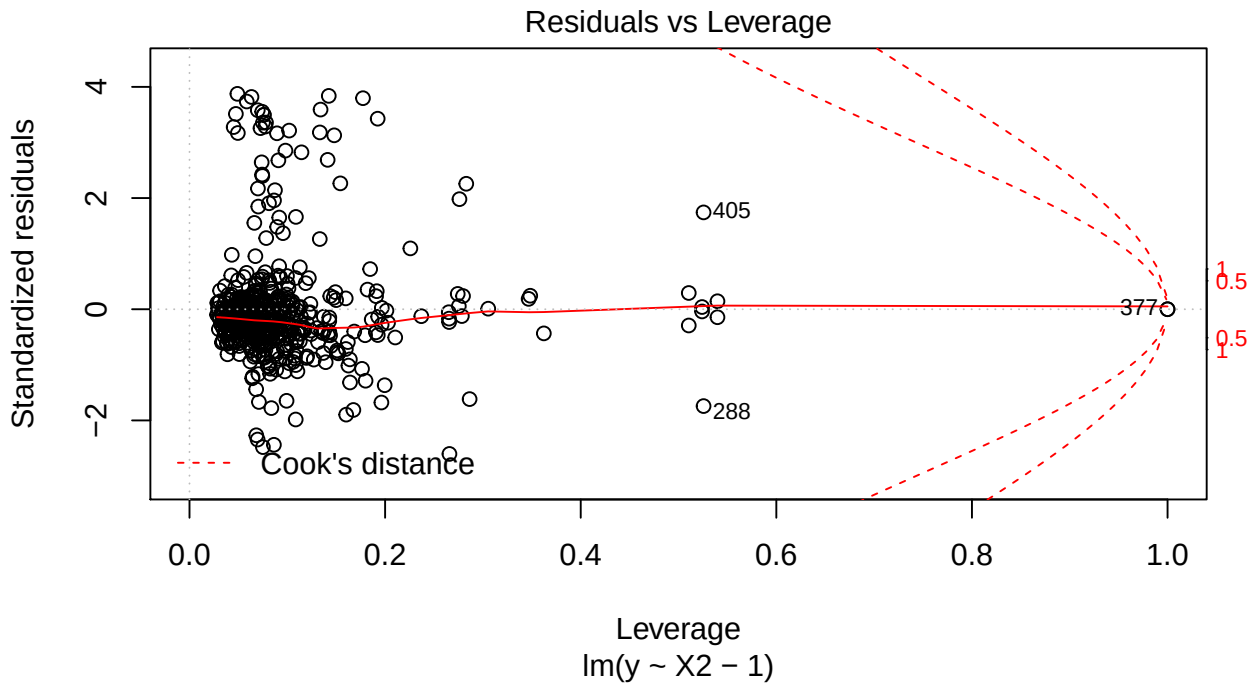





```
## Warning: not plotting observations with leverage one:
##   68
```




```
## Warning in sqrt(crit * p * (1 - hh)/hh): production de NaN
## Warning in sqrt(crit * p * (1 - hh)/hh): production de NaN
```



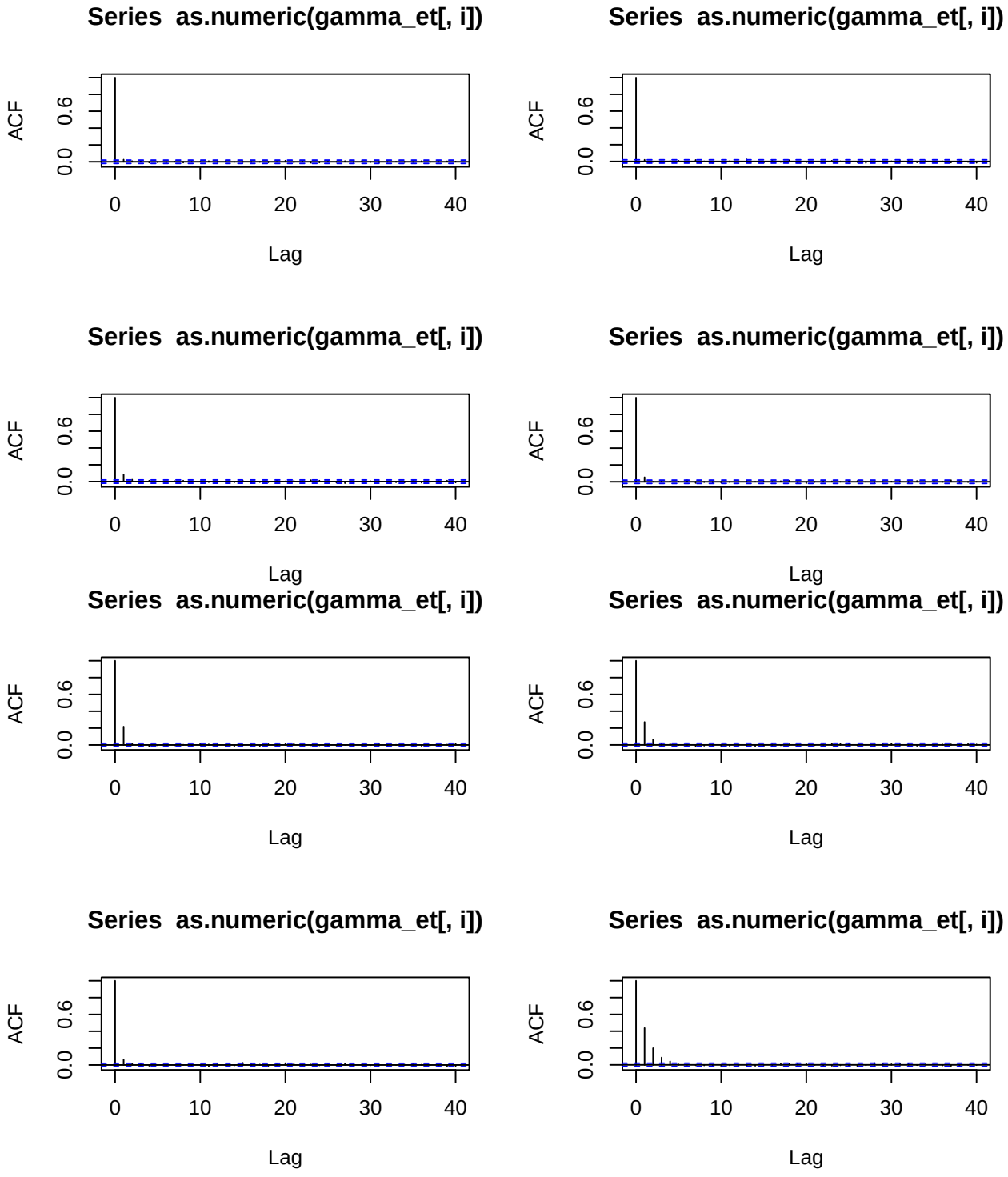

```
# les graphiques nous montrent : que les données possèdent plus de valeurs extrêmes par
# rapport à ce qui est théoriquement attendu (QQplot) que l'hypothèse d'homoscédasticité ne
# semble pas vérifiée (Scale-Location) qu'un point influent (377) est présent (residuals vs
# leverage)
```

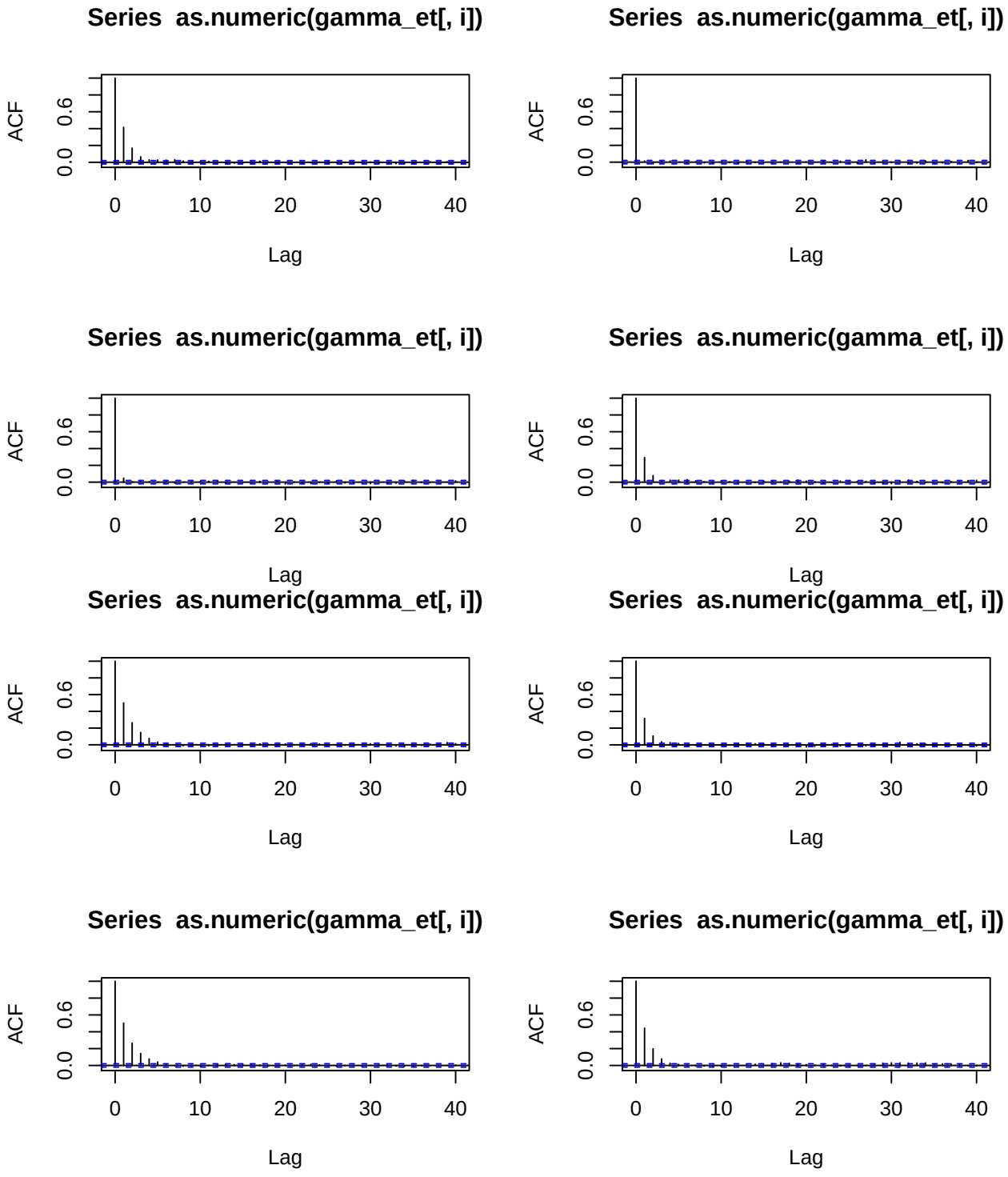
4.6 Annexe 6

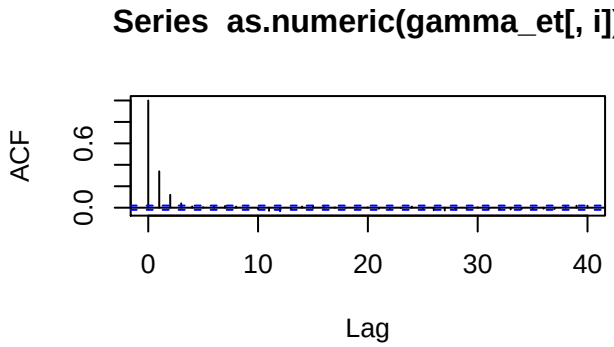
```
marglkd = function(gamma, X, g, y) {
    q = sum(gamma)
    X1 = X[, c(T, gamma)]
    if (q == 0) {
        return(q/2 * log(g + 1) - g/2 * log(t(y) %*% y))
    }
    m = -q/2 * log(g + 1) - n/2 * log(t(y) %*% y - g/(g + 1) * t(y) %*% X1 %*% solve(t(X1) %*%
        X1) %*% t(X1) %*% y)
    return(m)
}

set.seed(1)
niter = 10000  # nombre d'iterations
gamma_et = matrix(F, nrow = niter, ncol = 17)
gamma0 = sample(c(T, F), size = 17, replace = TRUE)  #valeur initiale aléatoire
lkd = rep(0, niter)
modelnumber = rep(0, niter)

oldgamma = gamma0
for (i in 1:niter) {
    newgamma = oldgamma
    for (j in 1:17) {
        g1 = newgamma
        g1[j] = TRUE
        g2 = newgamma
        g2[j] = FALSE
        ml1 = marglkd(g1, X_et, n_et, y_et)
        ml2 = marglkd(g2, X_et, n_et, y_et)
        p = c(ml1, ml2) - min(ml1, ml2)
        # On souhaite tirer depuis une Bernoulli, avec probabilité de tirer TRUE égale à
        # exp(p[1])/(exp(p[1])+exp(p[2])).  C'est ce que fait la ligne suivante. Notons que la
        # fonction sample() calcule la constante de normalisation.
        newgamma[j] = sample(c(T, F), size = 1, prob = exp(p))
    }
    gamma_et[i, ] = newgamma
    lkd[i] = marglkd(newgamma, X_et, n_et, y_et)
    modelnumber[i] = sum(newgamma * 2^(0:16))
    oldgamma = newgamma
}

meangamma_et = apply(gamma_et, 2, "mean")
result = data_frame(meangamma_et, row.names = colnames(X_et[, -c(1)]))
```

4.7 Annexe 7

```
par(mfrow = c(2, 2))
for (i in 1:17) acf(as.numeric(gamma_et[, i]))
```

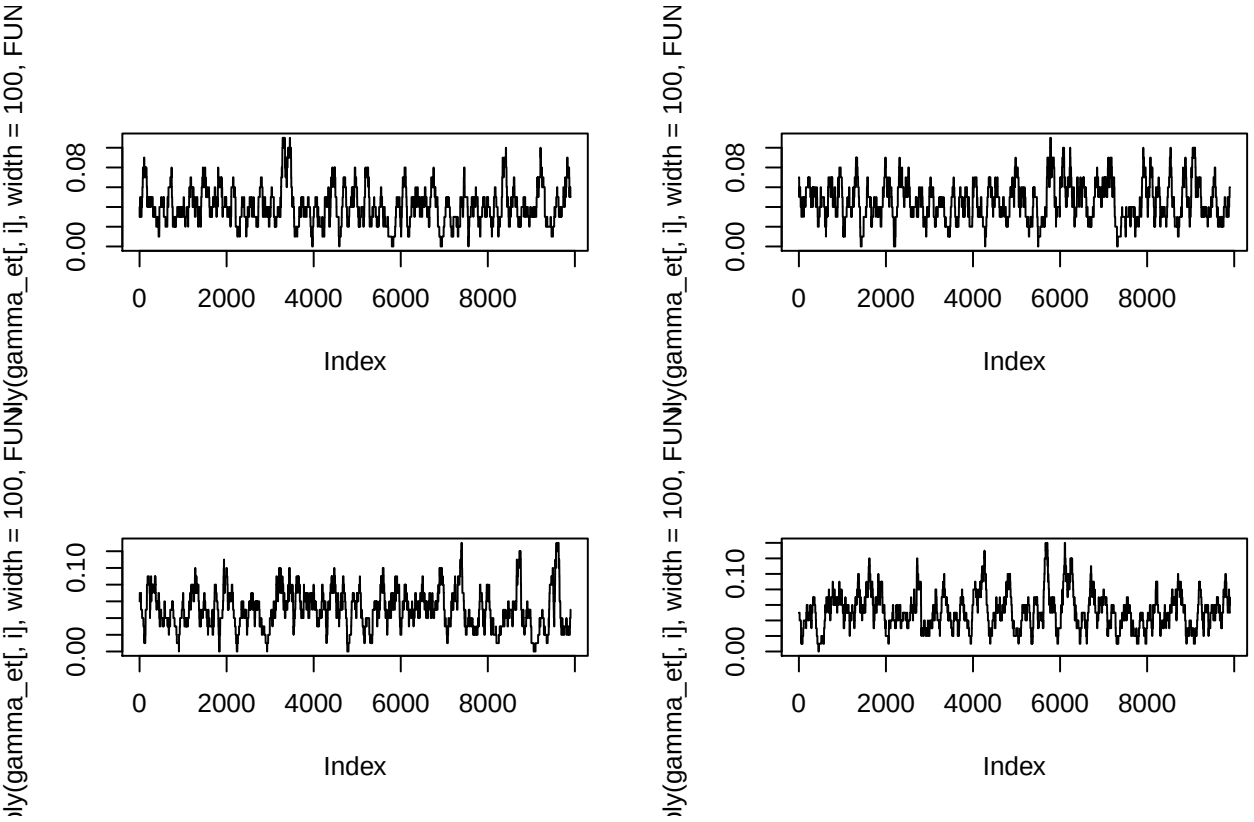



Series as.numeric(gamma_et[, i])
ACF
0.0
0.6
0
10
20
30
40
Lag


Series as.numeric(gamma_et[, i])
ACF
0.0
0.6
0
10
20
30
40
Lag

4.8 Annexe 8

```
require(zoo)
par(mfrow = c(2, 2))
for (i in 1:17) plot(rollapply(gamma_et[, i], width = 100, FUN = mean), type = "l")
```



ly(gamma_et[, i], width = 100, FUN
0.00 0.10
0 2000 4000 6000 8000
Index
0.00 0.20
0 2000 4000 6000 8000
Index
0.05 0.20
0 2000 4000 6000 8000
Index
0.00 0.15
0 2000 4000 6000 8000
Index

ly(gamma_et[, i], width = 100, FUN
0.00 0.25
0 2000 4000 6000 8000
Index
0.00 0.10
0 2000 4000 6000 8000
Index
0.00 0.25
0 2000 4000 6000 8000
Index
0.00 0.20
0 2000 4000 6000 8000
Index

ly(gamma_et[, i], width = 100, FUN
0.0 0.3
0.1 0.4
0.00 0.25
0 2000 4000 6000 8000
Index

ly(gamma_et[, i], width = 100, FUN
0.00 0.15
0.00 0.25
0 2000 4000 6000 8000
Index

4.9 Annexe 9

```r
summary(step(lm(Barre ~ ., data = df3), direction = "backward", trace = F))
```

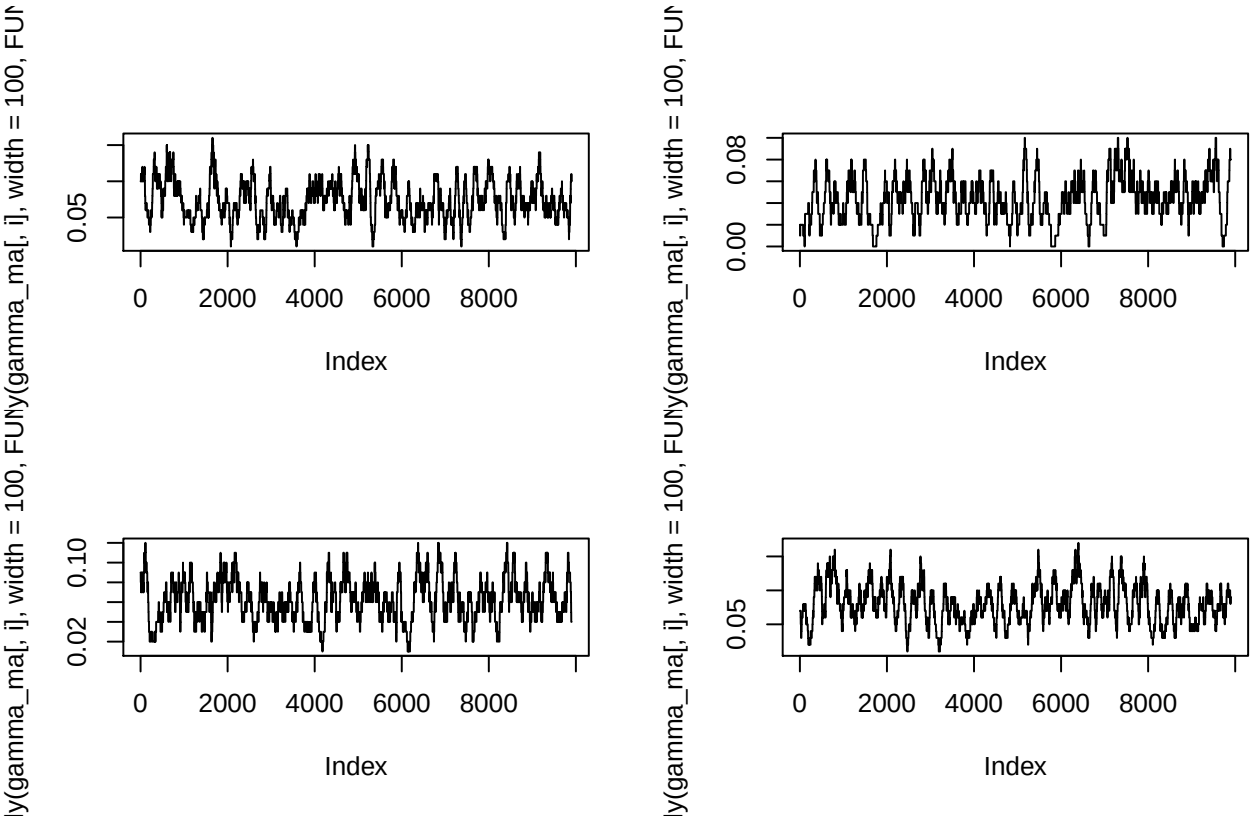
```
##
## Call:
## lm(formula = Barre ~ taux_reussite_attendu_serie_l + taux_acces_attendu_premiere_bac,
##     data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -387.32 -196.56 -130.83  -14.95 1696.20
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    -494.324    260.593  -1.897  0.05840 .
## taux_reussite_attendu_serie_l    -7.882      4.360  -1.808  0.07124 .
## taux_acces_attendu_premiere_bac  17.833      5.407   3.298  0.00104 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 419.5 on 513 degrees of freedom
## Multiple R-squared:  0.02539,    Adjusted R-squared:  0.02159
## F-statistic: 6.681 on 2 and 513 DF,  p-value: 0.001366
```

4.10 Annexe 10

```
set.seed(1)
niter = 10000  # nombre d'iterations
gamma_ma = matrix(F, nrow = niter, ncol = 34)
gamma0 = sample(c(T, F), size = 34, replace = TRUE)  #valeur initiale aléatoire
lkd = rep(0, niter)
modelnumber = rep(0, niter)

oldgamma = gamma0
for (i in 1:niter) {
    newgamma = oldgamma
    for (j in 1:34) {
        g1 = newgamma
        g1[j] = TRUE
        g2 = newgamma
        g2[j] = FALSE
        ml1 = marglkd(g1, X_ma, n_ma, y_ma)
        ml2 = marglkd(g2, X_ma, n_ma, y_ma)
        p = c(ml1, ml2) - min(ml1, ml2)
        # On souhaite tirer depuis une Bernoulli, avec probabilité de tirer TRUE égale à
        # exp(p[1])/(exp(p[1])+exp(p[2])).  C'est ce que fait la ligne suivante. Notons que la
        # fonction sample() calcule la constante de normalisation.
        newgamma[j] = sample(c(T, F), size = 1, prob = exp(p))
    }
    gamma_ma[i, ] = newgamma
    lkd[i] = marglkd(newgamma, X_ma, n_ma, y_ma)
    modelnumber[i] = sum(newgamma * 2^(0:33))
    oldgamma = newgamma
}

meangamma_ma = apply(gamma_ma, 2, "mean")
result = data_frame(meangamma_ma, row.names = colnames(X_ma[, -c(1)]))
```

4.11 Annexe 11

```
require(zoo)
par(mfrow = c(2, 2))
for (i in 1:34) plot(rollapply(gamma_ma[, i], width = 100, FUN = mean), type = "l")
```



y(gamma_ma[, i], width = 100, FUN
0.10 0.25
0.00 0.10
0.6 1.2
0.00 0.15
0 2000 4000 6000 8000
Index

y(gamma_ma[, i], width = 100, FUN
0.00 0.08
0.85
0.00 0.08
0.05 0.20
0 2000 4000 6000 8000
Index

y(gamma_ma[, i], width = 100, FUN
0.05
0.20
0.00
0.08
0.00
0.10
0.30
0.55
0
2000
4000
6000
8000
Index

y(gamma_ma[, i], width = 100, FUN
0.00
0.10
0.00
0.10
0.00
0.10
0.00
0.15
0
2000
4000
6000
8000
Index

y(gamma_ma[, i], width = 100, FUN
0.00
0.10
0.00
0.10
0.00
0.10
0.00
0.10
0
2000
4000
6000
8000
Index

y(gamma_ma[, i], width = 100, FUN
0.2
0.5
0.00
0.10
0.10
0.30
0.55
0.80
0
2000
4000
6000
8000
Index

y(gamma_ma[, i], width = 100, FUN
0.00
0.10
0
2000
4000
6000
8000
Index
0.05
0.35
0.60

y(gamma_ma[, i], width = 100, FUN
0.02
0.12
0.10
0.30
0.00
0.10

4.12 Annexe 12

```
summary(step(lm(y ~ X_ma - 1, data = df4), direction = "backward", trace = F))
```

```
##
## Call:
## lm(formula = y ~ X_ma - 1, data = df4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -928.81 -161.73  -64.02   42.28 1535.07
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## X_ma(Intercept)           615.70      97.72   6.300 6.71e-10 ***
## X_maMatiereANGLAIS       -389.22     111.75  -3.483 0.000541 ***
## X_maMatiereARTS PLAST    -453.20     293.17  -1.546 0.122800
## X_maMatiereBIOCH.BIOL    -474.17     187.13  -2.534 0.011596 *
## X_maMatiereBIOTECHNOL    -562.20     218.52  -2.573 0.010387 *
## X_maMatiereDOC LYCEES    -498.42     140.49  -3.548 0.000427 ***
## X_maMatiereE. P. S       -327.00     129.72  -2.521 0.012027 *
## X_maMatiereECO.GE.COM    -243.10     149.28  -1.629 0.104069
## X_maMatiereECO.GE.CPT     332.55     218.52   1.522 0.128706
## X_maMatiereECO.GE.FIN     362.11     134.31   2.696 0.007262 **
## X_maMatiereECO.GE.MK     -322.86     140.49  -2.298 0.021982 *
## X_maMatiereECO.GE.VEN    -447.30     187.13  -2.390 0.017216 *
## X_maMatiereEDUCATION     -435.79     132.64  -3.286 0.001092 **
## X_maMatiereESPAGNOL      -443.00     121.01  -3.661 0.000279 ***
## X_maMatiereESTH.COSME    -557.70     402.93  -1.384 0.166964
## X_maMatiereG.ELECTRON    -355.50     402.93  -0.882 0.378060
## X_maMatiereG.ELECTROT    -480.20     293.17  -1.638 0.102088
## X_maMatiereG.IND.BOIS    -573.70     402.93  -1.424 0.155145
## X_maMatiereHIST. GEO.    -343.57     113.14  -3.037 0.002522 **
## X_maMatiereITALIEN        351.90     293.17   1.200 0.230608
## X_maMatiereLET ANGLAI    -226.57     169.26  -1.339 0.181336
## X_maMatiereLET ESPAGN    -465.95     218.52  -2.132 0.033487 *
## X_maMatiereLET MODERN    -164.31     121.01  -1.358 0.175148
## X_maMatiereLET.HIS.GE    -412.07     145.96  -2.823 0.004952 **
## X_maMatiereLETT CLASS    -364.12     138.20  -2.635 0.008693 **
## X_maMatiereMATH.SC.PH    -357.43     162.87  -2.195 0.028674 *
## X_maMatiereMATHS         -427.80     110.18  -3.883 0.000118 ***
## X_maMatiereNRC           -456.20     293.17  -1.556 0.120348
## X_maMatierePHILO          -84.67     131.11  -0.646 0.518724
## X_maMatierePHY.CHIMIE    -284.64     129.72  -2.194 0.028688 *
## X_maMatiereS. V. T.      -263.42     124.21  -2.121 0.034450 *
## X_maMatiereSC.ECO.SOC    -234.55     136.16  -1.723 0.085596 .
## X_maMatiereSII.EE         -18.37     187.13  -0.098 0.921854
## X_maMatiereSII.ING.ME      21.43     169.26   0.127 0.899328
## X_maMatiereSII.SIN       -449.63     245.93  -1.828 0.068129 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 390.9 on 481 degrees of freedom
## Multiple R-squared:  0.4969, Adjusted R-squared:  0.4603
```

```
## F-statistic: 13.57 on 35 and 481 DF,  p-value: < 2.2e-16
```

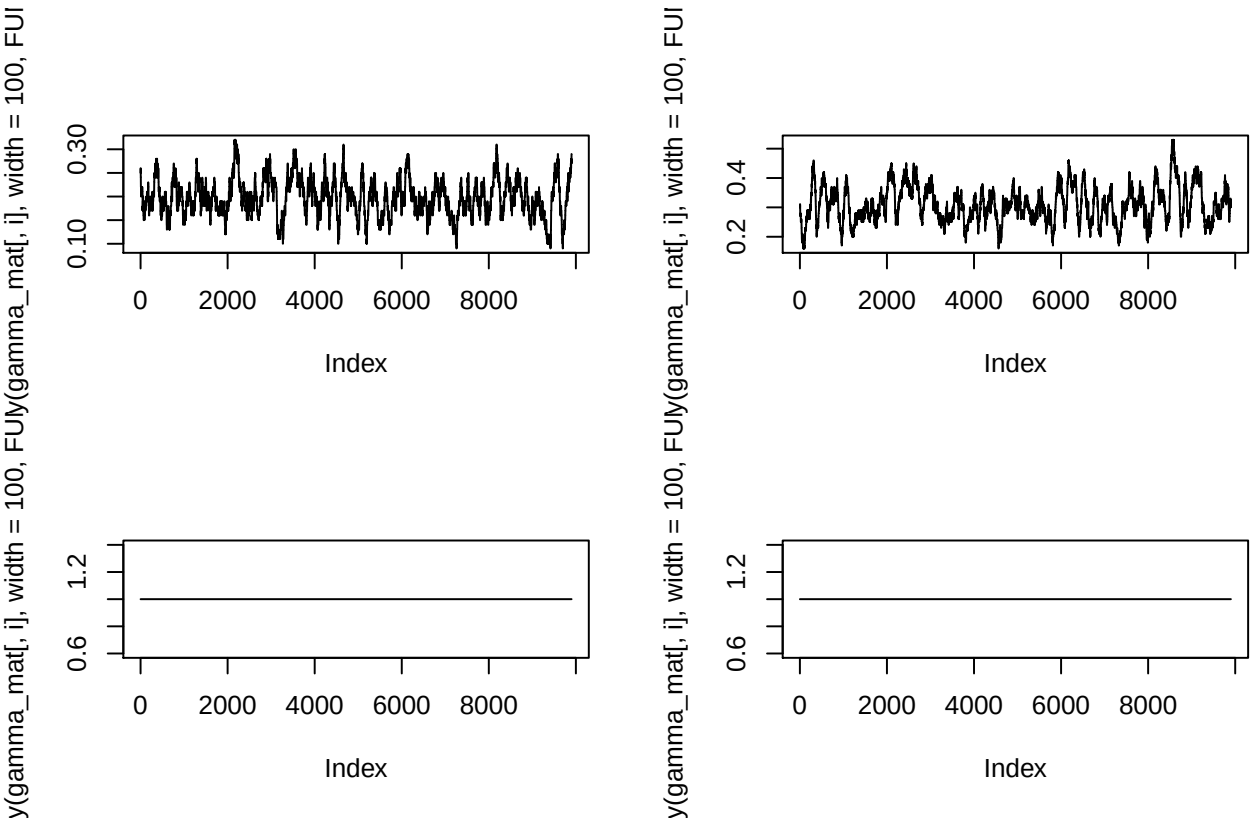
4.13 Annexe 13

```
set.seed(2)

niter = 10000  # nombre d'iterations
gamma_ang = matrix(F, nrow = niter, ncol = 17)
gamma0 = sample(c(T, F), size = 17, replace = TRUE)  #valeur initiale aléatoire
lkd_ang = rep(0, niter)
modelnumber_ang = rep(0, niter)

oldgamma = gamma0
for (i in 1:niter) {
    newgamma = oldgamma
    for (j in 1:17) {
        g1 = newgamma
        g1[j] = TRUE
        g2 = newgamma
        g2[j] = FALSE
        ml1 = marglkd(g1, X_ang, n_ang, y_ang)
        ml2 = marglkd(g2, X_ang, n_ang, y_ang)
        p = c(ml1, ml2) - min(ml1, ml2)
        # On souhaite tirer depuis une Bernoulli, avec probabilité de tirer TRUE égale à
        # exp(p[1])/(exp(p[1])+exp(p[2])).  C'est ce que fait la ligne suivante. Notons que la
        # fonction sample() calcule la constante de normalisation.
        newgamma[j] = sample(c(T, F), size = 1, prob = exp(p))
    }
    gamma_ang[i, ] = newgamma
    lkd_ang[i] = marglkd(newgamma, X_ang, n_ang, y_ang)
    modelnumber_ang[i] = sum(newgamma * 2^(0:16))
    oldgamma = newgamma
}

niter = 10000  # nombre d'iterations
gamma_mat = matrix(F, nrow = niter, ncol = 17)
gamma0 = sample(c(T, F), size = 17, replace = TRUE)  #valeur initiale aléatoire
lkd_mat = rep(0, niter)
modelnumber_mat = rep(0, niter)

oldgamma = gamma0
for (i in 1:niter) {
    newgamma = oldgamma
    for (j in 1:17) {
        g1 = newgamma
        g1[j] = TRUE
        g2 = newgamma
        g2[j] = FALSE
        ml1 = marglkd(g1, X_mat, n_mat, y_mat)
        ml2 = marglkd(g2, X_mat, n_mat, y_mat)
        p = c(ml1, ml2) - min(ml1, ml2)
        # On souhaite tirer depuis une Bernoulli, avec probabilité de tirer TRUE égale à
        # exp(p[1])/(exp(p[1])+exp(p[2])).  C'est ce que fait la ligne suivante. Notons que la
        # fonction sample() calcule la constante de normalisation.
        newgamma[j] = sample(c(T, F), size = 1, prob = exp(p))
```

```
    }
    gamma_mat[i, ] = newgamma
    lkd_mat[i] = marglkd(newgamma, X_mat, n_mat, y_mat)
    modelnumber_mat[i] = sum(newgamma * 2^(0:16))
    oldgamma = newgamma
}
meangamma_ang = apply(gamma_ang, 2, "mean")
meangamma_mat = apply(gamma_mat, 2, "mean")
result = data_frame(meangamma_ang, meangamma_mat, row.names = colnames(X_ang[, -c(1)]))
```

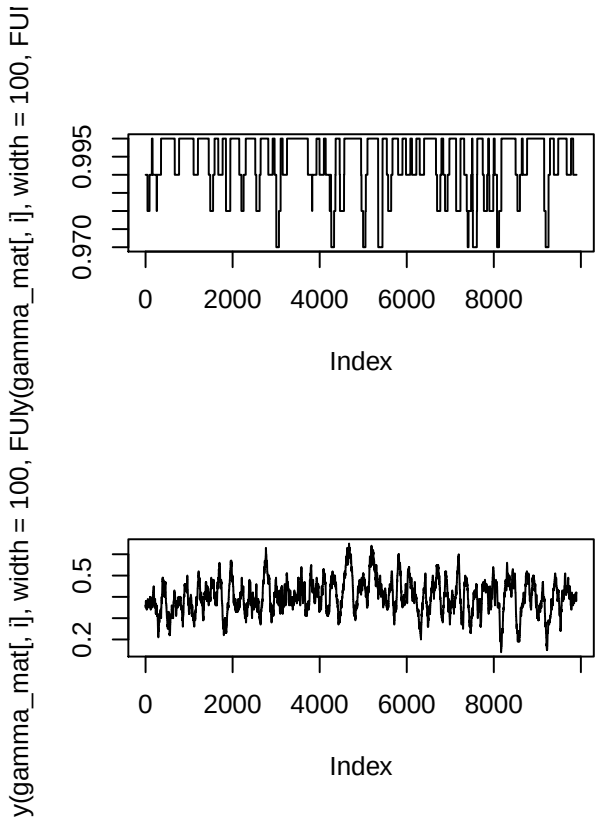
4.14 Annexe 14

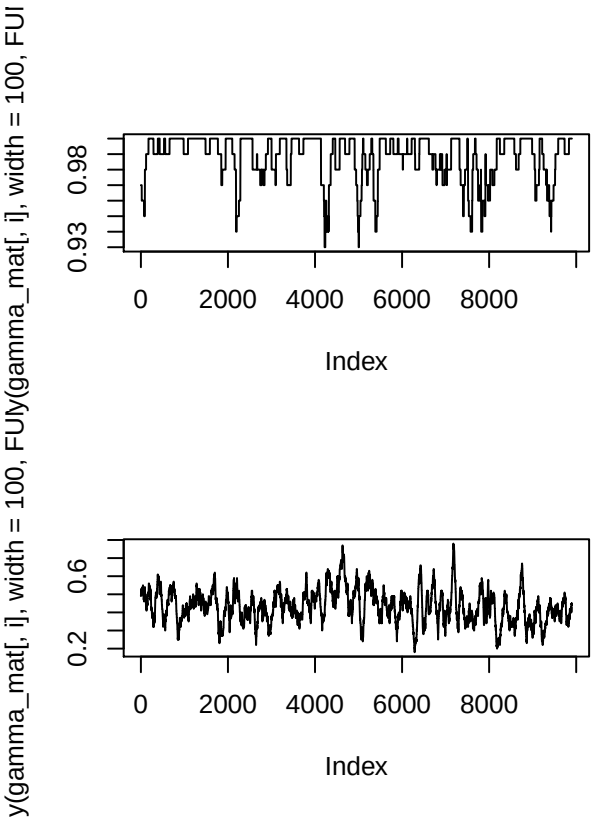
```
require(zoo)
par(mfrow = c(2, 2))
for (i in 1:17) plot(rollapply(gamma_mat[, i], width = 100, FUN = mean), type = "l")
```



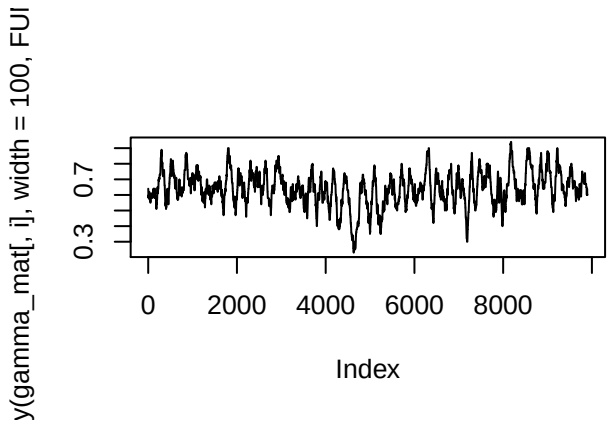
y(gamma_mat[, i], width = 100, FUIy(gamma_mat[, i], width = 100, FUIy(gamma_mat[, i], width = 100, FUIy(gamma_mat[, i], width = 100, FUI
0.6
1.2
0.3
0.6
0.05
0.25
0.6
1.2
0
2000
4000
6000
8000
Index

y(gamma_mat[, i], width = 100, FUIy(gamma_mat[, i], width = 100, FUIy(gamma_mat[, i], width = 100, FUI
0.10
0.35
0.6
0.9
0.990
1.000
0.6
1.2
0
2000
4000
6000
8000
Index






```
par(mfrow = c(2, 2))
```



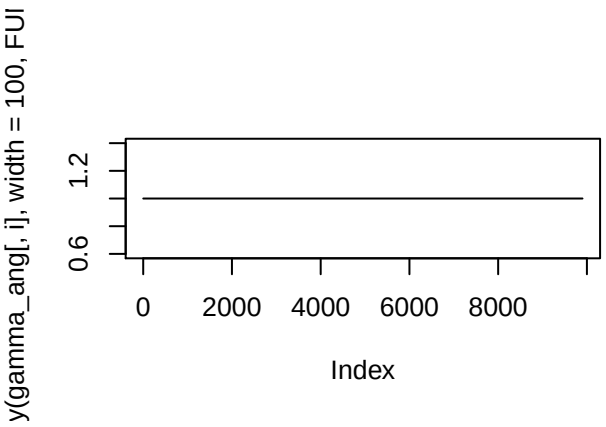

```
for (i in 1:17) plot(rollapply(gamma_ang[, i], width = 100, FUN = mean), type = "l")
```

y(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUI
0.990
1.000
0.05
0.25
0.86
0.96
0.05
0.25
0
2000
4000
6000
8000
Index

y(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUI
0.45
0.70
0.970
0.995
0.05
0.25
0.6
1.2
0
2000
4000
6000
8000
Index

y(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUI
0.6
1.2
0.15
0.40
0.990
1.000
0
2000
4000
6000
8000
Index

y(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUIy(gamma_ang[, i], width = 100, FUI
0.6
1.2
0.85
0.25
0.50
0
2000
4000
6000
8000
Index


y(gamma_ang[, i], width = 100, FUI
0.6
1.2
0
2000
4000
6000
8000
Index