Rapport - Statistique bayésienne

Philippe Real

11 mars, 2020

Contents

1 Introduction : Lecture des données - description statistique

On s'intéresse dans cette étude aux mutations des enseignants de collége et lycée de l'académie de Versaille. La variable réponse ou la variable à expliquer est la variable : *Barre*. Qui correspond au barême ou nombre de point nécessaire pour pouvoir obtenir un poste dans un établissement scolaire. Les co-variables sont composées des caractéristiques de l'établissement basées sur les effectifs de 2nd, 1ere et Terminale ainsi que les taux d'accès en 2nd, 1ere, Terminale et de réussites aux examens.

- Rennomage des colonnes

On peut vouloir obttenir parfois une notation plus compacte. On utilisera alors le nommage suivant:

Nouveau Nom	Ancien Nom
Prs_l	effectif_presents_serie_l
prs_es	effectif_presents_serie_es
Prs_s	effectif_presents_serie_s
Eff_2nd	effectif_de_seconde
Eff_1er	effectif_de_premiere
Suc.brt_l	taux_brut_de_reussite_serie_l
Suc.brt_es	taux_brut_de_reussite_serie_es
Suc.brt_s	taux_brut_de_reussite_serie_s
Suc.att_l	taux_reussite_attendu_serie_l
Suc.att_es	taux_reussite_attendu_serie_es
Suc.att_s	taux_reussite_attendu_serie_s
Acc.brt_bac.2	taux_acces_brut_seconde_bac
Acc.brt_bac.1	taux_acces_brut_premiere_bac
Acc.att_bac.1	taux_acces_attendu_premiere_bac)
Acc.att_bac.2	taux_acces_attendu_seconde_bac)
Suc.brt_Tot	taux_brut_de_reussite_total_series)
Suc.att_Tot	taux_reussite_attendu_total_series)

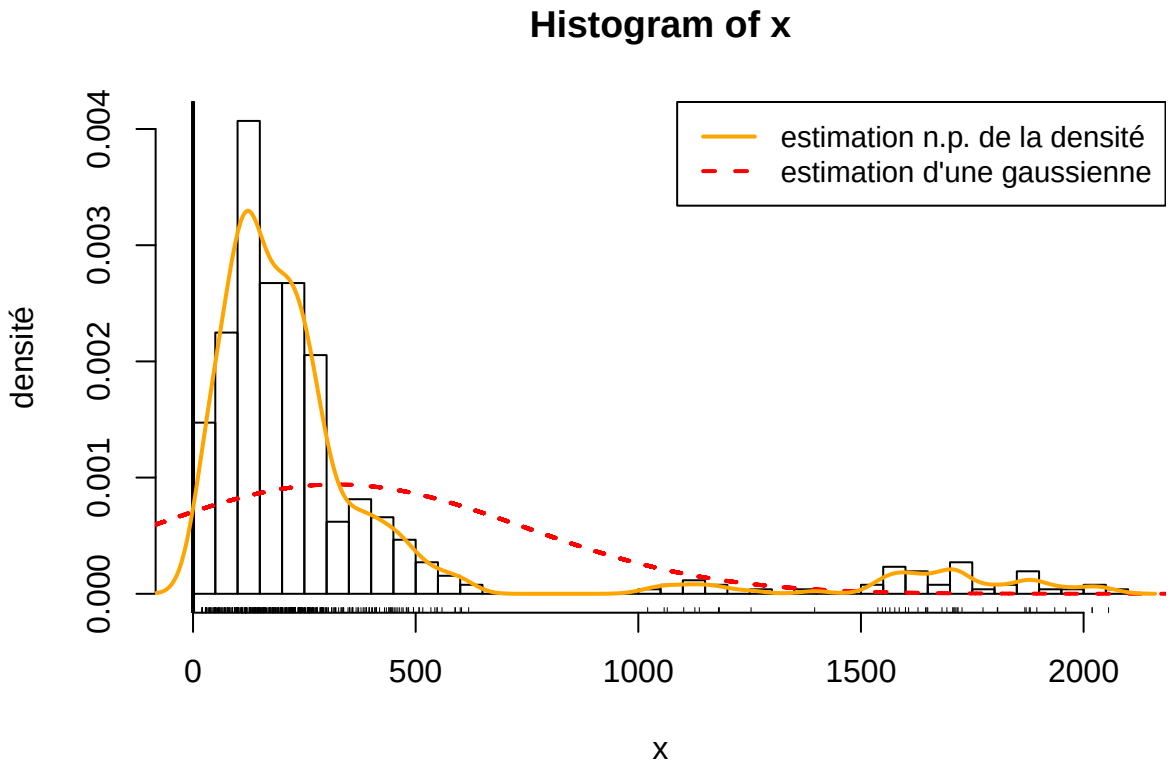
- Résumé des données :

On s'intéresse aux caractéristiques des lycées qui constituent nos vraiables explicatives et à la variable Barre qui est la variable à expliquer.

```
##      Barre          Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##  Min.   :  21.0   Min.   :  6.00   Min.   : 10.00   Min.   : 13.0
##  1st Qu.: 111.0   1st Qu.: 18.00   1st Qu.: 53.00   1st Qu.: 64.0
##  Median : 196.0   Median : 30.00   Median : 69.00   Median :100.0
##  Mean   : 321.9   Mean   : 34.24   Mean   : 74.42   Mean   :106.1
##  3rd Qu.: 292.0   3rd Qu.: 47.00   3rd Qu.: 99.00   3rd Qu.:140.0
##  Max.   :2056.0   Max.   :133.00   Max.   :192.00   Max.   :328.0
##   Tx_Suc.brt_l    Tx_Suc.brt_es    Tx_Suc.brt_s    Tx_Suc.att_l
##  Min.   : 36.00   Min.   : 51.0   Min.   :50.00   Min.   :65.00
##  1st Qu.: 82.00   1st Qu.: 81.0   1st Qu.:81.00   1st Qu.:84.00
##  Median : 89.00   Median : 88.0   Median :88.00   Median :89.00
##  Mean   : 86.35   Mean   : 86.4   Mean   :86.23   Mean   :86.91
##  3rd Qu.: 94.00   3rd Qu.: 94.0   3rd Qu.:93.00   3rd Qu.:92.00
##  Max.   :100.00   Max.   :100.0   Max.   :99.00   Max.   :98.00
##  Tx_Suc.att_es    Tx_Suc.att_s      Eff_2nd         Eff_1er
##  Min.   :61.00   Min.   :61.00   Min.   : 36.0   Min.   : 36.0
##  1st Qu.:86.00   1st Qu.:86.00   1st Qu.:268.0   1st Qu.:226.5
##  Median :90.00   Median :89.00   Median :336.0   Median :289.0
```

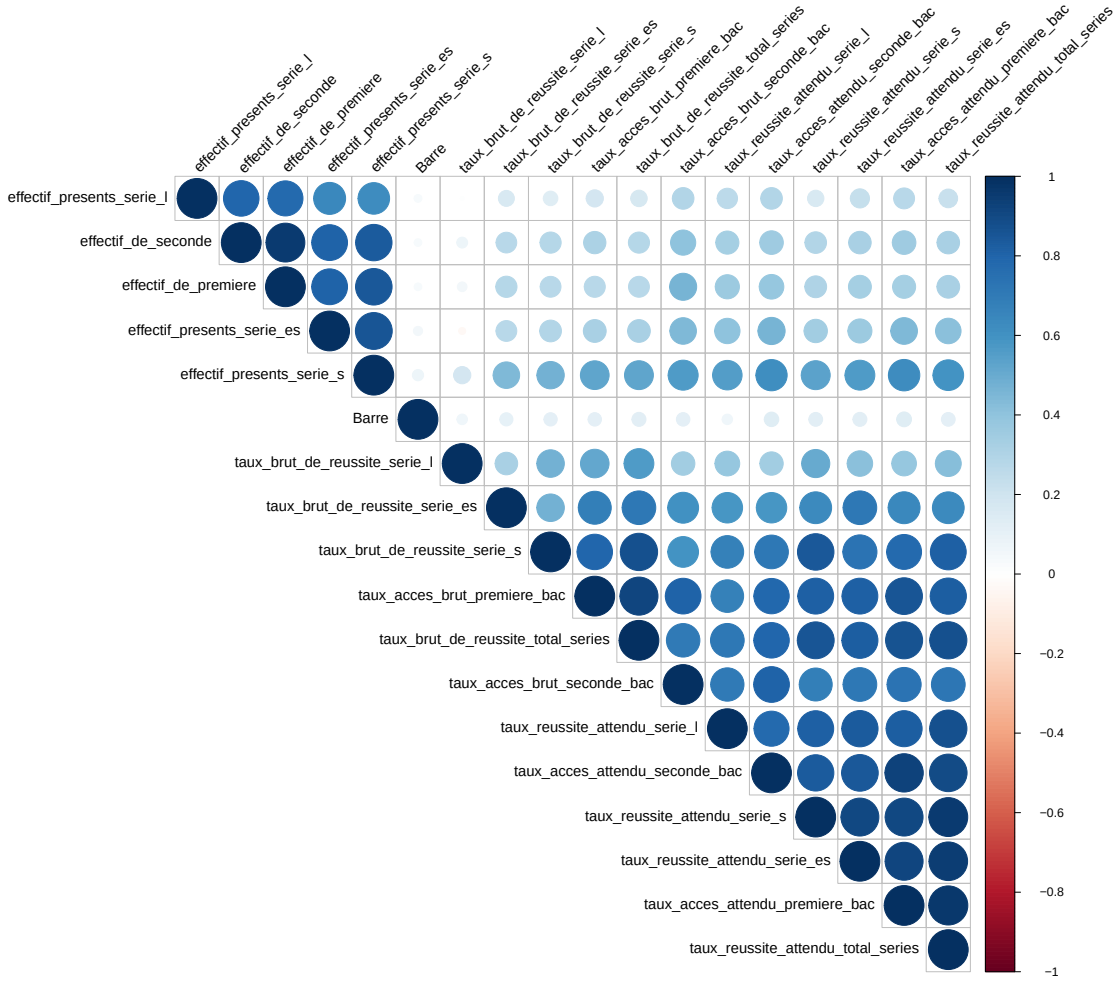
```
##  Mean   :87.97   Mean   :87.39   Mean   :351.6   Mean   :307.7
##  3rd Qu.:94.00   3rd Qu.:94.00   3rd Qu.:415.0   3rd Qu.:364.0
##  Max.   :98.00   Max.   :98.00   Max.   :764.0   Max.   :691.0
##  Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##  Min.   :49.00    Min.   :50.00    Min.   :65.00    Min.   :70.00
##  1st Qu.:64.00    1st Qu.:64.00    1st Qu.:82.00    1st Qu.:81.00
##  Median :71.00    Median :69.00    Median :85.00    Median :85.00
##  Mean   :69.61    Mean   :68.47    Mean   :84.53    Mean   :84.19
##  3rd Qu.:76.00    3rd Qu.:73.00    3rd Qu.:89.25    3rd Qu.:89.00
##  Max.   :87.00    Max.   :83.00    Max.   :97.00    Max.   :94.00
##  Tx_Suc.brt_Tot  Tx_Suc.att_Tot
##  Min.   :64.00   Min.   :67.0
##  1st Qu.:82.00   1st Qu.:84.0
##  Median :86.00   Median :88.0
##  Mean   :85.46   Mean   :86.8
##  3rd Qu.:91.00   3rd Qu.:92.0
##  Max.   :98.00   Max.   :98.0
```

- Histogramme de la variable à expliquer *Barre*



L'allure de la densité représentée par l'histogramme est très assymétrique et comporte une queue qui pourrait être épaisse. En tout cas à ne pas négliger. L'estimation de cette densité par une loi de Pareto proposée en partie II semble justifié.

- Corrélations 2 à 2 entre les variables



La variable a expliquer *Barre* est assez peu corrélée avec les variables constituant les caractéristiques de l'établissement. On remarque aussi deux groupes de variables dinstintcs, avec des corrélations inter-groupe faibles.

- Les variables de type effectifs (Effectifs et Effectifs présents, 5 variables en tout).
- Les variables de *taux* (Taux de réussite et Attendu, 12 variables en tout).

Par contre au sein de chacun des groupes, comme on peut s'y attendre les corrélations entre variables (intra-groupe) sont fortes. On remarque que le taux de réussite brute série L $Suc.brt_l$ est moins corrélés aux autres variables, et semble avoir une certaine indépendance.

2 Partie I - Régression linéaire Bayésienne

On cherche à expliquer le nombre de points nécessaire à une mutation (colonne Barre) par les caractéristiques du lycée. On considère un modéle de régression linéaire gaussien, que l'on rappelle ici.

2.1 Rappels définitions et notations

2.1.1 Modèle linéaire Gaussien

Le modèle linéaire, tente d'expliquer les observations (input) (y_i) par des covariables $(x_1, ..., x_p)$ à partir du modèle suivant :

$y_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} + \epsilon_i$ où $\epsilon_i \sim N(0, \sigma^2)$ et iid.

On note $y = (y_1, .., y_n)$ le vecteur des observations et $X = (x_{ik})_{1 \leq i \leq n, 1 \leq k \leq p}$ la matrice des covaraiables ou de design (predictor).

En notation matricielles le modèle se réécrit de la manière suivante:

$$y \mid \alpha, \beta, \sigma^2 \sim N_n \alpha 1_n + X\beta, \sigma^2 I_n)$$

où N_n est la distribution de la loi normale en dimension n.

Ainsi les y_i suivent des lois normales indépendantes avec :

$$E(y_i \mid \alpha, \beta, \sigma^2) = \alpha + \sum_{j=1}^{p} \beta_j x_{ij}$$

$$V(y_i \mid \alpha, \beta, \sigma^2) = \sigma^2$$

2.1.2 Contexte bayésien

On rappelle ici la formulation de la régression linaire dans le contexte bayésien.

On se place dans le cadre d'une expérience statistique paramétrique, où le vecteur des observations $Y = (y_1, ..., y_n)$ est iid et les $y_i \sim P_\theta$ une loi de paramètre θ.

Dans le contexte bayésien, on suppose que le paramètre inconnu θ est une v.a dont la loi de probabilité représente notre incertitude sur les valeurs possibles.

- Loi à priori $\pi(\theta)$

Cette loi du paramètre θ est la loi à priori, notée: $\pi(\theta)$. Elle représente "l'appriori" ou la croyance du statisticien avant le début de l'expérience. Sont choix est important, et on doit la choisir de manière à obtenir: une loi conjuguée pour faciliter les calculs, ou bien non informative (à priori de Jeffreys), fournit par un expert...

- Loi à postériori $\pi(\theta, y)$

On appelle la loi à postériori de θ sachant $y_1, y_2, ..., y_n$ la loi de distribution $\pi(\theta \mid Y) \propto \pi(\theta) L(\theta \mid Y)$

Cette définition découle de la formule de Bayes: $\pi(\theta \mid y) = \frac{\pi(\theta) f_{Y|\theta}(y|\theta)}{f_Y(y)}$

On retrouve l'équivalence des écritures avec $f_{Y|\theta}(y \mid \theta) = L(\theta \mid Y)$ Et $f_Y(y)$ ne dépend pas du paramètre θ, c'est une constante de normalisation qui est unique et que l'on peut retrouver une fois la loi à postériori déterminer analytiquement, qui doit s'intégrer à 1.

2.1.3 Régression linaire Bayésienne - Inférence bayésienne à l'aide de la loi a priori g de Zellner

On reprend les hypothèses et le contexte de définition du modèle linéaire gaussien, que l'on réinterprète avec l'approche Bayésienne. On considère la loi à priori $\pi(\theta)$ définit à partir des deux lois suivantes :

$$\beta \mid \sigma^2, X \sim N_{k+1}(\tilde{\beta}, \sigma^2 M^{-1})$$
$$\sigma^2 \mid X \sim IG(a, b)$$

L'idée principale de la modélisation de la G-prior de Zellner est de permettre d'introduire des informations (éventuellement faibles) sur le paramétre de localisation de la régression (commandé par le paramètre g) et surtout de contourner les aspects les plus difficiles de la définition de la prior, à savoir la structure de la corrélation (p76 Marin et Robert - bayesian essential with R).

En fixant la matrice M de la manière suivante dans l'approche de Zellner, on obtient la g-prior ou loi informative de Zellner :

$$\beta \mid \sigma^2, X \sim N_{k+1}(\tilde{\beta}, g\sigma^2({}^tXX)^{-1})$$
$$\sigma^2 \sim \pi(\sigma^2 \mid X) \propto \sigma^{-2}$$

Il reste à choisir le paramètre g, souvent g=1 ou g=n en fonction du poids que l'on veut accorder à la prior. Si g=2 celà revient à donner à la prior le même poids que 50% de l'échantilon. Avec g=n on donne à la loi à priori le même poids que 1-observation.

Pour l'espérance à priori $\tilde{\beta}$ ou pourra la prendre = 0 si l'on n'a pas d'information à priori.

La loi à priori $\pi(\theta)$ se déduit simplement à partir des deux lois précédentes:

$$\pi(\theta) = \pi(\beta, \sigma^2 \mid X) = \pi(\beta \mid \sigma^2, X)\pi(\sigma^2 \mid X)$$

Cette loi à la propriété remarquable d'être une loi conjugué et sa loi à postériori associée a l'expression analytique suivante:

$$\beta \mid \sigma^2, y, X \sim N_{k+1}(\frac{g}{g+1}\hat{\beta}, \frac{\sigma^2 g}{g+1}({}^tXX)^{-1})$$
$$\sigma^2 \mid y, X \sim IG(\frac{n}{2}\hat{\beta}, \frac{s^2}{2} + \frac{1}{2(g+1)}({}^t\hat{\beta}{}^tXX\hat{\beta})$$

donc :

$$\beta \mid y, X \sim Student_{k+1}(n, \frac{g}{g+1}\hat{\beta}, \frac{g(s^2 + ({}^t\hat{\beta}{}^tXX\hat{\beta})/(g+1))}{n(g+1)}({}^tXX)^{-1})$$

2.2 Résultats et interprétation des coefficients

Pour cette étude, on va s'appuyer sur les éléments du cours et les fonctions utilisées en TP et plus particulièrement du TP-N°4. On utilisera aussi des fonctions du package R-Bayess ainsi que le livre associé: "Bayesian essential with R" ou "Bayesian Core" de Marin et Robert. Comme suggéré en page 69 de cet ouvrage, on va centrer et réduire les éléments de la matrice de design X. Dans ce qui suit on va confronter les résultats obtenus à partir des fonctions pour l'essentiel vu ou adaptées du cours et des fonctions du package Bayess, plus particulièrement les fonctions: *BayesReg* et *ModChoBayesReg*.

2.2.1 Calcul explicite des coefficients

On se place dans le contexte Bayésien avec pour loi à prioiri $\pi(\theta) = \pi(\beta, \sigma^2 \mid X)$ la G-prior de Zellner :

$$\beta \mid \sigma^2, X \sim N_{k+1}(\tilde{\beta}, g\sigma^2({}^tXX)^{-1})$$
$$\sigma^2 \sim \pi(\sigma^2 \mid X) \propto \sigma^{-2}$$

On cherche à calculer la moyenne à priori, à partir de la formule suivante:

$$E^{\pi}(\beta \mid y) = \frac{g}{g+1}\left(\hat{\beta} + \tilde{\beta}/g\right)$$

Où $\hat{\beta}$ est le vecteur des coefficients du modèle linéaire classique obtenu par maximum de vraissemblance ou moindre carré ordinaire.

On peut justifier cette expression de la manière suivante, comme par définition de la prior on a :

$$E^{\pi}(\beta \mid \sigma^2, y) = \frac{g}{g+1}\left(\hat{\beta} + \tilde{\beta}/g\right)$$

Puis en prenant l'espérance et en conditionnant par rapport à y, on obtient :

$$E^{\pi}(E^{\pi}(\beta \mid \sigma^2, y) \mid y) = E^{\pi}\left(\frac{g}{g+1}\left(\hat{\beta} + \tilde{\beta}/g\right)\right) = \frac{g}{g+1}\left(\hat{\beta} + \tilde{\beta}/g\right)$$

Et comme par définition β ne dépend pas de σ on a :

$$E^{\pi}(E^{\pi}(\beta \mid \sigma^2, y) \mid y) = E^{\pi}(\beta \mid y)$$

On va calculer explicitement la quantité : $E^{\pi}(\beta \mid y)$

- calcul de $\hat{\beta}$ coefficient du modèle linéaire

On sait que $\hat{\beta}$ s'obtient comme solution du problème : $\hat{\beta} = (X^T X)^{-1} X^T y$

```
beta0.lm=mean(y)
beta.lm=solve(t(X)%*%X,t(X)%*%y)
betahat=beta.lm
betahat
```

```
##                          [,1]
## Eff_Prst_l          16.3770102
## Eff_Prst_es         10.0578749
## Eff_Prst_s           0.5621583
## Tx_Suc.brt_l        36.1191826
## Tx_Suc.brt_es       47.4496652
## Tx_Suc.brt_s        85.4422916
## Tx_Suc.att_l      -106.0647897
## Tx_Suc.att_es       32.3521086
## Tx_Suc.att_s       -40.3864199
## Eff_2nd              5.8396882
## Eff_1er            -44.5331083
## Tx_Acc.brt_bac.2    97.6265317
## Tx_Acc.att_bac.2   -51.1283516
## Tx_Acc.brt_bac.1  -140.2871142
## Tx_Acc.att_bac.1   206.2261510
## Tx_Suc.brt_Tot     -39.8727718
## Tx_Suc.att_Tot     -31.4216860
```

On peut aussi retrouver les coefficients $\hat{\beta}$ à partir de la fonction lm. On obtient quasiment les mêmes résultats:

```
reg.lm=lm(y~X)
summary(reg.lm)
```

```
##
## Call:
## lm(formula = y ~ X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -429.72 -205.90 -122.25   -8.55 1645.96
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        321.9155    18.5937  17.313   <2e-16 ***
## XEff_Prst_l         16.3770    34.4842   0.475   0.6351
## XEff_Prst_es        10.0579    42.4558   0.237   0.8128
## XEff_Prst_s          0.5622    59.0966   0.010   0.9924
## XTx_Suc.brt_l       36.1192    29.6131   1.220   0.2232
## XTx_Suc.brt_es      47.4497    41.4726   1.144   0.2531
## XTx_Suc.brt_s       85.4423    58.1080   1.470   0.1421
## XTx_Suc.att_l     -106.0648    51.0743  -2.077   0.0383 *
## XTx_Suc.att_es      32.3521    70.0697   0.462   0.6445
## XTx_Suc.att_s      -40.3864    90.0514  -0.448   0.6540
## XEff_2nd             5.8397    84.4786   0.069   0.9449
## XEff_1er           -44.5331    90.8498  -0.490   0.6242
## XTx_Acc.brt_bac.2   97.6265    51.3820   1.900   0.0580 .
## XTx_Acc.att_bac.2  -51.1284    65.2923  -0.783   0.4340
## XTx_Acc.brt_bac.1 -140.2871    73.6707  -1.904   0.0575 .
## XTx_Acc.att_bac.1  206.2262   114.7440   1.797   0.0729 .
## XTx_Suc.brt_Tot    -39.8728    95.2695  -0.419   0.6757
## XTx_Suc.att_Tot    -31.4217   169.9511  -0.185   0.8534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 422.4 on 498 degrees of freedom
## Multiple R-squared:  0.04068,    Adjusted R-squared:  0.007931
## F-statistic: 1.242 on 17 and 498 DF,  p-value: 0.2267
```

On a "éliminé" l'intercept en centrant ou sinon avec on aurait dû utiliser la formule: y~X-1

- Calcul de $E^{\pi}(\beta \mid y, X) = \frac{g}{g+1}(\hat{\beta} + \frac{\tilde{\beta}}{g})$ G-prior informative de Zellner

Avec comme Hypothèses Zellner G-prior: g=1 et $\tilde{\beta} = 0$

```
g=1
betatilde=rep(0,dim(X)[2])

mbetabayes=g/(g+1)*(beta.lm+betatilde/g)
postmean=rbind(Intercept=beta0.lm,mbetabayes)
postmean
```

```
##                        [,1]
## Intercept        321.9155039
## Eff_Prst_l         8.1885051
## Eff_Prst_es        5.0289374
## Eff_Prst_s         0.2810791
## Tx_Suc.brt_l      18.0595913
## Tx_Suc.brt_es     23.7248326
```

```
## Tx_Suc.brt_s      42.7211458
## Tx_Suc.att_l     -53.0323949
## Tx_Suc.att_es     16.1760543
## Tx_Suc.att_s     -20.1932099
## Eff_2nd            2.9198441
## Eff_1er          -22.2665542
## Tx_Acc.brt_bac.2  48.8132658
## Tx_Acc.att_bac.2 -25.5641758
## Tx_Acc.brt_bac.1 -70.1435571
## Tx_Acc.att_bac.1 103.1130755
## Tx_Suc.brt_Tot   -19.9363859
## Tx_Suc.att_Tot   -15.7108430
```

Avec comme Hypothèses Zellner G-prior: g=n et $\tilde{\beta} = 0$ On accorde ici moins d'importance à la prior et on se retrouve plus proche des coefficients obtenus à partir d'une régression classique.

```
g=length(y)
betatilde=rep(0,dim(X)[2])

mbetabayes=g/(g+1)*(beta.lm+betatilde/g)
postmean=rbind(Intercept=beta0.lm,mbetabayes)
postmean
```

```
##                         [,1]
## Intercept         321.9155039
## Eff_Prst_l         16.3453332
## Eff_Prst_es        10.0384206
## Eff_Prst_s          0.5610709
## Tx_Suc.brt_l       36.0493196
## Tx_Suc.brt_es      47.3578863
## Tx_Suc.brt_s       85.2770261
## Tx_Suc.att_l     -105.8596354
## Tx_Suc.att_es      32.2895320
## Tx_Suc.att_s      -40.3083030
## Eff_2nd             5.8283928
## Eff_1er           -44.4469708
## Tx_Acc.brt_bac.2   97.4376989
## Tx_Acc.att_bac.2  -51.0294573
## Tx_Acc.brt_bac.1 -140.0157659
## Tx_Acc.att_bac.1  205.8272610
## Tx_Suc.brt_Tot    -39.7956484
## Tx_Suc.att_Tot    -31.3609090
```

C'est cette dernière hypothèse que l'on va conserver.

2.3 Choix des covariables et comparaison au résultat obtenu par une analyse fréquentiste.

Pour choisir les covariables significatives, on peut se baser sur les facteurs de Bayes. Ils donnent une idée de l'importance d'une variable. En effet on peut tester l'hypothèse H_0 ={Modèle sans la variable i} conntre {Modèle avec la variable i}. Ceci pour chacune des variables.

2.3.1 Choix des covariables avec les Bayes factors

Pour comparer les modèles on peut utiliser les facteurs de Bayes: Test d'hypothèse $H_0 : \beta_i = 0$
On test l'hypothèse H_0, $\forall i = 1, ..., 17$ et on calcul le Bayes Factor. C'est ce que propose la fonction $BayesReg$ du package Bayess. Ce qui donne une indication de la pertinance de la variable, un peu à la manière de la fonction lm.

On va calculer tout d'abord les Bayes Factor à partir de la formule et de la fonction vue en du cours qui est reprise dans la fonction: $CalcBayesFactor$.

- A partir de la fonction $CalcBayesFactor$:

Avec $g = n$ on obtient :

```
##         colnames(X) bfactor
## 1        Eff_Prst_l -1.3251
## 2       Eff_Prst_es -1.3489
## 3        Eff_Prst_s -1.3567
## 4      Tx_Suc.brt_l -1.1484
## 5     Tx_Suc.brt_es -1.1734
## 6      Tx_Suc.brt_s -1.0541
## 7      Tx_Suc.att_l -0.7538
## 8     Tx_Suc.att_es -1.3269
## 9      Tx_Suc.att_s -1.3286
## 10          Eff_2nd -1.3561
## 11          Eff_1er -1.3231
## 12 Tx_Acc.brt_bac.2 -0.8518
## 13 Tx_Acc.att_bac.2 -1.2708
## 14 Tx_Acc.brt_bac.1 -0.8496
## 15 Tx_Acc.att_bac.1 -0.9049
## 16   Tx_Suc.brt_Tot -1.3322
## 17   Tx_Suc.att_Tot -1.3520
```

Avec $g = 1$ on obtient :

```
##         colnames(X) bfactor
## 1        Eff_Prst_l -0.1349
## 2       Eff_Prst_es -0.1466
## 3        Eff_Prst_s -0.1505
## 4      Tx_Suc.brt_l -0.0475
## 5     Tx_Suc.brt_es -0.0598
## 6      Tx_Suc.brt_s -0.0008
## 7      Tx_Suc.att_l  0.1480
## 8     Tx_Suc.att_es -0.1357
## 9      Tx_Suc.att_s -0.1366
## 10          Eff_2nd -0.1502
## 11          Eff_1er -0.1339
## 12 Tx_Acc.brt_bac.2  0.0994
## 13 Tx_Acc.att_bac.2 -0.1080
## 14 Tx_Acc.brt_bac.1  0.1005
## 15 Tx_Acc.att_bac.1  0.0731
## 16   Tx_Suc.brt_Tot -0.1384
## 17   Tx_Suc.att_Tot -0.1481
```

En donnant plus de poids à la prior certains coefficients commencent à être significatifs au sens de $Jeffrey$: 7, 12 14 et 15éme variable.

- Bayes Regression : *FonctionBayesReg* :

Pour estimer les β à postériori, on va utiliser la fonction (modifiée) *BayesReg* du package *Bayess* issue du livre de *Marin et Robert : Bayesian Essentials with R*. Le calcul détaillé a été exposé au § précédent. Comme on l'a vu ce calcul peut aussi être obtenu directement à partir de la fonction lm (residuals). On comparera le résultat obtenu avec le résultat renvoyé par la fonction du livre de P. Hoff: A First Course in Bayesian Statistical Methods.

Avec $g = n$ on obtient :

```
##
##            PostMean PostStError Log10bf EvidAgaH0
## Intercept  321.9155     18.3206
## x1          16.3453     33.9449 -1.3062
## x2          10.0384     41.7918 -1.3442
## x3           0.5611     58.1722 -1.3567
## x4          36.0493     29.1499 -1.0238
## x5          47.3579     40.8239 -1.0638
## x6          85.2770     57.1992 -0.8733
## x7        -105.8596     50.2754 -0.3944
## x8          32.2895     68.9737  -1.309
## x9         -40.3083     88.6429 -1.3117
## x10          5.8284     83.1573 -1.3557
## x11        -44.4470     89.4288 -1.3029
## x12         97.4377     50.5783 -0.5506
## x13        -51.0295     64.2711 -1.2194
## x14       -140.0158     72.5184  -0.547
## x15        205.8273    112.9493 -0.6352
## x16        -39.7956     93.7793 -1.3175
## x17        -31.3609    167.2928 -1.3491
##
##
## Posterior Mean of Sigma2: 173193.2688
## Posterior StError of Sigma2: 245171.3446
```

```
## $postmeancoeff
##  [1]  321.9155039   16.3453332   10.0384206    0.5610709   36.0493196
##  [6]   47.3578863   85.2770261 -105.8596354   32.2895320  -40.3083030
## [11]    5.8283928  -44.4469708   97.4376989  -51.0294573 -140.0157659
## [16]  205.8272610  -39.7956484  -31.3609090
##
## $postsqrtcoeff
##                        Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##         18.32064         33.94486         41.79178         58.17225
##     Tx_Suc.brt_l    Tx_Suc.brt_es    Tx_Suc.brt_s     Tx_Suc.att_l
##         29.14995         40.82392         57.19915         50.27543
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##         68.97375         88.64288         83.15726         89.42883
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##         50.57829         64.27109         72.51841        112.94926
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##         93.77934        167.29285
##
## $log10bf
##  [1] -1.3062110 -1.3441685 -1.3567250 -1.0238434 -1.0637704 -0.8732526
##  [7] -0.3944166 -1.3089805 -1.3116783 -1.3556744 -1.3029098 -0.5506177
## [13] -1.2194081 -0.5470375 -0.6351706 -1.3174966 -1.3490849
```

```
##
## $postmeansigma2
## [1] 173193.3
##
## $postvarsigma2
## [1] 60108988208
```

Les facteurs de bayes sont négatifs, et leur interprétation au sens de *Jeffrey* montre qu'ils ne sont pas significatifs.

Avec $g = 1$ on obtient :

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept 321.9155     18.5131
## x1          8.1885     24.2783 -0.1257
## x2          5.0289     29.8906 -0.1443
## x3          0.2811     41.6063 -0.1505
## x4         18.0596     20.8488  0.0129       (*)
## x5         23.7248     29.1983 -0.0067
## x6         42.7211     40.9103   0.087       (*)
## x7        -53.0324     35.9583  0.3226       (*)
## x8         16.1761     49.3318 -0.1271
## x9        -20.1932     63.3997 -0.1284
## x10         2.9198     59.4763   -0.15
## x11       -22.2666     63.9618 -0.1241
## x12        48.8133     36.1749  0.2457       (*)
## x13       -25.5642     45.9684 -0.0831
## x14       -70.1436     51.8671  0.2475       (*)
## x15       103.1131     80.7843  0.2041       (*)
## x16       -19.9364     67.0734 -0.1313
## x17       -15.7108    119.6522 -0.1468
##
##
## Posterior Mean of Sigma2: 176850.8134
## Posterior StError of Sigma2: 250348.9427
```

```
## $postmeancoeff
##  [1] 321.9155039   8.1885051   5.0289374   0.2810791  18.0595913
##  [6]  23.7248326  42.7211458 -53.0323949  16.1760543 -20.1932099
## [11]   2.9198441 -22.2665542  48.8132658 -25.5641758 -70.1435571
## [16] 103.1130755 -19.9363859 -15.7108430
##
## $postsqrtcoeff
##                        Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##         18.51308         24.27825         29.89058         41.60632
##     Tx_Suc.brt_l    Tx_Suc.brt_es    Tx_Suc.brt_s     Tx_Suc.att_l
##         20.84881         29.19834         40.91034         35.95831
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##         49.33183         63.39971         59.47625         63.96185
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##         36.17492         45.96837         51.86707         80.78428
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##         67.07345        119.65224
##
## $log10bf
##  [1] -0.125719637 -0.144344559 -0.150505049  0.012933199 -0.006683201
```

```
##  [6]  0.086951691  0.322646154 -0.127078682 -0.128402522 -0.149989616
## [11] -0.124099616  0.245703916 -0.083115184  0.247466860  0.204077270
## [16] -0.131257564 -0.146756674
##
## $postmeansigma2
## [1] 176850.8
##
## $postvarsigma2
## [1] 62674593126
```

En donnant plus d'importance à la prior, on voit que certaines variables se dégagent: les 4, 6, 7, 12 14 et 15éme.

- Conclusion

On obtient des résultats comparable avec les deux implémentations des facteurs de Bayes (Fonction du cours: *CalcBayesFactor* et Bayess: *BayesReg*) Les 7ème (Suc.att_l), 12ème (Acc.brt_bac.2), 14ème (Acc.brt_bac.1) et 15ème variables semblent être les plus significatives.

2.3.2 Choix de modèle : par calcul exact

On considère ici encore une implémentation de calcul exact vue en cours: *BayesModelChoice_Exact* que l'on rapproche de la fonction *ModChoBayesReg* du package Bayess.

- A partir de la méthode vue en cours qui est recodée ici dans la fonction : *BayesModelChoice_Exact*

```
##       model.name model.prob
## 16385        _15 0.12772218
## 4097         _13 0.08409933
## 32769        _16 0.04657771
## 129           _8 0.04448236
## 257           _9 0.03788892
## 65537        _17 0.03450863
## 16449      _7_15 0.02876012
## 2049         _12 0.02693390
## 8193         _14 0.02488491
## 33            _6 0.02198447
```

- A partir de la fonction (modifée) - ModChoBayesReg du package Bayess

Remarque: la valeur de la PostProb a été transformée aussi et n'est pas une plus une proba. Par contre le classement à partir de cette valeur reste valable. On a ajouté un paramètre *bCalcul*=TRUE par défaut, qui impose le calcul exact et par échantillonage de Gibbs sinon.

```
##
## bCalc = TRUE
## Model posterior probabilities are calculated exactly
##
##    Top10Models  PostProb
## 1           15 -2050.608
## 2           13 -2050.789
## 3           16 -2051.045
## 4            8 -2051.065
## 5            9 -2051.135
## 6           17 -2051.175
```

```
## 7         7 15 -2051.256
## 8           12 -2051.283
## 9           14 -2051.317
## 10           6 -2051.371
```

```
## $top10models
##  [1] "15"   "13"   "16"   "8"    "9"    "17"   "7 15" "12"   "14"   "6"
##
## $postprobtop10
##  [1] -2050.608 -2050.789 -2051.045 -2051.065 -2051.135 -2051.175 -2051.256
##  [8] -2051.283 -2051.317 -2051.371
```

On retrouve exactement les mêmes 10 meilleurs modèles. Plutôt que de faire un calcul exact on va maintenant utiliser l'algoritme d'echantillonnage de Gibbs. L'idée est d'obtenir la distribution d'intérêt à partir des lois conditionnelles, plus facile à calculer.

2.3.3 Choix de modèle : par échantillonnage de Gibbs

- Méthode N°1 - A partir de la fonction (modifée) *ModChoBayesReg* du package Bayess

```
##
## bCalc + false
## Model posterior probabilities are calculated by Gibbs
##
##    Top10Models PostProb
## 1           15   0.1240
## 2           13   0.0758
## 3            8   0.0458
## 4            9   0.0439
## 5           16   0.0429
## 6           17   0.0316
## 7           12   0.0293
## 8         7 15   0.0267
## 9           14   0.0251
## 10           5   0.0207
```

```
## $top10models
##  [1] "15"   "13"   "8"    "9"    "16"   "17"   "12"   "7 15" "14"   "5"
##
## $postprobtop10
##  [1] 0.1239750 0.0757500 0.0458500 0.0439000 0.0428750 0.0315875 0.0293000
##  [8] 0.0266750 0.0251125 0.0206750
```

Cette fois-ci la probabilité de chacun des modèles a pu être calculée. On retrouve des résultats très proches de ceux renvoyés par la fonction de calcul exact vue en cours: *BayesModelChoice_Exact*. Le classement des modèles est le même quelque soit les méthodes utilisées.

- Méthode N°2 - A partir de la méthode vue en cours *BayesModelChoice_Gibbs*

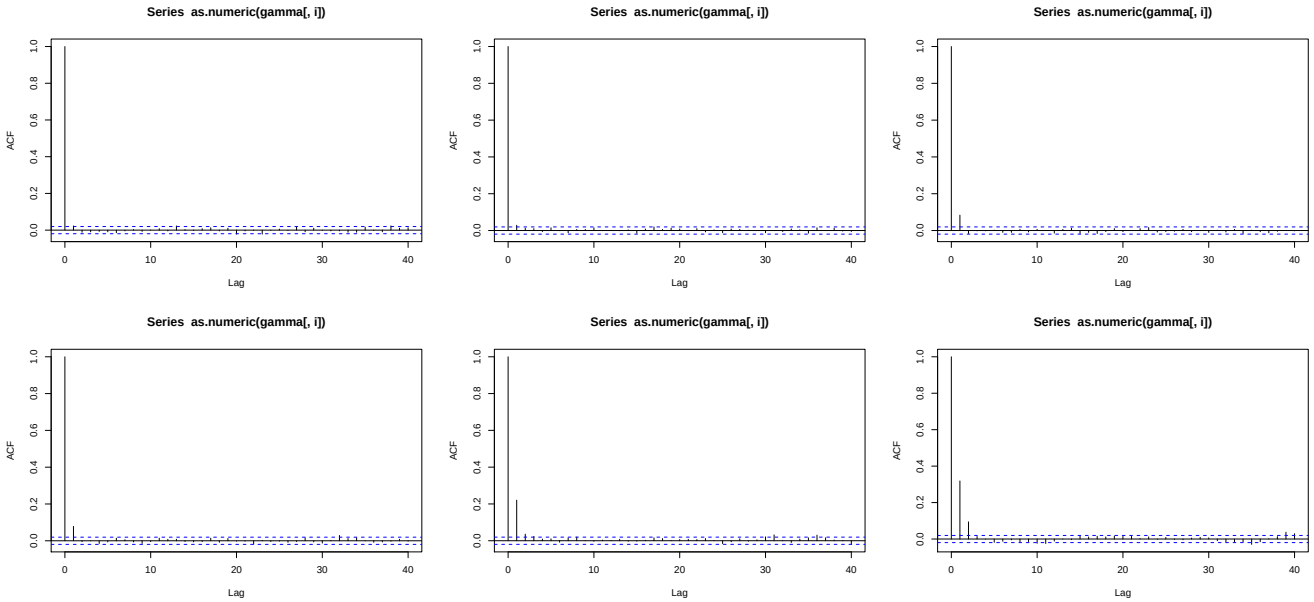
On va maintenant utiliser la fonction implémentée en cours: *BayesModelChoice_Gibbs* et comparer les résultats obtenus.

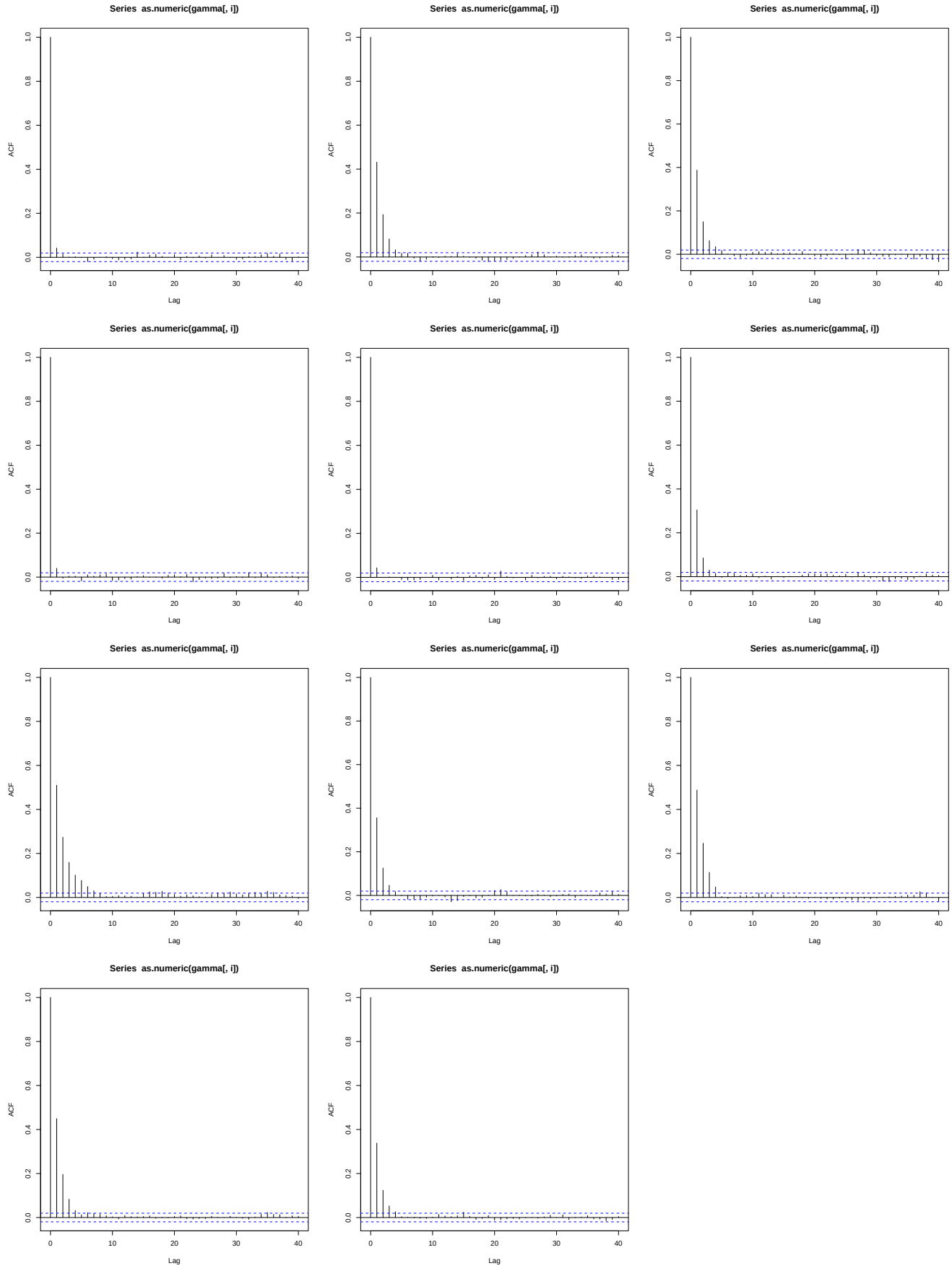
```
##                    X gamma.mean
## 15 Tx_Acc.att_bac.1     0.3225
## 13 Tx_Acc.att_bac.2     0.1966
## 7      Tx_Suc.att_l     0.1242
## 8     Tx_Suc.att_es     0.1195
## 16   Tx_Suc.brt_Tot     0.1174
## 17   Tx_Suc.att_Tot     0.1118
## 9      Tx_Suc.att_s     0.1095
## 12 Tx_Acc.brt_bac.2     0.0948
## 14 Tx_Acc.brt_bac.1     0.0839
## 5     Tx_Suc.brt_es     0.0819
## 6      Tx_Suc.brt_s     0.0815
## 10          Eff_2nd     0.0536
## 3        Eff_Prst_s     0.0519
## 4      Tx_Suc.brt_l     0.0511
## 2       Eff_Prst_es     0.0463
## 11          Eff_1er     0.0436
## 1        Eff_Prst_l     0.0421
```

On retrouve le même classement pour les 2 premières variables. Et un classement assez voisin pour les suivantes. On regarde maintenant, la convergence de la méthode.

- Vérication de la convergence et du mélange - autocorrélations:

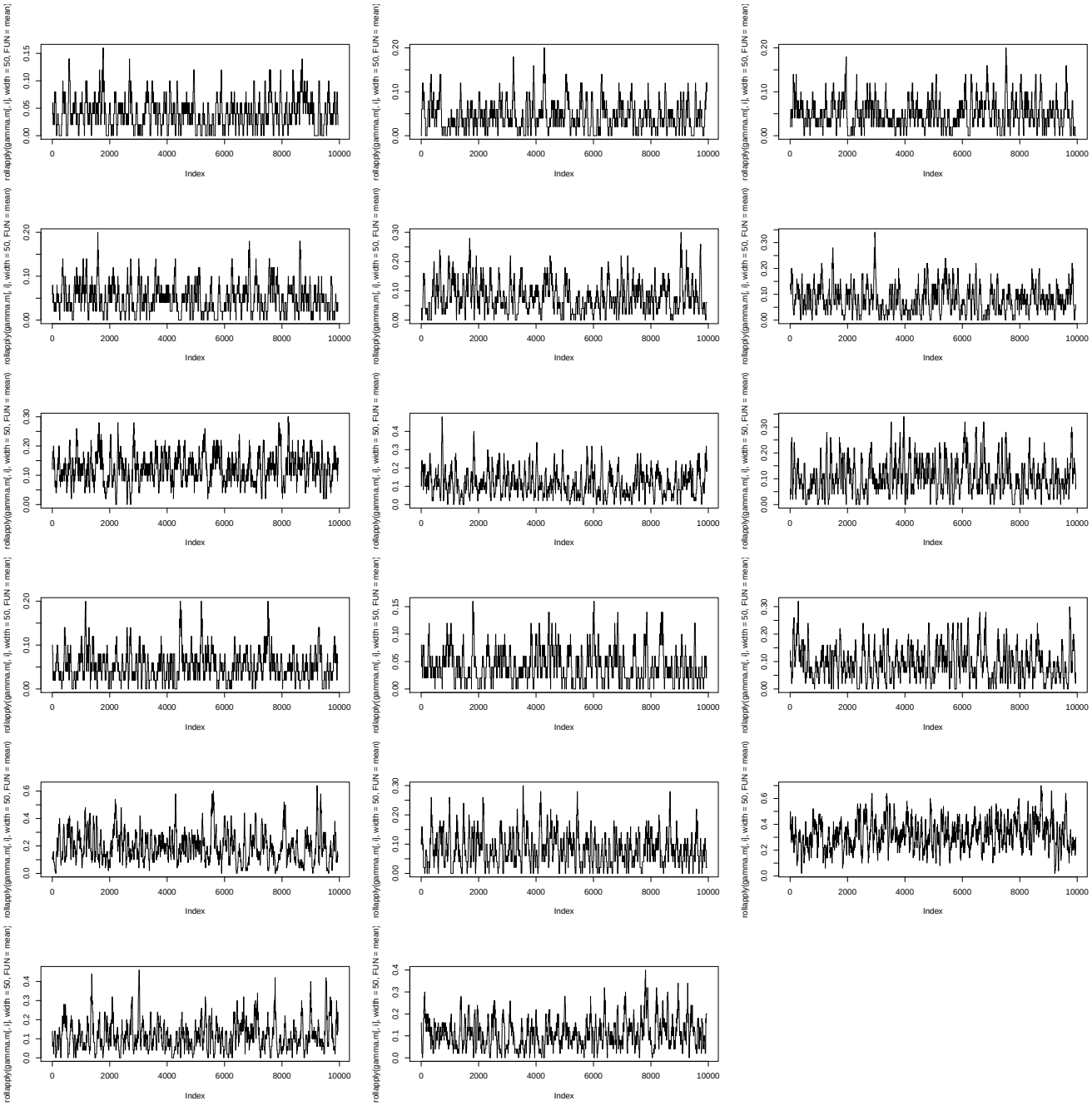
On vérifie le mélange de la chaine de Markov à l'aide des autocorrélations. Dans tous les cas les autocorrélations décroissent rapidement. On n'a pas besoin de sous-échantillonner.




Series as.numeric(gamma[, i])
ACF
Lag

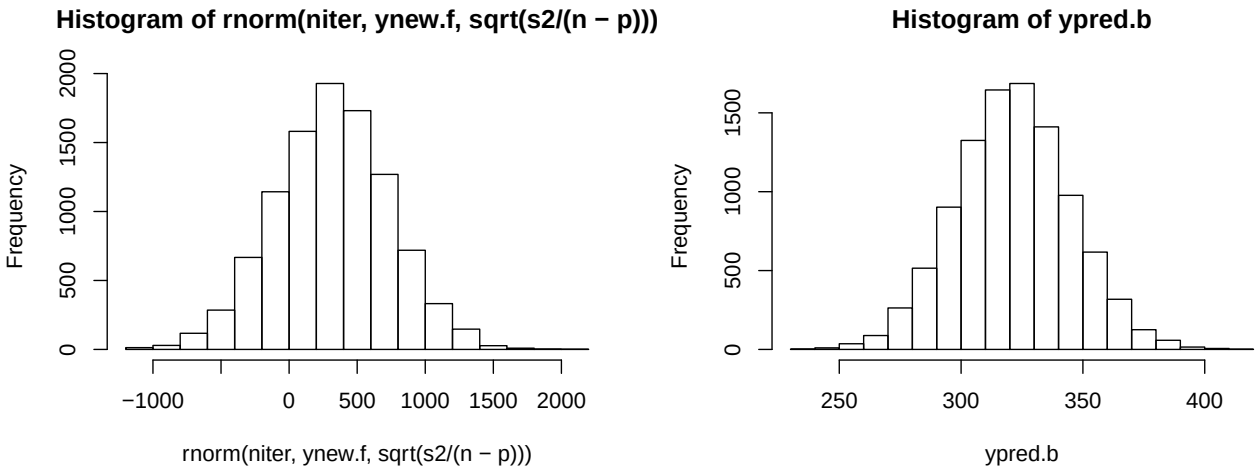
- Vérication de la convergence et du mélange - trace:

A l'aide de la trace (on utilise une moyenne glissante puisque les valeurs sont binaires).




```
##  probtop20          1          2          3          4          5
## 0.02810526 0.05000000 0.00000000 0.00000000 0.00000000 0.10000000
##          6          7          8          9         10         11
## 0.05000000 0.15000000 0.05000000 0.10000000 0.05000000 0.00000000
##         12         13         14         15         16         17
## 0.10000000 0.10000000 0.10000000 0.45000000 0.05000000 0.15000000
```

- Prédiction



Les histogrammes sont très similaires.

Pour comparaison, on va maintenant reprendre l'analyse et effectuer une analyse fréquentiste classique.

2.3.4 Comparaison au résultat obtenu par une analyse fréquentiste

- Analyse fréquentiste

On considère un modéle de régression linéaire gaussien i.e

$$y \mid \alpha, \beta, \sigma^2 \sim N_n(\alpha 1_n + X\beta, \sigma^2 I_n)$$

où N_n est la distribution de la loi normale en dimension n.

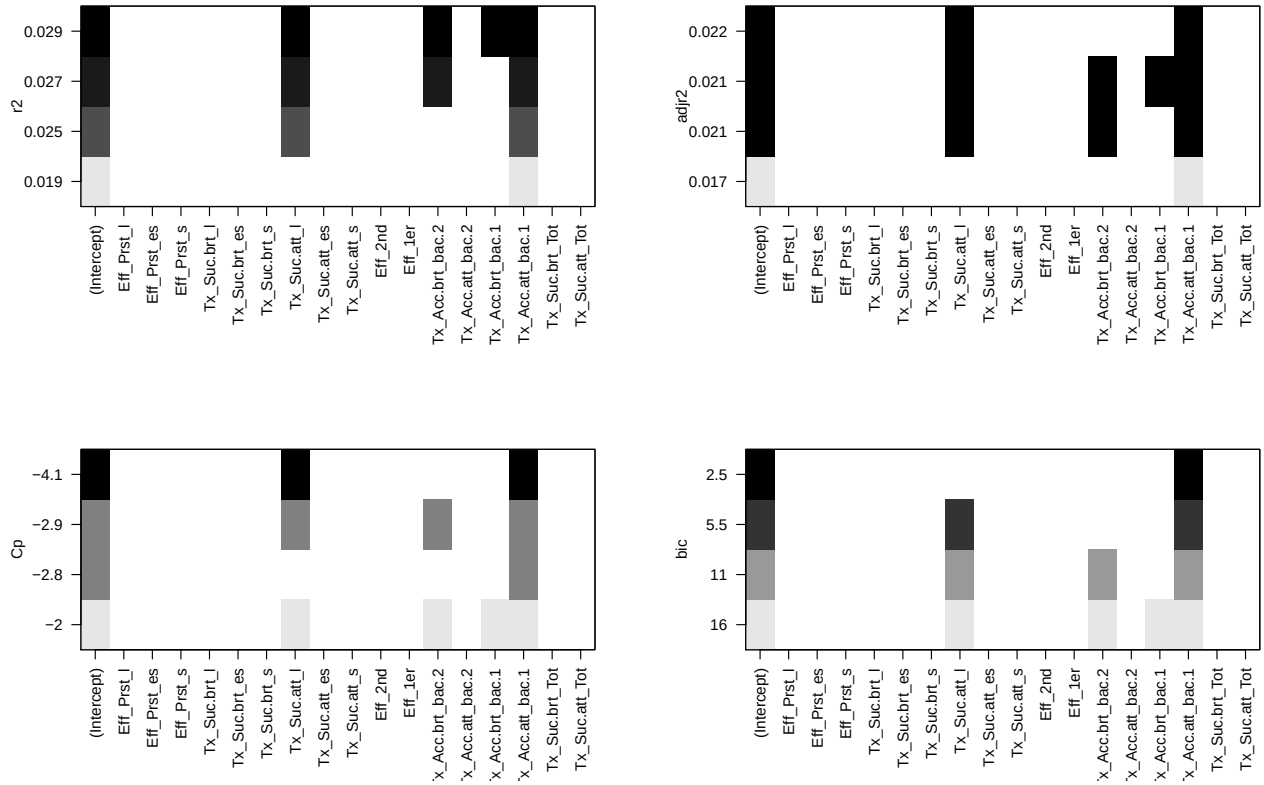
Ainsi les y_i suivent des lois normales indépendantes avec :

$$E(y_i \mid \alpha, \beta, \sigma^2) = \alpha + \sum_{j=1}^{p} \beta_j x_{ij}$$

$$V(y_i \mid \alpha, \beta, \sigma^2) = \sigma^2$$

```
##
## Call:
## lm(formula = Barre ~ ., data = d.reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -429.72 -205.90 -122.25   -8.55 1645.96
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       321.9155    18.5937  17.313   <2e-16 ***
## Eff_Prst_l         16.3770    34.4842   0.475   0.6351
## Eff_Prst_es        10.0579    42.4558   0.237   0.8128
## Eff_Prst_s          0.5622    59.0966   0.010   0.9924
## Tx_Suc.brt_l       36.1192    29.6131   1.220   0.2232
## Tx_Suc.brt_es      47.4497    41.4726   1.144   0.2531
```

```
## Tx_Suc.brt_s       85.4423    58.1080   1.470   0.1421
## Tx_Suc.att_l     -106.0648    51.0743  -2.077   0.0383 *
## Tx_Suc.att_es      32.3521    70.0697   0.462   0.6445
## Tx_Suc.att_s      -40.3864    90.0514  -0.448   0.6540
## Eff_2nd             5.8397    84.4786   0.069   0.9449
## Eff_1er           -44.5331    90.8498  -0.490   0.6242
## Tx_Acc.brt_bac.2   97.6265    51.3820   1.900   0.0580 .
## Tx_Acc.att_bac.2  -51.1284    65.2923  -0.783   0.4340
## Tx_Acc.brt_bac.1 -140.2871    73.6707  -1.904   0.0575 .
## Tx_Acc.att_bac.1  206.2262   114.7440   1.797   0.0729 .
## Tx_Suc.brt_Tot    -39.8728    95.2695  -0.419   0.6757
## Tx_Suc.att_Tot    -31.4217   169.9511  -0.185   0.8534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 422.4 on 498 degrees of freedom
## Multiple R-squared:  0.04068,    Adjusted R-squared:  0.007931
## F-statistic: 1.242 on 17 and 498 DF,  p-value: 0.2267
```




```
summary(step_mod)
```

```
##
## Call:
## lm(formula = Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1, data = d.reg)
##
## 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)        321.92      18.47  17.434  < 2e-16 ***
## Tx_Suc.att_l       -58.53      32.38  -1.808  0.07124 .
## Tx_Acc.att_bac.1   106.78      32.38   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
```

3 covariables qui se dégagent : taux_reussite_attendu_serie_l, taux_acces_attendu_premiere_bac, taux_acces_brut_seconde_

Au vu des p-valeurs des tests de Fisher, renvoyées par un test anova (cf. annexe) on peut envisager de se passer des variables : taux_acces_brut_premiere_bac et taux_acces_brut_seconde_bac on conserve donc le plus petit modèle *step_mod* composé de la variable *taux_acces_attendu_premiere_bac* qui est la plus significative et de la variable *taux_reussite_attendu_serie_l* qui est assez particulière du fait qu'elle est beaucoup moins corrélée que les autres (c. Introduction).

2.3.5 Préselection des covariables

On pourrait utiliser l'échantilloneur de Gibbs pour effectuer une préselection des variables ou bien les Bayes factor et ensuite faire un calcul exact de modèle. Mais ici ce n'est pas encore obligatoire, et on peut se passer de cette préselection. Le calcul exact incluant tous les modèles est encore rapide.

2.4 Mutations en mathématiques et anglais

2.4.1 Régression linéaire bayésienne et choix des covariables à l'aide des Bayes factors

Pour comparer les modèles on peut utiliser les facteurs de Bayes. On test l'hypothèse H_0, $\forall i = 1, ..., 17$ et on calcul le Bayes Factor à partir de la fonction *BayesReg*, pour g=n et g=1.

- Mutations en mathématiques - A partir de la fonction *BayesReg pour g=n*

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  86.1017      0.1652
## x1          0.0000      0.3021 -0.8891
## x2          0.0000      0.3863 -0.8891
## x3          0.0000      0.5842 -0.8891
## x4          0.0000      0.2491 -0.8891
## x5          0.0000      0.4013 -0.8891
## x6          9.4156      0.5787 21.0919    (****)
## x7          0.0000      0.4491 -0.8891
## x8          0.0000      0.6531 -0.8891
## x9          0.0000      0.8822 -0.8891
## x10         0.0000      0.7836 -0.8891
## x11         0.0000      0.8959 -0.8891
## x12         0.0000      0.5886 -0.8891
## x13         0.0000      0.6433 -0.8891
## x14         0.0000      0.7929 -0.8891
## x15         0.0000      1.0439 -0.8891
```

```
## x16         0.0000      0.9570 -0.8891
## x17         0.0000      1.5394 -0.8891
##
##
## Posterior Mean of Sigma2: 1.6099
## Posterior StError of Sigma2: 2.2974
```

```
## $postmeancoeff
##  [1]  8.610169e+01  3.318827e-14  2.910524e-13 -4.445481e-13 -3.286075e-14
##  [6]  2.620126e-14  9.415619e+00 -4.366877e-14  7.696621e-14  1.607011e-13
## [11] -1.021849e-13  3.777349e-14 -9.694467e-14  1.703082e-14  1.484738e-13
## [16]  1.050098e-14 -1.838455e-13 -3.283892e-13
##
## $postsqrtcoeff
##                       Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##        0.1651880        0.3020953        0.3862564        0.5841860
##     Tx_Suc.brt_l    Tx_Suc.brt_es    Tx_Suc.brt_s     Tx_Suc.att_l
##        0.2491424        0.4013360        0.5786809        0.4490823
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##        0.6530826        0.8822358        0.7836040        0.8959226
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##        0.5886290        0.6432968        0.7928656        1.0439175
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##        0.9570449        1.5393626
##
## $log10bf
##  [1] -0.8890756 -0.8890756 -0.8890756 -0.8890756 -0.8890756 21.0919120
##  [7] -0.8890756 -0.8890756 -0.8890756 -0.8890756 -0.8890756 -0.8890756
## [13] -0.8890756 -0.8890756 -0.8890756 -0.8890756 -0.8890756
##
## $postmeansigma2
## [1] 1.609937
##
## $postvarsigma2
## [1] 5.278048
```

- Mutations en mathématiques - A partir de la fonction *BayesReg pour g=1*

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  86.1017      0.9048
## x1          0.0000      1.1799 -0.1505
## x2          0.0000      1.5086 -0.1505
## x3          0.0000      2.2816 -0.1505
## x4          0.0000      0.9731 -0.1505
## x5          0.0000      1.5675 -0.1505
## x6          4.7876      2.2601  0.8203      (**)
## x7          0.0000      1.7540 -0.1505
## x8          0.0000      2.5507 -0.1505
## x9          0.0000      3.4457 -0.1505
## x10         0.0000      3.0605 -0.1505
## x11         0.0000      3.4992 -0.1505
## x12         0.0000      2.2990 -0.1505
## x13         0.0000      2.5125 -0.1505
## x14         0.0000      3.0967 -0.1505
## x15         0.0000      4.0772 -0.1505
```

```
## x16         0.0000      3.7379 -0.1505
## x17         0.0000      6.0122 -0.1505
##
##
## Posterior Mean of Sigma2: 48.2981
## Posterior StError of Sigma2: 68.922
```

```
## $postmeancoeff
##  [1]  8.610169e+01  1.687539e-14  1.479927e-13 -2.260414e-13 -1.670886e-14
##  [6]  1.332268e-14  4.787603e+00 -2.220446e-14  3.913536e-14  8.171241e-14
## [11] -5.195844e-14  1.920686e-14 -4.929390e-14  8.659740e-15  7.549517e-14
## [16]  5.339479e-15 -9.348078e-14 -1.669775e-13
##
## $postsqrtcoeff
##                        Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##        0.9047719        1.1798837        1.5085890        2.2816360
##     Tx_Suc.brt_l    Tx_Suc.brt_es    Tx_Suc.brt_s     Tx_Suc.att_l
##        0.9730674        1.5674849        2.2601349        1.7539660
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##        2.5507234        3.4457197        3.0604967        3.4991755
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##        2.2989892        2.5125034        3.0966693        4.0771944
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##        3.7378988        6.0122382
##
## $log10bf
##  [1] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150  0.8202562
##  [7] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150
## [13] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150
##
## $postmeansigma2
## [1] 48.29812
##
## $postvarsigma2
## [1] 4750.243
```

- Mutations en anglais - A partir de la fonction $BayesReg$ pour $g = n$

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  85.1346      0.1856
## x1          0.0000      0.4680 -0.8621
## x2          0.0000      0.4393 -0.8621
## x3          0.0000      0.6283 -0.8621
## x4          0.0000      0.3186 -0.8621
## x5          0.0000      0.4189 -0.8621
## x6          9.2800      0.6429 17.5057    (****)
## x7          0.0000      0.6551 -0.8621
## x8          0.0000      0.6518 -0.8621
## x9          0.0000      0.8641 -0.8621
## x10         0.0000      0.9345 -0.8621
## x11         0.0000      1.0954 -0.8621
## x12         0.0000      0.5832 -0.8621
## x13         0.0000      0.7794 -0.8621
## x14         0.0000      0.7603 -0.8621
## x15         0.0000      1.4612 -0.8621
```

```
## x16         0.0000      1.0542 -0.8621
## x17         0.0000      1.9166 -0.8621
##
##
## Posterior Mean of Sigma2: 1.7913
## Posterior StError of Sigma2: 2.5596
```

```
## $postmeancoeff
##  [1]  8.513462e+01 -6.448510e-14 -1.039713e-13  1.498843e-13  2.614261e-15
##  [6] -1.115418e-13  9.280008e+00  7.516000e-14  2.039124e-13  1.172060e-13
## [11] -8.539919e-14  1.254845e-13 -1.045704e-14 -1.080561e-13  2.439977e-14
## [16]  1.036990e-13  2.439977e-14 -4.147961e-13
##
## $postsqrtcoeff
##                        Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##        0.1856030        0.4679936        0.4393209        0.6282758
##     Tx_Suc.brt_l    Tx_Suc.brt_es     Tx_Suc.brt_s     Tx_Suc.att_l
##        0.3185938        0.4188504        0.6429324        0.6550707
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##        0.6517884        0.8641411        0.9344800        1.0953708
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##        0.5832310        0.7793598        0.7602870        1.4611691
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##        1.0541526        1.9165925
##
## $log10bf
##  [1] -0.8621379 -0.8621379 -0.8621379 -0.8621379 -0.8621379 17.5056625
##  [7] -0.8621379 -0.8621379 -0.8621379 -0.8621379 -0.8621379 -0.8621379
## [13] -0.8621379 -0.8621379 -0.8621379 -0.8621379 -0.8621379
##
## $postmeansigma2
## [1] 1.79132
##
## $postvarsigma2
## [1] 6.551355
```

- Mutations en anglais - A partir de la fonction $BayesReg$ pour $g = 1$

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  85.1346      0.9554
## x1          0.0000      1.7198 -0.1505
## x2          0.0000      1.6145 -0.1505
## x3          0.0000      2.3088 -0.1505
## x4          0.0000      1.1708 -0.1505
## x5          0.0000      1.5392 -0.1505
## x6          4.7292      2.3627  0.7199      (**)
## x7          0.0000      2.4073 -0.1505
## x8          0.0000      2.3952 -0.1505
## x9          0.0000      3.1756 -0.1505
## x10         0.0000      3.4341 -0.1505
## x11         0.0000      4.0254 -0.1505
## x12         0.0000      2.1433 -0.1505
## x13         0.0000      2.8641 -0.1505
## x14         0.0000      2.7940 -0.1505
## x15         0.0000      5.3696 -0.1505
```

```
## x16         0.0000      3.8739 -0.1505
## x17         0.0000      7.0433 -0.1505
##
##
## Posterior Mean of Sigma2: 47.47
## Posterior StError of Sigma2: 67.8284
```

```
## $postmeancoeff
##  [1]  8.513462e+01 -3.286260e-14 -5.298539e-14  7.638334e-14  1.332268e-15
##  [6] -5.684342e-14  4.729235e+00  3.830269e-14  1.039169e-13  5.973000e-14
## [11] -4.352074e-14  6.394885e-14 -5.329071e-15 -5.506706e-14  1.243450e-14
## [16]  5.284662e-14  1.243450e-14 -2.113865e-13
##
## $postsqrtcoeff
##                       Eff_Prst_l      Eff_Prst_es       Eff_Prst_s
##        0.9554497        1.7198243        1.6144555        2.3088435
##     Tx_Suc.brt_l    Tx_Suc.brt_es     Tx_Suc.brt_s     Tx_Suc.att_l
##        1.1707966        1.5392285        2.3627050        2.4073121
##    Tx_Suc.att_es     Tx_Suc.att_s          Eff_2nd          Eff_1er
##        2.3952498        3.1756225        3.4341100        4.0253659
## Tx_Acc.brt_bac.2 Tx_Acc.att_bac.2 Tx_Acc.brt_bac.1 Tx_Acc.att_bac.1
##        2.1433090        2.8640606        2.7939703        5.3696341
##   Tx_Suc.brt_Tot   Tx_Suc.att_Tot
##        3.8738935        7.0432642
##
## $log10bf
##  [1] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150  0.7198731
##  [7] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150
## [13] -0.1505150 -0.1505150 -0.1505150 -0.1505150 -0.1505150
##
## $postmeansigma2
## [1] 47.46998
##
## $postvarsigma2
## [1] 4600.689
```

- Conclusion

La 6ème variable: *taux_brut_de_reussite_serie_s* est prépondérante dans tous les cas.

2.4.2 Choix de modèles par test de tous les modèles ou Gibbs-sampler

On utilise la fonction *ModChoBayesReg* du package *Bayess*

- Mutations en Math

```
##
## Number of variables greather than 15
## Model posterior probabilities are estimated by using an MCMC algorithm
##
##    Top10Models PostProb
## 1            6   0.1499
## 2          6 9   0.0208
## 3          3 6   0.0205
## 4          2 6   0.0200
```

```
## 5          6 8   0.0192
## 6          5 6   0.0188
## 7          4 6   0.0187
## 8          1 6   0.0186
## 9         6 14   0.0184
## 10        6 13   0.0180

## $top10models
##  [1] "6"    "6 9"  "3 6"  "2 6"  "6 8"  "5 6"  "4 6"  "1 6"  "6 14" "6 13"
##
## $postprobtop10
##  [1] 0.1499125 0.0207875 0.0205250 0.0200500 0.0191875 0.0187750 0.0187375
##  [8] 0.0186500 0.0183625 0.0180500
```

La 6ème covariable est omniprésente dans tous les modèles. La probabilité à piriori du modèle constitué de cette seule variable est écrasante.

- Mutations en Anglais

```
y<-y.en
X<-X.en
```

```
##
## Number of variables greather than 15
## Model posterior probabilities are estimated by using an MCMC algorithm
##
##    Top10Models PostProb
## 1            6   0.1318
## 2         6 17   0.0224
## 3          2 6   0.0194
## 4         6 10   0.0194
## 5          4 6   0.0192
## 6         6 16   0.0181
## 7         6 13   0.0179
## 8          1 6   0.0176
## 9         6 11   0.0174
## 10        6 15   0.0172

## $top10models
##  [1] "6"    "6 17" "2 6"  "6 10" "4 6"  "6 16" "6 13" "1 6"  "6 11" "6 15"
##
## $postprobtop10
##  [1] 0.1317750 0.0224000 0.0194375 0.0193625 0.0192125 0.0181000 0.0179375
##  [8] 0.0176125 0.0174125 0.0172250
```

On retrouve la encore la prédominance de la 6ème variable : Suc.brt_s soit le *taux_brut_de_reussite_serie_s*.

2.4.3 Comparaison au résultat obtenu par une analyse fréquentiste

Analyse fréquentiste - Mutations en Mathématiques et en Anglais

- Régresssion linéaire - Résumé

Cas des mutations en mathématiques:

```
##
## Call:
## lm(formula = Barre ~ ., data = d.math.reg)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -6.973e-14 -1.009e-14 -3.688e-15  1.135e-14  1.451e-13
##
## Coefficients:
##                    Estimate Std. Error    t value Pr(>|t|)
## (Intercept)       8.610e+01  4.159e-15  2.070e+16  < 2e-16 ***
## Eff_Prst_l       -6.827e-15  7.736e-15 -8.820e-01  0.38266
## Eff_Prst_es      -2.159e-15  9.892e-15 -2.180e-01  0.82828
## Eff_Prst_s        2.502e-15  1.496e-14  1.670e-01  0.86799
## Tx_Suc.brt_l     -1.630e-14  6.380e-15 -2.555e+00  0.01444 *
## Tx_Suc.brt_es    -1.644e-14  1.028e-14 -1.600e+00  0.11728
## Tx_Suc.brt_s      9.657e+00  1.482e-14  6.517e+14  < 2e-16 ***
## Tx_Suc.att_l     -1.440e-14  1.150e-14 -1.252e+00  0.21775
## Tx_Suc.att_es     4.666e-14  1.672e-14  2.790e+00  0.00797 **
## Tx_Suc.att_s     -2.169e-14  2.259e-14 -9.600e-01  0.34276
## Eff_2nd           2.930e-15  2.007e-14  1.460e-01  0.88464
## Eff_1er           4.728e-15  2.294e-14  2.060e-01  0.83776
## Tx_Acc.brt_bac.2 -3.088e-14  1.507e-14 -2.048e+00  0.04698 *
## Tx_Acc.att_bac.2  2.235e-14  1.647e-14  1.357e+00  0.18229
## Tx_Acc.brt_bac.1  3.000e-14  2.030e-14  1.477e+00  0.14723
## Tx_Acc.att_bac.1  1.093e-14  2.673e-14  4.090e-01  0.68485
## Tx_Suc.brt_Tot    1.198e-14  2.451e-14  4.890e-01  0.62750
## Tx_Suc.att_Tot   -4.326e-14  3.942e-14 -1.097e+00  0.27894
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.195e-14 on 41 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 3.118e+29 on 17 and 41 DF,  p-value: < 2.2e-16
```

Cas des mutations en anglais:

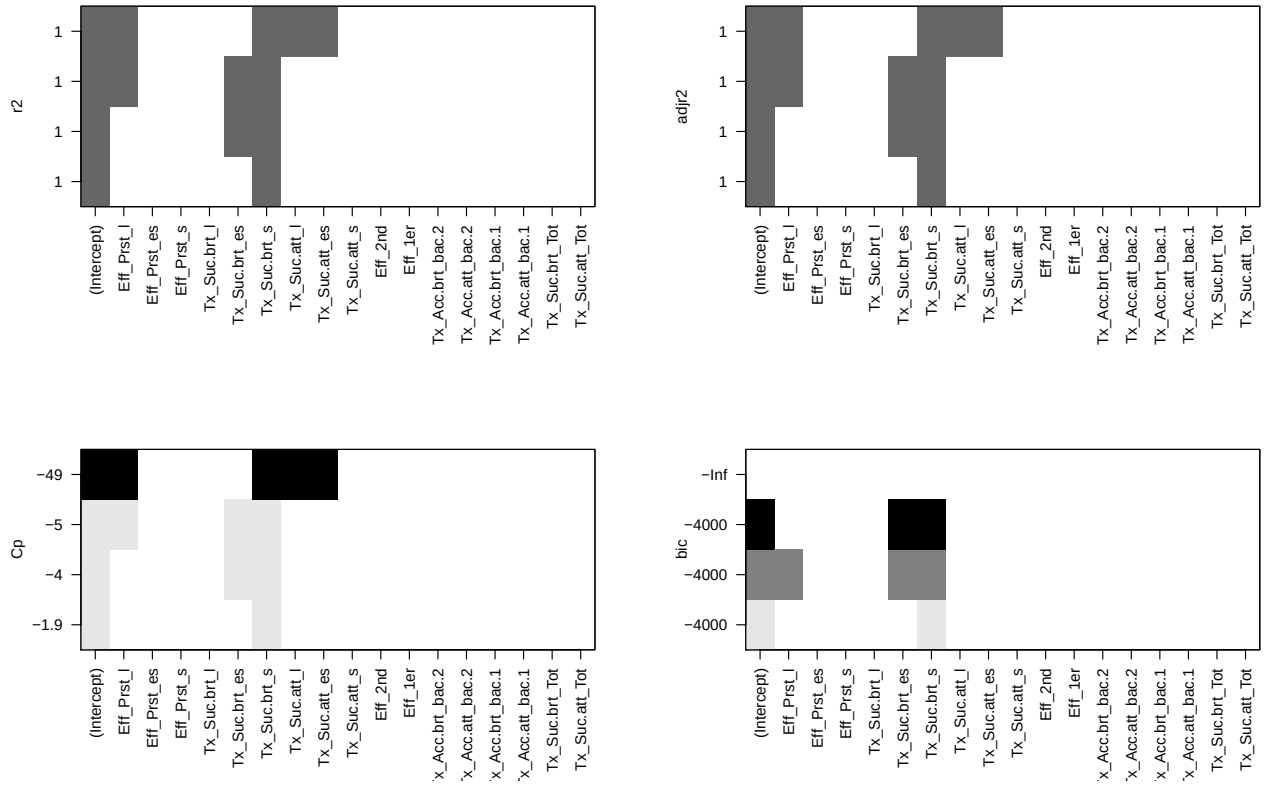
```
##
## Call:
## lm(formula = Barre ~ ., data = d.en.reg)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.381e-14 -9.472e-15 -4.810e-16  9.431e-15  6.801e-14
##
## Coefficients:
##                    Estimate Std. Error    t value Pr(>|t|)
## (Intercept)       8.513e+01  3.003e-15  2.835e+16   <2e-16 ***
## Eff_Prst_l       -4.546e-15  7.719e-15 -5.890e-01   0.5598
## Eff_Prst_es       4.722e-15  7.246e-15  6.520e-01   0.5190
## Eff_Prst_s        6.061e-15  1.036e-14  5.850e-01   0.5625
## Tx_Suc.brt_l      8.442e-15  5.255e-15  1.606e+00   0.1174
## Tx_Suc.brt_es     9.616e-15  6.909e-15  1.392e+00   0.1730
## Tx_Suc.brt_s      9.551e+00  1.060e-14  9.006e+14   <2e-16 ***
## Tx_Suc.att_l      1.628e-14  1.080e-14  1.507e+00   0.1411
## Tx_Suc.att_es     9.921e-15  1.075e-14  9.230e-01   0.3626
```

```
## Tx_Suc.att_s     -3.051e-14  1.425e-14 -2.141e+00   0.0395 *
## Eff_2nd          -2.187e-14  1.541e-14 -1.419e+00   0.1650
## Eff_1er           1.497e-14  1.807e-14  8.290e-01   0.4131
## Tx_Acc.brt_bac.2 -1.775e-14  9.620e-15 -1.845e+00   0.0738 .
## Tx_Acc.att_bac.2 -1.336e-15  1.285e-14 -1.040e-01   0.9178
## Tx_Acc.brt_bac.1  2.892e-14  1.254e-14  2.306e+00   0.0273 *
## Tx_Acc.att_bac.1 -4.673e-15  2.410e-14 -1.940e-01   0.8474
## Tx_Suc.brt_Tot   -2.876e-14  1.739e-14 -1.654e+00   0.1073
## Tx_Suc.att_Tot    9.699e-15  3.161e-14  3.070e-01   0.7609
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.166e-14 on 34 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.835e+29 on 17 and 34 DF,  p-value: < 2.2e-16
```

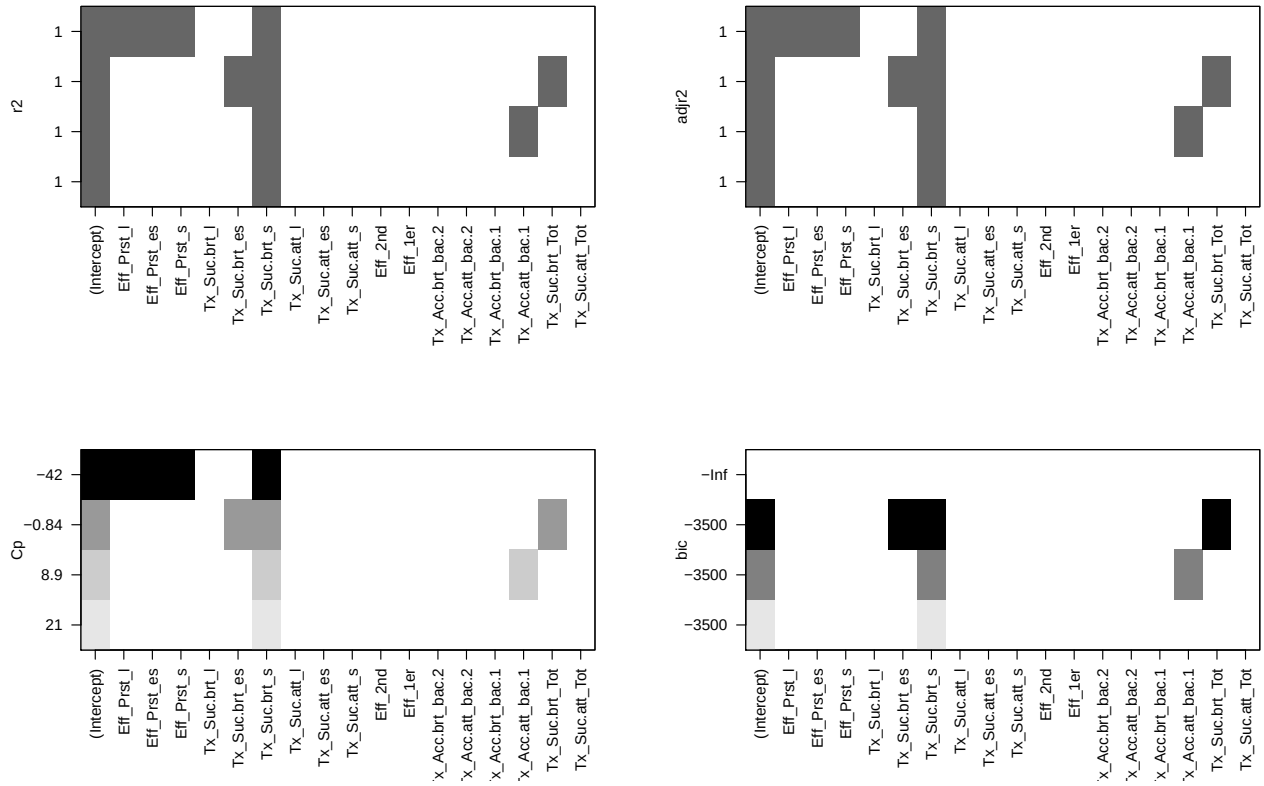
On trouve des résultats comparable en particulier pour la significativité de la variable: $Tx_Suc.brt_s$.

- Choix de modèles - méthode *regsubsets*

Cas des mutations en mathématiques:



Cas des mutations en anglais:



- Choix de modèles - méthode *step*

Cas des mutations en mathématiques:

```
summary(step_mod.math)
```

```
##
## Call:
## lm(formula = Barre ~ Eff_Prst_es + Eff_Prst_s + Tx_Suc.brt_l +
##     Tx_Suc.brt_es + Tx_Suc.brt_s + Tx_Suc.att_es + Tx_Acc.brt_bac.2 +
##     Tx_Acc.att_bac.2 + Tx_Acc.brt_bac.1 + Tx_Suc.att_Tot, data = d.math.reg)
##
## Residuals:
##        Min          1Q      Median          3Q         Max
## -7.245e-14 -1.177e-14 -1.458e-15  8.307e-15  1.522e-13
##
## Coefficients:
##                     Estimate Std. Error    t value Pr(>|t|)
## (Intercept)       8.610e+01  3.917e-15  2.198e+16  < 2e-16 ***
## Eff_Prst_es      -1.281e-14  8.705e-15 -1.471e+00 0.147802
## Eff_Prst_s        7.699e-15  9.517e-15  8.090e-01 0.422533
## Tx_Suc.brt_l     -1.862e-14  5.056e-15 -3.683e+00 0.000585 ***
## Tx_Suc.brt_es    -1.533e-14  7.571e-15 -2.025e+00 0.048456 *
## Tx_Suc.brt_s      9.657e+00  8.354e-15  1.156e+15  < 2e-16 ***
## Tx_Suc.att_es     4.588e-14  1.372e-14  3.345e+00 0.001605 **
## Tx_Acc.brt_bac.2 -3.152e-14  1.078e-14 -2.925e+00 0.005247 **
## Tx_Acc.att_bac.2  2.271e-14  1.241e-14  1.831e+00 0.073336 .
```

```
## Tx_Acc.brt_bac.1  4.647e-14  1.195e-14  3.888e+00 0.000310 ***
## Tx_Suc.att_Tot   -6.595e-14  1.793e-14 -3.679e+00 0.000591 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.009e-14 on 48 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.976e+29 on 10 and 48 DF,  p-value: < 2.2e-16
```

Cas des mutations en anglais:

```
summary(step_mod.en)
```

```
##
## Call:
## lm(formula = Barre ~ Eff_Prst_s + Tx_Suc.brt_l + Tx_Suc.brt_es +
##     Tx_Suc.brt_s + Tx_Suc.att_l + Tx_Suc.att_es + Tx_Suc.att_s +
##     Eff_2nd + Tx_Acc.brt_bac.2 + Tx_Acc.brt_bac.1 + Tx_Suc.brt_Tot,
##     data = d.en.reg)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.892e-14 -8.816e-15 -1.148e-15  9.297e-15  7.071e-14
##
## Coefficients:
##                     Estimate Std. Error    t value Pr(>|t|)
## (Intercept)       8.513e+01  2.804e-15  3.036e+16  < 2e-16 ***
## Eff_Prst_s        1.187e-14  7.162e-15  1.657e+00  0.10535
## Tx_Suc.brt_l      9.788e-15  3.930e-15  2.490e+00  0.01702 *
## Tx_Suc.brt_es     1.156e-14  4.849e-15  2.384e+00  0.02195 *
## Tx_Suc.brt_s      9.551e+00  8.100e-15  1.179e+15  < 2e-16 ***
## Tx_Suc.att_l      1.511e-14  6.277e-15  2.407e+00  0.02081 *
## Tx_Suc.att_es     1.177e-14  8.624e-15  1.365e+00  0.17989
## Tx_Suc.att_s     -2.374e-14  9.918e-15 -2.394e+00  0.02145 *
## Eff_2nd          -1.371e-14  5.594e-15 -2.451e+00  0.01869 *
## Tx_Acc.brt_bac.2 -1.339e-14  5.940e-15 -2.254e+00  0.02977 *
## Tx_Acc.brt_bac.1  2.127e-14  9.059e-15  2.348e+00  0.02390 *
## Tx_Suc.brt_Tot   -3.420e-14  1.226e-14 -2.791e+00  0.00802 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.022e-14 on 40 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.034e+30 on 11 and 40 DF,  p-value: < 2.2e-16
```

2.5 Conclusion

Pour les mutations en Math et en Anglais, dans l'approche bayésiennne une covariable ressort très nettement: *taux_brut_de_reussite_serie_s*. On retouve la significativité de cette variable, quelque soit la matière math ou anglais. Les résultats obtenus pour l'une ou l'autre des matière sont très proches. Dans l'approche cas fréquentiste (modèle linéaire classique) on trouve encore que cette variable: *taux_brut_de_reussite_serie_s* est très significative. Par contre de nombreuses autres variables sont elles aussi significaive. On a plus de difficulté à sélectionner les variables qui comme ont la vue sont très corrélées. Dans le cadre bayésien la loi à priori choisie g-prior de Zellner a la particularité d'éliminer les corrélations entre covariables. Ceci pouvant peu-être expliquer la différence notoire des deux approches.

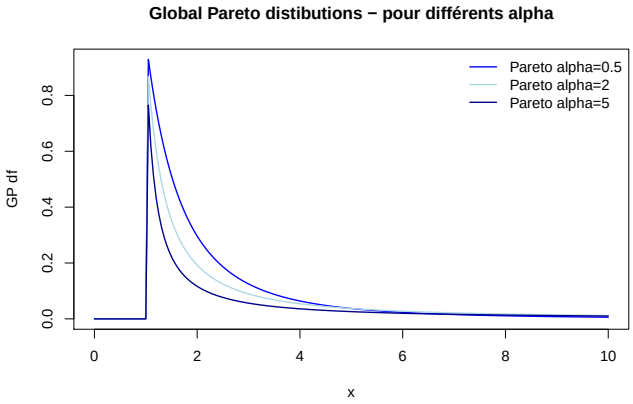
3 Partie II - Loi de Pareto

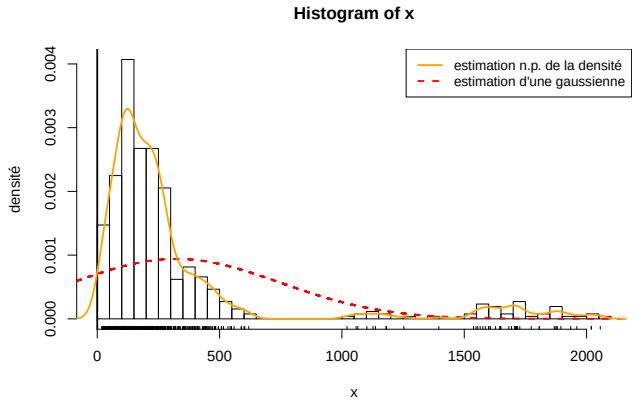
On ignore maintenant les covariables, et on s'intéresse uniquement à la loi du nombre de points nécessaire (colonne Barre). La loi gaussienne peut paraître peu pertinente pour ces données : on va plutôt proposer une loi de Pareto. Pour $m > 0$ et $\alpha > 0$, on dit que $ZPareto(m; \alpha)$ si Z est à valeurs dans $[m; +1[$ de densité:

$f(z \mid \alpha, m) = \alpha \frac{m^\alpha}{z^{\alpha+1}} \mathbb{1}_{[\geqslant, +\infty[}$

3.1 Package R pour générer des réalisation d'une loi de Paréto

On peut utiliser le package *extRemes* et la fonction *devd*





3.2 Choix d'une loi à priori pour α

- Loi de paréto :

$$f(z \mid \alpha, m) = \alpha \frac{m^\alpha}{z^{\alpha+1}} \mathbb{1}_{[\geqslant, +\infty[}$$

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    21.0   111.0   196.0   321.9   292.0  2056.0
```

le résumé des données nous ammène à choisir: m=21

A une constante multiplicative près et après transformation en log, on reconnaît une loi exponentielle de paramètre α.

$$f(z \mid \alpha, m) \propto \alpha e^{\alpha log(m/z)}$$

En applicant la transformation : $z \to ln(\frac{z}{m})$ a notre échantillon (Z_i), on a que $ln(\frac{Z}{m}) \sim Exp(\alpha)$

On peut alors estimer le paramètre α par mle à partir de la fonction R: $fitdist$ du package $fitdistrplus$.

```
m=21
y.exp<-log(y.tot/m)
fit.exp <- fitdist(y.exp, "exp", method="mle")
fit.exp
```

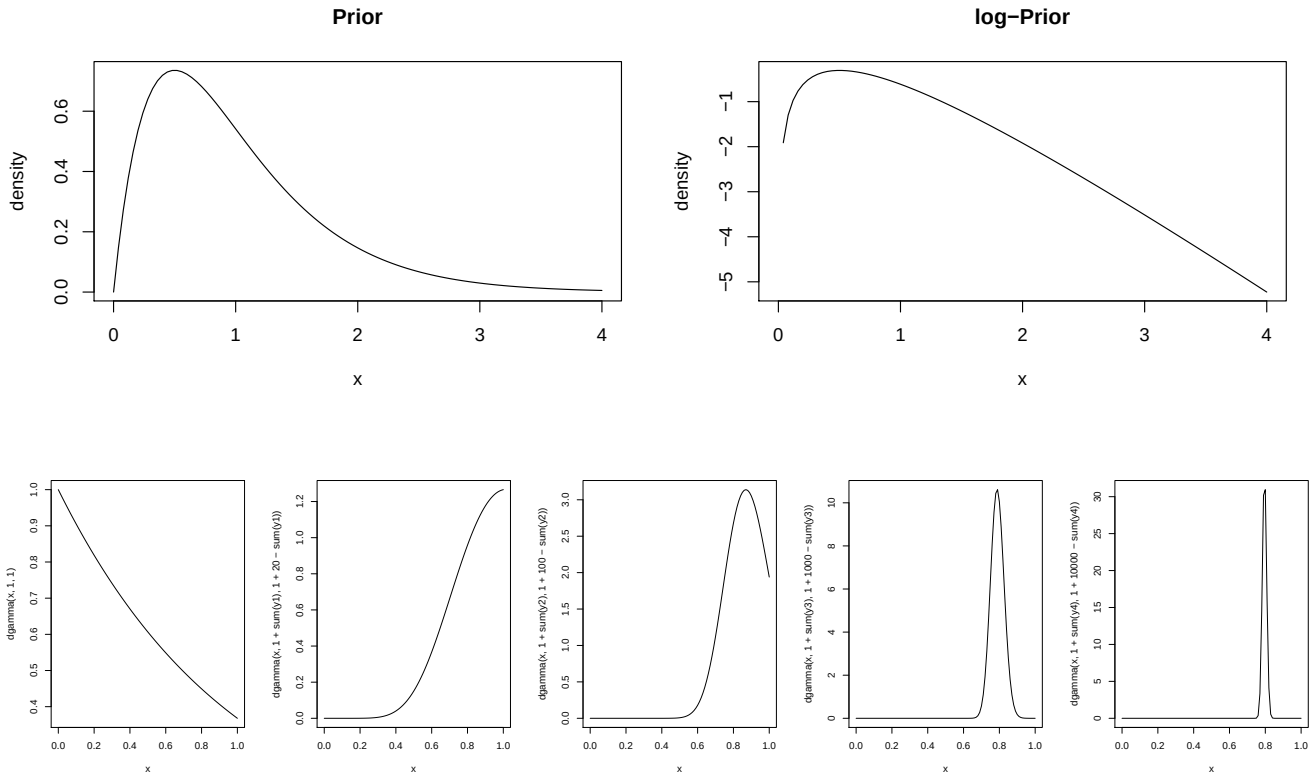
```
## Fitting of the distribution ' exp ' by maximum likelihood
## Parameters:
##       estimate Std. Error
## rate 0.4502063 0.01981913
```

On peut prendre pour loi à priori la loi $\Gamma(a, b)$ de manière à avoir une loi conjuguée. Nous allons tester une loi a priori avec un paramètre shape $= 2$ et scale $= 2$.

```
prior = function(alpha){
return(dgamma(alpha, 2, 2))}

logprior = function(alpha){
return(dgamma(alpha, 2, 2, log = T))}
```

```
par(mfrow = c(1, 2))
curve(dgamma(x, 2, 2), xlim=c(0, 4), main="Prior", ylab="density")
curve(dgamma(x, 2, 2, log = T), xlim=c(0, 4), main="log-Prior", ylab="density")
```



3.3 Loi à postériori de α

La loi à postériori correspondante est la loi : $\Gamma(a + n, b + \sum_{i=1}^{n} ln(\frac{Z_i}{m}))$

```
logposterior <- function(m,alpha,y){
n<-length(y)
loglkd <- n*log(alpha) + alpha*n*log(m)-(alpha+1)*sum(log(y))
if(!is.finite(loglkd)) return(-Inf)
return(loglkd+logprior(alpha))
}
```

3.4 Echantillon de la loi à postériori de α

Par la méthode de votre choix, tirer un échantillon de la loi a posteriori de α. Donner un intervalle de crédibilité à 95%.

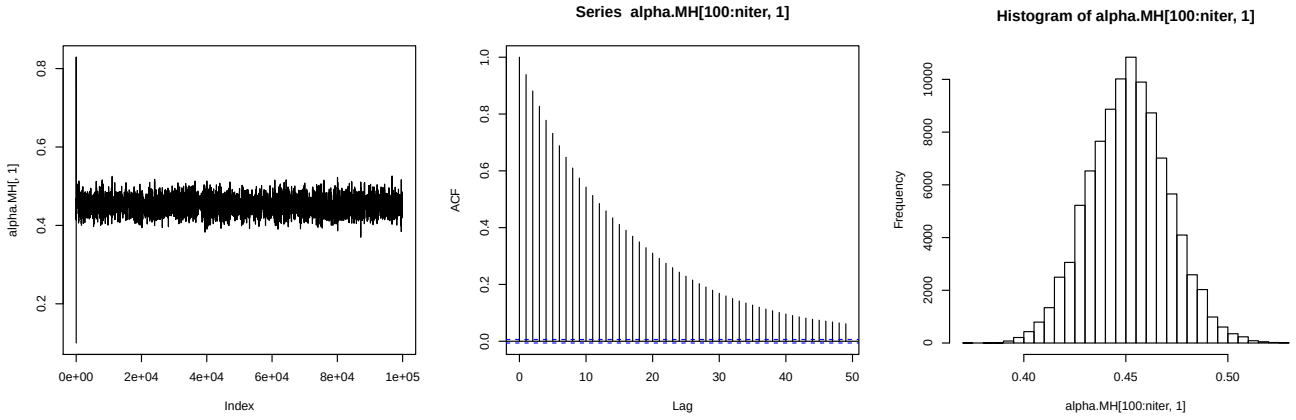
```
MH <- function(Y,alpha0, m, niter){
  alpha <- matrix(NA, nrow=niter, ncol=1)
  alpha[1] <- alpha0
  for(i in 2:niter){
    proposal <- rgamma(1, 2, 2)
    logalpha <- logposterior(m, proposal, Y)- logposterior(m, alpha[i-1,], Y)
    if(log(runif(1)) < logalpha){
      alpha[i] <- proposal
    }
    else{
      alpha[i] <- alpha[i-1]
    }
  }
  return(alpha)
}
```

```
alpha.MH <- MH(Y=y.tot, alpha0=0.1, m=21, niter=1e5)
```

```
niter=1e5
# Etudions la sortie de l'algorithme
par(mfcol=c(1,3))
# trace
plot(alpha.MH[, 1], type="l")

# autocorrÃ©lations
acf(alpha.MH[100:niter, 1])

# histogrammes
hist(alpha.MH[100:niter, 1], breaks=50)
```



Intervalle de confiance à 95% :

```
##       2.5%      97.5%
## 0.4141557 0.4892310
```

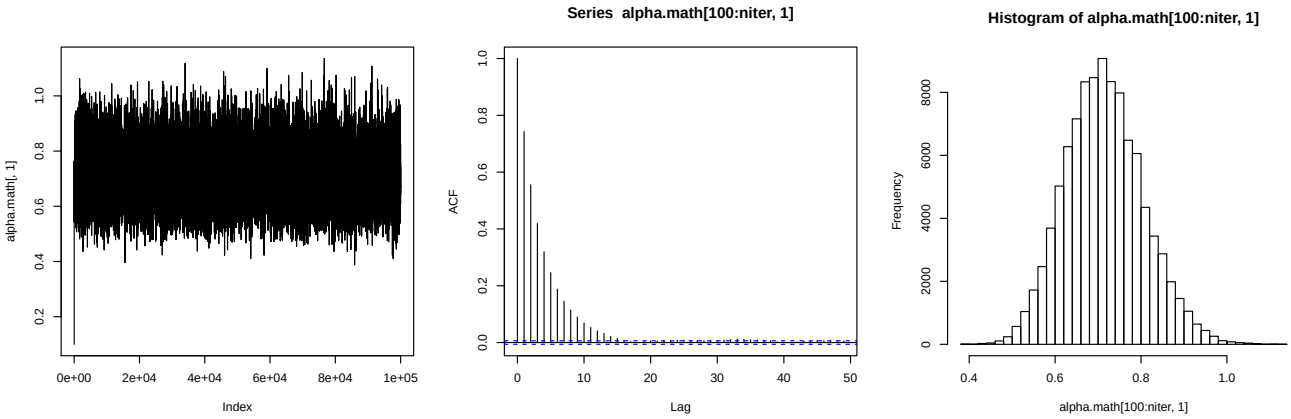
3.5 Analyse pour les mutation en anglais et en math

3.5.1 Calcul du *alpha* par l'alogorithme de Métropolis-Hastigs

```
niter <- 1e5
alpha.math <- MH(y.math, .1, 21, niter)
alpha.en <- MH(y.en, .1, 21, niter)
```

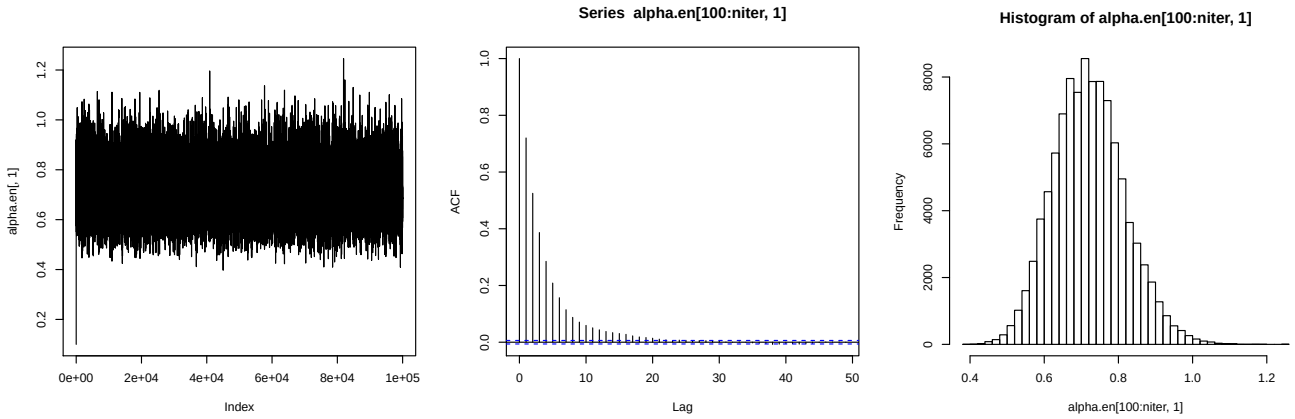
3.5.2 Convergence de l'algorithme de Metropolois-Hastings: mutations en mathématiques

```
# Etudions la sortie de l'algorithme
par(mfcol=c(1,3))
# trace
plot(alpha.math[, 1], type="l")
# autocorrélations
acf(alpha.math[100:niter, 1])
# histogrammes
hist(alpha.math[100:niter, 1], breaks=50)
```



3.5.3 Convergence de l'algorithme de Metropolois-Hastings: mutations en anglais

```
# Etudions la sortie de l'algorithme
par(mfcol=c(1,3))
# trace
plot(alpha.en[, 1], type="l")
# autocorrélations
acf(alpha.en[100:niter, 1])
# histogrammes
hist(alpha.en[100:niter,1], breaks=50)
```



Intervalle de confiance à 95% math et anglais

```
quantile(alpha.math , c(.025,.975))
```

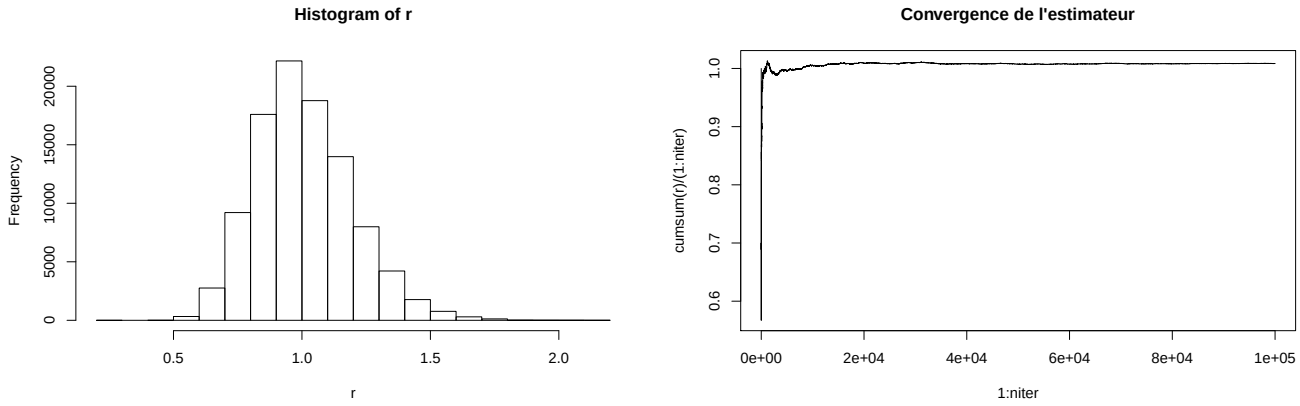
```
##      2.5%     97.5%
## 0.5458861 0.9034245
```

```
quantile(alpha.en , c(.025,.975))
```

```
##      2.5%     97.5%
## 0.5462845 0.9201473
```

On va tester l'hypothèse $\alpha_{math} = \alpha_{anglais}$. Pour celà on va estimer l'espérance à postériori du quotient $r_\alpha = \frac{\alpha_{math}}{\alpha_{anglais}}$. On utilise les approximations obtenues par Métropolis-Hastings précédemment pour chacun des α.

On regarde la convergence de l'estimateur, qu id'après les graphiques est obtenue à partir de 10000 path.




```
## [1] 1.008511
```

```
## [1] 0.1882724
```

```
##      2.5%     97.5%
## 0.6885549 1.4226057
```

A la vue des résultats on peut conclure à l'égalité des paramètre α pour les mutations en math et en anglais.

4 Annexes

Test des méthodes BayesReg du package Bayess et BayesReg2 version modifiée

```
data(faithful)
BayesReg(faithful[,1],faithful[,2])
```

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept   3.4878      0.0304
## x1          1.0225      0.0303     Inf    (****)
##
##
## Posterior Mean of Sigma2: 0.2513
## Posterior StError of Sigma2: 0.3561

## $postmeancoeff
## [1] 3.487783 1.022509
##
## $postsqrtcoeff
## [1] 0.03039825 0.03034252
##
## $log10bf
##      [,1]
## [1,]  Inf
##
## $postmeansigma2
## [1] 0.2513425
##
## $postvarsigma2
## [1] 0.1268176
```

```
BayesReg2(faithful[,1],faithful[,2])
```

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept   3.4878      0.0304
## x1          1.0244      0.0304     Inf    (****)
##
##
## Posterior Mean of Sigma2: 0.2513
## Posterior StError of Sigma2: 0.3561

## $postmeancoeff
## [1] 3.487783 1.024394
##
## $postsqrtcoeff
## [1] 0.03039825 0.03039845
##
## $log10bf
##      [,1]
## [1,]  Inf
##
## $postmeansigma2
```

```
## [1] 0.2513425
##
## $postvarsigma2
## [1] 0.1268176
```

```
data("caterpillar")
y.cat=log(caterpillar$y)
X.cat=as.matrix(caterpillar[,1:8])
```

- Fonction BayesReg

```
BayesReg(y.cat, scale(X.cat))
```

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  -0.8133      0.1407
## x1         -0.5039      0.1883  0.7224      (**)
## x2         -0.3755      0.1508  0.5392      (**)
## x3          0.6225      0.3436 -0.0443
## x4         -0.2776      0.2804 -0.5422
## x5         -0.2069      0.1499 -0.3378
## x6          0.2806      0.4760 -0.6857
## x7         -1.0420      0.4178  0.5435      (**)
## x8         -0.0221      0.1531 -0.7609
##
##
## Posterior Mean of Sigma2: 0.6528
## Posterior StError of Sigma2: 0.939
```

```
## $postmeancoeff
## [1] -0.81328069 -0.50390377 -0.37548142  0.62252447 -0.27762947 -0.20688023
## [7]  0.28061938 -1.04204277 -0.02209411
##
## $postsqrtcoeff
##                  x1        x2        x3        x4        x5        x6
## 0.1406514 0.1882559 0.1508271 0.3436217 0.2803657 0.1498641 0.4759505
##        x7        x8
## 0.4178148 0.1530573
##
## $log10bf
## [1]  0.72241000  0.53918250 -0.04430805 -0.54224765 -0.33779821 -0.68568404
## [7]  0.54353138 -0.76091468
##
## $postmeansigma2
## [1] 0.6528327
##
## $postvarsigma2
## [1] 0.8817734
```

```
BayesReg2(y.cat, scale(X.cat))
```

```
##
##           PostMean PostStError Log10bf EvidAgaH0
## Intercept  -0.8133      0.1407
```

```
## x1         -0.5117      0.1912  0.7224      (**)
## x2         -0.3813      0.1532  0.5392      (**)
## x3          0.6322      0.3489 -0.0443
## x4         -0.2819      0.2847 -0.5422
## x5         -0.2101      0.1522 -0.3378
## x6          0.2850      0.4833 -0.6857
## x7         -1.0582      0.4243  0.5435      (**)
## x8         -0.0224      0.1554 -0.7609
##
##
## Posterior Mean of Sigma2: 0.6528
## Posterior StError of Sigma2: 0.939
```

```
## $postmeancoeff
## [1] -0.81328069 -0.51171670 -0.38130319  0.63217659 -0.28193406 -0.21008787
## [7]  0.28497033 -1.05819944 -0.02243668
##
## $postsqrtcoeff
##                  x1        x2        x3        x4        x5        x6
## 0.1406514 0.1911748 0.1531656 0.3489495 0.2847127 0.1521877 0.4833300
##        x7        x8
## 0.4242930 0.1554305
##
## $log10bf
## [1]  0.72241000  0.53918250 -0.04430805 -0.54224765 -0.33779821 -0.68568404
## [7]  0.54353138 -0.76091468
##
## $postmeansigma2
## [1] 0.6528327
##
## $postvarsigma2
## [1] 0.8817734
```

Les légères différences s'expliquent par la fonction utilisée pour centrer et réduire.

- Fonction *ModChoBayesReg* pour le choix de modèle

```
ModChoBayesReg(y.cat,X.cat)
```

```
##
## Number of variables less than 15
## Model posterior probabilities are calculated exactly
##
##    Top10Models PostProb
## 1        1 2 7   0.0767
## 2          1 7   0.0689
## 3      1 2 3 7   0.0686
## 4        1 3 7   0.0376
## 5        1 2 6   0.0369
## 6    1 2 3 5 7   0.0326
## 7      1 2 5 7   0.0294
## 8          1 6   0.0205
## 9      1 2 4 7   0.0201
## 10           7   0.0198
```

```
## $top10models
##  [1] "1 2 7"     "1 7"       "1 2 3 7"   "1 3 7"     "1 2 6"
##  [6] "1 2 3 5 7" "1 2 5 7"   "1 6"       "1 2 4 7"   "7"
##
## $postprobtop10
##  [1] 0.07670048 0.06894313 0.06855427 0.03759751 0.03688912 0.03262797
##  [7] 0.02941759 0.02050185 0.02006371 0.01979095
```

```
ModChoBayesReg2(y.cat,X.cat,bCalc=FALSE)
```

```
##
## bCalc + false
## Model posterior probabilities are calculated by Gibbs
##
##    Top10Models PostProb
## 1        1 2 7   0.0789
## 2          1 7   0.0728
## 3      1 2 3 7   0.0685
## 4        1 3 7   0.0392
## 5        1 2 6   0.0370
## 6    1 2 3 5 7   0.0352
## 7      1 2 5 7   0.0311
## 8      1 2 4 7   0.0205
## 9          1 2   0.0200
## 10           7   0.0198

## $top10models
##  [1] "1 2 7"     "1 7"       "1 2 3 7"   "1 3 7"     "1 2 6"
##  [6] "1 2 3 5 7" "1 2 5 7"   "1 2 4 7"   "1 2"       "7"
##
## $postprobtop10
##  [1] 0.0788875 0.0727875 0.0685250 0.0391500 0.0370125 0.0351875 0.0311375
##  [8] 0.0204875 0.0199750 0.0198500
```

```
ModChoBayesReg2(y.cat,X.cat,bCalc=TRUE)
```

```
##
## bCalc = TRUE
## Model posterior probabilities are calculated exactly
##
##    Top10Models PostProb
## 1        1 2 7 -24.3915
## 2          1 7 -24.4378
## 3      1 2 3 7 -24.4402
## 4        1 3 7 -24.7011
## 5        1 2 6 -24.7094
## 6    1 2 3 5 7 -24.7627
## 7      1 2 5 7 -24.8076
## 8          1 6 -24.9645
## 9      1 2 4 7 -24.9738
## 10           7 -24.9798

## $top10models
##  [1] "1 2 7"     "1 7"       "1 2 3 7"   "1 3 7"     "1 2 6"
##  [6] "1 2 3 5 7" "1 2 5 7"   "1 6"       "1 2 4 7"   "7"
```

```
##
## $postprobtop10
##  [1] -24.39145 -24.43776 -24.44021 -24.70109 -24.70935 -24.76266 -24.80764
##  [8] -24.96446 -24.97384 -24.97978
```

Test anova des modèles linéaire du cas général

- On considère les 2 modèles suivants :

taux_reussite_attendu_serie_l + taux_acces_attendu_premiere_bac + taux_acces_brut_seconde_bac + taux_acces_brut_premiere_bac

```
reg.mod2 = lm(Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2 + Tx_Acc.brt_bac.1, data=d.reg)
summary(reg.mod2)
```

```
##
## Call:
## lm(formula = Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2 +
##     Tx_Acc.brt_bac.1, data = d.reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -410.82 -203.23 -128.06   -4.57 1670.03
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        321.92      18.47  17.432  < 2e-16 ***
## Tx_Suc.att_l       -75.79      34.68  -2.185  0.02934 *
## Tx_Acc.att_bac.1   124.40      46.07   2.700  0.00716 **
## Tx_Acc.brt_bac.2    45.31      33.69   1.345  0.17930
## Tx_Acc.brt_bac.1   -43.21      42.17  -1.025  0.30600
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 419.5 on 511 degrees of freedom
## Multiple R-squared:  0.02905,    Adjusted R-squared:  0.02145
## F-statistic: 3.822 on 4 and 511 DF,  p-value: 0.004514
```

taux_reussite_attendu_serie_l + taux_acces_attendu_premiere_bac + taux_acces_brut_seconde_bac

```
reg.mod1 = lm(Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2 , data=d.reg)
summary(reg.mod1)
```

```
##
## Call:
## lm(formula = Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2,
##     data = d.reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -379.54 -206.00 -132.06   -2.57 1674.19
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)        321.92      18.47  17.431  < 2e-16 ***
## Tx_Suc.att_l       -66.47      33.47  -1.986  0.04759 *
## Tx_Acc.att_bac.1    93.91      35.17   2.670  0.00783 **
## Tx_Acc.brt_bac.2    26.38      28.18   0.936  0.34963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 419.5 on 512 degrees of freedom
## Multiple R-squared:  0.02705,    Adjusted R-squared:  0.02135
## F-statistic: 4.745 on 3 and 512 DF,  p-value: 0.002831
```

- On réalise maintenant des tests entre modèles emboîtés :

```
anova(reg.mod2,reg.mod1)
```

```
## Analysis of Variance Table
##
## Model 1: Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2 +
##     Tx_Acc.brt_bac.1
## Model 2: Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2
##   Res.Df      RSS Df Sum of Sq    F Pr(>F)
## 1    511 89918104
## 2    512 90102863 -1   -184758 1.05  0.306
```

Au vu des p-valeurs des tests de Fisher, on peut envisager de se passer de la variable : taux_acces_brut_premiere_bac
On conserve le plus petit modèle : reg.mod1

On réalise à nouveaux un test anova, maintenant entre reg.mod1 et step_mod.

```
anova(step_mod,reg.mod1)
```

```
## Analysis of Variance Table
##
## Model 1: Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1
## Model 2: Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1 + Tx_Acc.brt_bac.2
##   Res.Df      RSS Df Sum of Sq      F Pr(>F)
## 1    513 90257096
## 2    512 90102863  1    154234 0.8764 0.3496
```

Au vu des p-valeurs des tests de Fisher, on peut envisager de se passer des variables : taux_acces_brut_premiere_bac et taux_acces_brut_seconde_bac On conserve le plus petit modèle : step_mod

Un estimateur sans biais de σ^2 est donnée par la formule suivante:

$$\hat{\sigma}^2 = \frac{1}{n-p-1}(y - \hat{\alpha}\mathbb{1}_n - X\hat{\beta})^T(y - \hat{\alpha}\mathbb{1}_n - X\hat{\beta}) = \frac{s^2}{n-p-1}$$

on obtient σ^2

```
##          [,1]
## [1,] 2654620
```

et les estimations par les moindres carrés des coefficients de régression :

```
##
## Call:
## lm(formula = Barre ~ Tx_Suc.att_l + Tx_Acc.att_bac.1, data = d.reg)
##
## 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)        321.92      18.47  17.434  < 2e-16 ***
## Tx_Suc.att_l       -58.53      32.38  -1.808  0.07124 .
## Tx_Acc.att_bac.1   106.78      32.38   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
```

effectif_presents_serie_l
effectif_presents_serie_es taux_reussite_attendu_serie_l
taux_brut_de_reussite_total_series

Nouveau Nom	Ancien Nom
Prs_l	effectif_presents_serie_l