

Trinôme : Carine Sauser, Mélanie Groisne, Xavier Milhaud

Projet Etienne Marceau
Méthodes statistiques en assurance non vie

Méthodes statistiques pour la finance et l'assurance

Table des matières

1	Introduction	4
2	Question 1 : Présentation de la loi Burr de type XII	4
2.1	Caractéristiques de la loi	4
2.2	Propriétés	4
3	Question 2	5
3.1	Analyse qq-plot, fonction d'excès moyen empirique et fonction de répartition empirique . . .	5
3.2	Loi exponentielle	6
3.2.1	Estimation des paramètres par la méthode des moments	6
3.2.2	Estimation des paramètres par la méthode du maximum de vraisemblance	7
3.2.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du maximum de vraisemblance	7
3.2.4	Adéquation de la loi aux données	7
3.3	Loi Gamma	8
3.3.1	Estimation des paramètres par la méthode des moments	8
3.3.2	Estimation des paramètres par la méthode du maximum de vraisemblance	8
3.3.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode des moments	9
3.3.4	Adéquation de la loi aux données	9
3.4	Loi Log-normale	9
3.4.1	Estimation des paramètres par la méthode des moments	9
3.4.2	Estimation des paramètres par la méthode du maximum de vraisemblance	10
3.4.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode des moments	10
3.4.4	Adéquation de la loi aux données	10
3.5	Loi Pareto	10
3.5.1	Estimation des paramètres par la méthode des moments	11
3.5.2	Estimation des paramètres par la méthode du maximum de vraisemblance	11
3.5.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode des moments	12
3.5.4	Adéquation de la loi aux données	12
3.6	Loi Burr	12
3.6.1	Estimation des paramètres par la méthode des moments	13
3.6.2	Estimation des paramètres par la méthode du maximum de vraisemblance	13
3.6.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode des moments	14
3.6.4	Adéquation de la loi aux données	14
3.7	Sélection de la loi	14
4	Question 3	14
4.1	Fonction d'excès moyen empirique et fonction de répartition empirique	14
4.2	Loi exponentielle	15
4.2.1	Analyse qq-plot	15
4.2.2	Estimation des paramètres par la méthode des moments	16
4.2.3	Estimation des paramètres par la méthode du maximum de vraisemblance	16
4.2.4	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du maximum de vraisemblance	17
4.2.5	Adéquation de la loi aux données	18
4.3	Loi Gamma	18
4.3.1	Analyse qq-plot	18
4.3.2	Estimation des paramètres par la méthode des moments	18
4.3.3	Estimation des paramètres par la méthode du maximum de vraisemblance	19

4.3.4	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du max de vraisemblance	19
4.3.5	Adéquation de la loi aux données	19
4.4	Loi Log-normale	20
4.4.1	Analyse qq-plot	20
4.4.2	Estimation des paramètres par la méthode des moments	21
4.4.3	Estimation des paramètres par la méthode du maximum de vraisemblance	21
4.4.4	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du max de vraisemblance	21
4.4.5	Adéquation de la loi aux données	22
4.5	Loi Pareto	22
4.5.1	Analyse qq-plot	22
4.5.2	Estimation des paramètres par la méthode des moments	22
4.5.3	Estimation des paramètres par la méthode du maximum de vraisemblance	23
4.5.4	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du max de vraisemblance	24
4.5.5	Adéquation de la loi aux données	24
4.6	Loi Burr de type XII	24
4.6.1	Analyse qq-plot	25
4.6.2	Estimation des paramètres par la méthode du maximum de vraisemblance	25
4.6.3	Tracé de la fonction <i>excess-of-loss</i> avec paramètres de la méthode du max de vraisemblance	25
4.6.4	Adéquation de la loi aux données	25
4.7	Résumé des résultats	26
4.8	Sélection de la loi	28
5	Question 4 : loi de comptage modifiée à zéro	28
5.1	Cas général	28
5.2	Cas particuliers	29
6	Question 5	30
6.1	Loi de Poisson	30
6.1.1	Estimation des paramètres par la méthode des moments	30
6.1.2	Estimation des paramètres par la méthode du maximum de vraisemblance	30
6.1.3	Fonction de masse de probabilité empirique et paramétrique	31
6.1.4	Adéquation de la loi aux données	31
6.2	Loi Binomiale Négative	31
6.2.1	Estimation des paramètres par la méthode des moments	31
6.2.2	Estimation des paramètres par la méthode du maximum de vraisemblance	32
6.2.3	Fonction de masse de probabilité empirique et paramétrique	32
6.2.4	Adéquation de la loi aux données	32
6.3	Loi de Poisson modifiée à 0	32
6.3.1	Estimation des paramètres par la méthode des moments	33
6.3.2	Estimation des paramètres par la méthode du maximum de vraisemblance	33
6.3.3	Fonction de masse de probabilité empirique et paramétrique	34
6.3.4	Adéquation de la loi aux données	34
6.4	Loi Binomiale Négative modifiée à 0	34
6.4.1	Estimation des paramètres par la méthode des moments	35
6.4.2	Estimation des paramètres par la méthode du maximum de vraisemblance	35
6.4.3	Fonction de masse de probabilité empirique et paramétrique	35

6.4.4	Adéquation de la loi aux données	36
6.5	Sélection de la loi	36
7	Question 6	36
7.1	Loi Gamma	36
7.2	Loi Lognormale	37
7.3	Loi Pareto	38
7.4	Loi Burr Type XII	38
7.5	Méthode Peak-over-threshold avec la loi Pareto généralisée	39
7.6	Loi Lognormale-Pareto composite	40
8	Question 7 : comparaison de méthodes paramétriques et semi-paramétrique pour l'estimation de copule	40
8.1	Introduction à l'article	40
8.2	Description des trois méthodes d'estimation étudiées	40
8.2.1	Méthode du maximum de vraisemblance (ML)	40
8.2.2	Méthode IFM	41
8.2.3	Méthode semi-paramétrique (SP)	41
8.3	Conclusions des analyses de l'article	41
8.3.1	Comparaison des résultats	41
8.3.2	Conclusion générale	43
9	Question 8 :test d'adéquation pour les copules	43
9.1	introduction	43
9.2	les différentes procédures d'adéquation	43
9.3	Résumé des conclusions de l'étude faite dans l'article	44
10	Question 9 :estimation des paramètres des copules	45
10.1	méthode du maximum de vraisemblance (ML)	45
10.2	méthode IFM	45
11	Question 10 : Tarification de contrats basée sur l'expérience	45
11.1	Modèle de crédibilité de Buhlmann-Straub	46
11.2	Détermination de la prime de crédibilité et estimation de ses paramètres	46
11.3	Evaluation de la prime pure basée sur l'expérience	47
12	Conclusion	48
13	Annexes	49
14	Annexes	49
14.1	Question 2	49
14.2	Question 3	60
14.3	Question 5	73
14.4	Question 6	79

1 Introduction

Ce projet concernant les méthodes statistiques en assurance non-vie est l'occasion pour nous de mettre en pratique les connaissances théoriques acquises en cours sur des problèmes classiques qui se posent en assurance. C'est également l'occasion de se familiariser avec des logiciels de statistiques tels que R, très utilisé dans le monde de la finance et de l'assurance car complet et performant.

Nous aborderons dans ce projet des notions générales et importantes telles que l'estimation de paramètres par diverses méthodes pour différents modèles connus, quelques analyses graphiques des données qui nous permettent de nous faire une première opinion sur la pertinence du modèle testé, et des tests d'adéquation. Nous aurons également l'occasion de peaufiner notre culture générale par l'intermédiaire d'articles publiés récemment sur des sujets aussi importants et actuels que la théorie des copules, et enfin nous appliquerons un modèle très connu du monde de l'actuariat pour la tarification de contrats IARD basée sur l'expérience. Bonne lecture.

2 Question 1 : Présentation de la loi Burr de type XII

2.1 Caractéristiques de la loi

C'est une loi à trois paramètres α, θ, γ .

– fonction de densité :

$$f(x) = \frac{\alpha \gamma (x/\theta)^\gamma}{x [1 + (x/\theta)^\gamma]^{\alpha+1}}.$$

– fonction de répartition :

$$F(x) = 1 - u^\alpha, \text{ avec } u = \frac{1}{1 + (x/\theta)^\gamma}.$$

– moments d'ordre k :

$$E[X^k] = \frac{\theta^k \Gamma(1+k/\gamma) \Gamma(\alpha-k/\gamma)}{\Gamma(\alpha)}, \quad -\gamma < k < \alpha\gamma.$$

– $E[(X \wedge x)^k] = \frac{\theta^k \Gamma(1+k/\gamma) \Gamma(\alpha-k/\gamma)}{\Gamma(\alpha)} \beta(1+k/\gamma, \alpha-k/\gamma, 1-u) + x^k u^\alpha, k > -\gamma.$

– $mode = \theta (\frac{\gamma-1}{\alpha\gamma+1})^{1/\gamma}$, si $\gamma > 1$, 0 sinon.

2.2 Propriétés

Il n'y a rien d'inhérent dans cette distribution qui rende cette loi meilleure qu'une autre en assurance. Mais elle a de bonnes propriétés :

1. Les fonctions de distribution cumulée et de répartition de masse ont toutes les deux des formes proches.
2. La loi Pareto est un cas particulier et est intéressante comme distribution en assurance.
3. La loi Pareto a une fonction de répartition de masse strictement décroissante, ce qui en fait un bon modèle pour les larges pertes, sauf lorsque l'on s'approche trop de zéro. La loi Burr peut avoir un mode non-nul.

La loi simulée est tracée sous Excel.

Ci-dessous trois représentations de simulation de la loi Burr type XII avec des paramétrisations différentes.

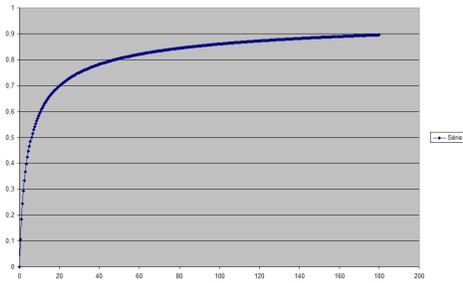


FIG. 1 – Graphe de la loi Burr type XII

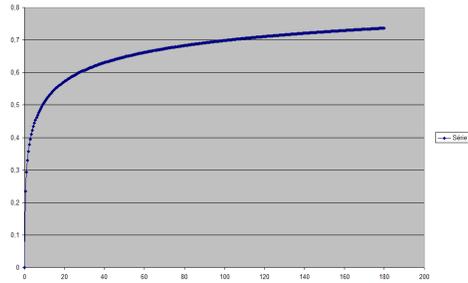


FIG. 2 – Graphe de la loi Burr type XII

3 Question 2

3.1 Analyse qq-plot, fonction d'excès moyen empirique et fonction de répartition empirique

On commence par faire une analyse graphique des données pour savoir quelle loi traduirait le mieux les données que nous avons. Pour cela nous traçons les qq-plots des lois exponentielle, gamma, lognormale, pareto et Burr de type XII, ainsi que les fonctions d'excès moyen et de répartition empirique.

Aucun des qq-plots ne donne une droite parfaite, par contre on s'en approche plus ou moins suivant les lois qu'on a choisi. Pour faire les qq-plots de Gamma, Pareto et Burr on avait besoin de valeurs, ces valeurs ont

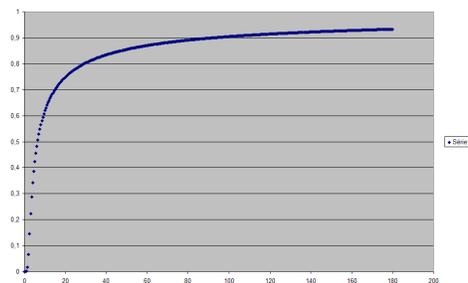


FIG. 3 – Graphe de la loi Burr type XII

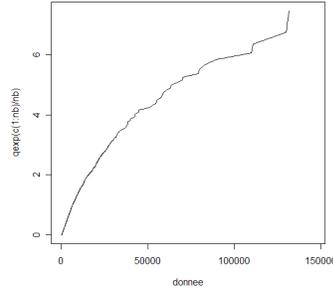


FIG. 4 – Graphiques du qq-plot de la loi exponentielle

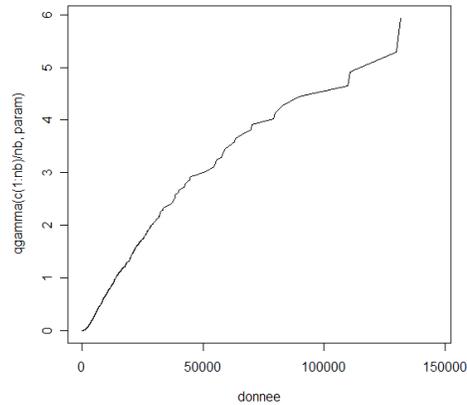


FIG. 5 – Graphiques du qq-plot de la loi Gamma

été choisi arbitrairement de façon à ce que la courbe s’approche le plus possible au graphe d’une droite. Après avoir tracer les qq-plots nous tracons le graphe de la fonction d’excès moyen puis le graphe de la fonction de répartition empirique.

A première vue ce graphe nous choque. Cependant lorsqu’on regarde les données d’n peu plus près cela peu s’expliquer par le fait que nous avons beaucoup de données de faibles valeurs et peu de valeurs élevées.

3.2 Loi exponentielle

On suppose que les données suivent une loi exponentielle dont nous allons estimer son paramètre de deux façons différentes, par la méthode des moments puis par la méthode du maximum de vraisemblance. Puis nous allons vérifier que ce modèle correspond effectivement à nos données pour cela nous feront le test Kolmogorov-Smirnov avec $\alpha = 5\%$.

3.2.1 Estimation des paramètres par la méthode des moments

D’une part l’espérance d’une loi exponentielle de paramètre (β) vaut $\frac{1}{\beta}$ et d’autre part l’espérance empirique vaut $\frac{1}{n} \sum x_i$. On obtient alors $\tilde{\beta}_{MM} = \frac{n}{\sum x_i}$

Application numérique : $\tilde{\beta}_{MM} = 0.0001227$

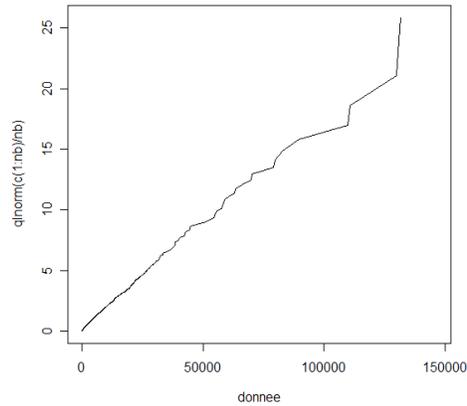


FIG. 6 – Graphiques du qq-plot de la loi Lognormale

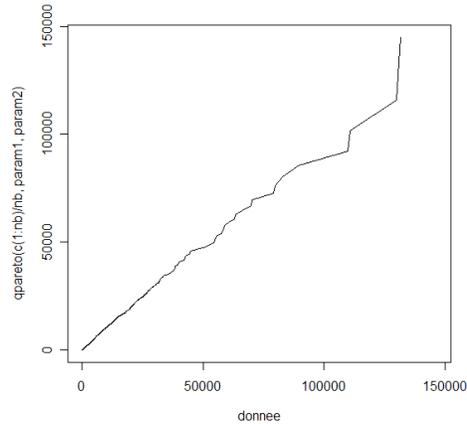


FIG. 7 – Graphiques du qq-plot de la loi Pareto

3.2.2 Estimation des paramètres par la méthode du maximum de vraisemblance

Pour estimer le paramètre β par la méthode du maximum de vraisemblance on maximise la fonction $L(\beta) = \prod \beta e^{-\beta x_i}$. On obtient le même résultat que précédemment.

On compare graphiquement la fonction de répartition de la loi exponentielle de paramètre (0.0001227) avec la fonction de répartition empirique. On remarque que la fonction de répartition de la loi exponentielle ne décrit pas bien les données. On devrait rejeter cette loi.

3.2.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du maximum de vraisemblance

La fonction de l'excès de perte est définie par $f(d) = E[X - d | X > d] = E[X | X > d] - d$.

3.2.4 Adéquation de la loi aux données

On fait un test de Kolmogorov-Smirnov pour savoir si la loi exponentielle de paramètre (0.0001227) décrit bien les données. On a $Dn = 0.09272$ et $c_{alpha} = 0.03261$

Dans notre cas présent l'hypothèse est rejetée. Ce test est en accord avec le graphique de comparaison

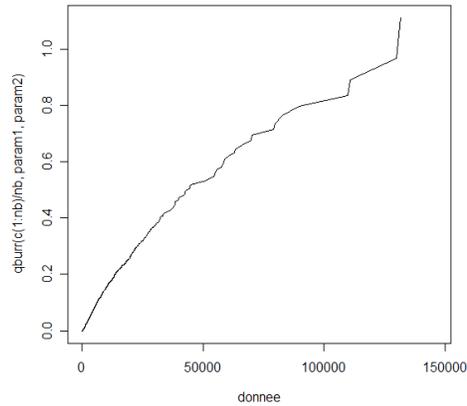


FIG. 8 – Graphiques du qq-plot de la loi Burr

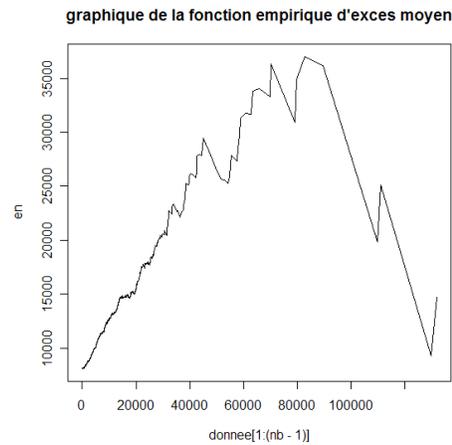


FIG. 9 – Graphique de la fonction d'excès moyen empirique

3.3 Loi Gamma

Nous suppose maintenant que les données suivent une loi gamma.

3.3.1 Estimation des paramètres par la méthode des moments

La loi gamma possèdent deux paramètres par conséquent pour pouvoir les déterminer on utilise l'espérance et la variance empiriques, sachant que l'espérance et la variance d'une loi gamma de paramètres (α, β) sont respectivement égaux à $\frac{\alpha}{\beta}$ et $\frac{\alpha}{\beta^2}$. On obtient donc $\tilde{\alpha} = \frac{(\frac{1}{n} \sum x_i)^2}{\frac{1}{n} \sum (x_i)^2}$ et $\tilde{\beta} = \frac{\frac{1}{n} \sum x_i}{\frac{1}{n} \sum (x_i)^2}$

Application numérique : $\tilde{\alpha}_{MM} = 0.4691$ et $\tilde{\beta}_{MM} = 5.7553$

3.3.2 Estimation des paramètres par la méthode du maximum de vraisemblance

Comme pour la loi exponentielle on maximise la fonction de maximum de vraisemblance qui s'écrit pour la loi Gamma de paramètres (α, β) : $\prod (\frac{\beta^\alpha}{\Gamma(\alpha)} x_i^{\alpha-1} e^{-\beta x_i})$.

Application numérique : $\hat{\alpha}_{MV} = 0.8455$ et $\hat{\beta}_{MV} = 0.0001037$

On compare graphiquement la fonction de répartition de la loi gamma de paramètre $(0.8455, 0.0001037)$ avec la fonction de répartition empirique.

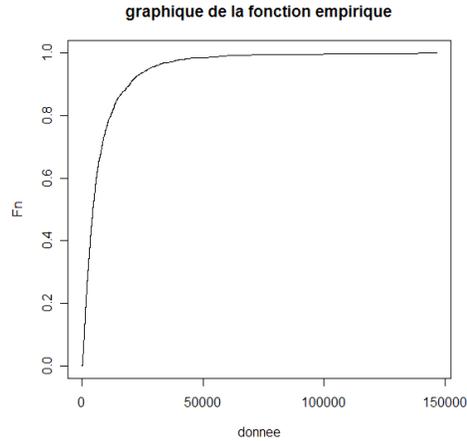


FIG. 10 – Graphique de la fonction de répartition empirique

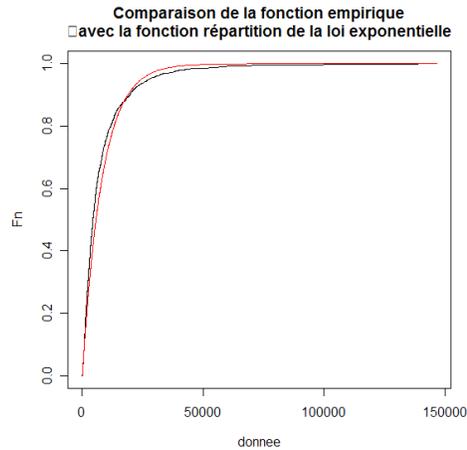


FIG. 11 – Graphe comparatif entre F_n et $F_{exponentielle}$

3.3.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode des moments

3.3.4 Adéquation de la loi aux données

Toujours par le test de Kolmogorov-Smirnov pour $\alpha = 5\%$, le test est à nouveau rejeté. En effet on a $Dn = 0.0728539131712503$ et $c_{alpha} = 0.0326128773643594$. La loi gamma(0.8455, 0.0001037) ne représente pas bien les données que nous avons.

3.4 Loi Log-normale

Nous faisons maintenant l'hypothèse que nos données suivent une loi Log normale.

3.4.1 Estimation des paramètres par la méthode des moments

Estimons les paramètres (μ, σ^2) de la loi log-normale, qui décrit nos données, par la méthode des moments. Nous avons à nouveau deux paramètres à estimer, nous nous servons alors de l'espérance et de la variance empiriques. $E[X] = e^{\mu + \frac{\sigma^2}{2}}$ et $Var[X] = e^{2\mu + \sigma^2}(e^{\sigma^2} - 1)$. Après calcul on obtient $\mu = \ln(\frac{1}{n} \sum x_i) - \frac{1}{2} \ln(\frac{\frac{1}{n} \sum (x_i)^2}{(\frac{1}{n} \sum x_i)^2} + 1)$ et $\sigma = \sqrt{\ln(\frac{\frac{1}{n} \sum (x_i)^2}{(\frac{1}{n} \sum x_i)^2} + 1)}$

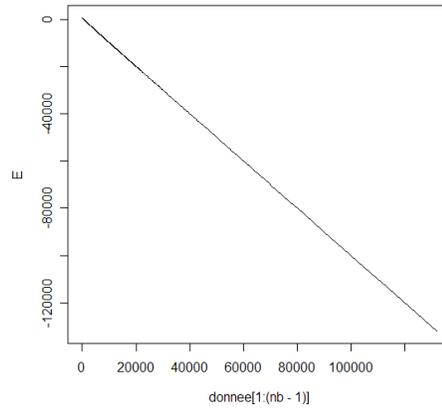


FIG. 12 – Graphique de la fonction excess of loss pour la loi exponentielle

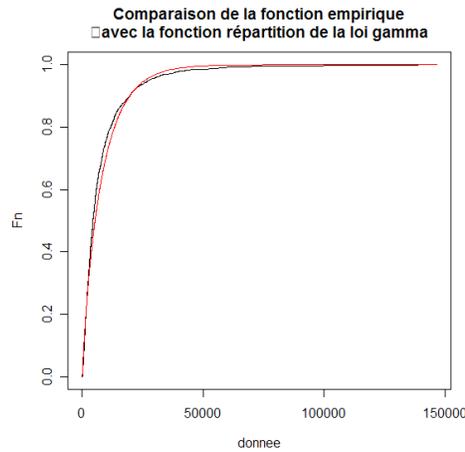


FIG. 13 – Graphe comparatif entre F_n et F_{gamma}

Application numérique : $\tilde{\mu}_{MM} = 8.4350$ et $\tilde{\sigma}_{MM} = 1.0685$

3.4.2 Estimation des paramètres par la méthode du maximum de vraisemblance

La fonction de maximum de vraisemblance pour la loi log-normale (μ, σ^2) s'écrit : $\prod \frac{1}{x_i \sqrt{2\pi}\sigma} e^{-\frac{(\ln(x_i) - \mu)^2}{2\sigma^2}}$

Application numérique : $\hat{\mu}_{MV} = 8.309$ et $\hat{\sigma}_{MV} = 0.8098$

L'estimation par cette méthode donne un résultat similaire à la méthode des moments.

3.4.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode des moments

3.4.4 Adéquation de la loi aux données

Toujours par le test de Kolmogorov-Smirnov pour $\alpha = 5\%$, l'hypothèse est à nouveau rejeté. En effet $D_n = 0.1069$ et $c_{alpha} = 0.03261$. La loi log-normale(8.309,0.8098) ne représente pas bien les données que nous avons.

3.5 Loi Pareto

Nous supposons cette fois que les données suivent une loi Pareto.

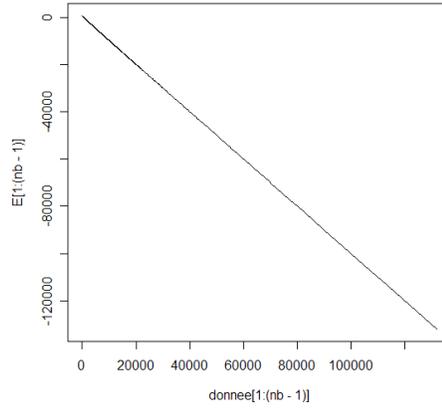


FIG. 14 – Graphe de la fonction excess of loss à partir de la fonction gamma

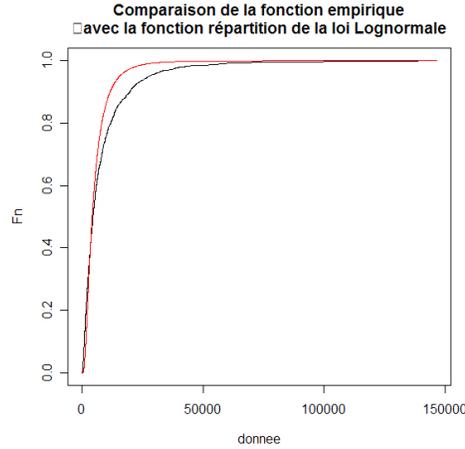


FIG. 15 – Graphe comparatif entre F_n et $F_{lognormale}$

3.5.1 Estimation des paramètres par la méthode des moments

La loi Pareto possède elle aussi deux paramètres donc comme pour les lois log-normale et gamma on utilise l'espérance et la variance empiriques. $E[X] = \frac{\lambda}{\alpha-1}$ et $Var[X] = \frac{\alpha\lambda^2}{(\alpha-1)^2(\alpha-2)}$

Pour simplifier les écritures on appellera on notera \bar{X}_n l'espérance empirique et S_n la variance empirique. Après calcul on obtient $\alpha = \frac{2S_n}{S_n - \bar{X}_n^2}$ et $\lambda = \bar{X}_n \left(\frac{2S_n}{S_n - \bar{X}_n^2} - 1 \right)$.

Application numérique : $\tilde{\alpha}_{MM} = 3.7669$ et $\tilde{\lambda}_{MM} = 22550.8$

3.5.2 Estimation des paramètres par la méthode du maximum de vraisemblance

La fonction de maximum de vraisemblance s'écrit $\prod \frac{\alpha\lambda^\alpha}{(\lambda+x_i)^{\alpha+1}}$. Après résolution numérique, on obtient :

Application numérique : $\hat{\alpha}_{MV} = 3.6505$ et $\hat{\lambda}_{MV} = 21529.6$

Les valeurs obtenue par la méthode du maximum de vraisemblance sont très proches des valeurs obtenues par la méthode des moments.

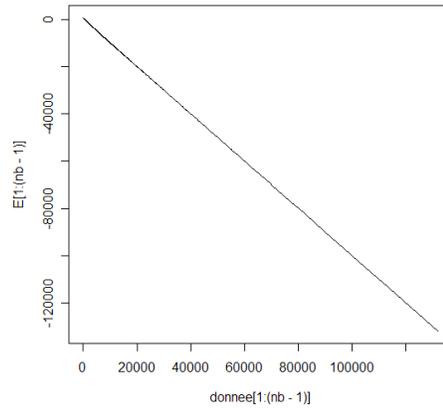


FIG. 16 – Graphe de la fonction excess of loss a partir de la loi lognormale

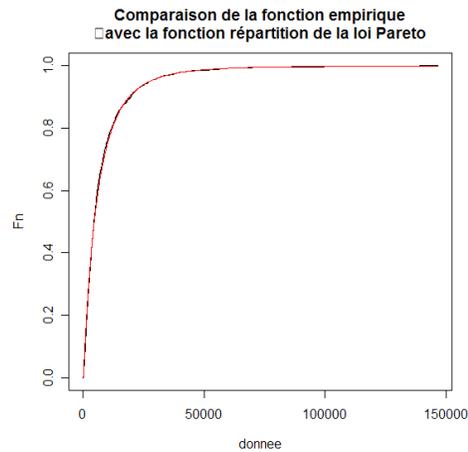


FIG. 17 – Graphe comparatif entre F_n et F_{pareto}

3.5.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode des moments

3.5.4 Adéquation de la loi aux données

Avec le test de Kolmogorov-Smirnov on obtient les valeurs suivantes $Dn = 0.02956$ et $c_{alpha} = 0.03261$ pour $\alpha = 5\%$. On ne rejette donc pas le modele.

3.6 Loi Burr

On suppose maintenant que les données suivent une loi de Burr XII.

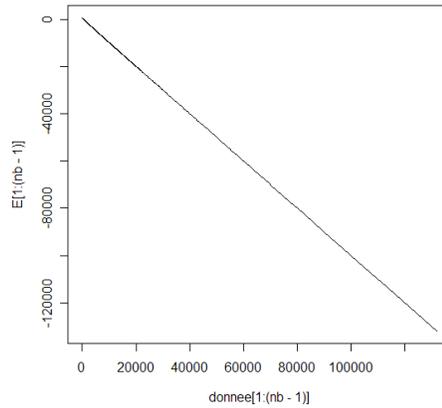


FIG. 18 – Graphe de la fonction excess of loss a partir de la loi Pareto

3.6.1 Estimation des paramètres par la méthode des moments

Pour estimer les paramètres de Burr par la méthode des moments on a besoin des trois premiers moments car il y a trois paramètres à estimer.

$$\begin{aligned}
 E[X] &= \frac{s \Gamma(a - \frac{1}{b}) \Gamma(1 + \frac{1}{b})}{\Gamma(a)} = \frac{1}{n} \sum x_i \\
 E[X^2] &= \frac{s^2 \Gamma(a - \frac{2}{b}) \Gamma(1 + \frac{2}{b})}{\Gamma(a)} = \frac{1}{n} \sum x_i^2 \\
 E[X^3] &= \frac{s^3 \Gamma(a - \frac{3}{b}) \Gamma(1 + \frac{3}{b})}{\Gamma(a)} = \frac{1}{n} \sum x_i^3
 \end{aligned}$$

3.6.2 Estimation des paramètres par la méthode du maximum de vraisemblance

Ne sachant pas résoudre les équations précédentes on fait trouver les paramètres de la loi Burr par le maximum de vraisemblance, en prenant des valeurs initiales arbitraires de manière à ce que la loi de Burr "fitte" bien avec la fonction de répartition.

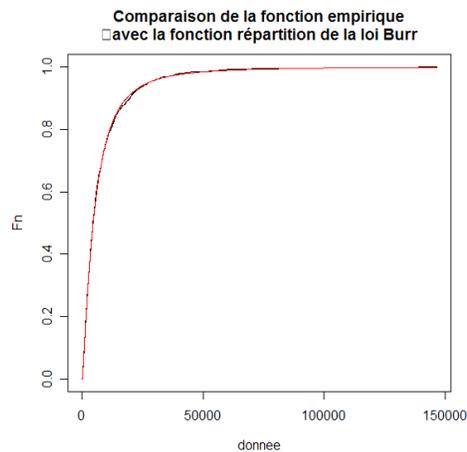


FIG. 19 – Graphe comparatif entre F_n et F_{burr}

3.6.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode des moments

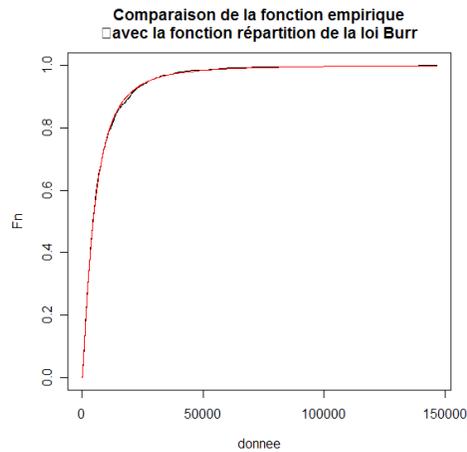


FIG. 20 – Graphe comparatif entre F_n et F_{burr}

3.6.4 Adéquation de la loi aux données

Avec le test de Kolmogorov-Smirnov on obtient les valeurs suivantes $D_n = 0.01389$ et $c_{alpha} = 0.03261$ pour $\alpha = 5\%$. On ne rejette donc pas le modèle.

3.7 Sélection de la loi

On sélectionne la loi en comparant les valeurs du test Bayésien Schwartz de $(\text{nombredeparametres}) * \ln(\text{nombrededonnees}) - 2 * \ln(\text{fonctiondevraisemblance})$. La loi sélectionnée est la loi qui a la plus petite valeur.

lois	exponentielle	gamma	lognormale	Pareto	Burr type XII
coeff bayésien	34807.59	34780.96	36057.80	34645.64	34632.55

La loi sélectionnée est la loi Burr ce qui correspond avec tout ce qu'on a vu jusqu'à présent. En effet la loi Burr a le plus petit D_n dans le test de Kolmogorov-Smirnov.

lois	exponentielle	gamma	lognormale	Pareto	Burr type XII
D_n	0.09272	0.0729	0.1069	0.02956	0.01389

4 Question 3

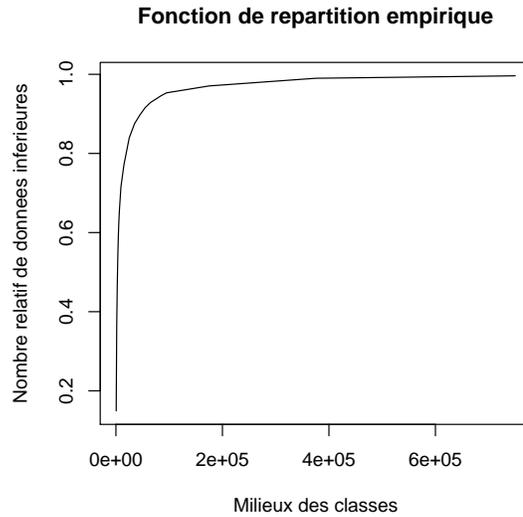
Il s'agit dans cette question de déterminer quelle loi est la mieux adaptée aux données groupées qui nous sont fournies. Nous allons pour cela procéder à l'étude de cinq lois : les lois Exponentielle, Gamma, Pareto, Lognormale et Burr XII.

4.1 Fonction d'excès moyen empirique et fonction de répartition empirique

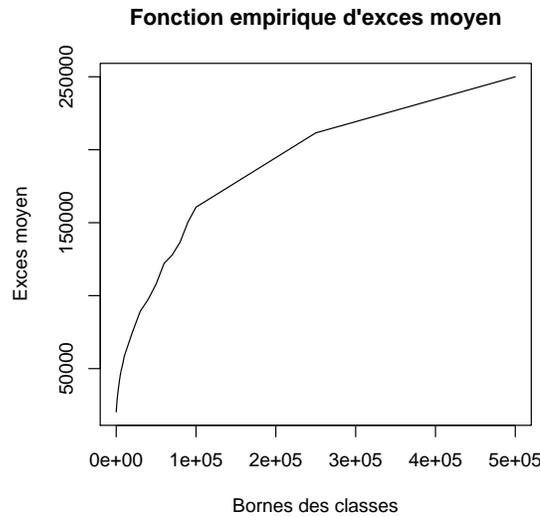
Définissons tout d'abord ces notions :

- La fonction de répartition empirique est définie par $F_n(x) = \frac{1}{n} \sum_{i=1}^n 1_{X_i \leq x}$ pour un échantillon de taille n .

Le code qui nous a servi pour obtenir le graphe de la FdR empirique suivant se trouve en annexes :



- La fonction empirique d'excès moyen suppose que les observations sont ordonnées de façon croissante, elle est définie par $\hat{e}_n(X) = \frac{\sum_{i=1}^n x_i I_{(X, \infty)}(x_i)}{\text{nbr de } x_i > X} - X$, ainsi on trace la courbe de $\hat{e}_n(X)$ pour $X \in (x_1, \dots, x_n)$ avec $\hat{e}_n(x_k) = \frac{\sum_{i=k+1}^n x_i}{n-k} - x_k$ ($k = 1, 2, \dots, n-1$).



4.2 Loi exponentielle

La loi exponentielle de paramètre β notée $Exp(\beta)$ ($\beta > 0$) a pour fonction de répartition :

$$F(x) = 1 - e^{-\beta x}, x > 0.$$

4.2.1 Analyse qq-plot

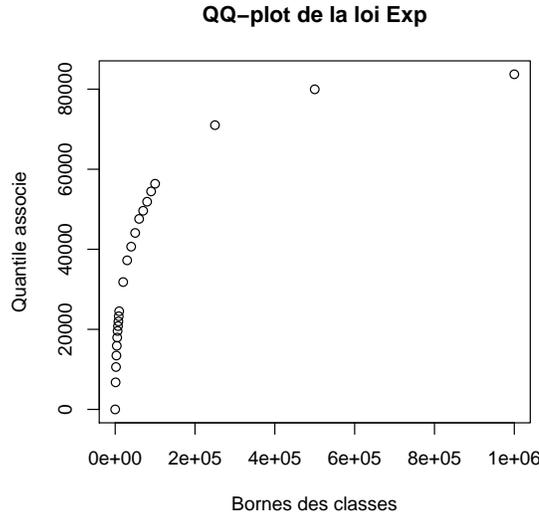
L'analyse *qq-plot* utilise la fonction quantile définie par la fonction inverse généralisée d'une fonction de répartition F elle-même définie par :

$$F^{-}(\alpha) = \inf(x \in \mathbb{R} : F(x) \geq \alpha) \text{ pour } 0 < \alpha < 1$$

Nous avons donc que $x_\alpha = F^{-}(\alpha)$ est le α -quantile de F.

Idée du qq-plot : Supposons que (x_1, \dots, x_n) sont des réalisations iid provenant d'une v.a. de FdR continue. Nous tracons en pratique les couples $(x_{(k)}, F^{-}(\frac{k}{n+1}))$ pour $k = 1, \dots, n$.

Nous obtenons comme graphique la figure suivante pour la loi exponentielle. Normalement nous sommes



censés obtenir plus ou moins une droite lors de l'analyse *qq-plot* des données, on constate qu'ici les premières données semblent alignées dans le cas de cette loi mais que par la suite ce n'est plus le cas.

4.2.2 Estimation des paramètres par la méthode des moments

L'estimation des paramètres par la méthode des moments consiste à trouver $\hat{\beta}$ tel que $\phi(\beta) = E(X)$, sachant que la moyenne empirique $\bar{X}_n = \frac{1}{n} \sum_i X_i$ converge vers cette espérance d'après la loi des grands nombres.

Note importante :

Etant donné que les ressources du cours et Internet ne nous permettent pas de connaître les observations à considérer dans le cas de données groupées pour cette méthode d'estimation, nous prendrons arbitrairement le milieu des classes pour les X_i . Ainsi $\sum_i X_i = \sum$ milieu de classe \times nb d'observations dans la classe considérée.

Pour la loi $Exp(\beta)$, l'espérance est donnée par $E(X) = \frac{1}{\beta}$. Ainsi, $\phi(\beta) = \frac{1}{\beta} = \bar{X}_n$, donc nous obtenons pour l'estimateur par la méthode des moments (EMM) : $\tilde{\beta}_n = \frac{1}{\bar{X}_n}$.

Le code du calcul de cet estimateur est donné dans les annexes et le résultat de son exécution est :

```
> EMM_exp(x)
L'estimateur de beta par la methode des moments de la loi Exp vaut : 8.223862e-07
```

4.2.3 Estimation des paramètres par la méthode du maximum de vraisemblance

L'estimation par la méthode du maximum de vraisemblance (EMV) consiste à trouver le β qui maximise la fonction de vraisemblance donnée par l'expression $L(\beta; X_1, X_2, \dots, X_n) = f(X_1, \dots, X_n; \beta) = \prod f(X_i, \beta)$ car les données sont iid (f est la densité).

Ce qui donne dans le cas de la loi exponentielle $L(\beta; X_1, X_2, \dots, X_n) = \prod \beta e^{-\beta X_i} = \beta^n e^{-\beta \sum_i X_i}$,

En dérivant le logarithme de cette expression par rapport à β et en l'annulant pour connaître le β optimal, nous avons $\frac{\delta}{\delta\beta} \ln(L) = \frac{\delta}{\delta\beta} (n \ln(\beta) - \beta \sum X_i) = \frac{n}{\beta} - \sum X_i = 0$, soit $\hat{\beta} = \frac{n}{\sum X_i} = \frac{1}{\bar{X}_n}$.

Remarque : c'est le même résultat que pour l'estimateur de la méthode des moments.

Dans le cas de données groupées, la fonction de vraisemblance est définie par

$$L(\beta) = \prod_{i=1}^r (F_X(c_i; \beta) - F_X(c_{i-1}; \beta))^{n_i} \text{ avec } c_i \text{ les limites des classes et } r \text{ le nombre de classes.}$$

Après calcul, nous obtenons finalement que l'EMV de β satisfait l'équation :

$$\sum_{i=1}^r n_i \frac{c_{i-1} \exp(-\beta c_i) - c_i \exp(-\beta c_{i-1})}{\exp(-\beta c_{i-1}) - \exp(-\beta c_i)} = 0, \text{ ce qui se calcule numériquement.}$$

Finalement en utilisant la forme explicite de la fonction de log-vraisemblance, nous obtenons (le code est en Annexes) :

```
> EMV_exp(x)
```

```
L'estimateur de beta par la methode du maximum de vraisemblance de la loi Exponentielle vaut  
[1] 5.165625e-05
```

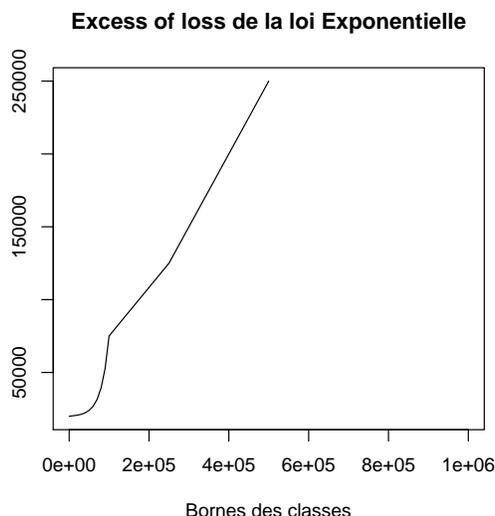
4.2.4 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du max de vraisemblance

La fonction *excess-of-loss* d'une variable aléatoire X est donnée par l'expression $E[W|X > d]$ avec $W = X - d$ et d un seuil fixé.

Le problème dans le cas de données groupées est de choisir les valeurs de X que nous considérons ainsi que les seuils d . En effet, nous ne disposons pas des valeurs de X mais bien de classes qui contiennent ces valeurs. Nous prendrons donc pour les X_i les milieux des classes.

La première idée qui nous vient à l'esprit afin d'évaluer cette espérance serait de prendre la différence des fonctions de répartition à la place de la densité dans la définition de l'espérance (de la même manière que dans l'estimation par le maximum de vraisemblance) et de faire varier le seuil d à chaque borne des classes. Ainsi la fonction *excess-of-loss* devient $E[W|X > d] = \frac{1}{(1-F_X(d))} \sum_{i=d+1}^n x_i [F_X(c_i) - F_X(c_{i-1})] - d$ avec x_i les milieux de classe. Nous adopterons cette méthode dans la suite du rapport pour le tracé de cette fonction. Nous obtenons comme graphique avec la commande :

```
> EoL_exp(x)
```



4.2.5 Adéquation de la loi aux données

Le test d'adéquation de la loi aux données pour des données groupées est le test du Khi-deux.

On essaie alors de montrer que le modèle choisi colle bien aux données (H_0). Pour cela, nous utilisons :

– Le nombre espéré d'observations E_i tel que $E_i = n(F_X(c_i; \theta) - F_X(c_{i-1}; \theta)) = n * P(c_{i-1} < X \leq c_i)$

Par convention, habituellement nous avons $E_i \geq 5$.

– La statistique du test : $Q = \sum_{i=1}^r \frac{(n_i - E_i)^2}{E_i}$

– Soit la variable aléatoire $Z \sim \chi^2$ dont le nombre de degré de liberté est

d.f. = nb de classes – nbparam – 1 = $r - 1 - 1 = r - 2$ (loi Exp). Ici $r = 23$ dans notre exemple.

On définit la valeur critique comme $Z_{\alpha, df}$ telle que $P(Z > Z_{\alpha, df}) = \alpha$ avec $\alpha = 5\%$ (seuil du test).

Finalement on rejette (H_0) si $Q > Z_{\alpha, df}$.

Une autre interprétation que nous utiliserons est que si $p_{value} < \alpha$ avec $p_{value} = P(Z > Q) = 1 - F_Z(Q)$ alors nous rejetons le modèle (H_0). Nous obtenons comme résultat en utilisant les estimateurs par la méthode du max de vraisemblance :

```
> Adeq_exp(x)
```

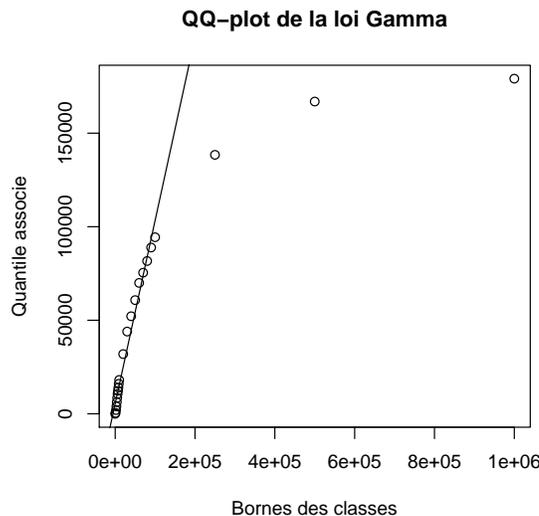
```
on rejette Ho, modele Exponentielle pas bien adapte, p-valeur trop faible= 0
```

4.3 Loi Gamma

La loi Gamma notée $Gamma(\alpha, \beta)$ n'admet pas de fonction de répartition explicite.

4.3.1 Analyse qq-plot

L'analyse *qq-plot* nous conduit au graphe suivant pour la loi Gamma :



4.3.2 Estimation des paramètres par la méthode des moments

Le principe de la méthode des moments est le même que dans la section précédente.

Ainsi, sachant que l'estimation des deux premiers moments de cette loi se fait grâce à la moyenne empirique de l'échantillon pour l'espérance et la variance empirique pour la variance, nous obtenons le système :

$$\begin{cases} E(X) = \frac{\alpha}{\beta}, (1) \\ Var(X) = \frac{\alpha}{\beta^2}, (2) \end{cases}$$

Soit,

$$\begin{cases} \bar{X}_n = \frac{\tilde{\alpha}_n}{\tilde{\beta}_n}, \\ \bar{S}_n^2 = \frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2 = \frac{\tilde{\alpha}_n}{\tilde{\beta}_n^2} = \frac{\bar{X}_n}{\tilde{\beta}_n}, \end{cases}$$

Nous obtenons ainsi pour estimateurs :

$$\begin{cases} \tilde{\alpha}_n = \frac{\bar{X}_n^2}{\frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2} \\ \tilde{\beta}_n = \frac{\bar{X}_n}{\frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2} \end{cases}$$

Il s'agit donc d'expressions déterministes calculables directement sous R à partir des données. Nous obtenons comme résultat les valeurs suivantes :

```
> EMM_gamma(x)
```

```
L'estimateur de alpha par la methode des moments de la loi Gamma vaut : 0.002476913
```

```
L'estimateur de beta par la methode des moments de la loi Gamma vaut : 2.036979e-09
```

4.3.3 Estimation des paramètres par la méthode du maximum de vraisemblance

Rappel : la méthode du maximum de vraisemblance pour des données groupées utilise la fonction de vraisemblance qui est donnée par l'expression :

$$L(\alpha, \beta) = \prod_{i=1}^r (F_X(c_i; \alpha, \beta) - F_X(c_{i-1}; \alpha, \beta))^{n_i}$$

Le problème avec cette loi est qu'elle n'admet pas de fonction de répartition déterministe. Cette approche semble donc devoir être écartée du point de vue de l'expression théorique.

Cependant, il est possible d'estimer grâce à R les paramètres en se servant directement des commandes qui calculent les fonctions de répartition (*pgamma* ici), c'est la technique que nous utiliserons le plus souvent (car pas de forme explicite). Nous obtenons :

```
> EMV_gamma(x)
```

```
Les estimateurs de alpha et beta par la methode du maximum de vraisemblance de la loi Gamma valent :
```

```
[1] 2.611672e-01 1.259555e-05
```

4.3.4 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du max de vraisemblance

En suivant la même démarche que dans le cas de loi exponentielle, nous obtenons :

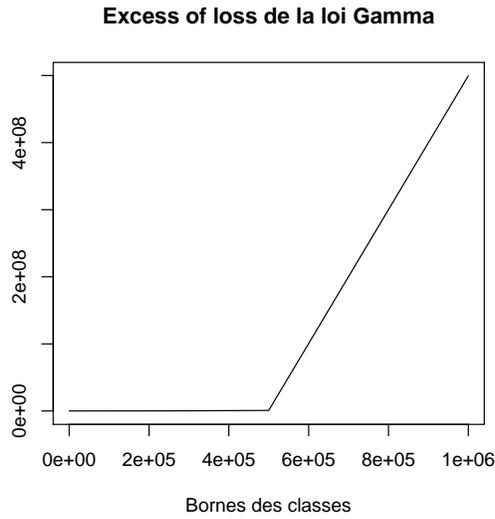
```
> EoL_gamma(x)
```

4.3.5 Adéquation de la loi aux données

Avec le même raisonnement que précédemment, nous obtenons comme résultat :

```
> Adeq_gamma(x)
```

```
on rejette Ho, modele Gamma pas bien adapte, p-valeur trop faible= 0
```



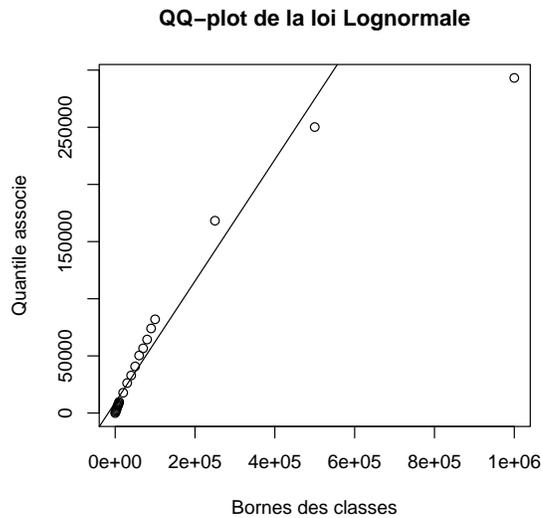
4.4 Loi Log-normale

L'expression de la fonction de répartition de la loi Log-normale notée $LN(\mu, \sigma^2)$ est exprimée en fonction de celle de la loi normale $N(\mu, \sigma^2)$ comme suit : $F_X(x) = \Phi_X\left(\frac{\ln(x)-\mu}{\sigma}\right)$, $x > 0$, avec $\Phi_X(x)$ la fonction de répartition de la loi normale $N(\mu, \sigma^2)$.

Autrement dit, nous recentrons et normalisons le logarithme de la la loi normale classique.

4.4.1 Analyse qq-plot

L'analyse *qq-plot* nous conduit au graphe suivant pour la loi Lognormale :



4.4.2 Estimation des paramètres par la méthode des moments

Par le même raisonnement que la section précédente et en nous servant de l'expression des deux premiers moments de la loi log-normale, nous obtenons le système suivant :

$$\begin{cases} E(X) = e^{\mu + \frac{\sigma^2}{2}} \\ Var(X) = e^{2\mu + \sigma^2} (e^{\sigma^2} - 1) \end{cases}$$

Donc,

$$\begin{cases} e^{\tilde{\mu}_n + \frac{\tilde{\sigma}_n^2}{2}} = \bar{X}_n, (1) \\ \bar{X}_n^2 (e^{\tilde{\sigma}_n^2} - 1) = \frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2, (2) \end{cases}$$

Finalement,

$$\begin{cases} \tilde{\mu}_n = \ln(\bar{X}_n) - \frac{\tilde{\sigma}_n^2}{2} \\ \tilde{\sigma}_n^2 = \ln\left(\frac{\frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2}{\bar{X}_n^2} + 1\right) \end{cases}$$

Ces estimateurs sont calculés directement sous R à partir des données. Les résultats sont :

```
> EMM_lognormale(x)
```

L'estimateur de mu par la methode des moments de la loi log-normale vaut : 11.00945

L'estimateur de sigma carre par la methode des moments de la loi log-normale vaut : 6.003216

4.4.3 Estimation des paramètres par la méthode du maximum de vraisemblance

L'expression de la fonction de vraisemblance est la suivante :

$$L(\mu, \sigma^2) = \prod_{i=1}^r (F_X(c_i; \mu, \sigma^2) - F_X(c_{i-1}; \mu, \sigma^2))^{n_i} = \prod_{i=1}^r \left(\Phi_X\left(\frac{\ln(c_i) - \mu}{\sigma}\right) - \Phi_X\left(\frac{\ln(c_{i-1}) - \mu}{\sigma}\right) \right)^{n_i}$$

Nous obtenons en développant cette expression qu'il s'agit en fait du produit des probabilités suivantes :

$$L(\mu, \sigma^2) = \prod_{i=1}^r \left[P\left(\frac{\ln(c_{i-1}) - \mu}{\sigma} < X \leq \frac{\ln(c_i) - \mu}{\sigma}\right) \right]^{n_i}$$

Par suite et en prenant le logarithme nous obtenons $\ln(L(\mu, \sigma^2)) = \sum_{i=1}^r n_i \ln\left(\Phi_X\left(\frac{\ln(c_i) - \mu}{\sigma}\right) - \Phi_X\left(\frac{\ln(c_{i-1}) - \mu}{\sigma}\right)\right)$, Soit en dérivant par rapport à chacune des variables le système suivant :

$$\begin{cases} \frac{\delta \ln(L(\mu, \sigma^2))}{\delta \mu} = \sum_{i=1}^r \frac{n_i}{\sigma} \left[\frac{\Phi'_X\left(\frac{\ln(c_{i-1}) - \mu}{\sigma}\right) - \Phi'_X\left(\frac{\ln(c_i) - \mu}{\sigma}\right)}{\Phi_X\left(\frac{\ln(c_i) - \mu}{\sigma}\right) - \Phi_X\left(\frac{\ln(c_{i-1}) - \mu}{\sigma}\right)} \right], (1) \\ \frac{\delta \ln(L(\mu, \sigma^2))}{\delta \sigma^2} = \dots, (2) \end{cases}$$

D'un point de vue théorique, nous réalisons que les dérivées partielles sont des fonctions compliquées. C'est pourquoi nous utiliserons l'expression de la définition de la vraisemblance pour données groupées lors de l'implémentation en maximisant la log-vraisemblance plutôt que de se servir d'une expression déterministe de l'estimateur (difficilement trouvable). Les résultats sont :

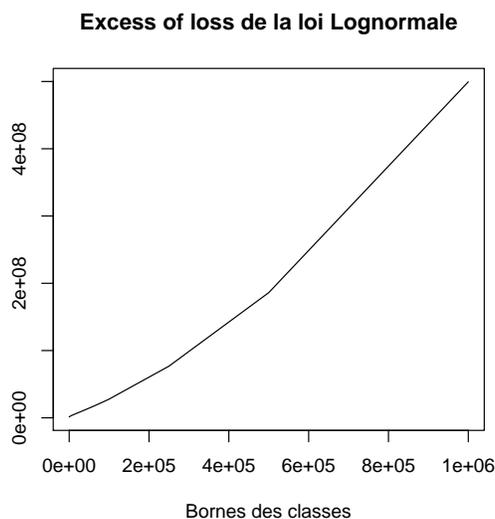
```
> EMV_lognormale(x)
```

Les estimateurs de mu et sigma par la methode du maximum de vraisemblance de la loi Lognormale valent
[1] 7.993165 2.070612

4.4.4 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du max de vraisemblance

En suivant toujours la même démarche, nous obtenons :

```
> EoL_lognormale(x)
```



4.4.5 Adéquation de la loi aux données

Avec le même raisonnement que précédemment, nous obtenons comme résultat :

```
> Adeq_lognormale(x)
Modele Lognormale adequat aux donnees donc accepte, p-valeur = 0.7429306
```

4.5 Loi Pareto

La loi de Pareto notée $Pa(\alpha, \lambda)$ admet pour fonction de répartition

$$F(x) = 1 - \left(\frac{\lambda}{\lambda+x} \right)^\alpha$$

Cette formule explicite va nous aider à déterminer les estimateurs par les différentes méthodes.

4.5.1 Analyse qq-plot

L'analyse *qq-plot* nous conduit au graphe suivant pour la loi Pareto :

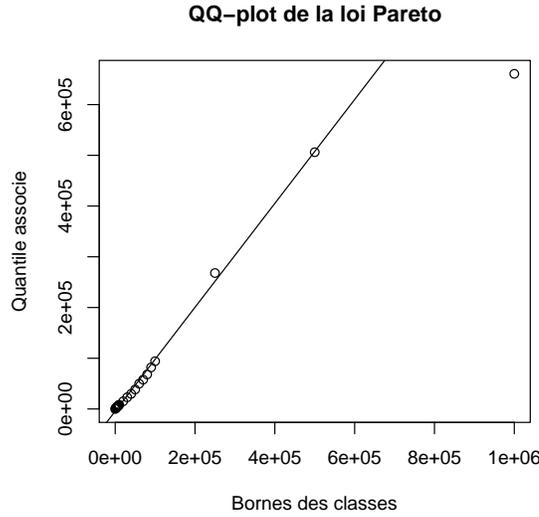
4.5.2 Estimation des paramètres par la méthode des moments

Toujours par la même technique, voici le système d'équations à résoudre (attention toutefois quelques contraintes sur les valeurs seront à respecter pour utiliser ces expressions des moments, à savoir $\alpha > 1$ pour l'espérance et $\alpha > 2$ pour la variance!)

$$\begin{cases} E(X) = \frac{\lambda}{\alpha-1} \\ Var(X) = \frac{\alpha\lambda^2}{(\alpha-1)^2(\alpha-2)} \end{cases}$$

D'où,

$$\begin{cases} \bar{X}_n = \frac{\tilde{\lambda}_n}{\tilde{\alpha}_n - 1} \\ \bar{S}_n^2 = \frac{1}{n} \sum_i X_i^2 - \bar{X}_n^2 = \frac{\tilde{\alpha}_n \bar{X}_n^2}{(\tilde{\alpha}_n - 2)} \end{cases}$$



En développant ce calcul, nous obtenons les estimateurs suivants :

$$\begin{cases} \tilde{\lambda}_n = \bar{X}_n \left(\frac{2}{1 - \frac{\bar{X}_n^2}{\bar{S}_n^2}} - 1 \right) \\ \tilde{\alpha}_n = \frac{2}{1 - \frac{\bar{X}_n^2}{\bar{S}_n^2}} \end{cases}$$

Il s'agit donc d'expressions déterministes calculables à partir des données et nous obtenons comme résultat les valeurs suivantes :

```
> EMM_pareto(x)
```

```
L'estimateur de alpha par la methode des moments de la loi Pareto vaut : 2.004966
```

```
L'estimateur de lambda par la methode des moments de la loi Pareto vaut : 1222012
```

4.5.3 Estimation des paramètres par la méthode du maximum de vraisemblance

Etant toujours dans le cas de données groupées, nous utilisons la même technique que précédemment pour estimer les paramètres de la loi de Pareto $Pa(\alpha, \lambda)$.

Ainsi, $L(\alpha, \lambda) = \prod_{i=1}^r \left((1 - (\frac{\lambda}{\lambda+c_i})^\alpha) - (1 - (\frac{\lambda}{\lambda+c_{i-1}})^\alpha) \right)^{n_i} = \prod_{i=1}^r \left((\frac{\lambda}{\lambda+c_{i-1}})^\alpha - (\frac{\lambda}{\lambda+c_i})^\alpha \right)^{n_i}$,

Donc $\ln(L(\alpha, \lambda)) = \sum_{i=1}^r n_i \ln \left((\frac{\lambda}{\lambda+c_{i-1}})^\alpha - (\frac{\lambda}{\lambda+c_i})^\alpha \right) = \sum_{i=1}^r n_i \ln \left(e^{\alpha \ln(\frac{\lambda}{\lambda+c_{i-1}})} - e^{\alpha \ln(\frac{\lambda}{\lambda+c_i})} \right)$,

En dérivant par rapport à chacune des variables, nous obtenons le système :

$$\begin{cases} \frac{\delta \ln(L(\alpha, \lambda))}{\delta \lambda} = \sum_{i=1}^r \frac{\alpha n_i}{\left(\frac{\lambda}{\lambda+c_{i-1}} \right)^\alpha - \left(\frac{\lambda}{\lambda+c_i} \right)^\alpha} \left[\left(\frac{\lambda}{\lambda+c_{i-1}} \right)^{\alpha-1} \frac{c_{i-1}}{(\lambda+c_{i-1})^2} - \left(\frac{\lambda}{\lambda+c_i} \right)^{\alpha-1} \frac{c_i}{(\lambda+c_i)^2} \right] = 0, (1) \\ \frac{\delta \ln(L(\alpha, \lambda))}{\delta \alpha} = \sum_{i=1}^r n_i \frac{\ln \left(\frac{\lambda}{\lambda+c_{i-1}} \right) \exp \left(\alpha \ln \left(\frac{\lambda}{\lambda+c_{i-1}} \right) \right) - \ln \left(\frac{\lambda}{\lambda+c_i} \right) \exp \left(\alpha \ln \left(\frac{\lambda}{\lambda+c_i} \right) \right)}{\left(\frac{\lambda}{\lambda+c_{i-1}} \right)^\alpha - \left(\frac{\lambda}{\lambda+c_i} \right)^\alpha} = 0, (2) \end{cases}$$

Numériquement il est possible de trouver des solutions à ces équations, les dérivées partielles sont des fonctions compliquées encore une fois. Nous avons calculé ces estimateurs de deux manières différentes : l'une utilise directement la définition de la log-vraisemblance pour données groupées (avec la différence des fonctions de répartition) auquel cas nous avons utilisé *ppareto* du package *actuar* de R ; l'autre utilise la

formule explicite de la log-vraisemblance issue du calcul à la main. Nous obtenons exactement les mêmes résultats :

```
> EMV_pareto(x)
```

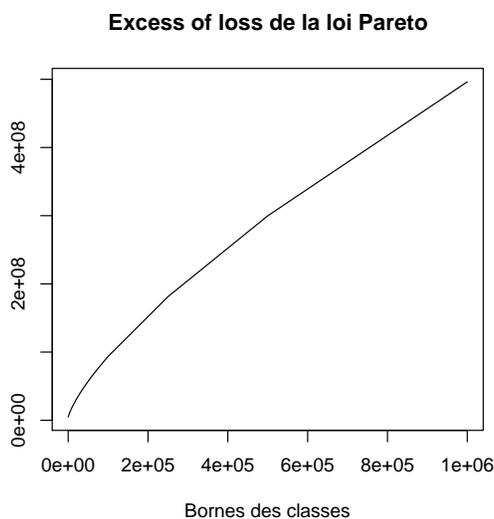
Les estimateurs de alpha et lambda par la methode du maximum de vraisemblance de la loi Pareto valent

```
[1] 0.7311205 1786.4534378
```

4.5.4 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du max de vraisemblance

En suivant toujours la même démarche, nous obtenons :

```
> EoL_pareto(x)
```



4.5.5 Adéquation de la loi aux données

Avec le même raisonnement que précédemment, nous obtenons comme résultat :

```
> Adeq_pareto(x)
```

on rejette Ho, modele Pareto pas bien adapte, p-valeur trop faible= 4.126888e-06

4.6 Loi Burr de type XII

La distribution Burr XII étendue à trois paramètres englobe la distribution de Pareto généralisée, qui est utilisée pour modéliser les dépassements de seuil ; la distribution log-logistique, qui est également utilisée dans l'analyse fréquentielle ; et la distribution de Weibull, qui s'inscrit dans la distribution généralisée des valeurs extrêmes, utilisée pour modéliser les maxima.

La distribution de Burr étendue est flexible pour l'approximation de la distribution des valeurs extrêmes. La distribution de Pareto généralisée et la distribution généralisée des valeurs extrêmes sont quant à elles limitées respectivement aux dépassements de seuil et aux ensembles extrêmes. Du point de vue de la modélisation, la généralisation peut se révéler nécessaire pour obtenir un meilleur ajustement.

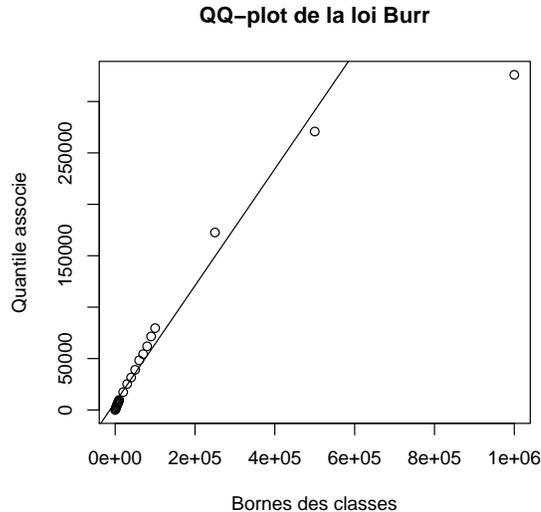
La distribution Burr XII étendue à trois paramètres est intéressante pour cela, en analyse fréquentielle.

Grâce au paquetage *actuar* de R, nous avons la densité de cette loi où a et b sont des paramètres de forme

et s un paramètre d'échelle : $f(x) = \left(\frac{ab\left(\frac{x}{s}\right)^b}{x\left[1 + \left(\frac{x}{s}\right)^b\right]^{a+1}} \right)$

4.6.1 Analyse qq-plot

L'analyse *qq-plot* nous conduit au graphe suivant pour la loi Burr XII :



4.6.2 Estimation des paramètres par la méthode du maximum de vraisemblance

L'estimation des paramètres par la méthode du maximum de vraisemblance s'est faite directement à partir de la définition de celle-ci et grâce à la fonction R *pburr* du package *actuar*. Les valeurs de départ de l'optimisation ont été testées afin de fitter au mieux la distribution empirique (n'ayant pas d'estimateurs par la méthode des moments pour nous servir de point de départ). Nous obtenons :

```
> EMV_burr(x)
```

```
Les estimateurs de alpha, lambda et tau par la methode du maximum de vraisemblance de la loi Burr XII  
[1] 1.6868849148 0.0001208446 0.6761348242
```

4.6.3 Tracé de la fonction *excess-of-loss* avec paramètres de la méthode du max de vraisemblance

En suivant toujours la même démarche, nous obtenons :

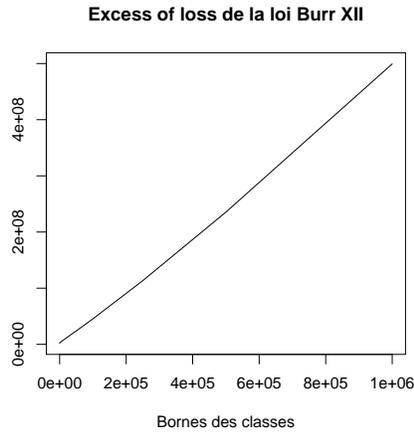
```
> EoL_burr(x)
```

4.6.4 Adéquation de la loi aux données

Avec le même raisonnement que précédemment, nous obtenons comme résultat (avec comme valeur de départ pour l'optimisation 0,5 pour tous les paramètres) :

```
> Adeq_burr(x)
```

```
Modele Burr XII adequat aux donnees donc accepte, p-valeur = 0.1985483
```



4.7 Résumé des résultats

Voici tous les résultats obtenus dans le tableau suivant :

Lois	Exp	Gamma	Lognormale	Pareto	Burr XII
EMM	8.22e-07	(0.0024 ; 2.03e-09)	(11 ; 6)	(2 ; 1222012)	
EMV	5.16e-05	(2.61e-01 ; 1.25e-05)	(7.99 ; 2.07)	(0.73 ; 1786.45)	(1.68 ; 0.0001 ; 0.67)

Ce tableau nous permet d'arriver à plusieurs conclusions qui étaient somme toute prévisibles :

- tout d'abord on se rend bien compte que l'estimation par la méthode des moments est totalement fautive. Cette estimation donne des résultats imprécis, ce qui est dû au fait que les données sont groupées. En effet, ce type d'estimation s'appuie sur les moments de la loi sous-jacente, or nos moments avec nos types de données sont totalement faussés, ce qui explique ces résultats absurdes. Il convient mieux d'utiliser la forme de la distribution comme dans la méthode du maximum de vraisemblance.
- il est intéressant de regarder comment les estimateurs par la méthode du max de vraisemblance fittent la distribution empirique des données. Nous faisons apparaître ces résultats dans les graphiques suivants qui confirment bien les tests d'adéquation, à savoir la loi Lognormale et Burr acceptées. :

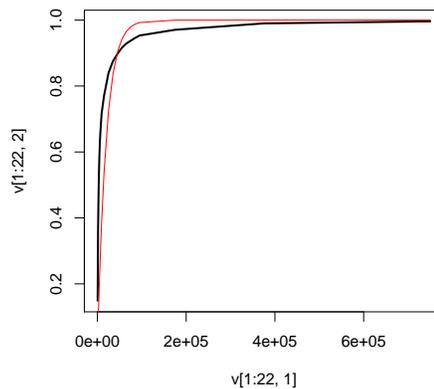


FIG. 21 – Fonctions de répartition empirique et théorique avec estimateurs MV de la loi Exp

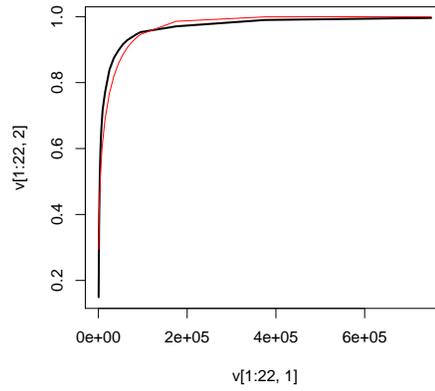


FIG. 22 – Fonctions de répartition empirique et théorique avec estimateurs MV de la loi Gamma

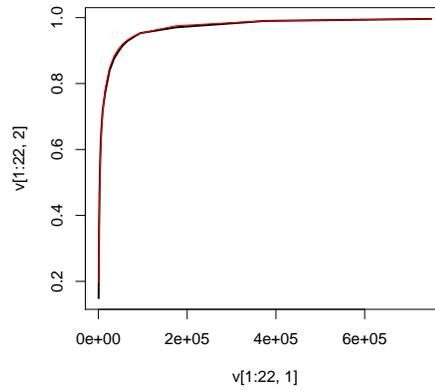


FIG. 23 – Fonctions de répartition empirique et théorique avec estimateurs MV de la loi Lognormale

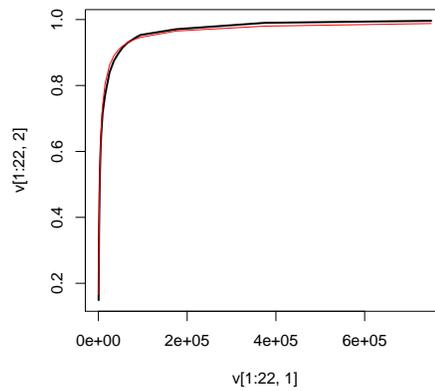


FIG. 24 – Fonctions de répartition empirique et théorique avec estimateurs MV de la loi Pareto

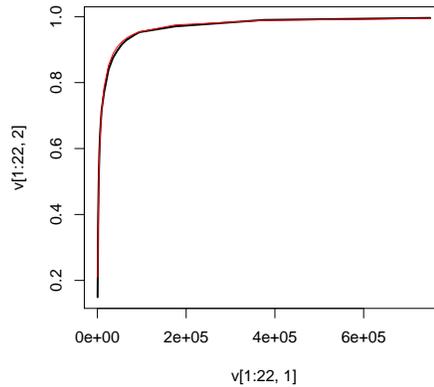


FIG. 25 – Fonctions de répartition empirique et théorique avec estimateurs MV de la loi Burr XII

4.8 Sélection de la loi

Nous avons vu dans les sections précédentes que deux lois semblaient s'accorder à nos données : la loi Lognormale et la loi Burr XII. Il apparait donc pertinent de savoir lequel de ces deux modèles est le mieux adapté.

Pour cela, nous allons procéder à un test formel de ratio de vraisemblance puisque nos deux modèles sont imbriqués, car la loi Lognormale a 2 paramètres (vecteur Θ_0) tandis que la Burr XII en a 3 (vect. Θ_1).

La statistique du test est définie par $R = 2[\ln L(\Theta_1) - \ln L(\Theta_0)]$.

Soit $W \sim \chi_2$ à nombre de degré de liberté ($nbparam(\Theta_1) - nbparam(\Theta_0) = 1$ ici).

Soit la valeur critique $\omega_{\alpha,df}$ telle que $1 - F_W(\omega_{\alpha,df}) = P(W > \omega_{\alpha,df}) = \alpha$ avec α le seuil du test (e.g. 5%).

On rejette le modèle avec Θ_0 si $R > \omega_{\alpha,df}$. Nous obtenons comme résultat de ce test :

```
> selectMod(x)
```

```
On rejette le modele Burr XII
```

Cette conclusion est compatible avec le test informel qui consiste à comparer les p-valeurs des tests d'adéquation aux données et à garder celui où la p-valeur est la plus grande, donc celui de la loi Lognormale.

5 Question 4 : loi de comptage modifiée à zéro

5.1 Cas général

Les lois de comptage (ou fréquence), sont des lois discrètes, elles sont utilisées dans divers domaines pour estimer la fréquence d'un événement. Il y a effet beaucoup de cas pour lesquels on veut estimer le nombre M d'événements survenus durant une période. Par exemple, en biologie, la fréquence d'une séquence d'acides aminés dans un brin d'ADN ; ou en sciences du langage, la fréquence d'un mot ou d'une syllabe dans une phrase ou un texte. Ces lois sont aussi très utilisés en assurance IARD quand l'on doit calculer le nombre de sinistres survenus durant une période. Les lois les plus utilisées sont la loi binomiale, la loi de Poisson et la loi binomiale négative.

Mais ces lois ont un comportement particulier en ce qui concerne les pics. En effet, par exemple pour certaines lois, il y a de faibles probabilités pour la plupart des valeurs, puis un pic beaucoup plus important autour d'une certaine valeur. Il y a des cas où l'on veut modéliser un phénomène dont l'occurrence est moins forte autour de cette valeur, et on obtient donc un lissage de la courbe (par exemple le nombre d'incendies diminue pour un ensemble de contrat si les souscripteurs ont installés des mesures de sécurité plus importantes : détecteurs de fumée...). Pour cela, on modifie la loi de comptage et on définit la loi de comptage

modifiée à zéro par :

- pour $k = 0$, $Pr^M(X = 0) = \alpha$, $0 < \alpha < 1$,
- pour $k \geq 1$, $Pr^M(X = k) = \frac{1-\alpha}{1-Pr(X=0)} Pr(X = k)$.

5.2 Cas particuliers

Pour les trois lois citées précédemment, les lois modifiées sont les suivantes, avec $Pr^M(X = 0) = \alpha$:

- loi binomiale (n, q) : $Pr^M(X = k) = \frac{1-\alpha}{(1-q)^n} C_n^k q^k (1-q)^{n-k}$
- loi de Poisson λ : $Pr^M(X = k) = \frac{1-\alpha}{1-e^{-\lambda}} \frac{\lambda^k e^{-\lambda}}{k!}$
- loi binomiale négative (r, β) : $Pr^M(X = k) = \frac{1-\alpha}{1-(1+\beta)^{-r}} \frac{\Gamma(r+k)}{\Gamma(r)k!} (1+\beta)^{-r} (\frac{\beta}{1+\beta})^k$.

Les graphiques des lois simulées ont été tracés sous Excel.

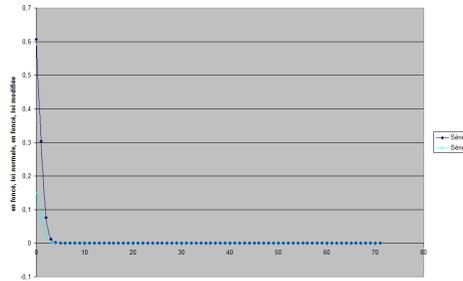


FIG. 26 – Graphe de la loi Poisson

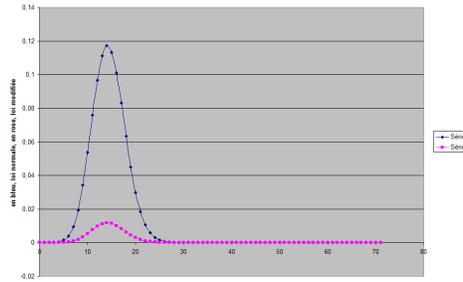


FIG. 27 – Graphe de la loi Binomiale

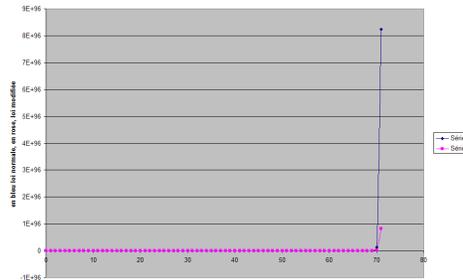


FIG. 28 – Graphe de la loi Binomiale Négative

6 Question 5

On cherche à trouver la loi de M (variable aléatoire correspondant à la fréquence des sinistres). Pour cela on fait comme hypothèse que M suit les lois de Poisson, Binomiale négative, Poisson modifiée à 0 et enfin Binomiale négative modifiée à 0. On estime leurs paramètres par deux méthodes différentes, la méthode des moments et la méthode du maximum de vraisemblance, puis on teste la pertinence de ces lois par rapport à l'échantillon.

On note n_k le nombre de contrats ayant eu k sinistres.

6.1 Loi de Poisson

On suppose que M suit une loi de Poisson de paramètre λ .

6.1.1 Estimation des paramètres par la méthode des moments

La méthode des moments consiste à estimer le paramètre λ grâce à l'espérance empirique de l'échantillon. Lorsqu'une variable aléatoire X suit une loi de poisson de paramètre λ , on a $E[X] = \lambda$. Or ici l'espérance est estimée par $\frac{1}{n} \sum_{i=1}^n k n_k$. En effet $E[X] = \sum k Pr(X = k)$ et on estime les $Pr(X = k)$ par $\frac{n_k}{n}$, avec $n = \sum n_k$. On obtient donc l'estimation de λ :

$$\tilde{\lambda}_{MM} = \frac{1}{\sum_{k=1}^{20} n_k} \sum_{k=1}^{20} k n_k = \frac{\text{Nombre total de sinistres}}{\text{Nombre total de contrats}}$$

Application numérique : $\tilde{\lambda}_{MM} = \frac{6495}{3869} = 1,6787$

6.1.2 Estimation des paramètres par la méthode du maximum de vraisemblance

La méthode du maximum de vraisemblance consiste à estimer le paramètre λ , en maximisant la fonction de vraisemblance qui s'écrit comme suit :

$$\begin{aligned} L(\lambda) &= \prod_{k=0}^{20} Pr(M = k)^{n_k} \\ &= \prod_{k=0}^{20} e^{-\lambda n_k} \frac{\lambda^{k n_k}}{(k!)^{n_k}} \end{aligned}$$

On prend le logarithme de la fonction de vraisemblance, cette opération ne change pas le maximum de la fonction.

$$\begin{aligned} l(\lambda) &= \ln L(\lambda) \\ &= \sum_{k=0}^{20} [-\lambda n_k + k n_k \ln(\lambda) - n_k \ln(k!)] \end{aligned}$$

Le λ qui maximise la fonction de vraisemblance est celui qui annule sa dérivée.

$$\frac{d l(\lambda)}{d \lambda} = \sum_{k=0}^{20} [-n_k + k n_k \frac{1}{\lambda}] = 0$$

d'où

$$\hat{\lambda}_{MV} = \frac{\sum_{k=1}^{20} k n_k}{\sum_{k=1}^{20} n_k} = \frac{\text{Nombre total de sinistres}}{\text{Nombre total de contrats}}$$

On obtient le même résultat qu'avec la méthode des moments.

Application numérique : $\hat{\lambda}_{MV} = 1,6787$

6.1.3 Fonction de masse de probabilité empirique et paramétrique

Graphique en bâtons de la fonction de masses de probabilité de la loi Poisson de paramètre 1,6787.

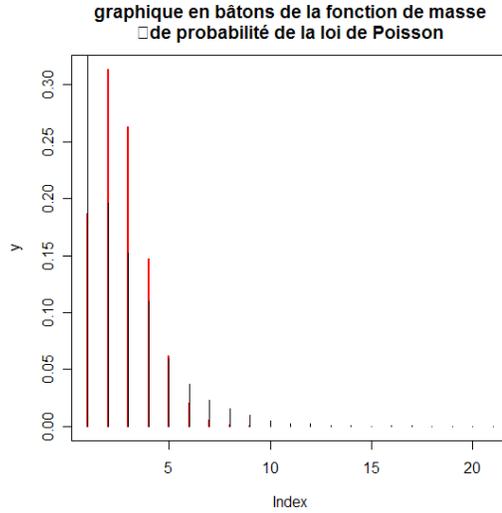


FIG. 29 – Graphe de comparaison de la loi de poisson avec nos données

Comparaison entre la fonction de masse de probabilité de la loi de poisson (en rouge) et la fonction empirique (en noire).

On remarque que la loi de poisson ne correspond pas du tout aux données.

6.1.4 Adéquation de la loi aux données

On utilise le test du khideux pour tester l'adéquation des données à la loi Poisson(1,6787). Dans notre cas $Q = \sum \frac{(n_k - E_k)^2}{E_k} = 10400078$ ce qui est bien plus grand que $z_{5\%,20} = 31,41$ donc on rejette le modèle (H0). La loi de Poisson ne représente pas bien la distribution que l'on a. On retrouve ce que l'on avait vu graphiquement que la loi de poisson n'est pas adaptée aux données.

6.2 Loi Binomiale Négative

On suppose maintenant que M suit une loi Binomiale Négative de paramètres r et β .

6.2.1 Estimation des paramètres par la méthode des moments

Si M est une variable aléatoire suivant une loi Binomiale Négative alors $E[M] = r\beta$ et $Var[M] = r\beta(1 + \beta)$. Après un rapide calcul on obtient $\beta = \frac{Var[M]}{E[M]} - 1$ et $r = \frac{E[M]}{\beta}$. Or on a vu que l'espérance empirique de M vaut $E[M] = \frac{\sum_{k=1}^{20} kn_k}{\sum_{k=1}^{20} n_k} = \frac{\text{Nombre total de sinistres}}{\text{Nombre total de contrats}}$. La variance empirique d'une variable aléatoire X vaut $\frac{1}{n} \sum_{k=0}^n k^2 n_k - (\frac{1}{n} \sum_{i=0}^n k n_k)^2$.

Application numérique : $\tilde{\beta}_{MM} = 1.4856$ et $\tilde{r}_{MM} = 1.1300$

6.2.2 Estimation des paramètres par la méthode du maximum de vraisemblance

La fonction de maximum de vraisemblance s'écrit comme ci-dessous :

$$L(r, \beta) = \prod_{k=0}^{20} \frac{\Gamma(r+k)^{n_k}}{(\Gamma(r) k!)^{n_k}} \left(\frac{1}{1+\beta}\right)^{rn_k} \left(\frac{\beta}{1+\beta}\right)^{kn_k}$$

En utilisant la propriété de la loi gamma : $\Gamma(r+1) = r\Gamma(r)$, on obtient par récurrence $\forall k \Gamma(r+k) = \prod_{i=0}^{k-1} (r+i) \Gamma(r)$. En passant au logarithme on obtient :

$$\begin{aligned} l(r, \beta) &= \ln(L(r, \beta)) \\ &= \sum_{k=0}^{20} [n_k \sum_{i=0}^{k-1} \ln(r+i) - n_k \ln(k!) - rn_k \ln(1+\beta) + kn_k \ln(\beta) - kn_k \ln(1+\beta)] \end{aligned}$$

Pour trouver le couple (r, β) qui maximise la fonction de maximum de vraisemblance, on dérive l'équation précédente par rapport à r puis par rapport à β et il faut que ces deux dérivées soient égales à 0.

$$\begin{aligned} \frac{\partial l(r, \beta)}{\partial r} &= \sum_{k=0}^{20} [n_k \sum_{i=0}^{k-1} \frac{1}{r+i} - n_k \ln(1+\beta)] = 0 \\ \frac{\partial l(r, \beta)}{\partial \beta} &= \sum_{k=0}^{20} [-rn_k \frac{1}{1+\beta} + kn_k \frac{1}{\beta} - kn_k \frac{1}{1+\beta}] = 0 \end{aligned}$$

Après un rapide calcul on obtient β en fonction de r : $\beta = \frac{1}{r} \frac{\sum kn_k}{\sum n_k}$

On résoud ce système par la fonction "optim" dans R. On obtient alors les valeurs suivantes pour r et β : $\hat{r}_{MV} = 0.9972$ et $\hat{\beta}_{MV} = 1.6834$.

6.2.3 Fonction de masse de probabilité empirique et paramétrique

Graphique en bâtons de la fonction de masses de probabilité de la loi Binomiale Négative de paramètres $(0.9972; 1.6834)$.

Comparaison entre la fonction de masse de probabilité de la loi Binomiale Négative (en rouge) et la fonction empirique (en noire).

Le loi binomiale négative décrit beaucoup mieux les données que nous avons bien que ce ne soit pas parfait.

6.2.4 Adéquation de la loi aux données

On utilise le test du khideux pour tester l'adéquation des données à la loi Binomiale Négative $(0.9972; 1.6834)$. Dans notre cas $Q = \sum \frac{(n_k - E_k)^2}{E_k} = 50.95$ c'est encore plus grand que $z_{5\%, 20} = 31,41$ mais nettement plus raisonnable comme valeur. Cependant on rejette encore le modèle (H_0) . Nous retrouvons nos observations précédentes : la loi binomiale négative représente bien mieux nos données que la loi de poisson cependant elle n'est pas adaptée aux données.

6.3 Loi de Poisson modifiée à 0

Les deux lois précédentes ont une forme qui ressemblent à la distribution que l'on a mais on a eu que leur adéquation ne convenait pas. L'idée est donc de les réutiliser intelligemment. On fixe la probabilité en 0 puis on garde les lois à un facteur multiplicatif, (on doit toujours avoir $\sum Proba = 1$). Pour cela on utilise donc les mêmes lois mais modifiée à 0.

La loi de Poisson modifiée à 0 de paramètre (α, λ) est la loi de Poisson de paramètre λ mais dont la probabilité $Pr(X=0) = \alpha$. Comme la somme des probabilité doit être égale à 1 on doit normé les autres

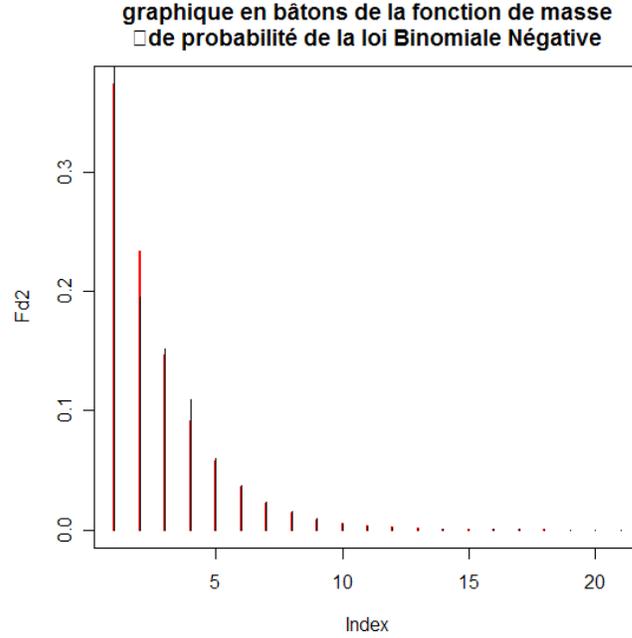


FIG. 30 – Graphe de comparaison des densités de la loi Binomiale négative et de la fonction empirique

probabilités. On note p_n la probabilité $Pr(X = n)$ dont X suit la loi de Poisson non modifiée et p_n^M la probabilité $Pr(X = n)$ avec X suivant la loi de Poisson modifiée à 0. On a donc l'équation suivante : $\sum p_n^M = 1$ sachant que $p_0^M = \alpha$ et $p_n^M = \kappa p_n$. Après un rapide calcul on obtient $\kappa = \frac{1-\alpha}{1-p_0}$. Ceci est valable pour toutes les lois. Dans le cas particulier de la loi de Poisson de paramètre λ , $\kappa = \frac{1-\alpha}{1-e^{-\lambda}}$

6.3.1 Estimation des paramètres par la méthode des moments

Pour déterminer les paramètres de la loi de Poisson modifiée à 0 on utilise les deux premiers moments.

$$E[M] = \sum k p_k^M = \sum_{k>0} k \frac{1-\alpha}{1-e^{-\lambda}} p_k = \frac{1-\alpha}{1-e^{-\lambda}} \sum k p_k = \frac{1-\alpha}{1-e^{-\lambda}} \lambda = \frac{\sum k n_k}{\sum n_k}$$

De meme

$$E[M^2] = \frac{1-\alpha}{1-e^{-\lambda}} \sum k^2 p_k = \frac{1-\alpha}{1-e^{-\lambda}} (\lambda + \lambda^2) = \frac{\sum k^2 n_k}{\sum n_k}$$

D'où $\lambda = \frac{E[M^2]}{E[M]} - 1$ puis $\alpha = 1 - \frac{E[M]}{\lambda} (1 - e^{-\lambda})$.

Application numérique : $\tilde{\lambda}_{MM} = 3.1643$ et $\tilde{\alpha}_{MM} = 0.4919$

6.3.2 Estimation des paramètres par la méthode du maximum de vraisemblance

pour simplifier l'écriture, on note $n = \sum n_k$.

On maximise la fonction de vraisemblance :

$$L(\alpha, \lambda) = \prod (p_k^M)^{n_k} = \alpha^{n_0} \left(\frac{1-\alpha}{1-e^{-\lambda}} \right)^{n-n_0} \prod p_k^{n_k}$$

On prend le logarithme :

$$l(\alpha, \lambda) = n_0 \ln(\alpha) + (n - n_0) \ln(1 - \alpha) - (n - n_0) \ln(1 - e^{-\lambda}) + \sum n_k (k \ln(\lambda) - \lambda - \ln(k!))$$

On résoud cette maximisation grâce à optim de R : on lui donne cette formule de $-L(\alpha, \lambda)$ (car optim cherche le minimum) et les valeurs trouvées avec la méthode des moments comme valeurs initiales.

Application numérique : $\hat{\alpha}_{MV} = 0.3882$ et $\hat{\lambda}_{MV} = 2.5243$

En dérivant par rapport à α le logarithme de la fonction de vraisemblance on trouve $\hat{\alpha}_{MV} = 0.3882$. Les deux résultats sont identiques, la maximisation doit donc être correcte.

6.3.3 Fonction de masse de probabilité empirique et paramétrique

Graphique en bâtons de la fonction de masses de probabilité de la loi de Poisson modifiée à 0 de paramètres (0.3866; 2.5310).

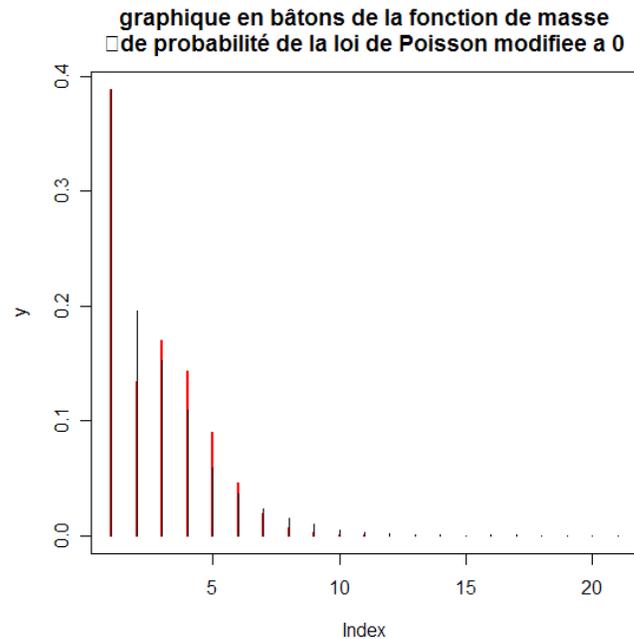


FIG. 31 – Graphe de comparaison des densités de la loi de poisson modifiée et de la fonction empirique

Comparaison entre la fonction de masse de probabilité de la loi de poisson modifiée à 0 (en rouge) et la fonction empirique (en noire).

Il y a adaquation parfaite pour $k=0$ (normale on a construit cette loi comme ça), mais par contre les autres valeur sont mal représentées.

6.3.4 Adéquation de la loi aux données

On obtient $Q = 60486.25$ valeur de Q nettement plus petite que pour la loi de Poisson mais encore beaucoup trop grand et beaucoup plus grande que celle de la loi Binomiale Négative. La loi de poisson moifiée à 0 est bien évidemment rejetée elle aussi.

6.4 Loi Binomiale Négative modifiée à 0

On fait donc la même chose que précédement mais cette fois ci avec la loi Binomiale Négative. Dans le cas particulier de la loi de Binomiale Négative de paramètre (r, β) , $\kappa = \frac{1-\alpha}{1-(1+\beta)^{-r}}$

6.4.1 Estimation des paramètres par la méthode des moments

Comme nous avons trois paramètres à estimer α , r , β , on utilise les trois premiers moments de la loi. On pose m_i le moment i de la loi Binomiale Négative, m_i^M le moment i de la loi Binomiale Négative modifiée à 0 et \hat{m}_i le moment empirique i . On a $\forall i m_i^M = \frac{1-\alpha}{1-(1+\beta)^{-r}} m_i$. Or le moment d'ordre i de la loi Binomiale Négative est la $i^{\text{ème}}$ dérivée de la fonction génératrice des moments prise en 0. La fonction génératrice des moments s'écrit $M_X(t) = (1 - \beta(e^t - 1))^{-r}$, d'où les dérivées suivantes :

$$\begin{aligned} \frac{d M_X(t)}{d t} &= r\beta e^t (1 - \beta(e^t - 1))^{-(r+1)} \\ \frac{d^2 M_X(t)}{d t^2} &= r\beta[(1 + \beta)e^t + r\beta e^{2t}](1 - \beta(e^t - 1))^{-(r+2)} \\ \frac{d^3 M_X(t)}{d t^3} &= r\beta[(1 + \beta)e^t + 2r\beta e^{2t}](1 - \beta(e^t - 1))^{-(r+2)} + r\beta[(1 + \beta)e^t + r\beta e^{2t}](r + 2)\beta(1 - \beta(e^t - 1))^{-(r+3)} \end{aligned}$$

Les moments sont les dérivées prises en 0, on obtient :

$$\begin{aligned} m_1 &= r\beta \\ m_2 &= r\beta(1 + \beta + r\beta) \\ m_3 &= r\beta(1 + \beta + 2r\beta) + r\beta(1 + \beta + r\beta)(r + 2)\beta \\ &= r\beta(1 + 3\beta + 2\beta^2 + 3r\beta + 3r\beta^2 + r^2\beta^2) \end{aligned}$$

Après calcul fastidieux on obtient r puis les deux autres coefficients en cascade. Application numérique : $\alpha_{MM} = 0.3155$, $r_{MM} = 0.5464$ et $\beta_{MM} = 2.046$

6.4.2 Estimation des paramètres par la méthode du maximum de vraisemblance

On maximise la fonction de vraisemblance :

$$L(\alpha, r, \beta) = \alpha^{n_0} \left(\frac{1 - \alpha}{1 - (1 + \beta)^{-r}} \right)^{n - n_0} \prod \left(\frac{\Gamma(r + k)}{\Gamma(r)k!} \left(\frac{1}{1 + \beta} \right)^r \left(\frac{\beta}{1 + \beta} \right)^k \right)^{n_k}$$

On prend le logarithme :

$$\begin{aligned} l(\alpha, r, \beta) &= n_0 \ln(\alpha) + (n - n_0) \ln(1 - \alpha) - (n - n_0) \ln(1 - (1 + \beta)^{-r}) \\ &+ \sum n_k (\ln(\Gamma(r + k)) - \ln(\Gamma(r)) - \ln(k!) - r \ln(1 + \beta) + k \ln(\beta) - k \ln(1 + \beta)) \end{aligned}$$

On résoud cette maximisation grâce à optim de R : on lui donne cette formule de $-L(\alpha, -r, \beta)$ (on met $-r$ comme valeur initiale sinon on ne peut pas évaluer la fonction) et les valeurs trouvées avec la méthode des moments comme valeurs initiales.

Application numérique : $\hat{\alpha}_{MV} = 0.3882$, $\hat{r}_{MV} = 1.919$ et $\hat{\beta}_{MV} = 1.079$

En dérivant par rapport à α le logarithme de la fonction de vraisemblance on trouve $\hat{\alpha}_{MV} = 0.3882$. Les deux résultats sont identiques, la maximisation doit donc être correcte.

6.4.3 Fonction de masse de probabilité empirique et paramétrique

Graphique en bâtons de la fonction de masses de probabilité de la loi Binomiale Négative modifiée à 0 de paramètres (0.3882; 28.5762; 0.1267).

Comparaison entre la fonction de masse de probabilité de la loi Binomiale Négative modifiée à 0 (en rouge) et la fonction empirique (en noire).

La loi binomiale négative modifiée à 0 représente très bien les données, c'est que nous avons prévu au départ.

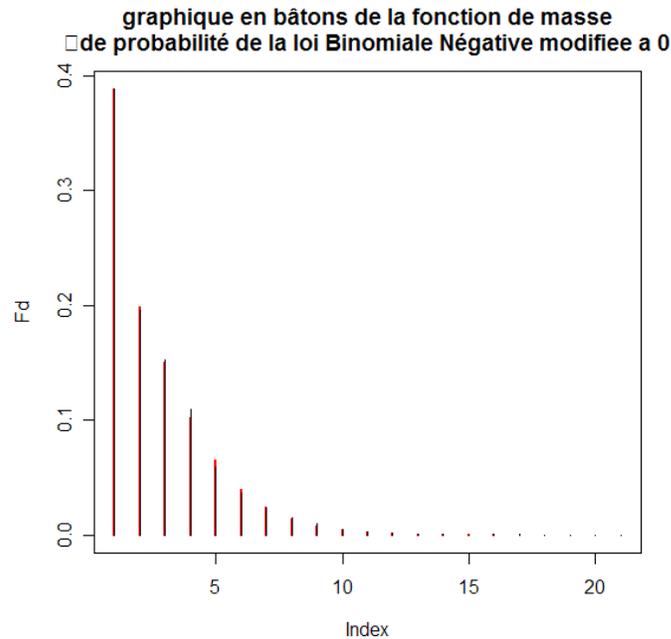


FIG. 32 – Graphe de comparaison des densités de la loi binomiale négative modifiée et de la fonction empirique

6.4.4 Adéquation de la loi aux données

On obtient $Q = 16$, la valeur de Q est plus petite que $c_{alpha,17} = 27.57$ prise dans une table. La loi n'est donc pas rejetée. Ce qui confirme l'adéquation de la loi que nous voyons sur le graphique.

6.5 Sélection de la loi

On sélectionne la loi qui a la valeur de Q la plus faible.

lois	Poisson	Binomiale négative	Poisson modifiée à 0	Binomiale négative modifiée à 0
Q	10400078	51	62560	16

on retient donc la loi Binomiale négative modifiée à 0 comme modèle pour décrire nos données. De plus elle est la seule loi non rejetée que nous avons.

7 Question 6

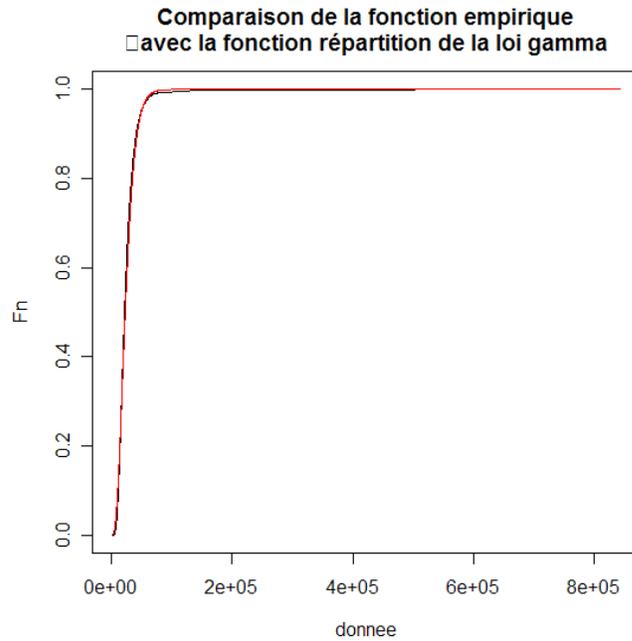
Les questions traitant de la fonction *excess-of-loss* n'ont pas été abordées ici car les tracés de l'exercice 2 nous paraissent anormaux et nous pensons donc que nous n'avons pas compris comment la tracer puisqu'elle ne nous semble pas juste dans l'exercice 2.

7.1 Loi Gamma

On suppose que les données suivent une loi Gamma dont on détermine les paramètres de la même façon qu'à la question 2. Nous ne tenons compte que du résultat obtenu avec la méthode du maximum de vraisemblance. Cependant la méthode des moments nous permet d'initialiser les paramètres dans la maximisation.

Application numérique : $\alpha = 3.7701$ et $\beta = 0.0001488$

Nous commençons par regarder l'adéquation graphique de la loi $\text{gamma}(3.7701, 0.0001488)$ avec nos données.



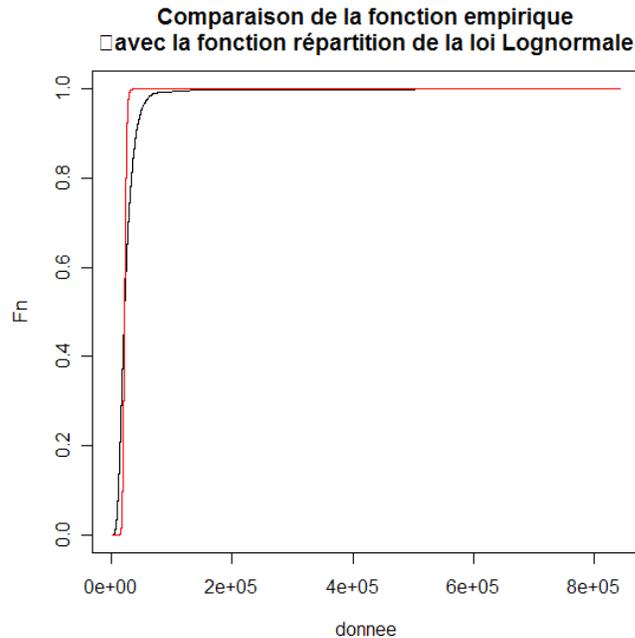
Nous remarquons que la loi gamma décrit assez bien nos données. Cependant le modèle est rejeté par le test de Kolmogorov-Smirnov. En effet $Dn = 0.04947$ alors que $c_{alpha} = 0.01349$, Dn est donc supérieur à c_{alpha} .

7.2 Loi Lognormale

On suppose maintenant que les données suivent une loi Lognormale. On détermine à nouveau les paramètres de la même façon que pour la question 2.

Application numérique : $\mu = 10.0015$ et $\sigma = 0.1275$

Nous commençons par regarder l'adéquation graphique de la loi lognormale(10.0015, 0.1275²) avec nos données.



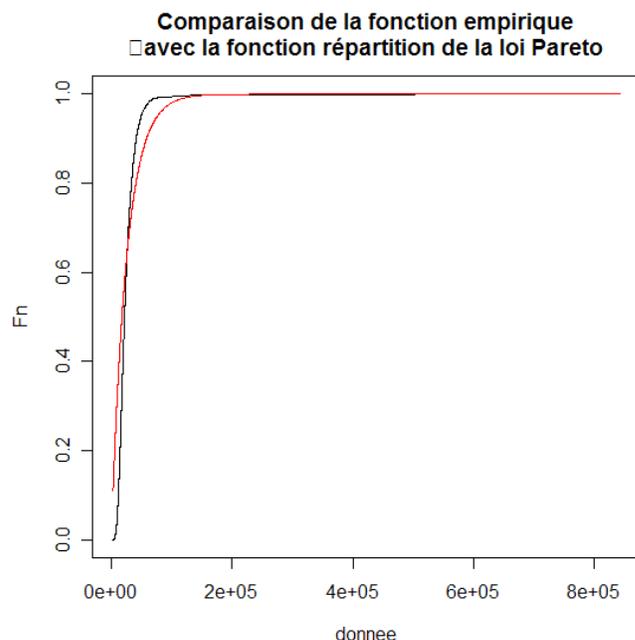
La loi lognormale décrit moins bien nos données, la courbe n'est pas assez arrondie. Le résultat du test de Kolmogorov-Smirnov donne un D_n plus grand donc plus mauvais, $D_n = 0.2912$. Le modèle est bien entendu rejeté.

7.3 Loi Pareto

On suppose maintenant que les données suivent une loi Pareto. On détermine à nouveau les paramètres de la même façon que pour la question 2.

Application numérique : $\alpha = 80154.9$ et $\lambda = 2030738553.5$

Nous commençons par regarder l'adéquation graphique de la loi Pareto(80154.9, 2030738553.5) avec nos données.



La loi Pareto décrit moins bien nos données que la loi gamma, la courbe sous-estime largement la fonction empirique lorsque la fonction empirique atteint 1. Le résultat du test de Kolmogorov-Smirnov donne bien un D_n plus grand que pour la loi gamma donc plus mauvais, $D_n = 0.2753$. Le modèle est bien entendu rejeté.

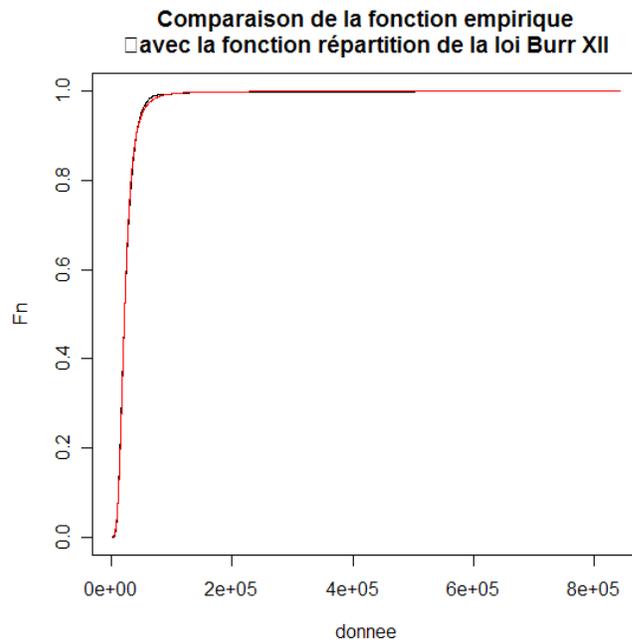
7.4 Loi Burr Type XII

On détermine les coefficients de la loi Burr type XII avec la méthode du maximum de vraisemblance. On initialise le programme avec des valeurs arbitraires. On trouve les valeurs suivantes pour les paramètres de la loi de Burr type XII.

Application numérique : $a = 0.9611$, $b = 0.9611$ et $s = 0.9610$

Nous commençons par regarder l'adéquation graphique de la loi Burr type XII avec nos données.

La loi Burr type XII décrit bien nos données à l'exception d'un petit décrochage en haut. Le résultat du test de Kolmogorov-Smirnov donne $D_n = 0.01581$ alors que $c_\alpha = 0.01349$. D_n étant supérieur à c_α , le modèle est donc rejeté cependant ces deux valeurs sont très proches.

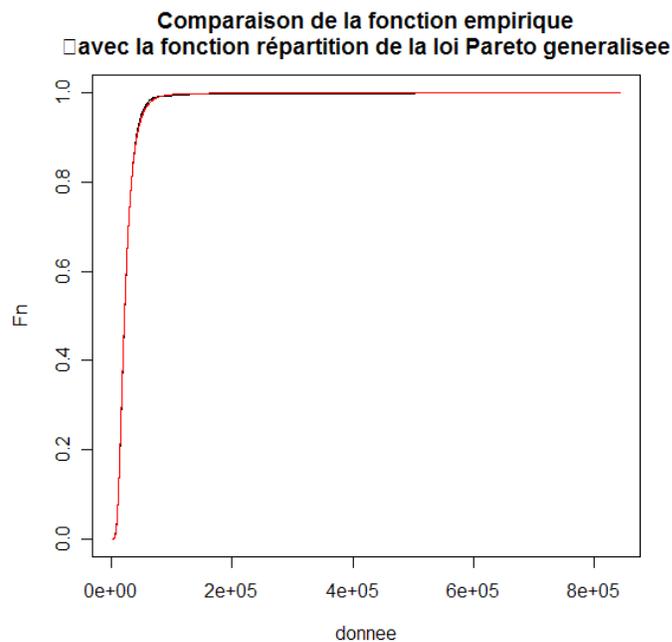


7.5 Méthode Peak-over-threshold avec la loi Pareto généralisée

Comme pour la loi Burr type XII on détermine les coefficients de la loi Pareto généralisée avec la méthode du maximum de vraisemblance. On initialise ici aussi le programme avec des valeurs arbitraires car nous avons le même problème que pour la loi Burr type XII on ne sait pas résoudre le système d'équations que l'on obtient avec la méthode des moments. On trouve les valeurs suivantes pour les paramètres de la loi de Burr type XII.

Application numérique : $a = 6.6494$, $b = 6.6494$ et $s = 6.6494$

Nous commençons par regarder l'adéquation graphique de la loi Pareto généralisée avec nos données.



La loi Pareto généralisée décrit très bien nos données. Le résultat du test de Kolmogorov-Smirnov confirme les conclusions que l'on pouvait tirer du graphe, en effet $Dn = 0.00946$ alors que $c_\alpha = 0.01349$. Dn est donc inférieur à c_α , le modèle n'est donc pas rejeté.

7.6 Loi Lognormale-Pareto composite

On cherche les paramètres de la loi lognormale-pareto comme dans l'article "Modelling actuarial data with a composite lognormal-Pareto". Mais nous obtenons un l'un des deux paramètres égale à l'infini. Nous avons surment fait une erreur quelque part que nous ne repérons pas. Nous n'étudions donc pas cette méthode. (Le code nos notre recherche est fourni en annexe).

8 Question 7 : comparaison de méthodes paramétriques et semi-paramétrique pour l'estimation de copule

8.1 Introduction à l'article

Les copules permettent d'identifier et de séparer les distributions marginales univariées de la distribution de la loi jointe, mais aussi d'établir un degré de dépendance entre ces marginales. C'est ainsi que la théorie des copules a connu un franc succès et les notions de dépendance se sont étendues hors du cadre de la classique loi normale multivariée, bien trop restrictive dans certains cas. En effet, les formes des distributions sont très réduites.

Aujourd'hui, les statisticiens utilisent les copules pour modéliser la dépendance entre des données et c'est pourquoi les études sur l'estimation des paramètres des copules se développe sans cesse et fait l'objet de nombreux articles. Cette approche est géniale car nous pouvons définir une fonction de répartition de loi jointe à partir des fonctions de répartition marginales et de la copule.

L'introduction de lois uniforme pour modéliser les fonctions de répartition marginales permet de spécifier les distributions marginales de la dépendance intrinsèque entre elles ainsi que la loi jointe, et ceci en trois étapes indépendantes.

8.2 Description des trois méthodes d'estimation étudiées

Le but de cet article est de comparer la robustesse des trois méthodes quand les distributions marginales sont inconnues pour estimer le paramètre de la copule de dépendance. Nous allons pour cela confronter l'utilisation de méthodes paramétriques (ML et IFM) à une méthode semi-paramétrique (SP).

8.2.1 Méthode du maximum de vraisemblance (ML)

Cette méthode paramétrique très connue du monde de la statistique est la méthode préférée des statisticiens lorsqu'ils ont besoin de résultats très précis. En effet, l'estimateur du paramètre de la copule par la méthode du maximum de vraisemblance possède des propriétés en terme d'optimalité et nous savons qu'asymptotiquement c'est le meilleur estimateur que nous puissions trouver.

Cependant, nous devons connaître les lois marginales pour utiliser cette méthode efficacement. Le cadre de cette estimation est le cadre classique suivant : nous sommes en dimension 2, il s'agit donc du cas de loi bivariée, les X_i sont i.i.d. Nous utiliserons les notations suivantes :

- $H(x_1, x_2; \xi)$ et $h(x_1, x_2; \xi)$ sont les fonctions de répartition et de densité jointes de (X_1, X_2) ,
- $C(u_1, u_2; \theta)$ est la fonction de répartition jointe de (U_1, U_2) avec $U_k = F_k(x_k; \alpha_k)$ pour $k = 1, 2$ où F_k est la fonction de répartition de X_k ,
- $c(u_1, u_2; \theta)$ est la dérivée de la copule de (X_1, X_2) notée $C(u_1, u_2; \theta)$.

Nous voulons estimer le paramètre θ de la copule.

D'après le théorème de Sklar, nous savons que $H(x_1, x_2) = C(F_1(x_1), F_2(x_2))$.

Ce qui donne $h(x_1, x_2; \xi) = c(F_1(x_1; \alpha_1), F_2(x_2; \alpha_2); \theta) f_1(x_1; \alpha_1) f_2(x_2; \alpha_2)$

La log-vraisemblance de la densité jointe s'écrit $L(\xi) = \sum_{i=1}^n \log [c(F_1(x_{1i}; \alpha_1), F_2(x_{2i}; \alpha_2); \theta) f_1(x_{1i}; \alpha_1) f_2(x_{2i}; \alpha_2)]$. Nous obtenons enfin l'estimateur en maximisant cette expression, mais il est important de noter que cette expression fait intervenir les fonctions de répartition marginales et que donc si elle ne sont pas correctes dans le modèle, l'estimateur devient un mauvais estimateur !

La bonne connaissance des paramètres α_k est donc également déterminante.

8.2.2 Méthode IFM

La méthode IFM (*Inference Function for Margins*) est une méthode paramétrique préférée à la méthode du maximum de vraisemblance lors de calculs et de simulations car elle est plus facile à implémenter. De plus, les concepts sous-jacents de cette méthode se rapprochent de ceux de la méthode ML et dans la plupart des cas, les calculs sont aussi efficaces.

Le même problème que pour la méthode ML apparait : la connaissance de la distribution des marginales a une forte influence sur la qualité de l'estimateur.

Par contre, l'estimation par cette méthode se fait en deux étapes cette fois : la première consiste à d'abord estimer les paramètres α_k des marginales F_{X_k} grâce aux observations disponibles pour chaque membre du couple (X_1, X_2) , nous obtenons les $\hat{\alpha}_k$. Puis nous remplaçons dans l'expression de la log-vraisemblance ci-dessus les α_k par leurs estimateurs et enfin nous maximisons cette nouvelle expression. C'est donc un peu la même démarche que pour la méthode ML mais qui fait intervenir une estimation de plus.

Ainsi, l'estimateur par la méthode IFM est le résultat de la maximisation de l'expression :

$$L(\xi) = \sum_{i=1}^n \log [c(F_1(x_{1i}; \hat{\alpha}_1), F_2(x_{2i}; \hat{\alpha}_2); \theta)]$$

Enfin, il est important de remarquer qu'il est nécessaire de connaître la forme des fonctions de répartition marginales (et leur nombre de paramètres) pour avoir un estimateur cohérent.

8.2.3 Méthode semi-paramétrique (SP)

La méthode semi-paramétrique que nous étudions ici est aussi connue sous le nom de *pseudo maximum de vraisemblance* (PML) ou encore de *maximum de vraisemblance canonique*, et a été développée par *Genest et al.*. Nous n'avons jamais entendu parlé de cette méthode. Pourtant il est évident qu'elle présente certains avantages, dont sa particularité essentielle qui est qu'elle considère les distributions marginales univariées comme des fonctions inconnues. L'estimateur obtenu par cette méthode (PMLE) est moins bon en général que celui obtenu par la méthode ML, mais des études ont montré que sous certaines conditions il est asymptotiquement efficace.

Le gros avantage de cette méthode est qu'elle ne suppose pas les distributions marginales connues, nous n'avons donc pas besoin de les spécifier, ni de spécifier un nombre de paramètres.

Le principe de cette estimation est d'estimer chaque distribution marginale par sa fonction de répartition empirique, en n'utilisant donc aucun paramètre, ce qui permet d'être très libre concernant la famille de lois de chaque distribution marginale. Puis nous estimons la dépendance entre les marginales en estimant le paramètre de la famille de copules choisie. Cette estimation se fait donc en deux étapes comme la méthode IFM.

Le paramètre θ de la copule est alors estimé en maximisant la pseudo-logvraisemblance

$$\sum_{i=1}^n \log [c(\tilde{F}_1(x_{1i}), \tilde{F}_2(x_{2i}); \theta)] \text{ où } \tilde{F}_k \text{ est la fonction de répartition empirique de la marginale } X_k.$$

Nous voyons très bien ici que les paramètres α_k qui intervenaient dans les marginales ont disparu, c'est pourquoi cette méthode d'estimation est semi-paramétrique et avantageuse si l'on ne connaît pas originellement l'expression des marginales.

8.3 Conclusions des analyses de l'article

8.3.1 Comparaison des résultats

Nous allons comparer nos résultats d'estimateurs par rapport à un estimateur de référence, le Benchmark Estimator (BM), qui est évalué dans le cas parfait où nous connaissons les fonctions de répartition marginales

et où nous appliquons la méthode du maximum de vraisemblance.

Nous savons donc qu'il est impossible que les trois estimateurs présentés ci-dessus soient aussi bons dans le cas où nous ne connaissons pas bien les marginales, et la perte de précision par rapport au BM nous donnera une idée de la qualité de chaque estimateur.

Pour effectuer cette comparaison, reportons nous aux tables de l'article qui récapitulent les résultats :

- La table 1 expose les résultats de l'erreur quadratique moyenne (MSE) des méthodes BM, ML et PML par rapport à celle de la méthode IFM dans le cas où les lois sont correctement spécifiées, ici les marginales suivent la loi normale. Nous remarquons que les nombres qui dépassent 100 sont ceux pour lesquels la MSE est plus grande dans la méthode IFM que pour la méthode étudiée, ce qui veut dire que l'approximation étudiée est meilleure. C'est évidemment l'inverse si le nombre est inférieur à 100. Nous ne sommes donc pas surpris de constater que lorsque les marginales sont connues et correctement spécifiées, les méthodes ML et BM sont moins erronées que la méthode SP (ou PML). En effet, les nombres sont proches de 100, avec même une tendance à avoir une meilleure approximation que la méthode IFM ($nb > 100$). La méthode PML est presque toujours moins bonne que ces 3 autres méthodes (2 exceptions en tout).
- La table 2 expose les résultats de biais de l'estimateur par rapport à la vraie valeur du paramètre pour les méthodes ML, IFM et PML dans le cas où les lois sont incorrectement spécifiées, ici les marginales suivent la loi normale pour les méthodes ML et IFM alors que les vraies lois sont différentes. Nous constatons que pour toutes les copules les résultats sont bien meilleurs pour la méthode PML qui approxime beaucoup mieux les paramètres (biais faible de l'ordre de 0,0x), ce qui est logique puisqu'elle s'appuie sur la fonction de répartition empirique qui est donc bien moins faussée que si nous prenons des lois normales pour une loi du chi-2 par exemple.
- Dans la table 3, nous sommes dans les mêmes hypothèses que pour la table 2 sauf que c'est la déviation standard des estimateurs qui est exprimée. Les conclusions sont les mêmes dans la grande majorité des cas même si la meilleure qualité de l'estimateur PML est moins flagrante. Nous remarquons (ce qui est valable pour la table précédente aussi) que la mauvaise spécification des lois n'aboutit pas forcément à des résultats absurdes et l'estimation est même parfois aussi bonne!
Remarque : il ne faut pas comparer le biais et la déviation, l'un peut être grand et l'autre petit (ou inversement) même dans le cas d'un mauvais estimateur (ils sont souvent assez fortement biaisés tout de même).
- La table 4 nous montre les résultats de la MSE de l'estimation ML ou PML par rapport à l'estimation IFM. Nous remarquons que la MSE de l'estimation IFM est 2 fois plus grande à peu près en moyenne que celle de la méthode PML, ce qui dénote une bien meilleure approximation par l'estimateur PML. La méthode ML est en général un peu moins précise que l'IFM au niveau de la MSE.
- La table 5 est la même que la 4 sauf que le nombre d'observations pour chaque échantillon est passé de 250 à 500. La grande différence est qu'on observe un gain de précision très important pour la méthode PML alors que ce n'est pas le cas pour la méthode ML. Ceci s'explique par le fait que la fonction empirique utilisée dans la méthode PML est bien plus précise quand le nombre de données augmente. L'ensemble de ces tableaux montrent donc que ces estimateurs ML et IFM ne sont pas robustes si les marginales sont mal spécifiées (ou pas spécifiées du tout donc!).
- La table 6 montre pour l'estimateur par la méthode PML le taux de couverture d'être dans un intervalle de confiance à 95% pour θ . Nous constatons qu'il est donc facile d'obtenir des intervalles de confiance par cette méthode.
- Nous avons ensuite un exemple d'application dans lequel nous regardons le couple (X_1 = dépenses ménagères, $X_2 = 1 - X_1$) pour un ménage particulier. Le premier travail est de choisir la bonne copule pour modéliser la dépendance entre ces 2 variables aléatoires. Nous utilisons une technique d'évaluation d'une fonction λ qui permet de tester quelle copule s'adapte au mieux à nos données. Il

est clair dans la figure 3 que la fonction λ de la copule de Franck est la mieux adaptée à l'empirique. D'autre part la figure 2 semble montrer qu'il n'y aurait pas de dépendance particulière entre les données. La colonne importante de la table 7 est la colonne de la p-valeur de la statistique du chi-2. On s'aperçoit que la copule de Franck est la meilleure sans ambiguïté, ce que nous aurions pu prédire auparavant car les estimations du rho de Spearman et de Kendall sont plus proches de la valeur empirique pour la copule de Franck, et ce quelque soit la méthode d'estimation. Nous remarquons de plus que les valeurs de la méthode PML sont les plus proches des valeurs empiriques (et proches des valeurs par les autres méthodes où les lois marginales sont supposées normales), ce qui montre que c'est ici la meilleure méthode d'estimation. Si nous supposons les marginales non normales pour la méthode IFM, les valeurs de l'estimateur varie énormément (de 2,47 à 1,83), ce qui confirme la non-robustesse de la méthode paramétrique.

8.3.2 Conclusion générale

Lorsque la loi des distributions marginales est inconnue, la méthode semi-paramétrique (PML) apparait comme la plus robuste d'après les résultats obtenus à travers les simulations. Ce résultat est important dans la pratique puisque les distributions marginales ne sont que très rarement connues. Elle est de plus facilement programmable, car pas loin de la méthode IFM finalement sauf que nous utilisons la fonction de répartition empirique.

De plus la méthode PML s'adapte à n'importe quelle données car la distribution empirique ne présume pas de forme spéciale pour les marginales, ce qui présente un réel avantage en terme de précision de l'estimateur du paramètre de la copule. Nous avons aussi pu nous rendre compte qu'il est important de choisir la bonne copule, celle qui colle le mieux aux données.

Enfin, ces problèmes de dépendance pour des lois jointes multivarées sont très étudiés dans le monde de la finance et de la gestion de risque, où l'on s'intéresse particulièrement à connaître l'allocation de richesse pour tel actif tout en contrôlant le risque global. La théorie des copules est très flexible et peut s'adapter à des données très diverses, d'où son importance.

9 Question 8 :test d'adéquation pour les copules

9.1 introduction

Les copules jouent un rôle très important en actuariat, il s'est donc avéré urgent de réaliser des tests pour estimer leur adéquation aux problèmes donnés. L'auteur récapitule les différentes méthodes proposées jusqu'à présent avant de se pencher sur une catégorie de tests : les 'tests de couverture'. Ils sont intéressants car ne requièrent pas d'ordre particulier à donner aux données, ni de choix arbitraire de paramètres de lissage, de fonction pondérante, de noyau ou autres.

9.2 les différentes procédures d'adéquation

Tout d'abord, il existe trois familles de tests :

- Des procédures en vue de tester des dépendances spécifiques (copule Normale, famille de copules de Clayton).
- Des méthodes statistiques dont l'implémentation nécessite des choix arbitraires de paramètres.
- Les méthodes de test de couverture que nous allons présenter.

Les tests de couvertures sont applicables à toutes sortes de copules. Nous posons quelques notations et généralités :

- Le vecteur $X = (X_1, \dots, X_2)$ est un vecteur aléatoire continu de fonction de répartition H et de marginales F_1, \dots, F_2 et H a pour copule représentative C , unique, et qui admet des marginales uniformes sur $(0, 1)$.

- C est inconnu mais $C \in \tilde{C} = C_\theta, \theta \in O$, ouvert de \mathcal{R}^p .
- On teste l'hypothèse $H_0 : C \in C_0$.
- $U_1 = (U_{1_1}, \dots, U_{1_2}), \dots, U_n$ sont des pseudo-observations déduites des rangs : $U_{i_j} = n\tilde{F}_j(X_{i_j})/(n+1)$, avec $\tilde{F}_j(t) = 1/n \sum_{i=1}^n \mathbf{1}(X_{i_j} \leq t)$.

Voici les diverses sortes de tests :

1. Deux sont basées sur la copule empirique

$$C_n(u) = 1/n \sum_{i=1}^n \mathbf{1}(U_{i1} \leq u_1, \dots, U_{id} \leq u_d)$$

avec $u = (u_1, \dots, u_d) \in [0, 1]^d$. On considère alors plusieurs quantités :

- $\tilde{C} = \sqrt{n} \times (C_n - C_{\theta_n})$, avec C_{θ_n} estimation de C sous H_0 , mais ce processus ne sont pas praticables en vrai.
- avec $S_n = \int_{[0,1]^d} \tilde{C}_n(u)^2 dC_n(u)$ et $T_n = \sup_{u \in [0,1]^d} |\tilde{C}_n(u)|$, nous avons des statistiques qui nous permettent de déduire des P-valeurs approchées dépendantes du comportement asymptotique de \tilde{C} .

2. Deux autres sont basés sur le même principe, mais on considère cette fois-ci la transformation de Kendall

$$X \mapsto V = H(X) = C(F_i(X_i)).$$

Soit K la fonction de distribution de V, on a alors, sous certaines conditions et à l'instar de C_n ,

$$K_n(v) = 1/n \sum_{i=1}^n \mathbf{1}(V_i \leq v), v \in [0, 1]$$

qui est un estimateur consistant de la distribution de K (ou de la copule C pour C_n).

On applique les mêmes procédés qu'avec C_n et en posant cette fois-ci :

- $\tilde{K} = \sqrt{n} \times (K_n - K_{\theta_n})$
- $S_n^K = \int_0^1 \tilde{K}_n(v)^2 dK_{\theta_n}(v)$; $T_n^K = \sup_{v \in [0,1]} |K_n(v)|$.

3. On peut aussi baser des tests sur la transformation de Rosenblatt. À U on associe la transformée de Rosenblatt $R_\theta(U)$ pour $\theta \in O$, et $E_i = R_{\theta_n}(U_i), i = 1, \dots, n$ sont interprétés comme un échantillon de la copule indépendante C_\perp où $C_\perp(e_1, \dots, e_d) = e_1 \times \dots \times e_d$.

Breymann et al les considèrent comme indépendantes et à peu près uniformes sur $[0, 1]^d$ et définissent :

$$\chi_i = \sum_{j=1}^d \phi^{-1}(E_{ij})^2, i = 1..n,$$

où ϕ est la fonction de répartition de $N(0,1)$ et les χ_i forment un échantillon de G, de fonction de distribution une variable aléatoire chi-deux à n degrés de liberté. La distribution empirique de χ_1, \dots, χ_n est une estimation naturelle de G. On pose ensuite :

$$\tilde{G} = \sqrt{n}(G_n - G).$$

En supposant de plus l'indépendance en θ on teste H_0 avec la statistique de Anderson-Darling :

$$A_n = -n - \frac{1}{n} \sum_{i=1}^n (2i-1) [\log(G(\chi(i)))] + \log(1 - G(\chi(n+1-i))).$$

4. On utilise aussi d'autres tests avec la transformation de Rosenblatt, en partant directement des E_i , et on définit sous l'hypothèse H_0 la fonction de distribution empirique

$$D_n(u) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(E_i \leq u), u \in [0, 1]^d.$$

Comme pour les cas antérieurs, les tests sont basés sur une estimation de la distance de la fonction de distribution empirique, ici D_n , et une estimation, ici C_\perp les pseudo-observations, ici les E_i qui sont proches de C_\perp . On utilise ensuite par exemple deux statistique des Cramér-von Mises.

9.3 Résumé des conclusions de l'étude faite dans l'article

De cet article, on peut retenir quelques remarques et conseils quant à l'utilisation des tests de couverture.

- Tout d'abord, comme d'habitude en statistiques, la pertinence du test augmente avec n, la taille de l'échantillon. Mais il y a des exceptions. Le pouvoir du test augmente beaucoup, par exemple quand on a un tau de Kendall de 0,25 ou 0,75 la plupart des exceptions se produisent, et au contraire il y a peu ou pas d'exception pour S_n, T_n et S_n^B .

- Pour que le double bootstrap soit efficace, il faut prendre m , le nombre de répétitions bien supérieur à n , obligatoirement quand C_{θ} et K_{θ} n'ont pas de formules explicites.
- Les tests reposants sur la transformation de Rosenblatt ont l'avantage de ne requérir qu'un simple bootstrap pour approximer leur distribution nulle et extraire les p-valeus. En revanche, l'ordre dans lequel les variables sont conditionnées importent beaucoup.
- Avec les statistiques de Cramér-von Mises, le même processus empirique est inévitablement plus puissant que si l'on utilise la statistique de Kolmogorov-Smirnov.
- S_n et S_n^B conduisent aux résultats les meilleurs pour les tests d'adéquation de couvertures pour les modèles de copules, bien que S_n^B soit plus consistant que S_n et ne recourt pas au double bootstrap, il repose sur une arbitraire et non unique transformée de Rosenblatt.
- S_n^C et S_n^K peuvent être utilisés, et surtout S_n^K quand l'hypothèse nulle est archimédienne, car alors la distribution de Kendall K est connue explicitement.
- Mais les tests basés sur A_n , T_n et T_n^K ne sont pas convaincants et peu recommandés, encore moins en ce qui concerne la version non corrigée du test proposé par Breymann et al. En effet, on ne peut négliger la dépendance entre les pseudo-observations, et de plus, il faut toujours vérifier les conditions de convergence pour les algorithmes bootstrap.

10 Question 9 : estimation des paramètres des copules

10.1 méthode du maximum de vraisemblance (ML)

Avec cette méthode, on maximise l'expression :

$$\sum_{i=1}^n \ln [(f1(x_{1i})f2(x_{2i})c(F1(x_{1i}), F2(x_{2i})), \alpha)],$$

avec α le paramètre de la copule que l'on cherche, $f1$ et $f2$ les fonctions de densités des marginales $F1$ et $F2$, et c la copule dérivée.

Le travail a été effectué sous Excel. Pour les échantillons A et B, on trouve par exemple, pour la copule de Frank avec comme marginales des lois exponentielles de lambdas tous égaux à 0,0001, les valeurs suivantes : $\alpha = -2,87976143$ pour l'échantillon A, et $\alpha = 3,58164676$ pour le B.

Pour cette même copule, avec comme marginales des lois normales $(0, 1; 2)$, on trouve pour l'échantillon C $\alpha = 5,37016687$.

En revanche, toujours avec C, cette méthode ne converge pas pour la copule de Plackett, et donne des paramètres négatifs, donc invalides, pour les copules de Clayton et de Gumbel.

10.2 méthode IFM

Cette fois-ci, on estime d'abord séparément les paramètres de chaque marginale, puis on maximise l'expression :

$$\sum_{i=1}^n \ln(c(u_1, u_2, \alpha)),$$

où u_i sont les marginales avec les paramètres trouvés.

Pour les échantillons A et B, cela marche bien, on trouve le même α qu'avec la méthode MV, avec les mêmes marginales et la même copule.

Pour l'échantillon C, le solveur d'Excel ne trouve pas de solution.

11 Question 10 : Tarification de contrats basée sur l'expérience

Nous sommes dans le cadre d'une assurance IARD qui émet des contrats d'assurance automobile avec une tarification basée sur l'expérience, donc ici sur le nombre de sinistres observés ces cinq dernières années pour chaque contrat. Les variables explicatives (ou critères de tarification) à considérer dans ce modèle de Poisson avec régression sont le sexe et l'âge des conducteurs. Ces variables sont catégoriques.

La variable Θ_i est le paramètre de risque non observable de l'assuré i lors de la signature du contrat.

Par hypothèse, nous considérons que les Θ_i sont i.i.d. et que $E[\Theta_i] = 1$ et $Var(\Theta_i) = \eta^2$.

Dans notre modèle nous avons trois variables explicatives (facteurs de risques) qui sont les x_i pour $i = 1, 2, 3$ et le nombre de sinistres (noté N_i ici) pour l'assuré i à la date t sachant $\Theta_i = \theta_i$ est modélisé par une loi de Poisson de moyenne $(\lambda_i \theta_i)$. Nous avons donc $N_i | \Theta_i = \theta_i \sim P(\lambda_i \theta_i)$

Ainsi, $E[N] = E[E[N|\Theta]] = E[\lambda * \Theta] = \lambda E[\Theta] = \lambda$

En outre, $E[N] = \lambda(i, t) = \exp(\beta_0 x_0(i, t) + \beta_1 x_1(i, t) + \beta_2 x_2(i, t))$ correspond à l'espérance du nombre de sinistres, connues selon la classe de l'assuré :

- pour la première classe qui correspond aux hommes de 25 ans ou moins, $\lambda(i, t) = e^{-1}$,
- la deuxième classe regroupe les hommes de 26 ans ou plus, $\lambda(i, t) = e^{-1-0,2} = e^{-1,2}$,
- la troisième classe contient les femmes de 25 ans ou moins, $\lambda(i, t) = e^{-1-0,5} = e^{-1,5}$,
- la quatrième classe représente les femmes de 26 ans ou plus, $\lambda(i, t) = e^{-1-0,5-0,2} = e^{-1,7}$.

Nous voyons bien que les risques sont modélisés de la même manière quelque soit l'année considérée.

De plus nous introduisons maintenant quelques définitions afin de s'adapter aux particularités de l'exercice :

- B = montant encouru d'un sinistre, par hypothèse de loi Lognormale ($\mu = 8, \sigma = 2$),
- C = montant remboursé d'un sinistre,
- d = franchise,
- a = coassurance,
- l = limite par sinistre.

Alors $C = \min(a * \max(B - d; 0); 1)$, ce qui donne dans notre cas $C = \min(0,8 * B; 250000)$ qui est la protection offerte dans le cadre de notre contrat.

11.1 Modèle de crédibilité de Buhlmann-Straub

Le but de cette section est de montrer comment nous pouvons nous placer dans le cadre du modèle de Buhlmann-Straub en l'adaptant à notre problème. Par suite et en appliquant ce modèle à nos données, nous serons en mesure d'estimer le paramètre de risque non-observable Θ_i .

Nous disposons du nombre de sinistres par assuré i et par année t , et ce pour les 3 479 contrats qui représentent chacun un risque. Nous avons donc 3 479 risques sur une période de 5 ans. On désigne par $N(i, t)$ le nombre total de sinistres pour le risque i pour l'année t , qui remplace le $S(i, t)$ du modèle classique. Le nombre de participants vaut 3 479. On définit par $Y(i, t)$ le coût moyen en sinistre par participant, $Y(i, t) = \frac{N(i, t)}{\lambda(i, t)}$. $\lambda(i, t)$ remplace donc $m(i, t)$.

La v.a. Θ_i représente le facteur de risque non-observable.

On suppose pour $i \in [1, n]$:

- $\Theta_1, \Theta_2, \dots, \Theta_n$ i.i.d. ,
- $E[Y(i, t) | \Theta_i = \theta_i] = E[\frac{N(i, t)}{\lambda(i, t)} | \Theta_i = \theta_i] = \frac{1}{\lambda(i, t)} E[N(i, t) | \Theta_i = \theta_i] = \frac{1}{\lambda(i, t)} \times \lambda(i, t) \theta_i = \theta_i = \mu(\theta_i)$,
- $Var(Y(i, t) | \Theta_i = \theta_i) = \frac{1}{\lambda^2(i, t)} Var(N(i, t) | \Theta_i = \theta_i) = \frac{1}{\lambda^2(i, t)} \lambda(i, t) \theta_i = \frac{\theta_i}{\lambda(i, t)} = \frac{v^2(\theta_i)}{\lambda(i, t)}$,
- $E[\mu(\Theta_i)] = E[\Theta_i] = 1 = \mu$
- $E[v^2(\Theta_i)] = E[\Theta_i] = 1 = v^2$,
- $Var(\mu(\Theta_i)) = Var(\Theta_i) = \eta^2 = a$.

Nous allons ainsi pouvoir appliquer ce modèle à nos données.

11.2 Détermination de la prime de crédibilité et estimation de ses paramètres

Il nous importe maintenant d'estimer le paramètre non-observable de risque Θ_i . Pour cela, en suivant le raisonnement de ce modèle, nous savons que

$$E[Y(i, t)] = E[E[Y(i, t) | \Theta_i = \theta_i]] = E[\mu(\Theta_i)] = \mu = 1 \text{ et } Var(Y(i, t)) = \frac{v^2}{\lambda(i, t)} + a = \frac{1}{\lambda(i, t)} + \eta^2.$$

Posons $\hat{\Theta}_i$ comme l'estimateur linéaire de Θ_i , $\hat{\Theta}_i = c_0 + \sum_{i=1}^n \sum_{t=1}^T c_{i,t} Y(i, t)$.

En cherchant les coefficients c qui minimisent l'erreur quadratique moyenne, nous obtenons comme estimateur pour le paramètre non-observable Θ_i l'expression suivante : $\hat{\Theta}_i = z_i \bar{Y}(i) + (1 - z_i) \mu$

$$\text{avec } z_i = \frac{\lambda(i)}{\lambda(i) + \frac{1}{\eta^2}}, \lambda(i) = \sum_{t=1}^T \lambda(i, t) \text{ et } \bar{Y}(i) = \sum_{t=1}^T \frac{\lambda(i, t) Y(i, t)}{\lambda(i)}.$$

Ce qui nous donne finalement après calculs : $\forall t \in [1, 5], \hat{\Theta}_i = \frac{\sum_t N(i, t)}{5\lambda(i) + \frac{1}{\eta^2}} + (1 - \frac{5\lambda(i)}{5\lambda(i) + \frac{1}{\eta^2}}) * 1 = \frac{\sum_t N(i, t)}{5\lambda(i) + \frac{1}{\eta^2}} + \frac{1}{1 + 5\eta^2 \lambda(i)}$

Nous avons donc une prime de crédibilité représentée par cet estimateur $\hat{\Theta}_i$. Nous voyons qu'elle dépend des paramètres $E[\mu(\Theta_i)] = E[\Theta_i] = \mu = 1$, $E[v^2(\Theta_i)] = E[\Theta_i] = v^2 = 1$ connus par hypothèse donc il n'est pas nécessaire de les estimer, et $Var(\mu(\Theta_i)) = Var(\Theta_i) = a = \eta^2$ qu'il convient donc maintenant d'estimer.

Ainsi, en notant $\hat{\eta}^2$ l'estimateur de $Var(\mu(\Theta_i))$, on a $\hat{\eta}^2 = \frac{\lambda_{..}}{\lambda_{..}^2 + \sum_i \lambda_i^2} \left[\sum_i \lambda_i (\bar{Y}(i) - \bar{Y})^2 - (n - 1) \right]$

$$\text{avec } \lambda_{i.} = \sum_t \lambda(i, t), m_{..} = \sum_i \sum_t \lambda(i, t) = \sum_i \lambda_{i.} \text{ et } \bar{Y}(i) = \frac{\sum_t \lambda(i, t) Y(i, t)}{\sum_t \lambda(i, t)}, \bar{Y} = E[\hat{\mu}(\Theta)] = 1.$$

Il vient donc par suite du calcul $\lambda_{i.} = 5\lambda(i)$, $\lambda_{..} = 5 \sum_i \lambda(i)$ et enfin $\bar{Y}(i) = \frac{\sum_t N(i, t)}{5\lambda(i)}$, d'où l'estimateur :

$$\begin{aligned} \hat{\eta}^2 &= \frac{5 \sum_i \lambda(i)}{(5 \sum_i \lambda(i))^2 + \sum_i (5\lambda(i))^2} \left[\sum_i 5\lambda(i) \left(\frac{\sum_t N(i, t)}{5\lambda(i)} - 1 \right)^2 - (n - 1) \right] \\ &= \frac{5 \sum_i \lambda(i)}{25(\sum_i \lambda(i))^2 + 25 \sum_i \lambda^2(i)} \left[\sum_i 5\lambda(i) \left(\frac{\sum_t N(i, t)}{5\lambda(i)} - 1 \right)^2 - (n - 1) \right] \end{aligned}$$

Nous pouvons donc maintenant estimer totalement la prime de crédibilité.

11.3 Evaluation de la prime pure basée sur l'expérience

La variable Θ_i représente la caractéristique non-observable du conducteur, et $P(\Theta = \theta_j) = f_{\Theta}(\theta_j) = \alpha_j$. Nous avons estimé ce paramètre dans la section précédente.

Nous savons que $(N_{i,t} | \Theta_i = \theta) \sim P(\lambda\theta)$ avec i le risque et t l'année de l'expérience.

Il nous est demandé d'établir la prime pure pour différents contrats.

Nous nous plaçons dans le cas de l'approche fréquence sévérité. Les coûts sont donc définis par la v.a. $X_{i,t}$ avec

$$\begin{cases} \sum_{k=1}^{N_{i,t}} B_{i,t,k}, N_{i,t} > 0 \\ 0, N_{i,t} = 0 \end{cases}$$

Avec $B_{i,t,k}$ le montant du k -ième sinistre pour le risque i la t -ième année, qui suit donc une loi Lognormale ($\mu = 8, \sigma = 2$).

Quand on connaît les caractéristiques du conducteur (sa classe, il y en a 4), on sait que le nombre de sinistres N obéit à une loi de Poisson (λ_j).

Ainsi, avec l'indice j correspondant à la classe :

- pour la première classe ($j=1$) qui correspond aux hommes de 25 ans ou moins, $\lambda_1 = e^{-1}$,
- la deuxième classe ($j=2$) regroupe les hommes de 26 ans ou plus, $\lambda_2 = e^{-1-0,2} = e^{-1,2}$,
- la troisième classe ($j=3$) contient les femmes de 25 ans ou moins, $\lambda_3 = e^{-1-0,5} = e^{-1,5}$,
- la quatrième classe ($j=4$) représente les femmes de 26 ans ou plus, $\lambda_4 = e^{-1-0,5-0,2} = e^{-1,7}$.

A l'émission du contrat, la prime pure pour l'année suivante correspond à $E(X_{i,t+1} | \Theta_i) = E[N_{i,t+1} | \Theta_i] E[B_i]$ (avec $E[B_i] = E[C]$ ici).

Connaissant l'espérance des sinistres, nous pouvons en déduire $E[C]$ qui vaut

$$\begin{aligned} E[C] &= E[\min(0, 8 * B; 250000)] \\ &= \int_0^{250000} (1 - F_{0,8B}(x)) dx \end{aligned}$$

On s'intéresse à estimer $E[N_{i,t+1} | \Theta_i]$, ce que nous faisons par $E[N_{i,t+1} | N_{i,1} = n_{i,1}, N_{i,2} = n_{i,2}, \dots, N_{i,T} = n_{i,T}] = \lambda_i E[\Theta_i | n_{i,1}, n_{i,2}, \dots, n_{i,T}]$ (1)

Ne connaissant pas la loi de $\hat{\Theta}_i$, c'est maintenant que nous utilisons l'expression de l'estimateur de la prime de crédibilité $\forall t \in [1, 5]$, $\hat{\Theta}_i = \frac{\sum_t N_{i,t}}{5\lambda_i + \frac{1}{\hat{\eta}^2}} + \frac{1}{1+5\hat{\eta}^2\lambda_i}$

Nous obtenons en remplaçant dans (1) l'expression

$$\begin{aligned} E[N_{i,t+1}|N_{i,1} = n_{i,1}, N_{i,2} = n_{i,2}, \dots, N_{i,T} = n_{i,T}] &= \lambda_i E[\hat{\Theta}_i | n_{i,1}, n_{i,2}, \dots, n_{i,T}] \\ &= \lambda_i E\left[\frac{\sum_t N_{i,t}}{5\lambda_i + \frac{1}{\hat{\eta}^2}} + \frac{1}{1 + 5\hat{\eta}^2\lambda_i} \middle| n_{i,1}, n_{i,2}, \dots, n_{i,T}\right] \\ &= \lambda_i \left(E\left[\frac{\sum_t N_{i,t}}{5\lambda_i + \frac{1}{\hat{\eta}^2}} \middle| n_{i,1}, n_{i,2}, \dots, n_{i,T}\right] + E\left[\frac{1}{1 + 5\hat{\eta}^2\lambda_i} \middle| n_{i,1}, n_{i,2}, \dots, n_{i,T}\right] \right) \end{aligned}$$

Or les expressions dans les espérances sont des constantes donc nous pouvons les sortir, ce qui nous donne $E[N_{i,t+1}|N_{i,1} = n_{i,1}, N_{i,2} = n_{i,2}, \dots, N_{i,T} = n_{i,T}] = \lambda_i \left(\frac{\sum_t N_{i,t}}{5\lambda_i + \frac{1}{\hat{\eta}^2}} + \frac{1}{1+5\hat{\eta}^2\lambda_i} \right)$

$$\text{avec } \hat{\eta}^2 = \frac{5 \sum_i \lambda(i)}{25(\sum_i \lambda(i))^2 + 25 \sum_i \lambda^2(i)} \left[\sum_i 5\lambda(i) \left(\frac{\sum_t N(i,t)}{5\lambda(i)} - 1 \right)^2 - (n-1) \right]$$

Nous connaissons les λ_j et nos données nous donnent le nombre de sinistres par risque par année. Nous détenons donc toutes les informations nous permettant de calculer cette prime pure.

D'autre part, les sinistres suivent la loi Lognormale ci-dessus, donc $E[B] = e^{8+2} = e^{10}$.

Nous avons calculé ces primes à l'aide du logiciel Excel. Voici donc maintenant l'évaluation de la prime pure pour les contrats demandés :

Contrats	1	23	687	751	2127	2130	2264	2302
Classe	H-25	H-25	H+26	H+26	F-25	F-25	F+26	F+26
Nb total sinistres	0	7	0	3	0	2	5	1
$E[B]$	e^{10}							
Prime pure	6040,12	13890,90	5184,49	8072,49	4071,36	5583,31	6634,43	4079,18

Nous pouvons remarquer en lisant ce tableau que globalement les femmes sont considérées comme moins risquées au volant puisqu'à même catégorie d'âge pour même nombre de sinistres, les hommes paie une prime plus chère. Quand l'âge est augmenté à nombre égal de sinistres, la prime baisse.

Cette assurance a donc un modèle qui considère que les femmes sont moins risquées que les hommes, et que les plus âgés sont moins risqués que les jeunes conducteurs.

En fait nous avons calculé ici la prime pure pour un contrat sans condition sur le montant remboursé (nous avons pris l'espérance de B), mais dans des proportions différentes les conclusions seront les mêmes si nous prenons un contrat tel que celui proposé au début de l'exercice, il suffira alors de prendre l'espérance de C au lieu de celle de B.

12 Conclusion

Ce projet nous a permis de nous familiariser avec certaines notions et méthodes classiques en assurances. Nous avons également constaté la difficulté de travailler en groupe lorsque le groupe n'est pas réuni, ce qui était le cas puisque les cours à l'ISFA se font rares en cette période dans notre master Recherche.

De plus, l'accès à certains documents s'est révélé très difficile (le livre de *Klugman et coll*) car seul l'ISFA les possédait à Lyon et nous étions plusieurs groupes à le vouloir.

Nous avons trouvé très intéressant les articles de recherche à étudier, c'est une nouvelle manière de travailler pour nous car nous n'étions pas en filière de recherche auparavant. Nous retiendrons donc de ce projet une familiarisation non négligeable avec R, un éveil de curiosité et certains points fondamentaux des statistiques dans le domaine des assurances.

13 Annexes

Cette partie du rapport contient le code des fonctions R implémentées pour effectuer presque tous les calculs du projet. Nous remercions au passage le paquetage **actuar** de R pour son aide précieuse... Seuls certains exercices implémentés sous Excel n'apparaissent pas dans cette section.

14 Annexes

Cette partie du rapport contient le code des fonctions R implémentées pour effectuer tous les calculs du projet.

14.1 Question 2

Voici le code R de toutes les fonctions qui nous ont servi dans l'exercice 2 :

```
#####
#####      Question 2      #####
#####

#### Lecture du fichier ####
tab=read.table("donnee_question2.txt",header=T,sep="\t")

#### Définition de variable qui servent pour toutes les lois ####
donnee=tab[,2]
nb=length(donnee)
E=mean(donnee)
V=(nb-1)/nb*var(donnee)

#### Definition d'un bouleen ####
#bool_fichier vaut 0 si on écrase tout ce qu'il y avait dans le fichier res_question5 pour réécrire
# sinon on écrit à la suite de qu'il y a déjà avant

#####
#####      Fonction qui lance toutes les lois et ecrit les resultats dans le fichier resQ2.txt
#####
q2<-function(){
a=loi_exp(0)
b=loi_gamma(1)
c=loi_lnorm(1)
d=loi_pareto(1)
e=loi_burr(1)

##### selection de la loi #####
write("",file="resQ2.txt",append=TRUE)
write("",file="resQ2.txt",append=TRUE)
write("Selection de la Loi",file="resQ2.txt",append=TRUE)

exp_baysien=1*log(nb)-2*(nb*log(a)-a*sum(donnee))
```

```

gamma_baysien=2*log(nb)-2*(nb*(b[1]*log(b[2])-log(gamma(b[1]))))+(b[1]-1)*sum(log(donnee))-b[2]*sum(
lnorm_baysien=2*log(nb)+2*(sum(log(donnee))+nb*(log(2*sqrt(pi))+log(c[2])))+sum((log(donnee)-c[1])^2
pareto_baysien=2*log(nb)-2*(nb*(log(d[1])+d[1]*log(d[2])))-(d[1]+1)*sum(log(d[2]+donnee)))
burr_baysien=3*log(nb)-2*(sum(log(dburr(donnee,e[1],e[2],e[3]))))

```

```

minimum=min(exp_baysien,gamma_baysien,lnorm_baysien,pareto_baysien,burr_baysien)

```

```

if(exp_baysien==minimum){
cat("exponentiel","\n")
write("La loi selectionnee est exponentielle",file="resQ2.txt",append=TRUE)
}else if(gamma_baysien==minimum){
cat("gamma","\n")
write("La loi selectionnee est gamma",file="resQ2.txt",append=TRUE)
}else if(lnorm_baysien==minimum){
cat("lnorm","\n")
write("La loi selectionnee est log-normale",file="resQ2.txt",append=TRUE)
}else if(pareto_baysien==minimum){
cat("pareto","\n")
write("La loi selectionnee est pareto",file="resQ2.txt",append=TRUE)
}else if(burr_baysien==minimum){
cat("burr","\n")
write("La loi selectionnee est burr",file="resQ2.txt",append=TRUE)
}

```

```

cat("exponentielle : ",exp_baysien,"\n")
cat("gamma : ",gamma_baysien,"\n")
cat("lnorm : ",lnorm_baysien,"\n")
cat("pareto : ",pareto_baysien,"\n")
cat("burr : ",burr_baysien,"\n")
cat(minimum,"\n")
}

```

```

#####
#####          Calcul et Graph de Fn          #####
#####
###Calcul
#Les données sont classées
Fn=c()
for (i in 1:nb){
j=i
while(j<nb && donnee[j]==donnee[j+1]){
j=j+1
}
Fn[i]=j/nb
}

```

```

###Gaph
graph_Fn<-function(){

```

```
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main = "graphique de la fonction empirique")
}
```

```
#####
#####          Les Q-Qplots          #####
#####
qqplot_exp<-function(){
plot(donnee,qexp(c(1:nb)/nb),type="l")
}
qqplot_gamma<-function(param){
plot(donnee,qgamma(c(1:nb)/nb,param),type="l")
}
qqplot_lnorm<-function(){
plot(donnee,qlnorm(c(1:nb)/nb),type="l")
}
qqplot_pareto<-function(param1, param2){
plot(donnee,qpareto(c(1:nb)/nb,param1,param2),type="l")
}
qqplot_burr<-function(param1, param2){
plot(donnee,qburr(c(1:nb)/nb,param1,param2),type="l")
}
}
```

```
#####
#####  Calcul et Graph de la fonction d'excès moyen  #####
#####
```

```
graph_en<-function(){
en=c()
for (i in 1:(nb-1)){
j=i+1
en[i]=sum(donnee[j:nb])/(nb-i)-donnee[i]
}
}
```

```
###Gaph
plot(donnee[1:(nb-1)],en, type = "l", col = "black", lwd = 1, main = "graphique de la fonction empirique")
}
```

```
#####
#####          Loi Exponentielle          #####
#####
loi_exp<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Exponentielle",file="resQ2.txt")
}else{
write("",file="resQ2.txt",append=TRUE)
write("",file="resQ2.txt",append=TRUE)
write("Loi Exponentielle",file="resQ2.txt",append=TRUE)
}
}
```

```

}

#### Methode des Moments ####
beta=1/E
write(rbind("beta", "=", beta), file="resQ2.txt", append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance ####
# l<-function(beta){
# -(nb*log(beta)-beta*sum(donnee))
# }
# o=optim(1/E,l)
# betaMV=o$par[1]
# write(c("betaMV", "=", o$par[1]), file="resQ2.txt", append=TRUE, ncolumns=3)

#### adequation graphique ####
#La courbe de Fn (fonction empirique est tracé en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi exponentielle")

#La courbe de répartition de la loi gamma est tracé en rouge
points(donnee,pexp(donnee, beta, lower.tail = TRUE,log.p = FALSE),col="red", type = "l", lwd = 1)

#### excess of loss ####
en=c()
for (i in 1:(nb-1)){
j=i+1
en[i]=sum(beta*exp(-beta*donnee[j:nb]))/(nb-i)-donnee[i]
}

###Gaph
plot(donnee[1:(nb-1)],en, type = "l", col = "black", lwd = 1, main = "graphique de la fonction empiri

#E=c()
#for (i in 1:(nb-1)){
# E[i]=sum(donnee[(i+1):nb]*dexp(donnee[(i+1):nb],beta))-donnee[i]
#}
#plot(donnee[1:(nb-1)],E,type="l")
#points(donnee[1:(nb-1)],1/beta-donnee[1:(nb-1)],type="l", col="red")

#### Test d'adequation ####
diff1=abs(Fn-pexp(donnee,beta))
diff2=c()
diff2[1]=abs(pexp(donnee[1],beta))
diff2[2:nb]=abs(Fn[1:(nb-1)]-pexp(donnee[2:nb],beta))
max1=max(diff1)
max2=max(diff2)

```

```

Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn), file="resQ2.txt", append=TRUE, ncolumns=3)
write(c("c_{alpha}", "=", calpha), file="resQ2.txt", append=TRUE, ncolumns=3)

if (Dn > calpha){
write("On rejette le modele", file="resQ2.txt", append=TRUE)
}else{
write("On ne rejette pas le modele", file="resQ2.txt", append=TRUE)
}
#cat(" Le max vaut : ", Dn, "\n")
return(beta)
}

#####
#####              Loi Gamma              #####
#####
loi_gamma<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Gamma", file="resQ2.txt")
}else{
write("", file="resQ2.txt", append=TRUE)
write("", file="resQ2.txt", append=TRUE)
write("Loi Gamma", file="resQ2.txt", append=TRUE)
}

#### Methode des Moments   ####
alphaMM=E^2/V
betaMM=E/V
write(c("alphaMM", "=", alphaMM, "betaMM", "=", betaMM), file="resQ2.txt", append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance   ####
l<-function(x){
alpha<-x[1]
beta<-x[2]
-(nb*(alpha*log(beta)-log(gamma(alpha)))+(alpha-1)*sum(log(donnee))-beta*sum(donnee))
}
o=optim(c(alphaMM,betaMM), l)
alphaMV=o$par[1]
betaMV=o$par[2]
write(c("alphaMV", "=", o$par[1], "betaMV", "=", o$par[2]), file="resQ2.txt", append=TRUE, ncolumns=3)

####  adequation graphique   ####
#La courbe de Fn (fonction empirique est tracé en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique

```

avec la fonction répartition de la loi gamma")

#La courbe de répartition de la loi gamma est tracé en rouge

```
points(donnee,pgamma(donnee, alphaMV, betaMV, lower.tail = TRUE,log.p = FALSE),col="red", type = "l")
```

```
#### excess of loss ####
```

```
en=c()
```

```
for (i in 1:(nb-1)){
```

```
  j=i+1
```

```
  en[i]=sum(betaMV/gamma(alphaMV)*donnee[j:nb]^(alphaMV-1)*exp(-betaMV*donnee[j:nb]))/(nb-i)-donnee[i]
```

```
}
```

```
###Gaph
```

```
plot(donnee[1:(nb-1)],en, type = "l", col = "black", lwd = 1, main = "graphique de la fonction empirique")
```

```
#E=c()
```

```
#for (i in 1:(nb-1)){
```

```
  # E[i]=sum(donnee[(i+1):nb]*dgamma(donnee[(i+1):nb],alphaMV,betaMV))-donnee[i]
```

```
  #}
```

```
  #plot(donnee[1:(nb-1)],E[1:(nb-1)],type="l")
```

```
  #points(donnee[1:(nb-1)],alphaMV/betaMV-donnee[1:(nb-1)],type="l", col="red")
```

```
#### Test d'adequation ####
```

```
diff1=abs(Fn-pgamma(donnee,alphaMV,betaMV))
```

```
diff2=c()
```

```
diff2[1]=abs(pgamma(donnee[1],alphaMV,betaMV))
```

```
diff2[2:nb]=abs(Fn[1:(nb-1)]-pgamma(donnee[2:nb],alphaMV,betaMV))
```

```
max1=max(diff1)
```

```
max2=max(diff2)
```

```
Dn=max(max1,max2)
```

```
calpha=1.36/sqrt(nb)
```

```
write(c("Dn", "=", Dn),file="resQ2.txt", append=TRUE,ncolumns=3)
```

```
write(c("c_{alpha}", "=", calpha),file="resQ2.txt", append=TRUE,ncolumns=3)
```

```
if (Dn > calpha){
```

```
  write("On rejette le modele",file="resQ2.txt",append=TRUE)
```

```
}else{
```

```
  write("On ne rejette pas le modele",file="resQ2.txt",append=TRUE)
```

```
}
```

```
return(c(alphaMV,betaMV))
```

```
}
```

```
#####  
#####          Loi Lognormale          #####  
#####
```

```

loi_lnorm<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Lognormale",file="resQ2.txt")
}else{
write("",file="resQ2.txt",append=TRUE)
write("",file="resQ2.txt",append=TRUE)
write("Loi Lognormale",file="resQ2.txt",append=TRUE)
}

#### Methode des Moments ####
muMM=log(E)-1/2*log(V/(E^2)+1)
sigmaMM=sqrt(log(V/(E^2)+1))
write(c("muMM","=",muMM,"sigmaMM","=",sigmaMM),file="resQ2.txt",append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance ####
l<-function(x){
mu<-x[1]
sigma<-x[2]
sum(log(donnee))+nb*(log(2*sqrt(pi))+log(sigma))+sum((log(donnee)-mu)^2/(2*sigma))
}
o=optim(c(muMM,sigmaMM),l)
muMV=o$par[1]
sigmaMV=o$par[2]
write(c("muMV","=",o$par[1],"sigmaMV","=",o$par[2]),file="resQ2.txt",append=TRUE,ncolumns=3)

#### adequation graphique ####
#La courbe de Fn (fonction empirique est tracé en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Lognormale")

#La courbe de répartition de la loi Lognormale est tracé en rouge
points(donnee,plnorm(donnee, muMV, sigmaMV, lower.tail = TRUE,log.p = FALSE),col="red", type = "l",

#### excess of loss ####
E=c()
for (i in 1:(nb-1)){
E[i]=sum(donnee[(i+1):nb]*dlnorm(donnee[(i+1):nb],muMV,sigmaMV))-donnee[i]
}
plot(donnee[1:(nb-1)],E[1:(nb-1)],type="l")
#points(donnee[1:(nb-1)],exp(muMV+sigmaMV^2/2)-donnee[1:(nb-1)],type="l", col="red")

#### Test d'adequation ####
diff1=abs(Fn-plnorm(donnee,muMV,sigmaMV))
diff2=c()
diff2[1]=abs(plnorm(donnee[1],muMV,sigmaMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-plnorm(donnee[2:nb],muMV,sigmaMV))
max1=max(diff1)

```

```

max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn), file="resQ2.txt", append=TRUE, ncolumns=3)
write(c("c_{alpha}", "=", calpha), file="resQ2.txt", append=TRUE, ncolumns=3)

if (Dn > calpha){
write("On rejette le modele", file="resQ2.txt", append=TRUE)
}else{
write("On ne rejette pas le modele", file="resQ2.txt", append=TRUE)
}
return(c(muMV, sigmaMV))
}

#####
#####          Loi Pareto          #####
#####
loi_pareto<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Pareto", file="resQ2.txt")
}else{
write("", file="resQ2.txt", append=TRUE)
write("", file="resQ2.txt", append=TRUE)
write("Loi Pareto", file="resQ2.txt", append=TRUE)
}

#### Methode des Moments ####
alphaMM=2*V/(V-E^2)
lambdaMM=E*(alphaMM-1)
write(c("alphaMM", "=", alphaMM, "lambdaMM", "=", lambdaMM), file="resQ2.txt", append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance ####
l<-function(x){
alpha<-x[1]
lambda<-x[2]
-(nb*(log(alpha)+alpha*log(lambda))-(alpha+1)*sum(log(lambda+donnee)))
}
o=optim(c(alphaMM, lambdaMM), l)
alphaMV=o$par[1]
lambdaMV=o$par[2]
write(c("alphaMV", "=", o$par[1], "lambdaMV", "=", o$par[2]), file="resQ2.txt", append=TRUE, ncolumns=3)

#### adéquation graphique ####
#La courbe de Fn (fonction empirique est tracé en noire
plot(donnee, Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Pareto")

```

```

#La courbe de répartition de la loi Pareto est tracé en rouge
points(donnee,ppareto(donnee, alphaMV, lambdaMV, lower.tail = TRUE,log.p = FALSE),col="red", type =

#### excess of loss ####
E=c()
for (i in 1:(nb-1)){
E[i]=sum(donnee[(i+1):nb]*dpareto(donnee[(i+1):nb],alphaMV,lambdaMV))-donnee[i]
}
plot(donnee[1:(nb-1)],E[1:(nb-1)],type="l")
#points(donnee[1:(nb-1)],lambdaMV/(alphaMV-1)-donnee[1:(nb-1)],type="l", col="red")

#### Test d'adequation ####
diff1=abs(Fn-ppareto(donnee,alphaMV,lambdaMV))
diff2=c()
diff2[1]=abs(ppareto(donnee[1],alphaMV,lambdaMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-ppareto(donnee[2:nb],alphaMV,lambdaMV))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn","=",Dn),file="resQ2.txt",append=TRUE,ncolumns=3)
write(c("c_{alpha}","=",calpha),file="resQ2.txt",append=TRUE,ncolumns=3)

if (Dn > calpha){
write("On rejette le modele",file="resQ2.txt",append=TRUE)
}else{
write("On ne rejette pas le modele",file="resQ2.txt",append=TRUE)
}
return(c(alphaMV,lambdaMV))
}

#####
#####          Loi Burr de type XII          #####
#####
loi_burr<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Burr de type XII",file="resQ2.txt")
}else{
write("",file="resQ2.txt",append=TRUE)
write("",file="resQ2.txt",append=TRUE)
write("Loi Burr de type XII",file="resQ2.txt",append=TRUE)
}

#### Methode des Moments ####

```

```

#### Methode du maximum de vraisemblance ####
l<-function(x){
a<-x[1]
b<-x[2]
s<-x[3]
#-(nb*log(a*b)+b*sum(log(donnee))-nb*b*log(s)-sum(log(donnee))-(a+1)*sum(log(1+(donnee/s)^b)))
-(sum(log(dburr(donnee,a,b,s))))
}
o=optim(c(1.5,1.5,.00015),l)
aMV=o$par[1]
bMV=o$par[2]
sMV=o$par[3]
write(c("aMV", "=", o$par[1], "bMV", "=", o$par[2], "sMV", "=", o$par[3]),file="resQ2.txt", append=TRUE,ncol=3)

####  adequation graphique  ####
#La courbe de Fn (fonction empirique est tracé en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Burr")

#La courbe de répartition de la loi Pareto est tracé en rouge
points(donnee,pburr(donnee, aMV, bMV, sMV, lower.tail = TRUE,log.p = FALSE),col="red", type = "l", lwd = 1)

#### excess of loss  ####
E=c()
for (i in 1:(nb-1)){
E[i]=sum(donnee[(i+1):nb]*dburr(donnee[(i+1):nb],aMV,bMV,sMV))-donnee[i]
}
plot(donnee[1:(nb-1)],E[1:(nb-1)],type="l")
#points(donnee[1:(nb-1)],exp(muMV+sigmaMV^2/2)-donnee[1:(nb-1)],type="l", col="red")

#### Test d'adequation  ####
diff1=abs(Fn-pburr(donnee,aMV,bMV,sMV))
diff2=c()
diff2[1]=abs(pburr(donnee[1],aMV,bMV,sMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-pburr(donnee[2:nb],aMV,bMV,sMV))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn),file="resQ2.txt", append=TRUE,ncolumns=3)
write(c("c_{alpha}", "=", calpha),file="resQ2.txt", append=TRUE,ncolumns=3)

if (Dn > calpha){
write("On rejette le modele",file="resQ2.txt",append=TRUE)
}else{
write("On ne rejette pas le modele",file="resQ2.txt",append=TRUE)
}
return(c(aMV,bMV,sMV))

```

}

Ci-dessous le fichier de reponse de la question 2 :

Loi Exponentielle

beta = 0.000122697687390143
Dn = 0.0927159853317877
c_{alpha} = 0.0326128773643594
On rejette le modele

Loi Gamma

alphaMM = 0.4690642875449
betaMM = 5.75531033190642e-05
alphaMV = 0.84550839741444
betaMV = 0.000103715230748097
Dn = 0.0728539131712503
c_{alpha} = 0.0326128773643594
On rejette le modele

Loi Lognormale

muMM = 8.43496650145032
sigmaMM = 1.06847606678934
muMV = 8.3089285776746
sigmaMV = 0.809838037161274
Dn = 0.106925149567132
c_{alpha} = 0.0326128773643594
On rejette le modele

Loi Pareto

alphaMM = 3.76693440106298
lambdaMM = 22550.8276473454
alphaMV = 3.65049668692617
lambdaMV = 21529.6278531219
Dn = 0.0295566413196886
c_{alpha} = 0.0326128773643594
On ne rejette pas le modele

Loi Burr de type XII

aMV = 2.03275876241943
bMV = 2.03275876241943
sMV = 2.03275876241943
Dn = 0.0138938540702532
c_{alpha} = 0.0326128773643594
On ne rejette pas le modele

Selection de la Loi
 La loi selectionnee est burr

14.2 Question 3

Voici le code R de toutes les fonctions qui nous ont servi dans l'exercice 3 : attention il n'est pas indenté comme dans le programme ici donc difficile à lire...

```
#Lecture et importation des donnees a partir du fichier texte
x<-read.table("DonneesQ3.txt", header=T, col.names=c("classe","borne inf","borne sup","nombre"), sep=";")

#####
##### PARTIE : TRACE DES FONCTIONS DE STATISTIQUES DESCRIPTIVES #####
#####

#Trace la fonction de repartition empirique
#x est l'echantillon de donnees groupees
FdR_emp<-function(x)
{
nb_lignes<-nrow(x)
v<-matrix(data=0,nrow=nb_lignes,ncol=2)
nbTotalSinistres<-sum(x[,4])
v[,1]<-(x[,2]+x[,3])/2 #chargement ds vect du milieu des classes
v[,2]=x[,4]/2
accNb=0
for (i in 2:nb_lignes){
accNb=accNb+x[i-1,4]
v[i,2] = accNb+x[i,4]/2
}
v[,2]=v[,2]/nbTotalSinistres
plot(v[1:22,1],v[1:22,2],xlab="Milieux des classes",ylab="Nombre relatif de donnees inferieures",main="")
return(v)
}

#Trace la fonction empirique d'excès moyen
#x est l'echantillon de donnees groupees, elles sont deja classees!
Exces_moyen<-function(x)
{
nb_lignes<-nrow(x) #nb de bornes pour les x_k
e_n<-matrix(data=0,nrow=nb_lignes-1,ncol=2)
e_n[,1]<-x[1:(nb_lignes-1),2]
w<-(x[1:(nb_lignes-1),2]+x[1:(nb_lignes-1),3])/2
e_n[,2]<-sum(x[,4]*w)/sum(x[,4])
for (i in 2:(nb_lignes-1)){
acc_Xj = 0
acc_nb = 0
for (j in i:(nb_lignes-1)){
acc_Xj = acc_Xj + x[j,4]*w[j]
acc_nb = acc_nb + x[j,4]
}
}
}
```

```

}
e_n[i,2] = (acc_Xj/acc_nb) - x[i,2]
}
plot(e_n[1:(nb_lignes-1),1],e_n[1:(nb_lignes-1),2],xlab="Bornes des classes",ylab="Exces moyen",mai
}

#Analyse graphique avec le q-q plot de la fonction Exp
QQplot_exp<-function(x)
{
beta_MV<-EMV_exp(x)
nb_lignes<-nrow(x)
qq<-matrix(data=0,nrow=nb_lignes,ncol=2)
qq[,1]<-x[,2]
qq[1,2]<-0
for (i in 2:nb_lignes) {
accNb=0
for (j in 2:i){
accNb = accNb + x[j-1,4]
}
qq[i,2]<-qexp(accNb/(sum(x[,4]+1)),beta_MV)
}
plot(qq[,1], qq[,2],xlab="Bornes des classes",ylab="Quantile associe",main="QQ-plot de la loi Exp")
# reg<-lm(qq[1:22,2]~qq[1:22,1])
# abline(reg)
}

#Analyse graphique avec le q-q plot de la fonction Gamma
QQplot_gamma<-function(x)
{
alpha_MV<-EMV_gamma(x)[1]
beta_MV<-EMV_gamma(x)[2]
nb_lignes<-nrow(x)
qq<-matrix(data=0,nrow=nb_lignes,ncol=2)
qq[,1]<-x[,2]
qq[1,2]<-0
for (i in 2:nb_lignes) {
accNb=0
for (j in 2:i){
accNb = accNb + x[j-1,4]
}
qq[i,2]<-qgamma(accNb/(sum(x[,4]+1)),alpha_MV,beta_MV)
}
plot(qq[,1], qq[,2],xlab="Bornes des classes",ylab="Quantile associe",main="QQ-plot de la loi Gamma")
reg<-lm(qq[1:20,2]~qq[1:20,1])
abline(reg)
}

```

```

#Analyse graphique avec le q-q plot de la fonction Lognormale
QQplot_lognormale<-function(x)
{
mu_MV<-EMV_lognormale(x)[1]
sigma_MV<-EMV_lognormale(x)[2]
nb_lignes<-nrow(x)
qq<-matrix(data=0,nrow=nb_lignes,ncol=2)
qq[,1]<-x[,2]
qq[1,2]<-0
for (i in 2:nb_lignes) {
accNb=0
for (j in 2:i){
accNb = accNb + x[j-1,4]
}
qq[i,2]<-qlnorm(accNb/(sum(x[,4]+1)),mu_MV,sigma_MV)
}
plot(qq[,1], qq[,2],xlab="Bornes des classes",ylab="Quantile associe",main="QQ-plot de la loi Lognormale")
reg<-lm(qq[1:22,2]~qq[1:22,1])
abline(reg)
}

```

```

#Analyse graphique avec le q-q plot de la fonction Pareto
QQplot_pareto<-function(x)
{
alpha_MV<-EMV_pareto(x)[1]
lambda_MV<-EMV_pareto(x)[2]
nb_lignes<-nrow(x)
qq<-matrix(data=0,nrow=nb_lignes,ncol=2)
qq[,1]<-x[,2]
qq[1,2]<-0
for (i in 2:nb_lignes) {
accNb=0
for (j in 2:i){
accNb = accNb + x[j-1,4]
}
qq[i,2]<-qpareto(accNb/(sum(x[,4]+1)),alpha_MV,lambda_MV)
}
plot(qq[,1], qq[,2],xlab="Bornes des classes",ylab="Quantile associe",main="QQ-plot de la loi Pareto")
reg<-lm(qq[1:22,2]~qq[1:22,1])
abline(reg)
}

```

```

#Analyse graphique avec le q-q plot de la fonction Burr XII
QQplot_burr<-function(x)
{
alpha_MV<-EMV_burr(x)[1]
lambda_MV<-EMV_burr(x)[2]
tau_MV<-EMV_burr(x)[3]

```

```

nb_lignes<-nrow(x)
qq<-matrix(data=0,nrow=nb_lignes,ncol=2)
qq[,1]<-x[,2]
qq[1,2]<-0
for (i in 2:nb_lignes) {
  accNb=0
  for (j in 2:i){
    accNb = accNb + x[j-1,4]
  }
  qq[i,2]<-qburr(accNb/(sum(x[,4]+1)),alpha_MV,tau_MV,lambda_MV)
}
plot(qq[,1], qq[,2],xlab="Bornes des classes",ylab="Quantile associe",main="QQ-plot de la loi Burr")
reg<-lm(qq[1:22,2]~qq[1:22,1])
abline(reg)
}

```

```

#####
##### PARTIE : ESTIMATEUR PAR LA METHODE DES MOMENTS #####
#####

```

```

##### FONCTIONS AUXILIAIRES DE CALCUL #####

```

```

#Fonction de calcul de la moyenne empirique dans un echantillon de donnees groupees
#cette fonction prend en argument l'echantillon

```

```

Moy_emp<-function(x)
{
  nb_lignes<-nrow(x)
  nb_donnees<-sum(x[,4]) #nb de donnees de l'echantillon
  v<-(x[1:nb_lignes,2]+x[1:nb_lignes,3])/2 #chargement ds vect du milieu des classes
  sommeXi=0 #initialisation
  for (i in 1:nb_lignes){ #calcul de la somme des Xi "ponderee"
    sommeXi = sommeXi+v[i]*x[i,4]
  }
  return((1/nb_donnees)*sommeXi)
}

```

```

#Fonction de calcul de la variance empirique dans un echantillon de donnees groupees
#cette fonction prend en argument l'echantillon

```

```

Var_emp<-function(x)
{
  n<-nrow(x)
  nb_donnees<-sum(x[,4]) #nb de donnees de l'echantillon
  v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
  somme=0
  for (i in 1:n){ #calcul de la somme des Xi carre "ponderee"
    somme = somme + (x[i,4])*(v[i]-Moy_emp(x))^2
  }
  return((1/nb_donnees)*somme)
}

```

```
#####
```

```
#Loi Exp
EMM_exp<-function(x)
{
beta_n<-1/Moy_emp(x) #calcul de l'estimateur
cat("L'estimateur de beta par la methode des moments de la loi Exp vaut : ", beta_n,"\n")
}
```

```
#Loi Gamma
EMM_gamma<-function(x)
{
alpha_n<-((Moy_emp(x))^2)/Var_emp(x) #calcul de l'estimateur de alpha
beta_n<-Moy_emp(x)/Var_emp(x) #calcul de l'estimateur de beta
cat("L'estimateur de alpha par la methode des moments de la loi Gamma vaut : ", alpha_n,"\n")
cat("L'estimateur de beta par la methode des moments de la loi Gamma vaut : ", beta_n,"\n")
}
```

```
#Loi log-normale
EMM_lognormale<-function(x)
{
mu_n<-log(Moy_emp(x))-(log((Var_emp(x)/(Moy_emp(x))^2)+1)/2)
sigma_n<-log((Var_emp(x)/(Moy_emp(x))^2)+1)
cat("L'estimateur de mu par la methode des moments de la loi log-normale vaut : ", mu_n,"\n")
cat("L'estimateur de sigma carre par la methode des moments de la loi log-normale vaut : ", sigma_n)
}
```

```
#Loi Pareto
EMM_pareto<-function(x)
{
alpha_n<-2/(1-((Moy_emp(x))^2/Var_emp(x)))
lambda_n<-Moy_emp(x)*(2/(1-((Moy_emp(x))^2/Var_emp(x)))-1)
cat("L'estimateur de alpha par la methode des moments de la loi Pareto vaut : ", alpha_n,"\n")
cat("L'estimateur de lambda par la methode des moments de la loi Pareto vaut : ", lambda_n,"\n")
}
```

```
#Loi Burr XII
EMM_burr<-function(x)
{
}
```

```
#####
##### PARTIE : ESTIMATEUR PAR LA METHODE DU MAXIMUM DE VRAISEMBLANCE #####
#####
```

```

#Loi Exp de parametre beta a estimer
#Ce calcul prend en argument le vecteur echantillon
EMV_exp<-function(x)
{
#retourne l'oppose de la logvraisemblance car nous minimisons ensuite avec optimize
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes
beta<-par
somme=0
for (i in 1:r){
somme = somme + e[i,4]*log(exp(-beta*e[i,2])-exp(-beta*e[i,3]))
}
v<-(-somme)
}
#par default optimize minimise par rapport au premier parametre de la fonction a optimiser
#EMV_exp<-optimize(log_vraisemblance,interval=c(.000001,.0001))
EMV_exp<-optim(c(0.00001),log_vraisemblance) #valeur initiale trouvee empiriquement avc FdR
EMV_exp<-EMV_exp$par
#v<-FdR_emp(x)
#points(v[1:22,1],pexp(v[1:22,1],EMV_exp),col="red",type="l",lwd=1)
#cat("L'estimateur de beta par la methode du maximum de vraisemblance de la loi Exponentielle vaut\
return(EMV_exp)
}

#Loi Gamma de parametre de forme alpha et de parametre d'echelle beta
EMV_gamma<-function(x)
{
#retourne l'oppose de la log-vraisemblance car optim minimise normalement
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes
alpha<-par[1]
beta<-par[2]
somme=0
for (i in 1:r){
somme<-somme+e[i,4]*log(pgamma(e[i,3],alpha,beta,lower=TRUE,log=FALSE) - pgamma(e[i,2],alpha,beta,1
}
v<-(-somme)
}
EMV_gamma<-optim(c(0.001,0.0000001),log_vraisemblance)
EMV_gamma<-EMV_gamma$par
#v<-FdR_emp(x)
#points(v[1:22,1],pgamma(v[1:22,1],EMV_gamma[1],EMV_gamma[2]),col="red",type="l",lwd=1)
#cat("Les estimateurs de alpha et beta par la methode du maximum de vraisemblance de la loi Gamma v
return(EMV_gamma)
}

#Loi Pareto de parametre alpha et lambda avec utilisation de la formule explicite de la log-vraisembl
EMV_pareto<-function(x)

```

```

{
#retourne l'oppose de la log-vraisemblance car optim minimise normalement
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes
alpha<-par[1]
lambda<-par[2]
somme=0
for (i in 1:r){
somme<-somme+e[i,4]*log((lambda/(lambda+e[i,2]))^alpha-(lambda/(lambda+e[i,3]))^alpha)
}
v<-(-somme)
}
EMV_pareto<-optim(c(2,10),log_vraisemblance)
EMV_pareto<-EMV_pareto$par
#v<-FdR_emp(x)
#points(v[1:22,1],ppareto(v[1:22,1],EMV_pareto[1],EMV_pareto[2]),col="red",type="l",lwd=1)
#cat("Les estimateurs de alpha et lambda par la methode du maximum de vraisemblance de la loi Pareto
return(EMV_pareto)
}

```

```

#Loi Pareto de parametre de forme alpha et d'echelle lambda avec utilisation de la FdR de pareto
EMV_pareto2<-function(x)
{
#retourne l'oppose de la log-vraisemblance car optim minimise normalement
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes
alpha<-par[1]
lambda<-par[2]
somme=0
for (i in 1:r){
somme<-somme+e[i,4]*log(ppareto(e[i,3],alpha,lambda,lower=TRUE,log=FALSE) - ppareto(e[i,2],alpha,lambda))
}
v<-(-somme)
}
EMV_pareto2<-optim(c(2,10),log_vraisemblance)
EMV_pareto2<-EMV_pareto2$par
#v<-FdR_emp(x)
#points(v[1:22,1],ppareto(v[1:22,1],EMV_pareto2[1],EMV_pareto2[2]),col="red",type="l",lwd=1)
#cat("Les estimateurs de alpha et lambda par la methode du maximum de vraisemblance de la loi Pareto
return(EMV_pareto2)
}

```

```

#Loi log-normale de parametre mu et sigma
EMV_lognormale<-function(x)
{
#retourne l'oppose de la log-vraisemblance car optim minimise normalement
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes

```

```

mu<-par[1]
sigma<-par[2]
somme=0
for (i in 1:r){
somme<-somme+x[i,4]*log( plnorm(e[i,3],mu,sigma) - plnorm(e[i,2],mu,sigma) )
}
v<-(-somme)
}
EMV_lognormale<-optim(c(11,6),log_vraisemblance)
EMV_lognormale<-EMV_lognormale$par
#v<-Fdr_emp(x)
#points(v[1:22,1],plnorm(v[1:22,1],EMV_lognormale[1],EMV_lognormale[2]),col="red",type="l",lwd=1)
#cat("Les estimateurs de mu et sigma par la methode du maximum de vraisemblance de la loi Lognormal
return(EMV_lognormale)
}

```

```

#Loi Burr XII de parametres de forme alpha et tau et de parametre d'echelle lambda
EMV_burr<-function(x)
{
#retourne l'oppose de la log-vraisemblance car optim minimise normalement
log_vraisemblance<-function(par,e=x){
r<-nrow(e) #nombre de classes
alpha<-par[1]
lambda<-par[2]
tau<-par[3]
somme=0
for (i in 1:r){
#avec les notations du poly pour la loi Burr
somme<-somme+e[i,4]*log( pburr(e[i,3],alpha,tau,lambda) - pburr(e[i,2],alpha,tau,lambda) )
}
v<-(-somme)
}
EMV_burr<-optim(c(0.5,0.5,0.5),log_vraisemblance)
EMV_burr<-EMV_burr$par
#v<-Fdr_emp(x)
#points(v[1:22,1],pburr(v[1:22,1],EMV_burr[1],EMV_burr[3], EMV_burr[2]),col="red",type="l",lwd=1)
#cat("Les estimateurs de alpha, lambda et tau par la methode du maximum de vraisemblance de la loi
return(EMV_burr)
}

```

```

#####
##### PARTIE : TRACE DES FONCTIONS EXCESS OF LOSS #####
#####

```

```

#Loi Exp de parametre Beta, calcule l'excess-of-loss pour differents seuils d
#on utilise les parametres obtenus par le MV
EoL_exp<-function(x)
{
beta_MV<-EMV_exp(x)

```

```

n<-nrow(x)
v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
excess<-matrix(data=0, nrow=n, ncol=2)
excess[,1]<-x[,2] # on charge les seuils d=0, 1000, 2000...
for (i in 1:n){
acc = 0
for (j in i:n){
acc = acc + v[j]*( pexp(x[j,3],beta_MV) - pexp(x[j,2],beta_MV) )
}
excess[i,2]<-acc/(1-pexp(x[i,2],beta_MV)) - x[i,2]
}
plot(excess[,1],excess[,2],xlab="Bornes des classes",ylab="",main="Excess of loss de la loi Exponent",
#points(excess[,1],1/beta_MV - excess[,1], type="l",col="red")
}

```

```

#Loi Gamma de parametres alpha et Beta, calcule l'excess-of-loss pour differents seuils d
#on utilise les parametres obtenus par le MV

```

```

EoL_gamma<-function(x)
{
alpha_MV<-EMV_gamma(x)[1]
beta_MV<-EMV_gamma(x)[2]
n<-nrow(x)
v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
excess<-matrix(data=0, nrow=n, ncol=2)
excess[,1]<-x[,2] # on charge les seuils d=0, 1000, 2000...
for (i in 1:n){
acc = 0
for (j in i:n){
acc = acc + v[j]*( pgamma(x[j,3],alpha_MV,beta_MV) - pgamma(x[j,2],alpha_MV,beta_MV) )
}
excess[i,2]<-acc/(1-pgamma(x[i,2],alpha_MV,beta_MV)) - x[i,2]
}
plot(excess[,1],excess[,2],xlab="Bornes des classes",ylab="",main="Excess of loss de la loi Gamma",
}

```

```

#Loi Pareto de parametres alpha et lambda, calcule l'excess-of-loss pour differents seuils d
#on utilise les parametres obtenus par le MV

```

```

EoL_pareto<-function(x)
{
alpha_MV<-EMV_pareto(x)[1]
lambda_MV<-EMV_pareto(x)[2]
n<-nrow(x)
v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
excess<-matrix(data=0, nrow=n, ncol=2)
excess[,1]<-x[,2] # on charge les seuils d=0, 1000, 2000...
for (i in 1:n){
acc = 0
for (j in i:n){

```

```

acc = acc + v[j]*( ppareto(x[j,3],alpha_MV,lambda_MV) - ppareto(x[j,2],alpha_MV,lambda_MV) )
}
excess[i,2]<-acc/(1-ppareto(x[i,2],alpha_MV,lambda_MV)) - x[i,2]
}
plot(excess[,1],excess[,2],xlab="Bornes des classes",ylab="",main="Excess of loss de la loi Pareto"
}

```

```

#Loi Lognormale de parametres mu et sigmaCarre, calcule l'excess-of-loss pour differents seuils d
#on utilise les parametres obtenus par le MV
EoL_lognormale<-function(x)
{
mu_MV<-EMV_lognormale(x)[1]
sigma_MV<-EMV_lognormale(x)[2]
n<-nrow(x)
v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
excess<-matrix(data=0, nrow=n, ncol=2)
excess[,1]<-x[,2] # on charge les seuils d=0, 1000, 2000...
for (i in 1:n){
acc = 0
for (j in i:n){
acc = acc + v[j]*( plnorm(x[j,3],mu_MV,sigma_MV) - plnorm(x[j,2],mu_MV,sigma_MV) )
}
excess[i,2]<-acc/(1-plnorm(x[i,2],mu_MV,sigma_MV)) - x[i,2]
}
plot(excess[,1],excess[,2],xlab="Bornes des classes",ylab="",main="Excess of loss de la loi Lognorm
}

```

```

#Loi Burr de parametres de forme alpha et tau et de parametre d'echelle lambda, calcule l'excess-of-
#on utilise les parametres obtenus par le MV
EoL_burr<-function(x)
{
alpha_MV<-EMV_burr(x)[1]
lambda_MV<-EMV_burr(x)[2]
tau_MV<-EMV_burr(x)[3]
n<-nrow(x)
v<-(x[1:n,2]+x[1:n,3])/2 #chargement ds vect du milieu des classes
excess<-matrix(data=0, nrow=n, ncol=2)
excess[,1]<-x[,2] # on charge les seuils d=0, 1000, 2000...
for (i in 1:n){
acc = 0
for (j in i:n){
acc = acc + v[j]*( pburr(x[j,3],alpha_MV,tau_MV,lambda_MV) - pburr(x[j,2],alpha_MV,tau_MV,lambda_MV)
}
excess[i,2]<-acc/(1-pburr(x[i,2],alpha_MV,tau_MV,lambda_MV)) - x[i,2]
}
plot(excess[,1],excess[,2],xlab="Bornes des classes",ylab="",main="Excess of loss de la loi Burr XI
}

```

```
#####
##### PARTIE : TEST D'ADEQUATION POUR DONNEES GROUPEES #####
#####
```

```
#Test d'adequation pour la loi Exp de parametre Beta
Adeq_exp<-function(x)
{
beta_MV<-EMV_exp(x)
#seuil du test alpha
alpha<-0.05
#Calcul de la statistique du test
E<-sum(x[,4])*(pexp(x[,3],beta_MV) - pexp(x[,2],beta_MV))
Q<-sum(((x[,4]-E)^2)/E)

#nombre de degre de liberte = 23-1-1=21
p_valeur<-1-pchisq(Q,21)
if (p_valeur < alpha){
cat("on rejette Ho, modele Exponentielle pas bien adapte, p-valeur trop faible= ", p_valeur ,"\n")
} else {
cat("Modele Exponentielle adequat aux donnees donc accepte, p-valeur = ", p_valeur ," \n")
}
}
}
```

```
#Test d'adequation pour la loi Gamma de parametres alpha et beta
Adeq_gamma<-function(x)
{
alpha_MV<-EMV_gamma(x)[1]
beta_MV<-EMV_gamma(x)[2]
#seuil du test alpha
alpha<-0.05
#Calcul de la statistique du test
E<-sum(x[,4])*(pgamma(x[,3],alpha_MV,beta_MV) - pgamma(x[,2],alpha_MV,beta_MV))
Q<-sum(((x[,4]-E)^2)/E)
```

```
#nombre de degre de liberte = 23-2-1=20
p_valeur<-1-pchisq(Q,20)
if (p_valeur < alpha){
cat("on rejette Ho, modele Gamma pas bien adapte, p-valeur trop faible= ", p_valeur ," \n")
} else {
cat("Modele Gamma adequat aux donnees donc accepte, p-valeur = ", p_valeur ," \n")
}
}
```

```
#Test d'adequation pour la loi Pareto de parametres alpha et lambda
Adeq_pareto<-function(x)
{
alpha_MV<-EMV_pareto(x)[1]
```

```

lambda_MV<-EMV_pareto(x)[2]
#seuil du test alpha
alpha<-0.05
#Calcul de la statistique du test
E<-sum(x[,4])*(ppareto(x[,3],alpha_MV,lambda_MV) - ppareto(x[,2],alpha_MV,lambda_MV))
Q<-sum(((x[,4]-E)^2)/E)

#nombre de degre de liberte = 23-2-1=20
p_valeur<-1-pchisq(Q,20)
if (p_valeur < alpha){
cat("on rejette Ho, modele Pareto pas bien adapte, p-valeur trop faible= ", p_valeur , " \n")
} else {
cat("Modele Pareto adequat aux donnees donc accepte, p-valeur = ", p_valeur , " \n")
}
}

#Test d'adequation pour la loi Lognormale de parametres mu et sigma
Adeq_lognormale<-function(x)
{
mu_MV<-EMV_lognormale(x)[1]
sigma_MV<-EMV_lognormale(x)[2]
#seuil du test alpha
alpha<-0.05
#Calcul de la statistique du test
E<-sum(x[,4])*(plnorm(x[,3],mu_MV,sigma_MV) - plnorm(x[,2],mu_MV,sigma_MV))
Q<-sum(((x[,4]-E)^2)/E)

#nombre de degre de liberte = 23-2-1=20
p_valeur<-1-pchisq(Q,20)
if (p_valeur < alpha){
cat("on rejette Ho, modele Lognormale pas bien adapte, p-valeur trop faible= ", p_valeur , " \n")
} else {
cat("Modele Lognormale adequat aux donnees donc accepte, p-valeur = ", p_valeur , " \n")
}
}

#Test d'adequation pour la loi Burr XII de parametres alpha, tau et lambda
Adeq_burr<-function(x)
{
alpha_MV<-EMV_burr(x)[1]
lambda_MV<-EMV_burr(x)[2]
tau_MV<-EMV_burr(x)[3]
#seuil du test alpha
alpha<-0.05
#Calcul de la statistique du test
E<-sum(x[,4])*(pburr(x[,3],alpha_MV,tau_MV,lambda_MV) - pburr(x[,2],alpha_MV,tau_MV,lambda_MV))
Q<-sum(((x[,4]-E)^2)/E)

```

```

#nombre de degre de liberte = 23-2-1=20
p_valeur<-1-pchisq(Q,20)
if (p_valeur < alpha){
cat("on rejette Ho, modele de Burr XII pas bien adapte, p-valeur trop faible= ", p_valeur ," \n")
} else {
cat("Modele Burr XII adequat aux donnees donc accepte, p-valeur = ", p_valeur ,"\n")
}
}

#####
##### PARTIE : SELECTION DE MODELE #####
#####

#Le but ici est de selectionner entre le modele Burr XII et Lognormale
selectMod<-function(x)
{
seuil<-0.05

log_vraisemblance_burr<-function(x){
r<-nrow(x) #nombre de classes
alpha<-EMV_burr(x)[1]
lambda<-EMV_burr(x)[2]
tau<-EMV_burr(x)[3]
somme=0
for (i in 1:r){
#avec les notations du poly pour la loi Burr
somme<-somme+x[i,4]*log( pburr(x[i,3],alpha,tau,lambda) - pburr(x[i,2],alpha,tau,lambda) )
}
return(somme)
}

log_vraisemblance_lognormale<-function(x){
r<-nrow(x) #nombre de classes
mu<-EMV_lognormale(x)[1]
sigma<-EMV_lognormale(x)[2]
somme=0
for (i in 1:r){
somme<-somme+x[i,4]*log( plnorm(x[i,3],mu,sigma) - plnorm(x[i,2],mu,sigma) )
}
return(somme)
}
#Calcul de la statistique du test
R<-2*(log_vraisemblance_burr(x) - log_vraisemblance_lognormale(x))
w<-qchisq(1-seuil,1)
if (R > w) {
cat("On rejette le modele Lognormale \n")
} else {
cat("On rejette le modele Burr XII \n")
}
}

```

14.3 Question 5

Voici le code R de toutes les fonctions qui nous ont servi dans l'exercice 5 :

```
#####
#####      Question 5      #####
#####

####  Lecture du fichier  ####
tab=read.table("donnee_question5.txt",header=T,sep="\t")

####  Définition de variable qui servent pour toutes les lois  ####

nb_sinistres=sum(tab[,1]*tab[,2])
nb_contrats=sum(tab[,2])
nk=tab[,2]
somme_carre=sum((tab[,1])^2*tab[,2])

# esperance empirique et deuxième moment empirique des donnees
E=nb_sinistres/nb_contrats
m2=somme_carre/nb_contrats

# définition de la factorielle
factoriel=c()
factoriel[1]=1
fac=1
for (i in 1:20){
  fac=fac*i
  factoriel[i+1]=fac
}

#### Definition d'un bouleen  ####
#bool_fichier vaut 0 si on écrase tout ce qu'il y avait dans le fichier res_question5 pour réécrire
# sinon on écrit à la suite de qu'il y a déjà avant

#####
#####      Fonction qui lance toutes les fonctions et ecrit les resultats dans le fichier resQ5.txt
#####
q5<-function(){
  pois_Q=pois(0)
  bin_Q=bin(1)
  pois_m_Q=pois_m(1)
  bin_m_Q=bin_m(1)

#####  selection de la loi  #####
write("",file="resQ5.txt",append=TRUE)
write("",file="resQ5.txt",append=TRUE)
write("Selection de la Loi",file="resQ5.txt",append=TRUE)
```

```

minimum=min(pois_Q, bin_Q, pois_m_Q, bin_m_Q)

if(pois_Q==minimum){
cat("Poisson","\n")
write("La loi selectionnee est la loi de Poisson",file="resQ5.txt",append=TRUE)
}else if(bin_Q==minimum){
cat("binomiale negative","\n")
write("La loi selectionnee est la loi Binomiale negative",file="resQ5.txt",append=TRUE)
}else if(pois_m_Q==minimum){
cat("poisson modifiee","\n")
write("La loi selectionnee est la loi de Poisson modifiée a 0",file="resQ5.txt",append=TRUE)
}else if(bin_m_Q==minimum){
cat("binomiale negative modifiee","\n")
write("La loi selectionnee est la loi Binomiale negative modifiee a 0",file="resQ5.txt",append=TRUE)
}
}

```

```

#####
#####      Graph de la fonction empirique      #####
#####
Fn<-function(){
plot(nk/nb_contrats, type = "h", col = "red", lwd=2, main="graphique en bâtons de la fonction de masse
de probabilité empirique")
}

```

```

#####
#####      Loi de Poisson      #####
#####
pois<-function(bool_fichier){
if (bool_fichier==0){
write("Loi_de_Poisson",file="resQ5.txt")
}else{
write("",file="resQ5.txt",append=TRUE)
write("",file="resQ5.txt",append=TRUE)
write("Loi_de_Poisson",file="resQ5.txt",append=TRUE)
}
lambda=E
write(rbind("lambda", "=", lambda),file="resQ5.txt",append=TRUE, ncolumns=3)

```

```

#### Graphique en batons ####
y=dpois(c(0:20),lambda)
plot(y, type = "h", col = "red", lwd=2, main="graphique en bâtons de la fonction de masse
de probabilité de la loi de Poisson")
points(nk/nb_contrats, type = "h", col = "black", lwd=1)

```

```

#### Test d'adequation ####
PT=chisq.test(nk,p=y)

```

```

Q=sum((nk-nb_contrats*y)^2/(nb_contrats*y))
write(rbind("Q", "=", Q), file="resQ5.txt", append=TRUE, ncolumns=3)
#cat(1-pchisq(Q, 21))
return(Q)
}

#####
#####      Loi Binomiale négative      #####
#####
bin<-function(bool_fichier){
if(bool_fichier==0){
write("Loi_Binomiale_negative", file="resQ5.txt")
}else{
write("", file="resQ5.txt", append=TRUE)
write("", file="resQ5.txt", append=TRUE)
write("Loi_Binomiale_negative", file="resQ5.txt", append=TRUE)
}
}

####  Méthode des Moments  ####
betaMM=(somme_carre/nb_contrats-E^2)/E-1
rMM=E/betaMM
write(c("rMM", "=", rMM, "betaMM", "=", betaMM), file="resQ5.txt", append=TRUE, ncolumns=3)
Fd=c()
Fd[1]=gamma(rMM+0)/(gamma(rMM)*1)*(1/(1+betaMM))^rMM*(betaMM/(1+betaMM))^0
for (i in 1:20){
Fd[i+1]=gamma(rMM+i)/(gamma(rMM)*fac)*(1/(1+betaMM))^rMM*(betaMM/(1+betaMM))^i
}

####  Méthode du maximum de vraisemblance  ####
l <- function(x) {
r <- x[1]
beta <- x[2]
-sum(nk*log(gamma(r+c(0:20)))-nk*log(gamma(r))-nk*log(factoriel)
-r*nk*log(1+beta)+c(0:20)*nk*log(beta/(1+beta)))
}
o=optim(c(rMM, betaMM), l)
rMV=o$par[1]
betaMV=o$par[2]
write(c("rMV", "=", o$par[1], "betaMV", "=", o$par[2]), file="resQ5.txt", append=TRUE, ncolumns=3)

Fd2=c()
for (i in 0:20){
Fd2[i+1]=gamma(rMV+i)/(gamma(rMV)*factoriel[i+1])*(1/(1+betaMV))^rMV*(betaMV/(1+betaMV))^i
}

####  Graphique en batons  ####

```

```
plot(Fd2, type = "h", col = "red", lwd=2, main="graphique en bâtons de la fonction de masse
de probabilité de la loi Binomiale Négative")
points(nk/nb_contrats, type = "h", col = "black", lwd=1)
```

```
#### Test d'adequation ####
Q=sum((nk-nb_contrats*Fd2)^2/(nb_contrats*Fd2))
write(rbind("Q", "=", Q), file="resQ5.txt", append=TRUE, ncolumns=3)
#cat(1-pchisq(Q, 21))
return(Q)
}
```

```
#####
#####      Loi de Poisson modifiée à 0      #####
#####
pois_m<-function(bool_fichier){
if (bool_fichier==0){
write("Loi_de_Poisson_modifiee_a_0", file="resQ5.txt")
}else{
write("", file="resQ5.txt", append=TRUE)
write("", file="resQ5.txt", append=TRUE)
write("Loi_de_Poisson_modifiee_a_0", file="resQ5.txt", append=TRUE)
}
}
```

```
#### Méthode des moments ####
lambdaMM=m2/E-1
alphaMM=1-(E/lambdaMM)*(1-exp(-lambdaMM))
write(rbind("lambdaMM", "=", lambdaMM, "alphaMM", "=", alphaMM), file="resQ5.txt", append=TRUE, ncolumns=3)
```

```
#### Méthode du maximum de vraisemblance ####
alphaMV=tab[1,2]/nb_contrats
l <- function(x) {
alpha<-x[1]
lambda<-x[2]
-(nk[1]*log(alpha)+(nb_contrats-nk[1])*log(1-alpha)
+sum(nk[2:21]*(c(1:20)*log(lambda)-lambda-log(factoriel[2:21])-log(1-exp(-lambda))))))
}
o=optim(c(alphaMM, lambdaMM), l)
alpha2MV=o$par[1]
lambdaMV=o$par[2]
write(c("alphaMV", "=", alphaMV, "alpha2MV", "=", o$par[1], "lambdaMV", "=", o$par[2]), file="resQ5.txt", app
```

```
#### Graphique en batons ####
y=c()
```

```

y[1]=alpha2MV
y[2:21]=(1-alphaMV)/(1-exp(-lambdaMV))*dpois(c(1:20),lambdaMV)
plot(y, type = "h", col = "red", lwd=2, main="graphique en bâtons de la fonction de masse
de probabilité de la loi de Poisson modifiée a 0")
points(nk/nb_contrats, type = "h", col = "black", lwd=1)

#### Test d'adequation ####
#PT=chisq.test(nk,p=y)
Q=sum((nk-nb_contrats*y)^2/(nb_contrats*y))
write(rbind("Q","=",Q),file="resQ5.txt",append=TRUE, ncolumns=3)
return(Q)
}

#####
##### Loi Binomiale négative modifiée à 0 #####
#####
bin_m<-function(bool_fichier){
if(bool_fichier==0){
write("Loi_Binomiale_negative_modifiee_a_0",file="resQ5.txt")
}else{
write("",file="resQ5.txt",append=TRUE)
write("",file="resQ5.txt",append=TRUE)
write("Loi_Binomiale_negative_modifiee_a_0",file="resQ5.txt",append=TRUE)
}

m3=sum((tab[,1])^3*tab[,2])/nb_contrats

#a=(m2-E)*(2*E+m2)
#b=3*(m2^2-E^2)+E-m3
#c=2*m2+2*E-m3
#delta=b^2-4*a*c
#x=c((-b-(delta)^(.5))/(2*a),(-b+(delta)^(.5))/(2*a))
#rMM=(-b+(delta)^(.5))/(2*a)

rMM=(2*m2^2-E*(m2+m3))/(2*E^2-3*E*m2+E*m3+E-m2)

betaMM=(m2/E-1)/(rMM+1)
alphaMM=1-E/(rMM*betaMM)*(1-(1+betaMM)^(-rMM))
write(c("alphaMM","=",alphaMM,"rMM","=",rMM,"betaMM","=",betaMM),file="resQ5.txt",append=TRUE,ncolun

#### Maximum de vraisemblance ####
l <- function(x) {
alpha<-x[1]
r<-x[2]
beta<-x[3]
-(nk[1]*log(alpha)+(nb_contrats-nk[1])*log(1-alpha)+sum(nk[2:21]*(log(gamma(r+c(1:20))))
-log(gamma(r))-log(factoriel[2:21])-r*log(1+beta)+c(1:20)*log(beta)-c(1:20)*log(1+beta)-log(1-(1+be

```

```

}
o=optim(c(alphaMM,rMM,betaMM), 1)
alphaMV=o$par[1]
rMV=o$par[2]
betaMV=o$par[3]
write(c("alphaMV", "=", o$par[1], "rMV", "=", o$par[2], "betaMV", "=", o$par[3]), file="resQ5.txt", append=TRUE)

#### Graphique en batons ####
Fd=c()
Fd[1]=alphaMV
for (i in 1:20){
Fd[i+1]=(1-alphaMV)/(1-(1+betaMV)^(-rMV))*gamma(rMV+i)/(gamma(rMV)*factoriel[i+1])*(1/(1+betaMV))^(rMV+i)
}
plot(Fd, type = "h", col = "red", lwd=2, main="graphique en bâtons de la fonction de masse
de probabilité de la loi Binomiale Négative modifiée a 0")
points(nk/nb_contrats, type = "h", col = "black", lwd=1)

#### Test d'adequation ####
Q=sum((nk-nb_contrats*Fd)^2/(nb_contrats*Fd))
write(rbind("Q", "=", Q), file="resQ5.txt", append=TRUE, ncolumns=3)
return(Q)

}

```

Ci-dessous le fichier de réponse de la question 5 :

```

Loi_de_Poisson
lambda = 1.67872835357974
Q = 10400078.4984484

```

```

Loi_Binomiale_negative
rMM = 1.13003685468967
betaMM = 1.48555186197069
rMV = 0.997378140680519
betaMV = 1.68291292581295
Q = 50.9637283262316

```

```

Loi_de_Poisson_modifiee_a_0
lambdaMM = 3.16428021555042
alphaMM = 0.491887209536304
alphaMV = 0.388214008787800
alpha2MV = 0.388217012177196
lambdaMV = 2.5242733780895
Q = 62560.0011742003

```

```

Loi_Binomiale_negative_modifiee_a_0
alphaMM = 0.315467760390042
rMM = 0.546394659556197
betaMM = 2.04623069278999
alphaMV = 0.388194442438583
rMV = 1.91888038522401
betaMV = 1.07898594256429
Q = 15.6691823101408

```

```

Selection de la Loi
La loi selectionnee est la loi Binomiale negative modifiee a 0

```

14.4 Question 6

Voici le code R de toutes les fonctions qui nous ont servi dans l'exercice 6 :

```

tab=read.table("donnee_question6.txt",header=T,sep="\t")
donnee=tab[,2]
nb=length(donnee)
E=mean(donnee)
var=(nb-1)/nb*var(donnee)

#Les données sont classées
Fn=c()
for (i in 1:nb){
  j=i
  while(j<nb && donnee[j]==donnee[j+1]){
    j=j+1
  }
  Fn[i]=j/nb
}
graph_fn<-function(){
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main = "graphique de la fonction empirique")
}

#####
#####      Fonction qui lance toutes les lois et ecrit les resultats dans le fichier resQ2.txt
#####

q6<-function(){
loi_gamma(0)
loi_lnorm(1)
loi_pareto(1)
loi_burr(1)
loi_pareto_gen(1)
}

#####

```

```

#####              Loi Gamma              #####
#####

loi_gamma<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Gamma",file="resQ6.txt")
}else{
write("",file="resQ6.txt",append=TRUE)
write("",file="resQ6.txt",append=TRUE)
write("Loi Gamma",file="resQ6.txt",append=TRUE)
}

#### Methode des Moments ####
alphaMM=E^2/V
betaMM=E/V
write(c("alphaMM", "=", alphaMM, "betaMM", "=", betaMM),file="resQ6.txt",append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance ####
l<-function(x){
alpha<-x[1]
beta<-x[2]
-(nb*(alpha*log(beta)-log(gamma(alpha)))+(alpha-1)*sum(log(donnee))-beta*sum(donnee))
}
o=optim(c(alphaMM,betaMM),l)
alpha=o$par[1]
beta=o$par[2]
write(c("alphaMV", "=", o$par[1], "betaMV", "=", o$par[2]),file="resQ6.txt",append=TRUE,ncolumns=3)

####  adequation graphique ####
#La courbe de Fn (fonction empirique est tracée en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi gamma")

#La courbe de répartition de la loi gamma est tracée en rouge
points(donnee,pgamma(donnee, alpha, beta, lower.tail = TRUE,log.p = FALSE),col="red", type = "l", l

#### Test d'adequation ####
diff1=abs(Fn-pgamma(donnee,alpha,beta))
diff2=c()
diff2[1]=abs(pgamma(donnee[1],alpha,beta))
diff2[2:nb]=abs(Fn[1:(nb-1)]-pgamma(donnee[2:nb],alpha,beta))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn),file="resQ6.txt",append=TRUE,ncolumns=3)
write(c("c_{alpha}", "=", calpha),file="resQ6.txt",append=TRUE,ncolumns=3)

```

```

if (Dn > calpha){
write("On rejette le modele",file="resQ6.txt",append=TRUE)
}else{
write("On ne rejette pas le modele",file="resQ6.txt",append=TRUE)
}

}

#####
#####          Loi Lognormale          #####
#####
loi_lnorm<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Lognormale",file="resQ6.txt")
}else{
write("",file="resQ6.txt",append=TRUE)
write("",file="resQ6.txt",append=TRUE)
write("Loi Lognormale",file="resQ6.txt",append=TRUE)
}

#### Methode des Moments ####
muMM=log(E)-1/2*log(V/(E^2)+1)
sigmaMM=sqrt(log(V/(E^2)+1))
write(c("muMM","=",muMM,"sigmaMM","=",sigmaMM),file="resQ6.txt",append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance ####
l<-function(x){
mu<-x[1]
sigma<-x[2]
sum(log(donnee))+nb*(log(2*sqrt(pi))+log(sigma))+sum((log(donnee)-mu)^2/(2*sigma))
}
o=optim(c(muMM,sigmaMM),l)
muMV=o$par[1]
sigmaMV=o$par[2]
write(c("muMV","=",o$par[1],"sigmaMV","=",o$par[2]),file="resQ6.txt",append=TRUE,ncolumns=3)

####  adequation graphique  ####
#La courbe de Fn (fonction empirique est tracée en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Lognormale")

#La courbe  de répartition de la loi Lognormale est tracée en rouge
points(donnee,plnorm(donnee, muMV, sigmaMV, lower.tail = TRUE,log.p = FALSE),col="red", type = "l",

#### Test d'adequation ####
diff1=abs(Fn-plnorm(donnee,muMV,sigmaMV))
diff2=c()
diff2[1]=abs(plnorm(donnee[1],muMV,sigmaMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-plnorm(donnee[2:nb],muMV,sigmaMV))
max1=max(diff1)

```

```

max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn), file="resQ6.txt", append=TRUE, ncolumns=3)
write(c("c_{alpha}", "=", calpha), file="resQ6.txt", append=TRUE, ncolumns=3)

if (Dn > calpha){
write("On rejette le modele", file="resQ6.txt", append=TRUE)
}else{
write("On ne rejette pas le modele", file="resQ6.txt", append=TRUE)
}
}

#####
#####          Loi Pareto          #####
#####
loi_pareto<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Pareto", file="resQ6.txt")
}else{
write("", file="resQ6.txt", append=TRUE)
write("", file="resQ6.txt", append=TRUE)
write("Loi Pareto", file="resQ6.txt", append=TRUE)
}
}

#### Methode des Moments   ####
alphaMM=2*V/(V-E^2)
lambdaMM=E*(alphaMM-1)
write(c("alphaMM", "=", alphaMM, "lambdaMM", "=", lambdaMM), file="resQ6.txt", append=TRUE, ncolumns=3)

#### Methode du maximum de vraisemblance   ####
l<-function(x){
alpha<-x[1]
lambda<-x[2]
-(nb*(log(alpha)+alpha*log(lambda))-(alpha+1)*sum(log(lambda+donnee)))
}
o=optim(c(-alphaMM, -lambdaMM), l)
alphaMV=o$par[1]
lambdaMV=o$par[2]
write(c("alphaMV", "=", o$par[1], "lambdaMV", "=", o$par[2]), file="resQ6.txt", append=TRUE, ncolumns=3)

####  adequation graphique   ####
#La courbe de Fn (fonction empirique est tracée en noire
plot(donnee, Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Pareto")

#La courbe de répartition de la loi Pareto est tracée en rouge
points(donnee, ppareto(donnee, alphaMV, lambdaMV, lower.tail = TRUE, log.p = FALSE), col="red", type =

```

```

#### Test d'adequation ####
diff1=abs(Fn-ppareto(donnee,alphaMV,lambdaMV))
diff2=c()
diff2[1]=abs(ppareto(donnee[1],alphaMV,lambdaMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-ppareto(donnee[2:nb],alphaMV,lambdaMV))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn","=",Dn),file="resQ6.txt",append=TRUE,ncolumns=3)
write(c("c_{alpha}","=",calpha),file="resQ6.txt",append=TRUE,ncolumns=3)

if (Dn > calpha){
write("On rejette le modele",file="resQ6.txt",append=TRUE)
}else{
write("On ne rejette pas le modele",file="resQ6.txt",append=TRUE)
}

}

#####
#####          Loi Burr de type XII          #####
#####
loi_burr<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Burr de type XII",file="resQ6.txt")
}else{
write("",file="resQ6.txt",append=TRUE)
write("",file="resQ6.txt",append=TRUE)
write("Loi Burr de type XII",file="resQ6.txt",append=TRUE)
}

}

#### Methode du maximum de vraisemblance ####
l<-function(x){
a<-x[1]
b<-x[2]
s<-x[3]
-(sum(log(dburr(donnee,a,b,s))))
}
o=optim(c(1.5,1.5,.00015),l)
aMV=o$par[1]
bMV=o$par[2]
sMV=o$par[3]
write(c("aMV","=",o$par[1],"bMV","=",o$par[2],"sMV","=",o$par[3]),file="resQ6.txt",append=TRUE,ncol=3)

####  adequation graphique  ####
#La courbe de Fn (fonction empirique est tracée en noire

```

```
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Burr XII")
```

```
#La courbe de répartition de la loi Burr XII est tracée en rouge
points(donnee,pburr(donnee, aMV, bMV, sMV, lower.tail = TRUE,log.p = FALSE),col="red", type = "l",
```

```
#### Test d'adequation ####
diff1=abs(Fn-pburr(donnee,aMV,bMV,sMV))
diff2=c()
diff2[1]=abs(pburr(donnee[1],aMV,bMV,sMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-pburr(donnee[2:nb],aMV,bMV,sMV))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)
```

```
write(c("Dn", "=", Dn),file="resQ6.txt",append=TRUE,ncolumns=3)
write(c("c_{alpha}", "=", calpha),file="resQ6.txt",append=TRUE,ncolumns=3)
```

```
if (Dn > calpha){
write("On rejette le modele",file="resQ6.txt",append=TRUE)
}else{
write("On ne rejette pas le modele",file="resQ6.txt",append=TRUE)
}
}
```

```
}
```

```
#####
##### Pareto généralisé #####
#####
```

```
loi_pareto_gen<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Pareto généralisée",file="resQ6.txt")
}else{
write("",file="resQ6.txt",append=TRUE)
write("",file="resQ6.txt",append=TRUE)
write("Loi Pareto généralisée",file="resQ6.txt",append=TRUE)
}
}
```

```
#### Methode du maximum de vraisemblance ####
```

```
l<-function(x){
a<-x[1]
b<-x[2]
s<-x[3]
-(sum(log(dgenpareto(donnee,a,b,s))))
}
o=optim(c(1.5,1.5,.00015),l)
aMV=o$par[1]
bMV=o$par[2]
```

```

sMV=o$par[3]
write(c("aMV", "=", o$par[1], "bMV", "=", o$par[1], "sMV", "=", o$par[1]), file="resQ6.txt", append=TRUE, ncol=3)

####  adequation graphique  ####
#La courbe de Fn (fonction empirique est tracée en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi Pareto generalisee")

#La courbe de répartition de la loi pareto generalisee est tracée en rouge
points(donnee,pgenpareto(donnee, aMV, bMV, sMV, lower.tail = TRUE,log.p = FALSE),col="red", type =

#### Test d'adequation  ####
diff1=abs(Fn-pgenpareto(donnee,aMV,bMV,sMV))
diff2=c()
diff2[1]=abs(pgenpareto(donnee[1],aMV,bMV,sMV))
diff2[2:nb]=abs(Fn[1:(nb-1)]-pgenpareto(donnee[2:nb],aMV,bMV,sMV))
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)
calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn), file="resQ6.txt", append=TRUE, ncolumns=3)
write(c("c_{alpha}", "=", calpha), file="resQ6.txt", append=TRUE, ncolumns=3)

if (Dn > calpha){
write("On rejette le modele", file="resQ6.txt", append=TRUE)
}else{
write("On ne rejette pas le modele", file="resQ6.txt", append=TRUE)
}

}

#####
#####          Lognormale-Pareto          #####
#####
loi_lnorm_pareto<-function(bool_fichier){
if (bool_fichier==0){
write("Loi Lognormale-Pareto", file="resQ6.txt")
}else{
write("", file="resQ6.txt", append=TRUE)
write("", file="resQ6.txt", append=TRUE)
write("Loi Lognormale-Pareto", file="resQ6.txt", append=TRUE)
}

#### Methode du maximum de vraisemblance  ####
alpha_tem[1]=nb*(sum(log(donnee/donnee[1])))^(-1)
theta_tem[1]=donnee[1]*prod((donnee/donnee[1])^(0.372238898^2))
m=1
if (theta_tem[m] > donnee[m]){

```

```

if (theta_tem[m] < donnee[m+1]) {
alphaMV=alpha_tem[m]
thetaMV=theta_tem[m]
}
}
m=m+1
bool=nb
while (m<nb){
alpha_tem[m]=(0.372238898^2)*(nb*sum(log(donnee[1:m]))-m*sum(log(donnee)))/(2*(m*sum((log(donnee[1:m])
+sqrt(0.372238898^4*(nb*sum(log(donnee[1:m]))-m*sum(log(donnee)))^2+4*m*nb*0.372238898^2*(m*sum((log
theta_tem[m]=(prod(donnee[1:m]))^(1/m)*exp(nb*0.372238898^2/(m*alpha_tem))
if (theta_tem[m] > donnee[m]){
if (theta_tem[m] < donnee[m+1]) {
alphaMV=alpha_tem[m]
thetaMV=theta_tem[m]
bool=m
}
}
m=m+1
}
y=c()
if (bool < nb){
y[1:bool]=alphaMV*thetaMV^alphaMV/((1+pnorm(0.372238898))*donnee[1:bool]^(alphaMV+1))*exp(-alphaMV^2/(2
y[bool+1:nb]=alphaMV*thetaMV^alphaMV/((1+pnorm(0.372238898))*donnee[1:bool]^(alphaMV+1))
}else{
alphaMV=nb*0.372238898/sqrt(nb*sum((log(donnee))^2)-(sum(log(donnee)))^2)
thetaMV=(prod(donnee))^(1/nb)*exp(0.372238898^2/alphaMV)
y[1:nb]=alphaMV*thetaMV^alphaMV/((1+pnorm(0.372238898))*donnee[1:nb]^(alphaMV+1))*exp(-alphaMV^2/(2
}

write(c("alphaMV", "=", alphaMV, "thetaMV", "=", thetaMV), file="resQ6.txt", append=TRUE, ncolumns=3)

####  adequation graphique  ####
#La courbe de Fn (fonction empirique est tracée en noire
plot(donnee,Fn, type = "l", col = "black", lwd = 1, main="Comparaison de la fonction empirique
avec la fonction répartition de la loi lognormale-pareto")

#La courbe de répartition de la loi lognormale-pareto est tracée en rouge
points(donnee,y,col="red", type = "l", lwd = 1)

####  Test d'adequation  ####
diff1=abs(Fn-y)
diff2=c()
diff2[1]=abs(y[1])
diff2[2:nb]=abs(Fn[1:(nb-1)]-y[2:nb])
max1=max(diff1)
max2=max(diff2)
Dn=max(max1,max2)

```

```

calpha=1.36/sqrt(nb)

write(c("Dn", "=", Dn), file="resQ6.txt", append=TRUE, ncolumns=3)
write(c("c_{alpha}", "=", calpha), file="resQ6.txt", append=TRUE, ncolumns=3)

if (Dn > calpha){
write("On rejette le modele", file="resQ6.txt", append=TRUE)
}else{
write("On ne rejette pas le modele", file="resQ6.txt", append=TRUE)
}

}

```

Ci-dessous le fichier de réponse pour la question 6 :

Loi Gamma

```

alphaMM = 4.53159499872428
betaMM = 0.000178886722069259
alphaMV = 3.77011957000094
betaMV = 0.000148841940706752
Dn = 0.0494720868669304
c_{alpha} = 0.0134898332145886
On rejette le modele

```

Loi Lognormale

```

muMM = 10.0401306513698
sigmaMM = 0.446544763834161
muMV = 10.0014726189440
sigmaMV = 0.127460264168653
Dn = 0.291233099242919
c_{alpha} = 0.0134898332145886
On rejette le modele

```

Loi Pareto

```

alphaMM = -0.566316353013995
lambdaMM = -39678.2459292323
alphaMV = 80154.8585342456
lambdaMV = 2030738553.50695
Dn = 0.275302843540434
c_{alpha} = 0.0134898332145886
On rejette le modele

```

Loi Burr de type XII

```

aMV = 0.961054997941461
bMV = 0.961054997941461
sMV = 0.961054997941461
Dn = 0.0158149826376913
c_{alpha} = 0.0134898332145886

```

On rejette le modele

Loi Pareto généralisée

aMV = 6.64938596565111

bMV = 6.64938596565111

sMV = 6.64938596565111

Dn = 0.00945532702448681

c_{alpha} = 0.0134898332145886

On ne rejette pas le modele

Loi Lognormale-Pareto

alphaMV = 0.737179219191782

thetaMV = Inf

Dn = NaN

c_{alpha} = 0.0134898332145886