

Act-3000 Théorie du risque - Algorithmes récurrents et études de cas

avec Christopher Blier-Wong et Ihsan Chaoubi

Illustrations numériques

Étienne Marceau

École d'actuariat
Université Laval, Québec, Canada

2018-10-10



UNIVERSITÉ
LAVAL

Faculté des sciences et de génie
École d'actuariat

Agenda

1 Contexte

2 Algorithmes récurrents

- Algorithme de DePril
- Algorithme de Panjer
- Illustration no1
- Illustration no2
- Illustration no3

3 Études de cas

- Rappel de la procédure
- Données no1

4 References

Les illustrations de ce document portent sur les notions suivantes :

- Algorithme de Panjer, algorithme de DePril, Erlang et exponentielle
- Études de cas, combinant estimation et algorithmes récursifs

Les notions sont expliquées en détails dans [Marceau, 2013] et [Cossette and Marceau, 2018]

Agenda

1 Contexte

2 Algorithmes rékursifs

- Algorithme de DePril
- Algorithme de Panjer
- Illustration no1
- Illustration no2
- Illustration no3

3 Études de cas

- Rappel de la procédure
- Données no1

4 References

Algorithmes récursifs

Algorithme de DePril

On considère une v.a. X discrète où $X \in \{0, 1h, 2h, \dots\}$ avec

$$f_X(kh) = \Pr(X = kh),$$

pour $k \in \mathbb{N}$.

On définit

$$S_n = X_1 + \dots + X_n,$$

où les v.a. X_1, \dots, X_n sont i.i.d. et se comportent comme la v.a. X ($X_i \sim X, i = 1, 2, \dots, n$).

Algorithme de DePril :

- Point de départ:

$$f_{S_n}(0) = f_X(0)^n.$$

- Relation récursive:

$$f_{S_n}(kh) = \frac{1}{f_X(0)} \sum_{j=1}^k \left((n+1) \frac{j}{k} - 1 \right) f_X(jh) f_{S_n}((k-j)h)$$

pour $k \in \mathbb{N}^+$.

Algorithmes récurrents

Algorithme de Panjer

Soit la v.a. X définie selon l'approche fréquence sévérité

$$X = \begin{cases} \sum_{i=1}^M B_i & , \quad M > 0 \\ 0, & , \quad M = 0 \end{cases}$$

avec les hypothèses suivantes :

- $\underline{B} = \{B_i, i \in \mathbb{N}^+\}$;
- $B_i \sim B \in A_h = \{0, 1h, 2h, \dots\}$;
- \underline{B} et M sont indépendantes ;
- fonction de masse de probabilité de B :

$$\Pr(B = hj) = f_B(hj),$$

pour $j \in \mathbb{N}$ avec un pas de discrétisation $h > 0$;

Algorithmes récursifs

Algorithme de Panjer

L'algorithme de Panjer s'applique à la condition que la loi de M fasse partie de la classe $(a,b,0)$ dont la fonction de masse de probabilité satisfait la relation récursive suivante :

$$f_M(k) = \left(a + \frac{b}{k}\right) f_M(k-1),$$

pour $k \in \mathbb{N}^+$

Seules les lois Poisson, Binomiale et Binomiale Négative sont membres de cette famille.

Algorithmes récursifs

Algorithme de Panjer

Les valeurs de a et b pour les membres de la famille $(a, b, 0)$ sont les suivantes :

- loi de Poisson: $a = 0$ et $b = \lambda$;
- loi binomiale négative (1ère paramétrisation): $a = 1 - q$ et $b = (1 - q)(r - 1)$;
- loi binomiale: $a = -\frac{q}{1-q}$ et $b = (n + 1) \frac{q}{1-q}$.

Algorithme de Panjer – Forme générale

- Point de départ:

$$f_X(0) = \Pr(X = 0) = \mathcal{P}_M(f_B(0)).$$

- Relation récurrente:

$$f_X(hk) = \frac{\sum_{j=1}^k \left(a + \frac{bj}{k}\right) f_B(hj) f_X(h(k-j))}{1 - af_B(0)},$$

pour $k \in \mathbb{N}^+$

Loi Poisson :

$$N \sim \text{Pois}(\lambda).$$

■ **Point de départ :**

$$f_X(0) = \Pr(X = 0) = e^{-\lambda(1-f_B(0))}.$$

■ **Relation récursive:**

$$f_X(hk) = \frac{\lambda}{k} \sum_{j=1}^k (j) f_B(hj) f_X(h(k-j)),$$

pour $k \in \mathbb{N}^+$

Loi Binomiale Négative (1^{ère} paramétrisation):

$$N \sim BNeg(r, q).$$

■ **Point de départ:**

$$f_X(0) = \left(\frac{q}{1 - (1 - q) f_B(0)} \right)^r,$$

■ **Relation récurrente:**

$$f_X(kh) = \frac{\sum_{j=1}^k \left(1 - q + \frac{(1-q)(r-1)j}{k} \right) f_B(jh) f_X((k-j)h)}{1 - (1 - q) f_B(0)},$$

pour $k \in \mathbb{N}^+$

Loi Binomiale :

$$N \sim \text{Binom}(n, q).$$

■ Point de départ:

$$f_X(0) = \Pr(X = 0) = (1 - q + qf_B(0))^n$$

■ Relation récursive:

$$\begin{aligned} f_X(hk) &= \frac{\sum_{j=1}^k \left(\frac{q}{q-1} + \frac{(n+1)qj}{(1-q)k} \right) f_B(j) f_X(k-j)}{1 + \frac{q}{1-q} f_B(0)} \\ &= \frac{\sum_{j=1}^k \left(-q + \frac{(n+1)qj}{k} \right) f_B(hj) f_X(h(k-j))}{1 - q + qf_B(0)}, \end{aligned}$$

pour $k \in \mathbb{N}^+$

Algorithmes récurrents

Illustration no1

Soit la v.a. X avec

$$\mathcal{L}_X(t) = \mathcal{P}_M(\mathcal{L}_B(t)), \quad t \geq 0,$$

avec

- $B \sim LNorm(\mu, \sigma)$, $\mu = \ln(10) - 0.32$ et $\sigma = 0.8$;
- $\mathcal{P}_M(r) = \alpha \exp(\lambda_1(r-1)) + (1-\alpha) \exp(\lambda_2(r-1))$, $r \in [0,1]$,
 $\alpha = 0.8$, $\lambda_1 = 1$, $\alpha = 0.8$, $\lambda_2 = 6$.

On déduit :

- Espérance de X :

$$E[X] = 20$$

- Variance de X :

$$Var(X) = 267.269$$

Algorithmes récursifs

Illustration no1

Soit les v.a. indépendantes $K_i \sim \text{Pois}(\lambda_i)$, $i = 1, 2$, avec

$$P_{K_i}(r) = \exp(\lambda_i(r-1)), \quad r \in [0, 1], \quad i = 1, 2.$$

Soit les v.a. indépendantes $Y_i \sim \text{PoisComp}(\lambda_i, F_B)$, $i = 1, 2$, avec

$$\mathcal{L}_{Y_i}(t) = \mathcal{P}_{K_i}(\mathcal{L}_B(t)), \quad t \geq 0, \quad i = 1, 2.$$

On déduit :

■ fgp de M :

$$\mathcal{P}_M(r) = \alpha \mathcal{P}_{K_1}(r) + (1 - \alpha) \mathcal{P}_{K_2}(r), \quad r \in [0, 1] ;$$

■ TLS de X :

$$\begin{aligned} \mathcal{L}_X(t) &= \alpha \mathcal{P}_{K_1}(\mathcal{L}_B(t)) + (1 - \alpha) \mathcal{P}_{K_2}(\mathcal{L}_B(t)) \\ &= \alpha \mathcal{L}_{Y_1}(t) + (1 - \alpha) \mathcal{L}_{Y_2}(t), \quad t \geq 0 ; \end{aligned}$$

■ F_X :

$$F_X(x) = \alpha F_{Y_1}(x) + (1 - \alpha) F_{Y_2}(x)$$

Algorithmes récurrents

Illustration no1

On applique les outils suivants pour évaluer approximativement $F_X(x)$:

- discrétisation *upper* et *lower* ($h = 1, 0.1$) ;
- algorithme de Panjer.

Algorithmes récursifs

Illustration no1

Soit les v.a. discrètes $\tilde{B}^{(up,h)}$ et $\tilde{B}^{(low,h)}$ résultant de l'approximation par discrétisation de la distribution de la v.a. continue B

Soit les v.a. correspondantes $\tilde{Y}_1^{(up,h)}$, $\tilde{Y}_1^{(low,h)}$, $\tilde{Y}_2^{(up,h)}$, $\tilde{Y}_2^{(low,h)}$, $\tilde{X}^{(up,h)}$ et $\tilde{X}^{(low,h)}$.

Algorithmes récursifs

Illustration no1

Étapes pour l'évaluation des approximations de F_X :

- 1 Discrétisation de la v.a. $B \Rightarrow f_{\tilde{B}(up,h)}$ et $f_{\tilde{B}(low,h)}$;
- 2 Algo de Panjer : calcul des valeurs de $f_{\tilde{Y}_i^{(up,h)}}(kh)$ et $f_{\tilde{Y}_i^{(low,h)}}(kh)$, $k \in \{0,1,\dots,k_0\}$, $i = 1,2$;
- 3 Calcul des valeurs de $f_{\tilde{X}(up,h)}(kh)$ et $f_{\tilde{X}(low,h)}(kh)$, $k \in \{0,1,\dots,k_0\}$, avec

$$f_{\tilde{X}(up,h)}(kh) = \alpha \times f_{\tilde{Y}_1^{(up,h)}}(kh) + (1 - \alpha) f_{\tilde{Y}_2^{(up,h)}}(kh),$$

et

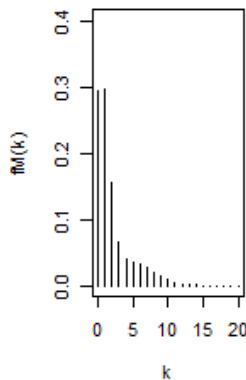
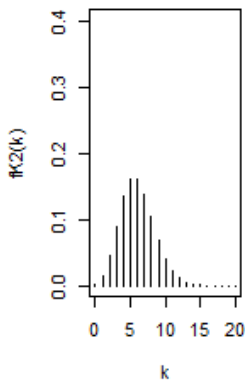
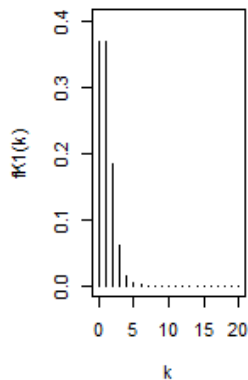
$$f_{\tilde{X}(low,h)}(kh) = \alpha \times f_{\tilde{Y}_1^{(low,h)}}(kh) + (1 - \alpha) f_{\tilde{Y}_2^{(low,h)}}(kh),$$

pour $k \in \{0,1,\dots,k_0\}$,

Algorithmes récursifs

Illustration no1

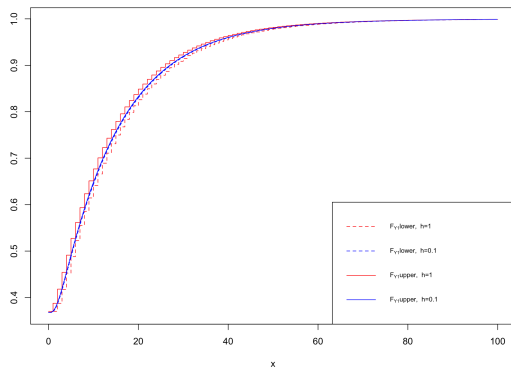
Valeurs des fonctions de masse de probabilité f_{K_1} , f_{K_2} , et f_M :



Algorithmes récursifs

Illustration no1

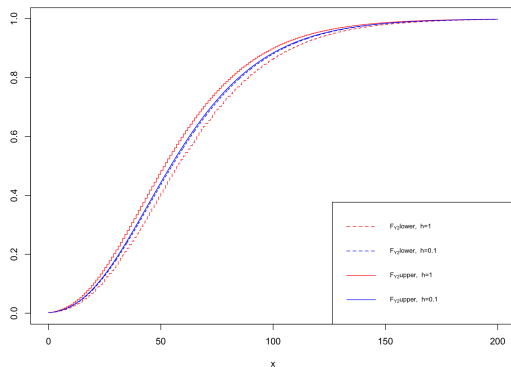
Valeurs de $F_{\widehat{Y}_1}^{(up,h)}$ et $F_{\widehat{Y}_1}^{(low,h)}$, $h = 1, 0.1$:



Algorithmes récursifs

Illustration no1

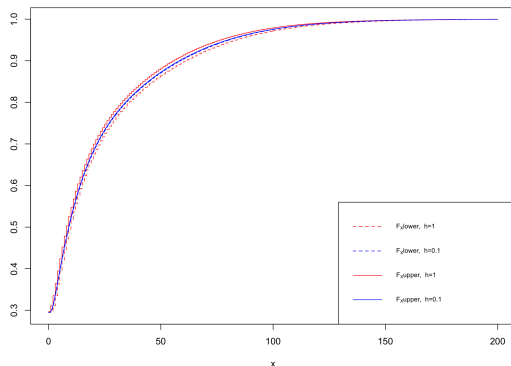
Valeurs de $F_{\widehat{Y}_2^{(up,h)}}$ et $F_{\widehat{Y}_2^{(low,h)}}$, $h = 1, 0.1$:



Algorithmes récursifs

Illustration no1

Valeurs de $F_{\tilde{X}(up,h)}$ et $F_{\tilde{X}(low,h)}$, $h = 1, 0.1$:



Algorithmes récursifs

Illustration no2

Soit les v.a. indépendantes X_1 et X_2 avec

$$\mathcal{L}_{X_i}(t) = \mathcal{P}_{M_i}(\mathcal{L}_{B_i}(t)), t \geq 0,$$

avec

- $B_i \sim \text{Exp}(\beta_i)$, $i = 1, 2$, $\beta_1 = \frac{1}{10}$ et $\beta_2 = \frac{1}{2}$ (note : $\beta_2 > \beta_1$) ;
- $\mathcal{P}_{M_i}(r) = \exp(\lambda_i(r-1))$, $r \in [0, 1]$, $\lambda_1 = 2$, $\lambda_2 = 10$.

On définit

$$S = X_1 + X_2.$$

On déduit :

- Espérance de X_1 et X_2 : $E[X_1] = 20$, $E[X_2] = 20$.
- Variance de X_1 et X_2 : $\text{Var}(X_1) = 600$, $\text{Var}(X_2) = 120$.
- Espérance de S : $E[S] = 40$.
- Variance de S : $\text{Var}(S) = 720$.

Objectif : Évaluer F_S (et les mesures de risque associée à S)

Stratégie :

1 Étape 1: Démontrer que

$$F_S(x) = \gamma_0 + \sum_{k=1}^{\infty} \gamma_k H(x; k, \beta_2), \quad x \geq 0;$$

2 Étape 2: Utiliser l'algorithme de Panjer pour évaluer γ_k , $k \in \{0, 1, 2, \dots, k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

3 Étape 3: On évalue $F_S(x)$, avec γ_k , $k \in \{0, 1, 2, \dots, k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

Étape 1.

La TLS de S est donnée par

$$\mathcal{L}_S(t) = \mathcal{L}_{X_1}(t) \times \mathcal{L}_{X_2}(t) = \mathcal{P}_N(\mathcal{L}_C(t)), t \geq 0,$$

où

- fgp de N :

$$\mathcal{P}_N = \exp(\lambda_N(r-1)), r \in [0,1] ;$$

- $\lambda_N = \lambda_1 + \lambda_2 = 12$;

- TLS de C

$$\begin{aligned}\mathcal{L}_C(t) &= p_1 \mathcal{L}_{B_1}(t) + p_2 \mathcal{L}_{B_2}(t) \\ &= p_1 \left(\frac{\beta_1}{\beta_1 + t} \right) + p_2 \left(\frac{\beta_2}{\beta_2 + t} \right), t \geq 0;\end{aligned}$$

Algorithmes récursifs

Illustration no2

On réarrange les termes de $\mathcal{L}_C(t)$.

La TLS de la v.a. C est

$$\mathcal{L}_C(t) = \alpha \left(\frac{\beta_1}{\beta_1 + t} \right) + (1 - \alpha) \left(\frac{\beta_2}{\beta_2 + t} \right). \quad (1)$$

Dans (1), on a

$$\left(\frac{\beta_1}{\beta_1 + t} \right) = q \left(\frac{\beta_2}{\beta_2 + t} \right) \frac{1}{1 - (1 - q) \left(\frac{\beta_2}{\beta_2 + t} \right)}$$

où $q = \frac{\beta_1}{\beta_2}$.

Algorithmes récursifs

Illustration no2

On introduit la v.a. discrète J (avec support \mathbb{N}^+) avec

$$\mathcal{P}_J(r) = qr \frac{1}{1 - (1 - q)r}, \quad r \in [0, 1].$$

On observe

$$\begin{aligned} \mathcal{P}_J(r) &= qr \frac{1}{1 - (1 - q)r} \\ &= r \sum_{k=0}^{\infty} q(1 - q)^k \times r^k \end{aligned}$$

avec $q = \frac{\beta_1}{\beta_2} \in (0, 1)$.

Algorithmes récursifs

Illustration no2

Ainsi, dans (1), on remplace

$$\left(\frac{\beta_1}{\beta_1 + t} \right)$$

par

$$\left(\frac{\beta_1}{\beta_1 + t} \right) = \mathcal{P}_J \left(\frac{\beta_2}{\beta_2 + t} \right) = \mathcal{P}_J (\mathcal{L}_D (t)),$$

où

$$\mathcal{L}_D (t) = \left(\frac{\beta_2}{\beta_2 + t} \right).$$

Alors, (1) devient

$$\begin{aligned}\mathcal{L}_C(t) &= \alpha \left(\frac{\beta_1}{\beta_1 + t} \right) + (1 - \alpha) \left(\frac{\beta_2}{\beta_2 + t} \right) \\ &= \alpha \mathcal{P}_J \left(\frac{\beta_2}{\beta_2 + t} \right) + (1 - \alpha) \left(\frac{\beta_2}{\beta_2 + t} \right).\end{aligned}\quad (2)$$

On introduit une v.a. discrète K sur le support $\mathbb{N}^+ = \{1, 2, \dots\}$ dont la f.g.p. est

$$\mathcal{P}_K(r) = \alpha \mathcal{P}_J(r) + (1 - \alpha) \times r, \quad (3)$$

pour $r \in [0, 1]$. [Note : $\Pr(K = 0) = 0$.]

La fgp de la v.a. K est

$$\begin{aligned}\mathcal{P}_K(r) &= \sum_{k=0}^{\infty} \eta_k \times r^k \\ &= \alpha \mathcal{P}_J(r) + (1 - \alpha) \times r \\ &= \alpha \sum_{k=0}^{\infty} f_J(k) \times r^k + (1 - \alpha) \times r\end{aligned}\tag{4}$$

où

$$f_J(k) = \begin{cases} 0 & , \quad k = 0 \\ q(1 - q)^{k-1} & , \quad k \in \mathbb{N}^+ \end{cases} .\tag{5}$$

Algorithmes récursifs

Illustration no2

En combinant (4) et (3), on déduit que

$$f_K(k) = \eta_k = \begin{cases} 0 & , \quad k = 0 \\ \alpha \times q + (1 - \alpha) & , \quad k = 1 \\ \alpha \times q (1 - q)^{k-1} & , \quad k = 2, 3, \dots \end{cases}.$$

Avec (2) et (3), et puisque $\left(\frac{\beta_2}{\beta_2 + t}\right) \in [0, 1]$ pour $t \geq 0$, on conclut que

$$\mathcal{L}_C(t) = \mathcal{P}_K\left(\frac{\beta_2}{\beta_2 + t}\right)$$

Clairement,

$$\begin{aligned} \sum_{k=1}^{\infty} \eta_k &= \alpha \sum_{k=1}^{\infty} f_J(k) + (1 - \alpha) \\ &= \alpha \times 1 + (1 - \alpha) = 1. \end{aligned}$$

Algorithmes récurrents

Illustration no2

Maintenant, on revient à la TLS de S

$$\mathcal{L}_S(t) = \mathcal{L}_{X_1}(t) \times \mathcal{L}_{X_2}(t) = \mathcal{P}_N(\mathcal{L}_C(t)), t \geq 0,$$

où

- fgp de N :

$$\mathcal{P}_N = \exp(\lambda_N(r-1)), r \in [0,1] ;$$

- $\lambda_N = \lambda_1 + \lambda_2 = 12$;
- TLS de C

$$\begin{aligned}\mathcal{L}_C(t) &= p_1 \mathcal{L}_{B_1}(t) + p_2 \mathcal{L}_{B_2}(t) \\ &= p_1 \left(\frac{\beta_1}{\beta_1 + t} \right) + p_2 \left(\frac{\beta_2}{\beta_2 + t} \right) \\ &= \mathcal{P}_K \left(\frac{\beta_2}{\beta_2 + t} \right), t \geq 0;\end{aligned}$$

Alors, la TLS de S devient

$$\begin{aligned}\mathcal{L}_S(t) &= \mathcal{L}_{X_1}(t) \times \mathcal{L}_{X_2}(t) \\ &= \mathcal{P}_N(\mathcal{L}_C(t)) \\ &= \mathcal{P}_N\left(\mathcal{P}_K\left(\frac{\beta_2}{\beta_2 + t}\right)\right), t \geq 0,\end{aligned}\tag{6}$$

On introduit la v.a. discrète L dont la fonction de masse de probabilité et la fgp sont respectivement

$$\Pr(L = k) = \gamma_k, k \in \mathbb{N},$$

et

$$\mathcal{P}_L(r) = \mathcal{P}_N(\mathcal{P}_K(r)) = \sum_{k=0}^{\infty} \gamma_k r^k.$$

Algorithmes récurrents

Illustration no2

Puisque $N \sim \text{Pois}(\lambda_N)$, on utilise l'algorithme de Panjer pour calculer les valeurs de γ_k , $k \in \{0, 1, 2, \dots, k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

En combinant (6) et (7), on obtient

$$\begin{aligned}\mathcal{L}_S(t) &= \mathcal{P}_N\left(\mathcal{P}_K\left(\frac{\beta_2}{\beta_2+t}\right)\right) \\ &= \mathcal{P}_L\left(\frac{\beta_2}{\beta_2+t}\right) \\ &= \sum_{k=0}^{\infty} \gamma_k \left(\frac{\beta_2}{\beta_2+t}\right)^k, \quad t \geq 0.\end{aligned}\tag{8}$$

De la TLS de S en (8), on déduit

$$F_S(x) = \gamma_0 + \sum_{k=1}^{\infty} \gamma_k H(x; k, \beta_2), \quad x \geq 0.$$

Étape 2.

On applique l'algorithme de Panjer pour calculer γ_k , $k \in \{0, 1, 2, \dots, k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

k	0	1	5	10	20
γ_k	6.144212×10^{-6}	0.000026543	0.000348132	0.001599818	0.007537266

Étape 3.

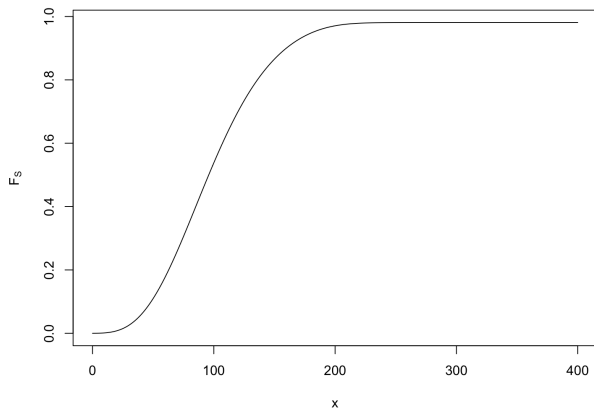
On évalue $F_S(x)$, avec γ_k , $k \in \{0,1,2,\dots,k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

x	0	5	10	20	50
$F_S(x)$	6.144212×10^{-6}	0.00026746	0.00125063	0.00788859	0.10987205

Algorithmes récursifs

Illustration no2

Valeurs de F_S :



Algorithmes récursifs

Illustration no3

Soit les v.a. indépendantes $X_1 \sim \text{Gamma}(\alpha_1, \beta_1)$ et $X_2 \sim \text{Gamma}(\alpha_2, \beta_2)$ avec

$$\mathcal{L}_{X_i}(t) = \left(\frac{\beta_i}{\beta_i + t} \right)^{\alpha_i}, \quad t \geq 0,$$

avec $\beta_2 > \beta_1 > 0$.

On définit la v.a. $S = X_1 + X_2$.

Algorithmes récurrents

Illustration no3

Objectif : Évaluer $F_S(x)$, $x \geq 0$.

Stratégie :

1 Étape 1 : Transformer $\left(\frac{\beta_i}{\beta_i+t}\right)^{\alpha_i}$ adéquatement.

2 Étape 2 : Démontrer

$$F_S(x) = \sum_{k=0}^{\infty} \gamma_k H(x; \alpha_1 + \alpha_2 + k; \beta_2), \quad x \geq 0.$$

3 Étape 3 : Évaluer $F_S(x)$, avec γ_k , $k \in \{0,1,2,\dots,k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

Étape 1.

La TLS de S est donnée par

$$\begin{aligned}\mathcal{L}_S(t) &= \mathcal{L}_{X_1}(t) \times \mathcal{L}_{X_2}(t) \\ &= \left(\frac{\beta_1}{\beta_1 + t}\right)^{\alpha_1} \times \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_2}, \quad t \geq 0.\end{aligned}\tag{9}$$

Dans (9), on a

$$\mathcal{L}_{X_1}(t) = \left(\frac{\beta_1}{\beta_1 + t}\right)^{\alpha_1} = \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_1} \left(\frac{q}{1 - (1 - q)\left(\frac{\beta_2}{\beta_2 + t}\right)}\right)^{\alpha_1}$$

où $q = \frac{\beta_1}{\beta_2} \in (0,1)$.

On introduit la v.a. discrète J (avec support \mathbb{N}^+) avec

$$\begin{aligned}\mathcal{P}_J(r) &= \left(\frac{q}{1 - (1 - q)r} \right)^{\alpha_1} \\ &= \sum_{\kappa=0}^{\infty} \gamma_{\kappa} r^{\kappa}, \quad r \in [0, 1].\end{aligned}$$

On reconnaît la fgp de la loi binomiale négative de paramètres α_1 et $q = \frac{\beta_1}{\beta_2}$

$$f_J(k) = \gamma_k = \frac{\Gamma(\alpha_1 + k)}{\Gamma(\alpha_1)k!} q^{\alpha_1} (1 - q)^k$$

pour $k \in \mathbb{N}$.

La TLS de X_1 devient

$$\begin{aligned}\mathcal{L}_{X_1}(t) &= \left(\frac{\beta_1}{\beta_1 + t}\right)^{\alpha_1} \\ &= \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_1} \mathcal{P}_J\left(\frac{\beta_2}{\beta_2 + t}\right) \\ &= \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_1} \sum_{k=0}^{\infty} \gamma_k \left(\frac{\beta_2}{\beta_2 + t}\right)^k,\end{aligned}$$

où $q = \frac{\beta_1}{\beta_2} \in (0,1)$.

Étape 2.

Alors, l'expression en (9) de la TLS de la v.a. S devient

$$\begin{aligned}\mathcal{L}_S(t) &= \left(\frac{\beta_1}{\beta_1 + t}\right)^{\alpha_1} \times \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_2} \\ &= \left(\left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_1} \sum_{k=0}^{\infty} \gamma_k \left(\frac{\beta_2}{\beta_2 + t}\right)^k\right) \times \left(\frac{\beta_2}{\beta_2 + t}\right)^{\alpha_2}.\end{aligned}$$

Algorithmes récursifs

Illustration no3

On réarrange les termes

$$\mathcal{L}_S(t) = \sum_{k=0}^{\infty} \gamma_k \left(\frac{\beta_2}{\beta_2 + t} \right)^{\alpha_1 + \alpha_2 + k}.$$

On déduit que

$$F_S(x) = \sum_{k=0}^{\infty} \gamma_k H(x; \alpha_1 + \alpha_2 + k, \beta_2)$$

ou

$$f_S(x) = \sum_{k=0}^{\infty} \gamma_k h(x; \alpha_1 + \alpha_2 + k, \beta_2),$$

pour $x \geq 0$.

Étape 3.

On évalue $F_S(x)$, avec γ_k , $k \in \{0,1,2,\dots,k_0\}$, où k_0 est fixé de telle sorte que $1 - \sum_{k=1}^{\infty} \gamma_k \leq \varepsilon$ pour un ε fixé très petit (e.g., $\varepsilon = 10^{-10}$).

Hypothèses de calculs : $\alpha_1 = 1.2$, $\alpha_2 = 4.5$, $\beta_1 = \frac{1.2}{10}$ et $\beta_2 = \frac{4.5}{30}$.

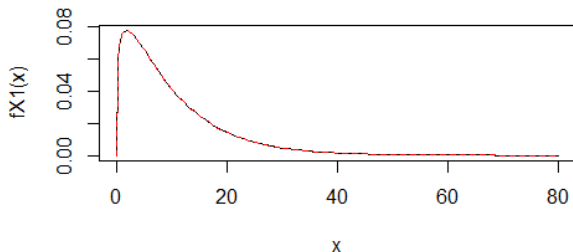
Valeurs numériques :

- $E[S] = 40$, $q = 0.8$
- $\gamma_k, (k = 0,1,2,3)$: 0.765082000; 0.183619680; 0.040396330; 0.008617884
- $F_S(x)$, $(x = 40,80)$: 0.5564092; 0.9767901; 0.9995224 (valeurs calculées en R avec $k_0 = 1000$)

Algorithmes récursifs

Illustration no3

Valeurs de $f_{X_1}(x) = h(x; \alpha_1, \beta_1)$ et $f_{X_1} = \sum_{k=0}^{\infty} \gamma_k h(x; \alpha_1 + k, \beta_2)$:

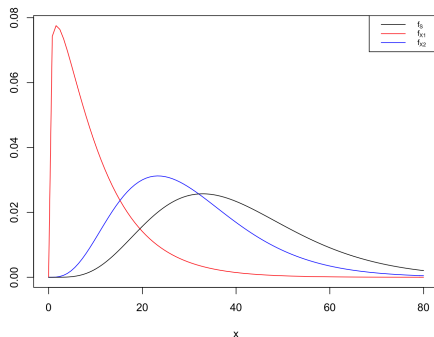


Comme prévu, les deux courbes se superposent parfaitement.

Algorithmes récursifs

Illustration no3

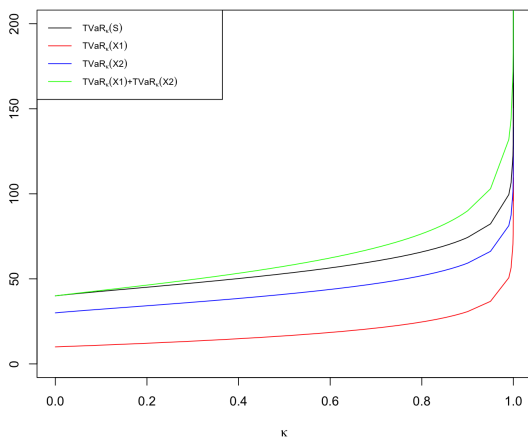
Valeurs de f_{X_1} , f_{X_2} et f_S :



Algorithmes récursifs

Illustration no3

Valeurs de $TVaR_{\kappa}(X_1)$, $TVaR_{\kappa}(X_2)$, $TVaR_{\kappa}(S)$ et $TVaR_{\kappa}(X_1) + TVaR_{\kappa}(X_2)$:



Agenda

1 Contexte

2 Algorithmes récurrents

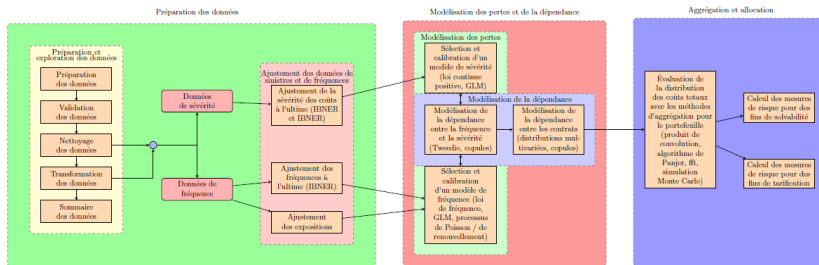
- Algorithme de DePril
- Algorithme de Panjer
- Illustration no1
- Illustration no2
- Illustration no3

3 Études de cas

- Rappel de la procédure
- Données no1

4 References

Illustration de la procédure

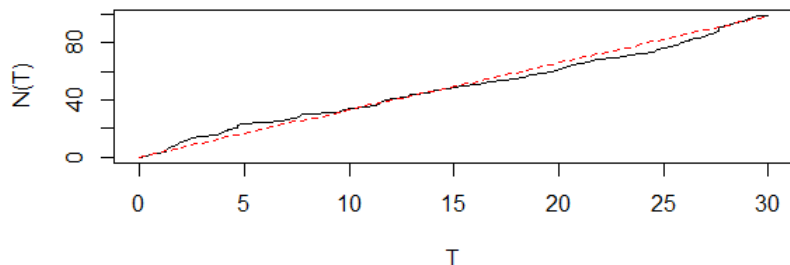


Contexte :

- Montants complets de sinistres
- Temps d'occurrence
- Période d'observation = $(0,30]$
- 99 observations : $(x_i, t_i), i = 1, 2, \dots, 99$

Parcours du processus de comptage vs intensité cumulée pour un processus de Poisson

Parcours du processus de comptage

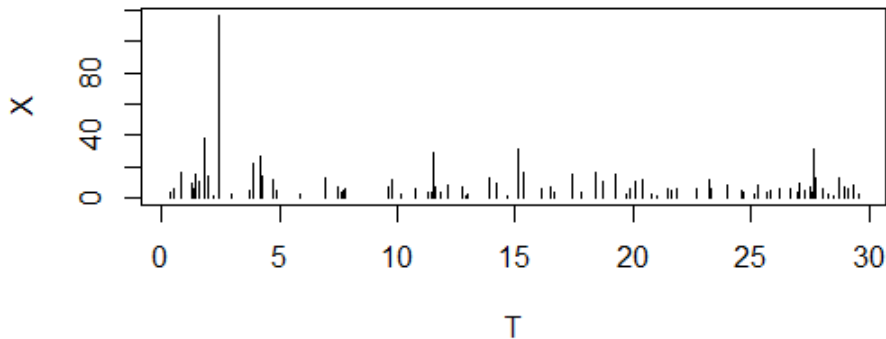


Intensité cumulée du processus de Poisson homogène

Études de cas

Données no1

Montants de sinistres vs temps d'occurrence

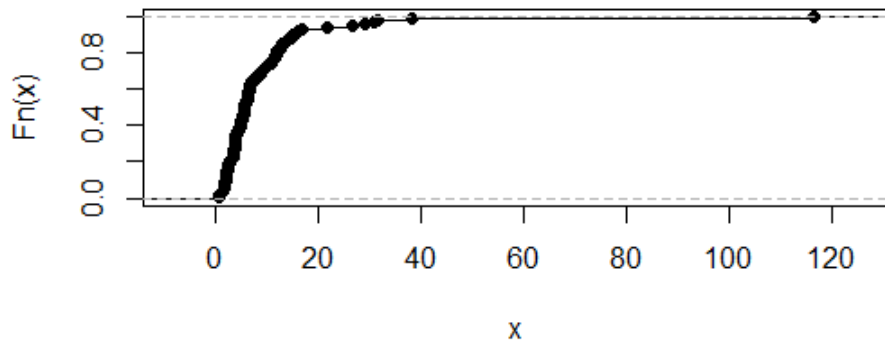


Études de cas

Données no1

Fonction de répartition empirique - Sinistres

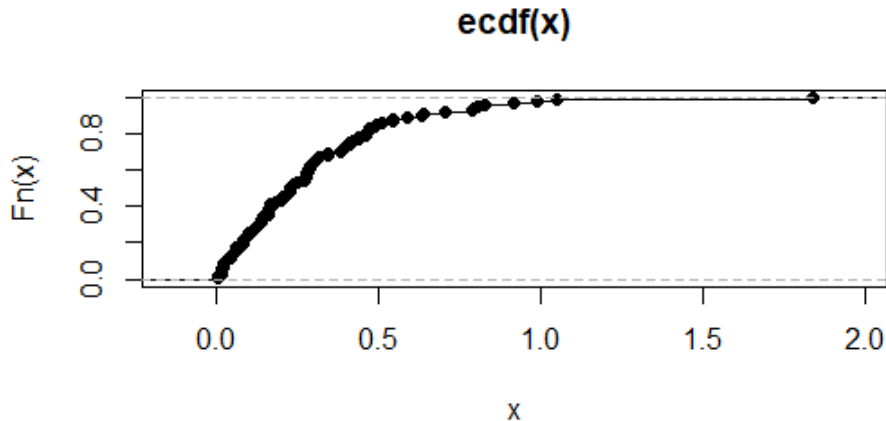
ecdf(x)



Études de cas

Données no1

Fonction de répartition empirique - Temps inter-inistres



Fonctions d'excès moyen: [Embrechts and Schmidli, 1994]

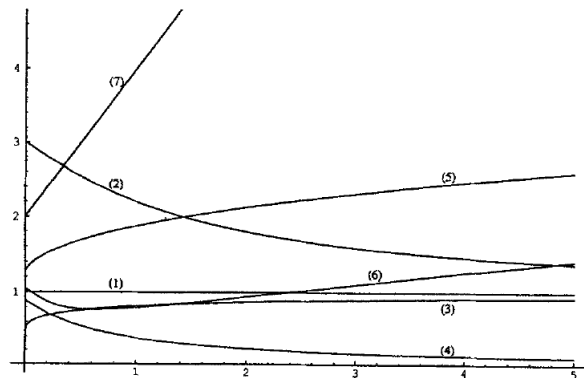
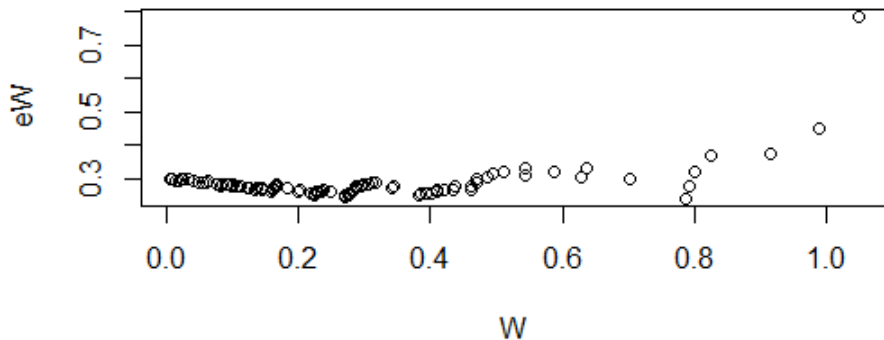


Fig. 2.1. Mean-residual-life function $e(x)$ for a wide class of distributions: (1) exponential (1), (2) gamma (3), (3) gamma (0.5), (4) Weibull (2), (5) Weibull (0.7), (6) lognormal $(-0.2, 1)$ and (7) Pareto (1.5)

Études de cas

Données no1

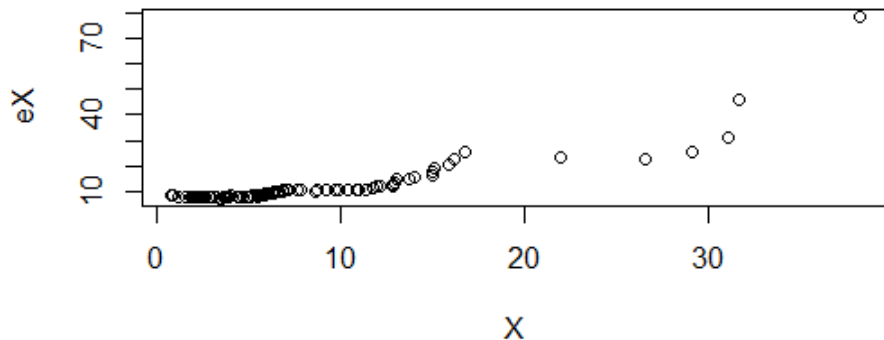
Fonctions d'excès moyen empirique : temps inter-sinistres



Études de cas

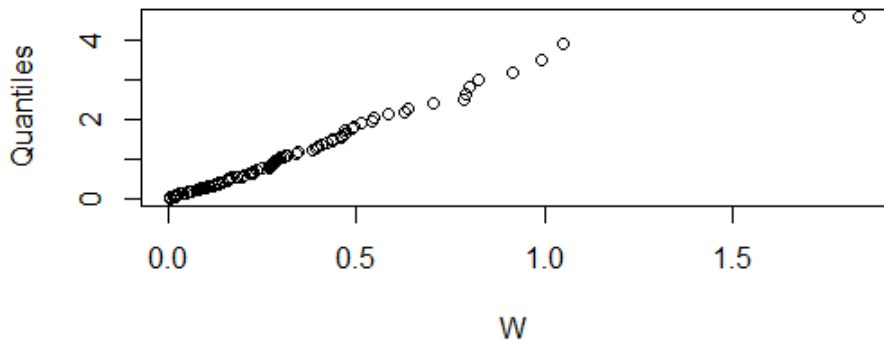
Données no1

Fonctions d'excès moyen empirique : montants de sinistres



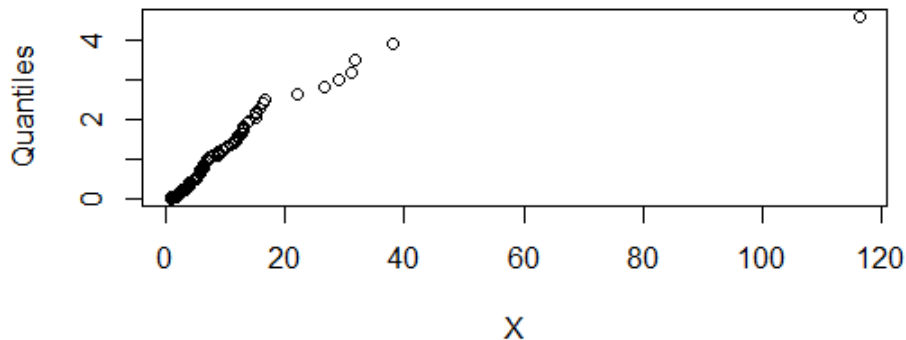
QQ-plot - Loi exponentielle : temps inter-sinistres

QQ-plot - Loi Exponentielle



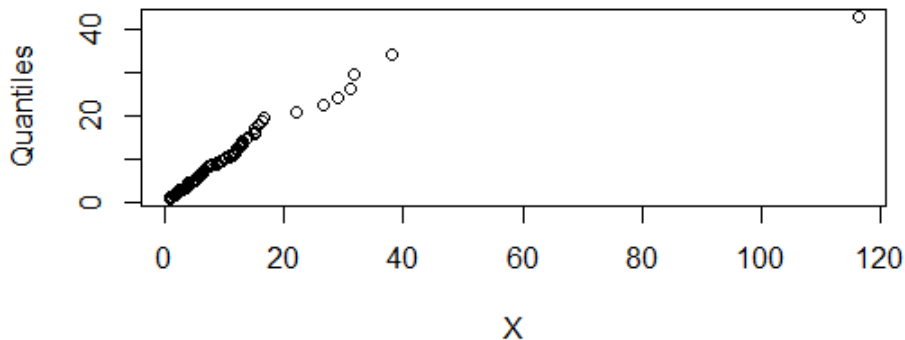
QQ-plot - Loi exponentielle : montants de sinistres

QQ-plot - Loi Exponentielle



QQ-plot - Loi lognormale : montants de sinistres

QQ-plot - Loi Lognormale



Estimation MV - montant de sinistre X :

- Loi lognormale : $X \sim LNorm(\mu, \sigma)$
- $\mu = 1.796876, \sigma = 0.8439212$

Estimation MV - processus de comptage \underline{N} :

- Processus de Poisson homogène avec intensité λ
- Temps inter-sinistre : loi exponentielle avec paramètre λ
- $\lambda = \frac{99}{33} = 3.3$

On examine le comportement de l'accroissement $S(30,31]$ du processus de Poisson composé \underline{S}

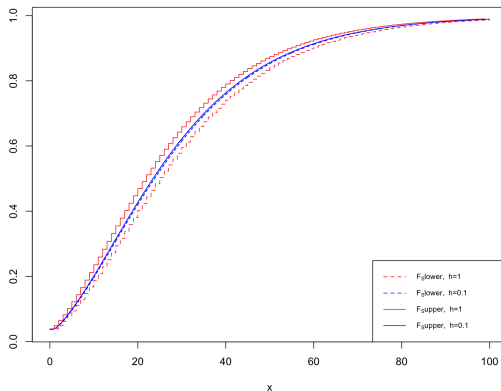
On applique les outils suivants pour évaluer approximativement $F_{S(30,31]}(x)$:

- discrétisation *upper* et *lower* ($h = 1, 0.1$) ;
- algorithme de Panjer.

Études de cas

Données no1

Valeurs de $F_{\widetilde{S(30,31]}}^{(up,h)}$ et $F_{\widetilde{S(30,31]}}^{(low,h)}$, $h = 1, 0.1$:



Agenda

1 Contexte

2 Algorithmes récurrents




- Algorithme de DePril
- Algorithme de Panjer
- Illustration no1
- Illustration no2
- Illustration no3

3 Études de cas

- Rappel de la procédure
- Données no1

4 References

Références I

-  Cossette, H. and Marceau, E. (2018).
Mathématiques actuarielles du risque : modèles, mesures de risque et méthodes quantitatives.
-  Embrechts, P. and Schmidli, H. (1994).
Modelling of extremal events in insurance and finance.
Zeitschrift für Operations Research, 39(1):1–34.
-  Marceau, E. (2013).
Modélisation et évaluation quantitative des risques en actuariat: Modèles sur une période.
Springer.