

Travail pratique 2

Olivier Bourret et Marianne Chouinard

20 décembre 2020

Préambule

Dans l'optique de consolider nos apprentissages du cours de processus stochastiques, il nous est demandé de faire l'analyse de deux cas. L'un des deux met plus l'accent sur la théorie des processus de Poisson et de renouvellement, tandis que l'autre se concentre davantage sur les Martingales et le mouvement Brownien. Pour chaque équipe, les paramètres ainsi que des lois particulières étaient attribuées, afin de travailler chacun sur un choix unique. Pour l'équipe numéro 9, pour la Poisson conditionnelle mixte $N_i(t) \sim \text{Poisson}(\Lambda_i)$, où $\Lambda_i \sim \text{Uniforme}(a, b)$. Voici le reste des paramètres assignés:

Table 1: Paramètres du numéro 1 a) et b)

i	n_i	a_i	b_i	α_i	β_i	M_i	δ_i
1	35	0	4	1	1.5	1500	2%
2	25	0	5	1	2.0	2500	2%
3	40	0	6	1	2.5	1000	2%

Table 2: Paramètres du numéro 1 c)

i	n_i	θ_i	λ_i	α_i	β_i	M_i	δ_i
1	35	2	0.03	1	1.5	1500	2%
2	25	3	0.04	1	2.0	2500	2%
3	40	4	0.05	1	2.5	1000	2%

Table 3: Paramètres du numéro 2

μ	σ	δ	K_1	K_2	T_1	T_2	λ	α	β_i
0.05	0.1	0.02	115	127	1	2	0.1	?	0.5

Note: Les fonctions utilisées en R dont l'algorithme est présenté dans le rapport sont contenues dans le fil du texte. Tout le code utilisé et nécessaire pour ce travail est fourni dans l'annexe.

Question 1

a) Moyenne théorique de la perte actualisée

Pour cette section, il est demandé de trouver la moyenne théorique de la perte actualisée pour une période de longueur t , dont la force d'intérêt est nulle. Il est important de mentionner que puisque la force d'intérêt est de 0%, la formule qui exprime la perte totale sera légèrement modifiée si $N_i(t) > 0$.

$$\begin{aligned} L_i(t) &= \sum_{k=1}^{N_i(t)} e^{-\delta S_{i,k}} M_i X_{i,k} \\ &= \sum_{k=1}^{N_i(t)} e^{-0 \cdot S_{i,k}} M_i X_{i,k} \\ &= \sum_{k=1}^{N_i(t)} M_i X_{i,k} \end{aligned}$$

Ainsi, la solution de la moyenne théorique se simplifie, puisque que $S_{i,k}$ disparaît. On aura alors,

$$\begin{aligned} E[L_{TOT}(t)] &= E[L_1(t) + L_2(t) + \dots + L_{100}(t)] \\ &= E[L_1(t)] + \dots + E[L_{35}(t)] + E[L_{36}(t)] + \dots + E[L_{60}(t)] + E[L_{61}(t)] + \dots + E[L_{100}(t)] \\ &= 35 \cdot E[L_1(t)] + 25 \cdot E[L_{36}(t)] + 40 \cdot E[L_{61}(t)] \end{aligned}$$

Le dernier résultat est possible, car pour $i = 1, \dots, 35, i = 36, \dots, 60, i = 61, \dots, 100$, chacun des i de chaque groupe possèdent la même distribution avec les mêmes paramètres, alors l'espérance demeure la même. Il est donc intéressant de généraliser et trouver $E[L_i(t)]$ peu importe le i .

$$\begin{aligned} E[L_i(t)] &= E[E[L_i(t)|N_i(t)]] \\ &= E[L_i(t)|N_i(t) = k] \cdot Pr(N_i(t) = k) \\ &= E \left[\sum_{k=1}^{N_i(t)} M_i \cdot X_{i,k} | N_i(t) = k \right] \cdot Pr(N_i(t) = k) \\ &= M_i \cdot E \left[\sum_{k=1}^{N_i(t)} X_{i,k} | N_i(t) = k \right] \cdot Pr(N_i(t) = k) \\ &= M_i \cdot (E[X_{i,1}|N_i(t) = k] + \dots + E[X_{i,k}|N_i(t) = k]) \cdot Pr(N_i(t) = k) \\ &= M_i(k \cdot E[X_{i,k}])Pr(N_i(t) = k) \\ \text{où } X_{i,k} &\sim Beta(\alpha = 1, \beta = \beta_i) \Rightarrow E[X_{i,k}] = \frac{1}{1 + \beta_i} \\ &= M_i \left(\frac{1}{1 + \beta_i} \right) Pr(N_i(t) = k) \end{aligned} \tag{1}$$

Il est également possible de trouver $Pr(N_i(t) = k)$

$$\begin{aligned}
Pr(N_i(t) = k) &= \int_0^{b_i} Pr(N_i(t) = k | \Lambda = \lambda) \cdot g_\Lambda(\lambda) d\lambda \\
&= \int_0^{b_i} \frac{(\lambda t)^k \cdot e^{-\lambda t}}{k!} \cdot \frac{1}{b_i} d\lambda \\
&= \frac{1}{b_i t} \int_0^{b_i} \frac{t^{k+1} \lambda^k e^{-t\lambda}}{k!} d\lambda \rightarrow \text{Gamma}(k+1, t) \\
&= \frac{1}{b_i t} \left(1 - \sum_{j=0}^k \frac{(tb_i)^j e^{-tb_i}}{j!} \right)
\end{aligned} \tag{2}$$

En appliquant le résultat de l'équation (2) dans l'équation (1) on retrouve,

$$\begin{aligned}
E[L_i(t)] &= M_i \left(\frac{1}{1 + \beta_i} \right) \frac{1}{b_i t} \left(1 - \sum_{j=0}^k \frac{(tb_i)^j e^{-tb_i}}{j!} \right) \\
&= \frac{M_i \cdot k}{(1 + \beta_i) b_i t} \left(1 - e^{-tb_i} \sum_{j=0}^k \frac{(tb_i)^j}{j!} \right)
\end{aligned} \tag{3}$$

Pour les différents i on aura:

$$\begin{aligned}
\bullet i = 1, \dots, 35 &\Rightarrow \frac{150k}{t} \left(1 - e^{-4t} \sum_{j=0}^k \frac{(4t)^j}{j!} \right) \\
\bullet i = 36, \dots, 100 &\Rightarrow \frac{500k}{3t} \left(1 - e^{-5t} \sum_{j=0}^k \frac{(5t)^j}{j!} \right) \\
\bullet i = 61, \dots, 100 &\Rightarrow \frac{1000k}{21t} \left(1 - e^{-6t} \sum_{j=0}^k \frac{(6t)^j}{j!} \right)
\end{aligned}$$

La perte moyenne théorique sera donc,

$$\begin{aligned}
E[L_{TOT}(t)] &= 35 \cdot E[L_1(t)] + 25 \cdot E[L_{36}(t)] + 40 \cdot E[L_{61}(t)] \\
&= 35 \left[\frac{150k}{t} \left(1 - e^{-4t} \sum_{j=0}^k \frac{(4t)^j}{j!} \right) \right] + 25 \left[\frac{500k}{3t} \left(1 - e^{-5t} \sum_{j=0}^k \frac{(5t)^j}{j!} \right) \right] + 40 \left[\frac{1000k}{21t} \left(1 - e^{-6t} \sum_{j=0}^k \frac{(6t)^j}{j!} \right) \right] \\
&= \frac{5250k}{t} \left(1 - e^{-4t} \sum_{j=0}^k \frac{(4t)^j}{j!} \right) + \frac{12500k}{3t} \left(1 - e^{-5t} \sum_{j=0}^k \frac{(5t)^j}{j!} \right) + \frac{40000k}{21t} \left(1 - e^{-6t} \sum_{j=0}^k \frac{(6t)^j}{j!} \right) \\
&= \frac{5250k}{t} \Gamma(4; k+1, t) + \frac{12500k}{3t} \Gamma(5; k+1, t) + \frac{40000k}{21t} \Gamma(6; k+1, t)
\end{aligned}$$

■

b) Simulation de 1000 pertes avec $\Lambda_i \sim \text{Uniforme}(a_i, b_i)$ où $t = 10$

Pour la simulation, il fallait tout d'abord se questionner sur la méthode utilisée pour simuler la perte totale des 1000 cas. Chaque contrat suit un processus de Poisson mixte où le paramètre Λ_i est une variable aléatoire d'une loi $U(a_i, b_i)$. Ainsi, lorsque Λ_i est connu, il conserve la même valeur pour le reste du processus. Au lieu de simuler le nombre de pertes pour les 10 prochaines années, il est plus simple de simuler chaque perte jusqu'à ce qu'on arrive à 10 ans. En effet, de cette manière, le nombre de pertes est connu ainsi que le moment auquel la perte survient. Cette dernière information est réutilisée dans l'actualisation de la perte $(S_{i,k})$.

Pour connaître le temps entre la $(n - 1)^{\text{ième}}$ et la $n^{\text{ième}}$ perte, T_n , lorsque Λ_i est connu, $T_n \sim \text{Exp}(\lambda)$. Pour ce faire, nous avons créé une fonction qui suit cette algorithme tout simple:

Algorithme

1. Simulation d'un nombre aléatoire d'un λ à partir d'une $\text{Uniforme}(a_1, b_1)$.
2. Création d'une boucle:
 - i. Simulation d'un nombre aléatoire d'une $\text{Exp}(\lambda)$ qui représente T_n .
 - ii. Simulation d'un nombre aléatoire d'une $\text{Beta}(\alpha_1, \beta_1)$ qui représente taux de la perte qui vient de survenir.
 - iii. Somme des T_n préalablement simulés qui donne $S_{i,k}$, soit le temps depuis le début du processus pour avoir n pertes.
 - iv. Si $S_{i,k} \geq 10$ fin de la boucle
3. Somme des pertes actualisées en fonction du temps où la perte est survenue depuis $t = 0, (S_{i,k})$, du taux de la perte $(X_{i,k})$ et du montant associé au produit dérivé (M_i) .

Fin de l'algorithme.

Le code de cette fonction a été généré ainsi:

```
# Fonction de perte des contrats type 1
Simulation_1 <- function(){

  lambda <- runif(1,0,4)
  T_i <- c()
  X_i <- c()
  S_ik <- c()
  perte <- c()
  temps <- 0
  while(temps<10){
    T_i <- append(T_i, rexp(1,rate = lambda)) ## Temps entre 2 pertes
    X_i <- append(X_i,rbeta(1, shape1 = 1, shape2 = 1.5)) ## Taux de la perte
    S_ik <- append(S_ik, sum(T_i)) ## Somme des temps de perte
    temps <- sum(T_i)
  }
  T_i <- T_i[-length(T_i)]
  S_ik <- S_ik[-length(S_ik)]
  X_i <- X_i[-length(X_i)]

  sum(exp(-0.02*S_ik)*1500*X_i)
}
```

Le même processus s'applique pour la création des fonctions des valeurs actualisées des pertes pour les contrats $i = 36, \dots, 60$ et $i = 61, \dots, 100$, mais avec les valeurs des paramètres associés aux contrats i . Il suffit maintenant de réappliquer ces fonctions le nombre de fois nécessaire pour les 100 contrats du portefeuille d'en faire la somme pour avoir la perte totale actualisée. Cette procédure est appliquée 1000 fois, afin d'avoir nos 1000 simulations désirées.

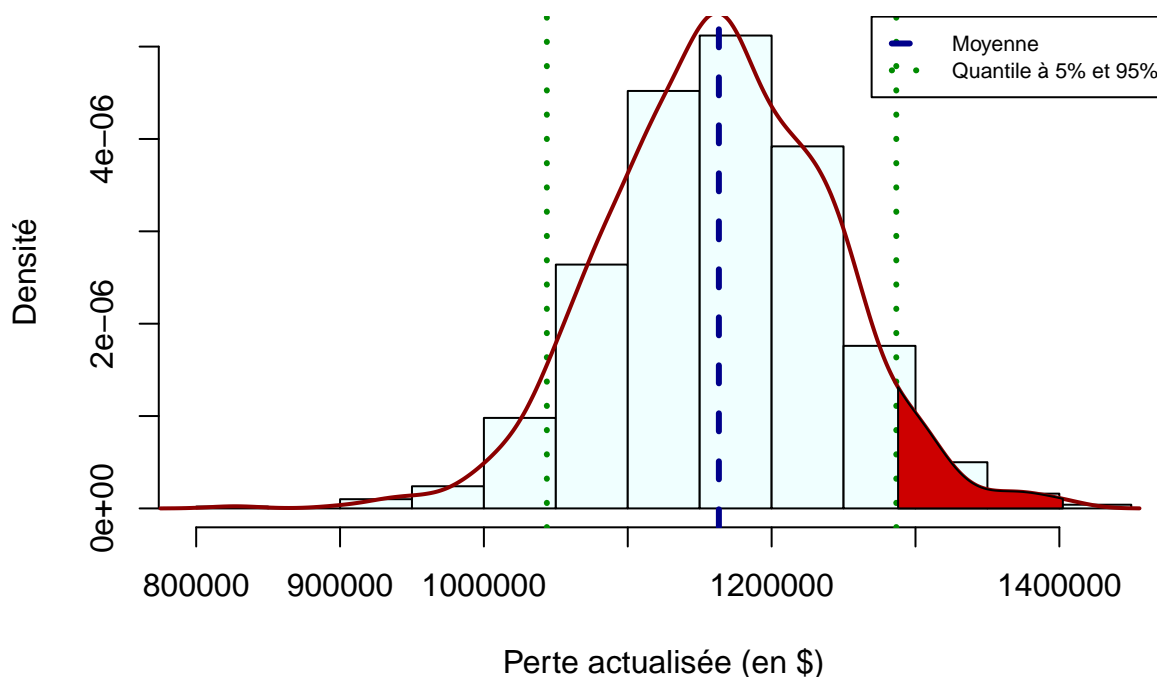
Ces 1000 simulations peuvent donner une idée de la répartition des valeurs possibles. Il est intéressant de connaître certaines statistiques telles que la moyenne, la variance, et l'approximation de la VaR et de la TVaR.

Table 4: Statistiques des 1000 simulations

	Moyenne	Variance	α	$VaR(1 - \alpha)$	$TVaR(1 - \alpha)$
90%	1163338	5867997240	10%	1257301	1295546
95%			5%	1286660	1321652
99%			1%	1345997	1379867

Afin de mieux voir la répartition des données, voici la répartition des pertes totales actualisées pour une longueur de $t = 10$.

Distribution des 1000 simulations des pertes



La distribution des 1000 pertes totales actualisées ressemble à une distribution d'une loi normale de moyenne $\mu = 1163338$ et de variance $\sigma^2 = 5867997240$. Les valeurs approximatives que les pertes peuvent prendre sont entre 830000 et 1400000.

c) Simulation de 1000 pertes avec renouvellement Gamma où $t = 10$

La question c) revient essentiellement au même principe utilisé qu'au numéro 1 b), cependant le temps entre les renouvellements est donné par une loi $\text{Gamma}(\theta_i, \lambda_i)$ dont les paramètres ont été énoncés dans le préambule. Donc, la création des fonctions pour simuler les 1000 pertes utilise pratiquement le même algorithme. Ce qui diffère, c'est que le lambda n'est pas simulé et que T_n est une valeur aléatoire provenant d'un loi Gamma. La fonction pour évaluer la perte d'un contrat où $i = 1, \dots, 35$ sera la suivante:

```
## Fonction pour simuler Gamma type 1
SimGamma_1 <- function(){

  T_i <- c()
  X_i <- c()
  S_ik <- c()
  perte <- c()
  temps <- 0
  while(temps<10){
    T_i <- append(T_i, rgamma(1, shape = 2, rate = 0.03))
    X_i <- append(X_i,rbeta(1, shape1 = 1, shape2 = 1.5))
    S_ik <- append(S_ik, sum(T_i))
    temps <- sum(T_i)
  }
  T_i <- T_i[-length(T_i)]
  S_ik <- S_ik[-length(S_ik)]
  X_i <- X_i[-length(X_i)]

  sum(exp(-0.02*S_ik)*1500*X_i)
}
```

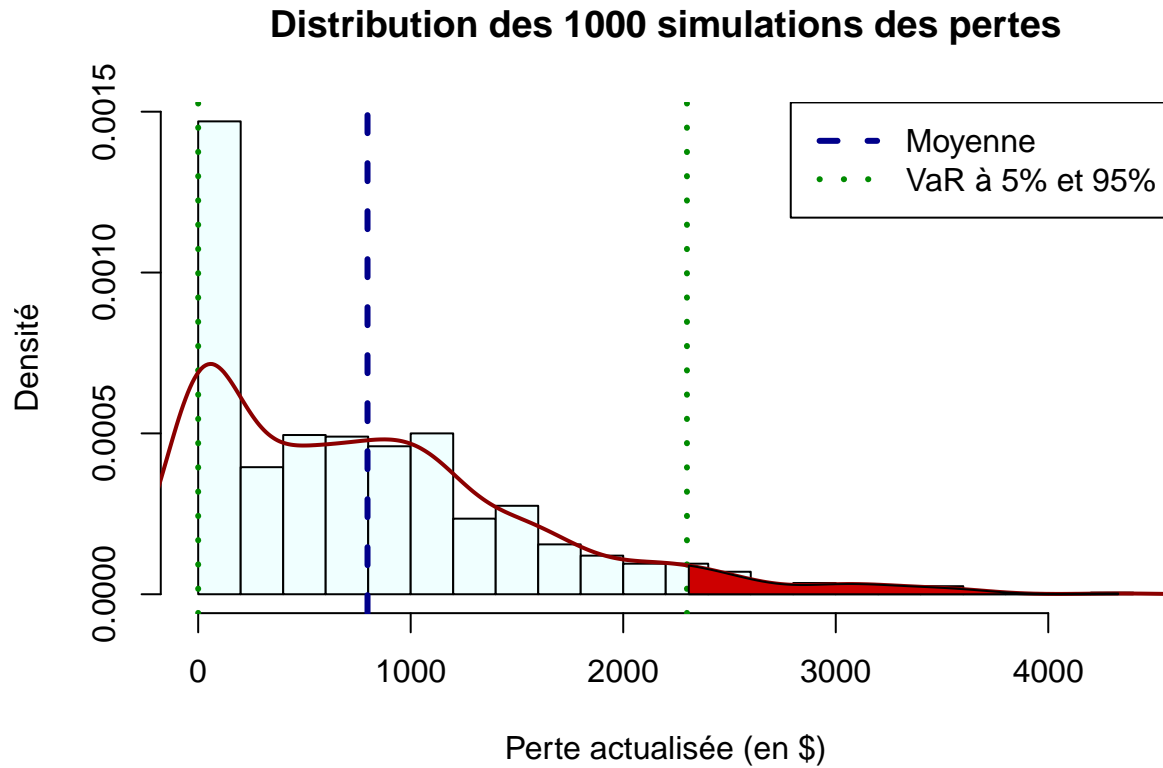
Ensuite, le même processus s'applique pour la création des fonctions des valeurs actualisées des pertes pour les contrats $i = 36, \dots, 60$ et $i = 61, \dots, 100$, mais avec les valeurs des paramètres associés aux contrats i . Ainsi, il reste à appliquer ces fonctions le nombre de fois nécessaire pour simuler les 100 pertes actualisées. En appliquant 1000 fois cette procédure, nous obtenons les valeurs recherchées.

Encore une fois, il est intéressant de regarder les statistiques de cet échantillon. On obtient:

Table 5: Statistiques des 1000 simulations

	Moyenne	Variance	α	$VaR(1 - \alpha)$	$TVaR(1 - \alpha)$
90%	797	592595	10%	1817	2469
95%			5%	2300	2870
99%			1%	3271	3568

Question d'avoir un visuel de la distribution des valeurs obtenues, nous avons fait l'histogramme des 1000 simulations en plus d'y insérer la moyenne et les quantiles à 5% et à 95% et la VaR à 95%.



Comparaison des résultats en b) et en c)

Le premier réflexe qu'il est possible d'avoir, en regardant les résultats c'est de se questionner sur la validité des résultats et à savoir si tout a été fait correctement. En effet, l'énorme différence entre les valeurs simulées en b) et en c) peuvent semer le doute dans notre esprit, mais il faut garder son calme et réfléchir aux valeurs obtenues.

En effet, en b) nous avons des processus de Poisson mixte dont la valeur de λ varie en fonction d'une loi $Uniforme(0, b_i)$. Ainsi, en 10 ans, si l'on a un λ qui peut se situer entre 0 et 6, bien l'espérance du nombre total de pertes peut se retrouver entre 0 et 60. De plus, le temps de renouvellement entre deux évènement suit une $Exp(\lambda)$, alors le temps espéré entre 2 évènements pourrait être dans l'intervalle $(\frac{1}{6}, \infty)$. Donc, en 10 ans, si le temps moyen entre 2 renouvellements se trouve à être $\frac{1}{2}$, alors en 10 ans, il peut tout de même y avoir plusieurs pertes.

Pour la Gamma, en prenant la même logique, l'espérance du temps entre les renouvellements est beaucoup plus grand ($E[T_n] = \frac{\theta_i}{\lambda_i}$). Avec les données fournies à notre équipe, les espérances du temps de renouvellement sont de $\{66.6, 75, 85\}$ pour les trois combinaisons des paramètres fournis. Ainsi, les évènements deviennent très rares dans une période de 10 ans, ce qui fait en sorte que les valeurs totales actualisées des pertes demeurent très basses.

Question 2

La deuxième question, a pour but de nous faire travailler deux concepts vus dans cours, soit les mouvements Brownien et les martingales. On s'intéresse ici à la valeur d'un action ainsi qu'au prix d'option d'achat de deux types d'options.

a) Approximation des prix de l'option européenne et l'option exotique

Tout d'abord, il est proposé dans l'exercice deux options d'achats différents. Il faut en déterminer le prix et pour ce faire, il faut calculer l'espérance de la valeur du gain au temps t actualisée au temps $t = 0$ avec une force d'intérêt sans risque. Dans la situation, la valeur de l'action $S_0 = 100\$$ et sa valeur suit un mouvement Brownien géométrique tel que:

$$S_t = S_0 e^{\mu t + \sigma Z(t)}, \quad \text{où } Z(t) \sim \text{mouvement Brownien standard}$$

Ou encore, pour des sauts d'une période de temps d'une année:

$$S_t = S_{t-1} e^{\mu t + (\sigma Z(t) - \sigma Z(t-1))} = S_{t-1} e^{\mu t + (\sigma Z'(1))}$$

La première option d'achat est dite européenne (*European Call option*) et elle permet à son détenteur de décider au temps T , la date d'échéance de l'option d'acheter l'action à la valeur K prédéterminée au moment de l'action. (**Rappel:** les valeurs K et T qui ont été attribués à l'équipe 9 sont $K_1 = 115\$$, $K_2 = 127\$$, $T_1 = 1$ et $T_2 = 2$.) Le détenteur de l'option d'achat a également le droit de refuser le droit d'exercer son pouvoir d'achat. Donc, si la valeur de l'action est en dessous de la valeur prédéterminée dans l'option d'achat au temps convenu, l'individu est mieux de ne pas compléter la transaction. Ainsi, son gain, qui ne comprend pas le prix d'acquisition de l'option pourra être écrit comme:

$$\text{Gain au temps } T = \text{Max}(0, S_T - K)$$

La deuxième qui est proposée c'est l'option d'achat exotique. Dans ce contexte, son principe demeure relativement simple, mais dans un contexte réel, il peut être plus complexe. En effet, dans ce numéro on propose une option d'achat à deux temps. Le détenteur de l'option peut exercer son pouvoir d'achat à 2 moments distincts. Il peut décider d'appliquer son option aux temps T_1 et T_2 à l'un de ces deux moments ou même de ne rien faire (tout dépendamment si la valeur de l'action au temps T est supérieur à celle prédéterminée au moment de contracter l'option d'achat). Le gain de l'option exotique au temps est présenté comme suit:

$$\text{Gain au temps } T_2 = \text{Max}(0, S_{T_1} - K_1) + \text{Max}(0, S_{T_2} - K_2), \quad T_1 < T_2$$

Il est important de garder en tête que le prix de l'option n'est pas considéré dans le calcul de gain et qu'aucun facteur d'actualisation ou d'accumulation n'est appliqué.

Nous voulons déterminer le prix de chaque combinaison possible des options européennes ($K_1 T_1, K_2 T_1, K_1 T_2$ et $K_2 T_2$) et de l'option exotique. Pour y parvenir, nous avons simulé 1000 scénarios de valeurs boursières à T_1 et à T_2 . Nous avons conservé ces valeurs pour qu'il soit possible de comparer les différentes options d'achat en fonction de la même valeur boursière.

Ainsi, avec les valeurs financières, il était simple de calculer le gain de chacune des options. Puisque la valeur de l'action au temps T_2 dépend de la valeur au temps T_1 , il fallait en tenir compte dans la création de notre

formule pour simuler les résultats. Cependant, le rendement entre les deux temps sont indépendants et stationnaires (propriété des mouvements Brownien avec dérive), alors il restait d'appliquer ce rendement à la valeur de l'action à T_1 . Ensuite, afin de déterminer le prix, nous avons fait la moyenne des 1000 gains que nous avons actualisée à $t = 0$ pour chaque situation. Nous obtenons les prix approximatifs suivant:

Table 6: Prix des options d'achat

$OptEuroT_1K_1$	$OptEuroT_1K_2$	$OptEuroT_2K_1$	$OptEuroT_2K_2$	Option Exotique
1.21	0.14	4.73	1.53	2.74

Quand on analyse et l'on compare les prix, il est tout à fait normal que plus le temps est grand, plus la valeur de l'action est grande. Également, plus le prix d'exercice (\mathbf{K}) est élevé, plus le prix de l'option est petite. En effet, la probabilité d'atteindre le prix \mathbf{K} au moment T est plus faible plus le prix d'exercice est élevé. Qui dit faible risque dit faible prix.

Si l'on compare le prix de l'option exotique aux autres, on peut remarquer que le prix de cette option est très près de la somme des prix des options européennes T_1, K_1 et T_2, K_2 . La logique derrière tout ça, c'est que l'option exotique est une combinaison de ces deux options européennes.

b) Statistiques et graphiques du profit des options

En reprenant les simulations effectuées en a), nous nous intéressons au profit de l'option d'achat européenne à T_1 au prix d'exercice $K_1 = 115$ et de l'option d'achat exotique. Le profit est calculé de la manière suivante:

$$\text{Profit} = (\text{valeur actualisée à } t = 0 \text{ du Gain au temps } T) - (\text{prix d'achat})$$

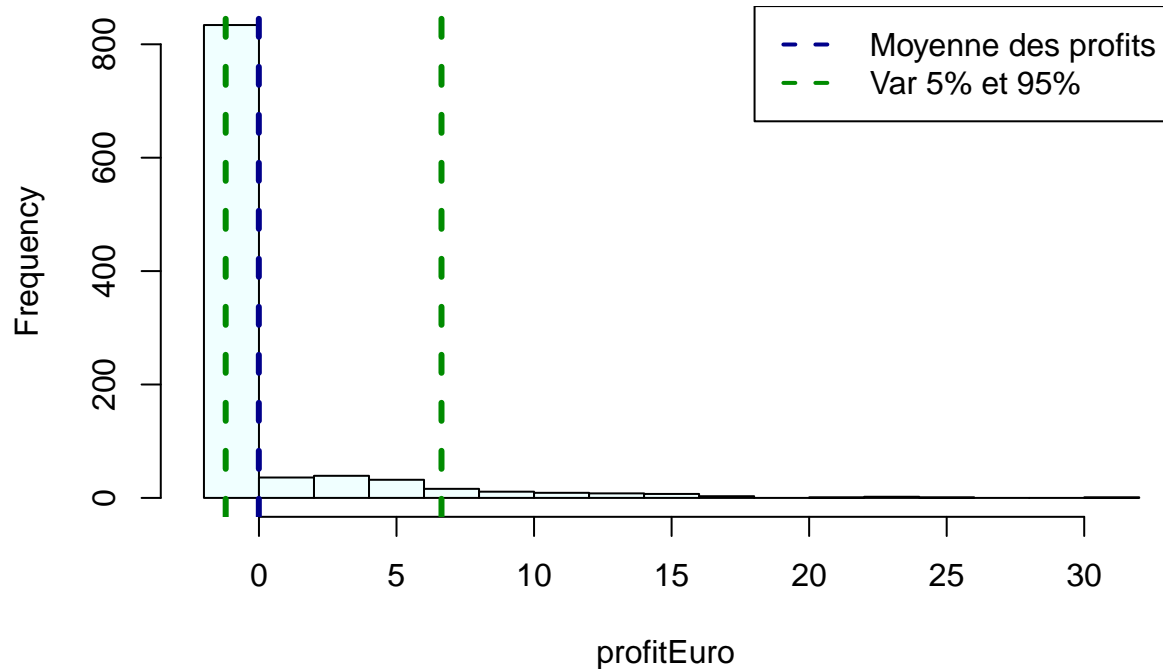
*On utilise la force d'intérêt sans risque δ pour actualiser le gain dans l'équation précédente

Nous nous intéressons à la "Value-at-risk" (VaR) à 5% et à 95%, mais nous avons décidé d'y inclure la moyenne. Voici les valeurs auxquels on s'intéresse dans un table, en plus des graphiques de la distribution des résultats trouvés:

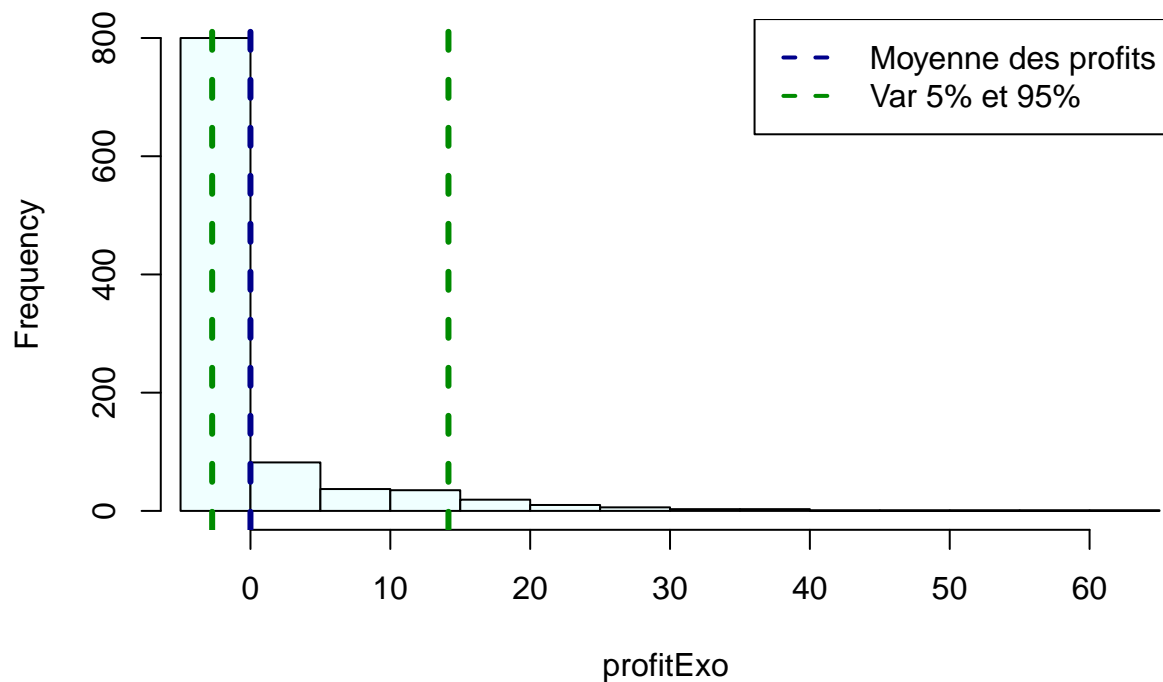
Table 7: Statistiques des différentes options

Type d'option	Profit Moyen	VaR(5)	VaR(95)
Européenne	0	-1.21	6.80
Exotique	0	-2.74	14.61

Distribution des profits de l'option d'achat Européenne



Distribution des profits de l'option d'achat exotique



Lorsqu'on regarde la moyenne, il est tout à fait normal d'obtenir une valeur de 0, puisque le prix est calculé en fonction de l'espérance des gains. Dans la situation, il est peu avantageux de prendre les options d'achats, car la probabilité de faire un profit est très faible. En effet, pour faire un profit avec l'option européenne, il faudrait obtenir un rendement de 15% en une seule année, mais le rendement moyen est d'environ 5.65% par année. (Cette valeur est calculée, sachant que $S_1 \sim$ mouvement Brownien géométrique et que $S_t \sim \log\text{-Normale}(\mu, \sigma^2 t)$). C'est essentiellement semblable pour l'option d'achat exotique, mais elle

présente légèrement plus de résultats positifs. En effet, puisqu'il y a deux moments où le détenteur de l'option peut exercer son pouvoir d'achat, le gain d'un des temps peut compenser la perte de l'autre.

c) Trouver α de sorte que le processus de la valeur de l'action suive une Martingale

On veut trouver α tel que le processus de la valeur de l'action suive une martingale. On a une martingale à condition que

$$E[S_t | S_{t-1}, \dots, S_0] = S_{t-1}$$

On a le processus suivant, où S_t représente la valeurs de l'action au temps t pour un processus qui inclut des chocs tel que:

$$S_t = S_{t-1} e^{\mu + \sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i}$$

Donc, pour obtenir une martingale avec notre processus, on pose :

$$\begin{aligned} E \left[S_{t-1} e^{\mu + \sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i} | S_{t-1} \right] &= S_{t-1} \\ \cancel{S_{t-1}} E \left[e^{\mu + \sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i} | S_{t-1} \right] &= \cancel{S_{t-1}} \\ E \left[e^{\mu + \sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i} \right] &= 1 \\ e^\mu E \left[e^{\sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i} \right] &= 1 \\ E \left[e^{\sigma(Z(t) - Z(t-1)) + \sum_{i=N(t-1)}^{N(t)} X_i} \right] &= e^{-\mu} \\ E \left[e^{\sigma Z(t) - \sigma Z(t-1) + \sum_{i=N(t-1)}^{N(t)} X_i} \right] &= e^{-\mu} \end{aligned}$$

On sait que $Z(t)$ suit un mouvement brownien standard, donc $Z(t) \sim N(0, t)$. On pose $W(t) = \sigma Z(t)$ et l'on obtient $W(t)$, un mouvement Brownien, tel que $W(t) \sim N(0, t\sigma^2)$. Par les propriétés des mouvements Brownien il est possible de développer l'expression suivante, sachant que les accroissements sont stationnaires.

$$\begin{aligned} (W(t) - W(t-1)) &\sim N(0, \sigma^2) \\ W(1) &\sim N(0, \sigma^2) \end{aligned}$$

Ceci nous permet de simplifier dans l'équation :

$$\begin{aligned}
E[e^{W(1)} e^{\sum_{i=N(t-1)}^{N(t)} X_i}] &= e^{-\mu} \\
E[S^*(1) e^{\sum_{i=N(t-1)}^{N(t)} X_i}] &= e^{-\mu}, \quad \text{posons } S^*(t) = e^{W(t)} \text{ un mouvement Brownien géométrique} \\
E[S^*(1) e^{\sum_{i=N(t-1)}^{N(t)} X_i}] &= e^{-\mu} \longrightarrow S^*(t) \sim \text{Log-normale}(0, \sigma^2 t) \\
E[S^*(1)] E[e^{\sum_{i=N(t-1)}^{N(t)} X_i}] &= e^{-\mu} \quad \text{car } S^*(t) \perp \sum_{i=N(t-1)}^{N(t)} X_i
\end{aligned}$$

Sachant que l'espérance de la loi log-normale est $E[S^*(1)] = e^{\frac{1}{2}\sigma^2}$, il suffit de la remplacer dans la dernière équation. D'ailleurs, nous poserons $Y(t) = \sum_{i=1}^{N(t)} X_i$ et par la propriété des accroissements stationnaires $\sum_{i=1}^{N(t)} X_i = Y(1)$. On se retrouve avec:

$$e^{\frac{1}{2}\sigma^2} E[e^{Y(1)}] = e^{-\mu} \quad (4)$$

On remarque que le terme $E[e^{Y(1)}]$ est la fonction génératrice des moments de $Y(t)$ évaluée à $t = 1$. Pour un processus de Poisson composé, la fonction génératrice des moments s'obtient avec la formule suivante :

$$M_{Y(t)}(u) = e^{\lambda t(M_X(u)-1)}$$

Pour $X_i \sim U(0, \alpha)$, la fonction génératrice des moments s'exprime ainsi :

$$\begin{aligned}
M_X(u) &= \frac{e^{\alpha u} - 1}{\alpha u} \\
\Rightarrow M_{Y(1)}(1) &= e^{\lambda \left[\frac{e^\alpha - 1}{\alpha} - 1 \right]} \\
&= e^{\frac{\lambda}{\alpha} (e^\alpha - 1 - \alpha)}
\end{aligned}$$

Ainsi, en remplaçant dans l'équation (1),

$$\begin{aligned}
M_{Y(t)}(u) &= e^{\lambda t(M_X(u)-1)} \\
\ln(e^{\frac{1}{2}\sigma^2 + \frac{\lambda}{\alpha}(e^\alpha - 1 - \alpha)}) &= \ln(e^{-\mu}) \\
\frac{1}{2}\sigma^2 + \frac{\lambda}{\alpha}(e^\alpha - 1 - \alpha) &= -\mu
\end{aligned}$$

Avec l'utilisation d'un solveur, on peut trouver la valeur de α en posant $\lambda = 0.1$, $\mu = 0.05$ et $\sigma = 0.1$. On obtient :

$$\alpha \approx -1.885$$

d) Trouver α de sorte que le processus de la valeur de l'action suive une sous-Martingale

Pour cette question, il est demandé de trouver la valeur de α afin que le processus de la valeur de l'action suive une sous-martingale avec un rendement annuel espéré de $e^\mu - 1$. La démarche afin d'arriver au résultat est similaire à la question précédente, puisque nous avons opté pour stratégies sables.

$$\begin{aligned}
E[S_t|S_{t-1}] &= S_{t-1}e^\mu \\
E\left[S_{t-1}e^{\mu+\sigma(Z(t)-Z(t-1))+\sum_{i=N(t-1)}^{N(t)} X_i}\right] &= S_{t-1}e^\mu \\
S_{t-1}E\left[e^{\mu+\sigma(Z(t)-Z(t-1))+\sum_{i=N(t-1)}^{N(t)} X_i}\right] &= S_{t-1}e^\mu \\
e^\mu E\left[e^{\sigma(Z(t)-Z(t-1))+\sum_{i=N(t-1)}^{N(t)} X_i}\right] &= e^\mu \\
E\left[e^{W(1)+\sum_{i=N(t-1)}^{N(t)} X_i}\right] &= 1 \\
E\left[e^{W(1)}\right] E\left[e^{\sum_{i=N(t-1)}^{N(t)} X_i}\right] &= 1 \\
e^{\sigma^2} e^{\lambda\left[\frac{e^\lambda-1-\alpha}{\alpha}\right]} &= 1 \rightarrow \text{car } M_{Y(1)}(1) = e^{\frac{\lambda}{\alpha}(e^\alpha-1-\alpha)} \\
e^{\sigma^2+\lambda\left[\frac{e^\lambda-1-\alpha}{\alpha}\right]} &= 1 \\
\sigma^2 + \lambda\left[\frac{e^\alpha-1-\alpha}{\alpha}\right] &= \ln(1) = 0 \\
\alpha &\approx -0.215 \rightarrow \text{Solution du solveur}
\end{aligned}$$

Ce résultat sera utilisé pour faire le numéro 2 e).

e) Simuler 1000 scénarios du temps nécessaire pour une augmentation de β % de la valeur

On se rappelle que la formule pour la valeur d'une action au temps t pour un processus avec des chocs est la suivante :

$$S_t = S_0 e^{\mu+\sigma Z(t)+\sum_{i=1}^{N(t)} X_i} \quad (5)$$

À la question d) la valeur de α trouvée était de -0.215. Ainsi, les chocs X_i sont uniformes sur $(-0.215, 0)$, et $N(t) \sim \text{Poisson}(0.1t)$. Par les propriétés d'un processus de Poisson homogène, pour un processus dont le nombre d'événements sur l'intervalle $(0, t)$ suit un processus de Poisson de paramètres (λt) , le temps d'attente entre chaque événement suit une loi exponentielle de moyenne $\frac{1}{\lambda}$. Pour les simulations, on part de 1000 actions pour lesquels on pose une valeur initiale $S_0 = 1\$$, et on s'intéresse au temps nécessaire pour que S_t atteigne une valeur de $1 \times (1 + \beta\%) = 1\$ \times 1.5 = 1.50\$$. On peut donc poser $S_0 = 1$ dans l'équation (1). Puisque les intervalles t se mesurent en années, nous regarderons la valeur de l'action pour des intervalles d'une journée, soit $\frac{t}{365}$, pour obtenir plus de précision. De cette manière nous nous rapprochons d'un modèle de l'évolution continue de la valeur de l'action.

À partir de l'équation (1), on a développé l'algorithme suivant qui permet de simuler les temps d'attente pour $n = 1000$ simulations.

Algorithme

1. Création d'une boucle
 - i. Simulation d'un nombre aléatoire d'une $Exp(\lambda = 0.1)$ qui représente le temps d'attente entre les chocs.
 - a. Si le premier choc se produit dans le prochain intervalle ($t < \frac{1}{365}$), simulation de la valeur de l'action à $t = \frac{1}{365}$ en simulant un nombre provenant d'une loi $Normale(\mu = 0.05, \sigma = 0.1)$ et un nombre d'une $Uniforme(-0.215, 0)$. Remplacer ces valeurs dans l'équation (1).
 - b. Sinon, simuler un nombre provenant d'une loi $Normale(\mu = 0.05, \sigma = 0.1)$, et obtenir la valeur de l'action à $t = \frac{1}{365}$. Utiliser ces valeurs dans l'équation (1).
 - ii. Si la valeur de l'action atteint 1.50, fin de la boucle. Sinon, poser $t^* = t + \frac{1}{365}$ et recommencer à i.
2. Exécution de la boucle $n = 1000$ fois

Le code de cette fonction a été généré ainsi:

```
## Fonction pour calculer le temps pour atteindre un rendement de 50%
SimRendement <- function(){
  t <- 0
  T_i <- c()
  temps <- 0
  rendement <- 1
  while(rendement<1.5){
    if(temps<=0){
      temps <- temps+rexp(1,0.1)
    }

    temps <- temps-1/365
    if(temps<=0){
      rendement <- rendement*exp(0.05*(1/365)+0.1*rnorm(1,0,sqrt(1/365))+runif(1,-0.215,0))
    }
    else{
      rendement <- rendement*exp(0.05*(1/365)+0.1*rnorm(1,0,sqrt(1/365)))
    }
    t <- t+1/365
  }

  t
}

SimTemps <- sort(replicate(1000,SimRendement()))
```

On peut représenter graphiquement les simulations:

Distribution des 1000 simulations pour voir un rendement de 50%

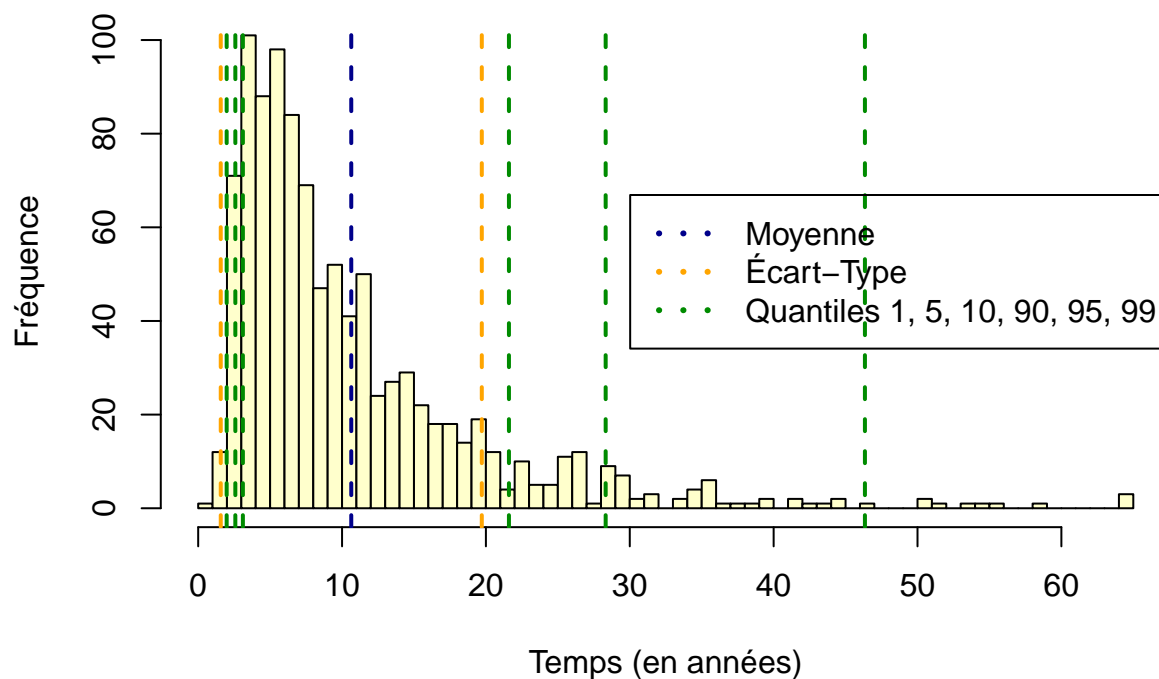


Table 8: Statistiques de temps

Statistiques	Valeurs
<i>Moyenne</i>	10.640808
<i>Écart-type</i>	9.072379
<i>1^{er} Percentile</i>	1.980630
<i>5^e Percentile</i>	2.582329
<i>10^e Percentile</i>	3.106575
<i>90^e Percentile</i>	21.600548
<i>95^e Percentile</i>	28.323836
<i>99^e Percentile</i>	46.344411

On remarque qu'il y a une asymétrie de la courbe de distribution des temps vers la gauche. La moyenne est de 10.64 années, mais pour atteindre un rendement de 50% en grande partie, les données se trouvent sous cette moyenne. On observe des valeurs extrêmes très hautes, par exemple le 95^e percentile est de 28.32 ans, qui se retrouve significativement au dessus de la moyenne.

Annexe

Code informatique

```
### Table 1
param_ab <- data.frame("i" = c(1:3),
                        "n_i"= c(35,25,40),
                        "a_i"= c(0,0,0),
                        "b_i"= c(4,5,6),
                        "alpha_i"= c(1,1,1),
                        "beta_i"= c(1.5,2,2.5),
                        "M_i"= c(1500,2500,1000),
                        "delta_i" = c("2%", "2%", "2%"))

library(knitr)
library(xtable)

kable(param_ab, caption = "Paramètres du numéro 1 a) et b)",
      escape = FALSE, align = "l",
      col.names = c("$i$", "$n_i$", "$a_i$", "$b_i$", "$\\alpha_i$",
                    "$\\beta_i$", "$M_i$", "$\\delta_i$"))

### Table 2
param_c <- data.frame("i" = c(1:3),
                      "n_i"= c(35,25,40),
                      "theta_i"= c(2,3,4),
                      "lambda_i"= c(0.03,0.04,0.05),
                      "alpha_i"= c(1,1,1),
                      "beta_i"= c(1.5,2,2.5),
                      "M_i"= c(1500,2500,1000),
                      "delta_i" = c("2%", "2%", "2%"))

kable(param_c, caption = "Paramètres du numéro 1 c)",
      escape = FALSE, align = "r",
      col.names = c("$i$", "$n_i$", "$\\theta_i$", "$\\lambda_i$",
                    "$\\alpha_i$", "$\\beta_i$", "$M_i$", "$\\delta_i$"))

## Table 3
param_2ecas <- data.frame("mu"= 0.05, "sigma" = 0.1, "force_interet"= 0.02,
                          "k_1"= 115, "k_2"= 127, "T_1"= 1, "T_2"= 2,
                          "lambda"= 0.1, "alpha"= "?", "beta"= 0.5)

kable(param_2ecas, caption = "Paramètres du numéro 2", escape = FALSE,
      col.names = c("$\\mu$", "$\\sigma$", "$\\delta$", "$K_1$",
                    "$K_2$", "$T_1$", "$T_2$", "$\\lambda$",
                    "$\\alpha$", "$\\beta_i$"))

# Fonction de perte des contrats type 1
Simulation_1 <- function(){

  lambda <- runif(1,0,4)
  T_i <- c()
```



```

X_i <- c()
S_ik <- c()
perte <- c()
temps <- 0
while(temps<10){
  T_i <- append(T_i, rexp(1,rate = lambda)) ## Temps entre 2 pertes
  X_i <- append(X_i,rbeta(1, shape1 = 1, shape2 = 1.5)) ## Taux de la perte
  S_ik <- append(S_ik, sum(T_i)) ## Somme des temps de perte
  temps <- sum(T_i)
}
T_i <- T_i[-length(T_i)]
S_ik <- S_ik[-length(S_ik)]
X_i <- X_i[-length(X_i)]

sum(exp(-0.02*S_ik)*1500*X_i)
}

```

```

Simulation_2 <- function(){

lambda <- runif(1,0,5)
T_i <- c()
X_i <- c()
S_ik <- c()
perte <- c()
temps <- 0
while(temps<10){
  T_i <- append(T_i, rexp(1,rate = lambda))
  X_i <- append(X_i,rbeta(1, shape1 = 1, shape2 = 2))
  S_ik <- append(S_ik, sum(T_i))
  temps <- sum(T_i)
}
T_i <- T_i[-length(T_i)]
S_ik <- S_ik[-length(S_ik)]
X_i <- X_i[-length(X_i)]

sum(exp(-0.02*S_ik)*2500*X_i)
}

```

Fonction de perte des contrats type 3

```

Simulation_3 <- function(){

lambda <- runif(1,0,6)
T_i <- c()
X_i <- c()
S_ik <- c()
perte <- c()
temps <- 0
while(temps<10){
  T_i <- append(T_i, rexp(1,rate = lambda))
  X_i <- append(X_i,rbeta(1, shape1 = 1, shape2 = 2.5))
  S_ik <- append(S_ik, sum(T_i))
}

```

```

    temps <- sum(T_i)
  }
  T_i <- T_i[-length(T_i)]
  S_ik <- S_ik[-length(S_ik)]
  X_i <- X_i[-length(X_i)]

  sum(exp(-0.02*S_ik)*1000*X_i)
}

## Fonction pour simuler n échantillons
simPerteTot <- function(n){
  i <- 1
  pertetetot <- c()

  for(i in 1:n){
    a <- sum(replicate(35, Simulation_1()))
    b <- sum(replicate(25, Simulation_2()))
    c <- sum(replicate(40, Simulation_3()))
    pertetetot <- append(pertetetot,sum(a,b,c))
    i+1
  }
  pertetetot
}

simUnif <- simPerteTot(1000)

Moyenne <- round(mean(simUnif),0)
Variance <- round(sd(simUnif)^2,0)
VaR90 <- quantile(simUnif,0.9)
VaR95 <- quantile(simUnif,0.95)
VaR99 <- quantile(simUnif,0.99)
TVaR90 <- mean(tail(sort(simUnif),100))
TVaR95 <- mean(tail(sort(simUnif),50))
TVaR99 <- mean(tail(sort(simUnif),10))

TableStats <- data.frame("Moyenne"= c(Moyenne,"",""),
                          "Variance"= c(Variance,"",""),
                          "alpha"= c("10%","5%","1%"),
                          "VaR(1-alpha)" = c(VaR90, VaR95,VaR99),
                          "TVaR(1-alpha)" = c(TVaR90, TVaR95,TVaR99))

kable(TableStats, caption = "Statistiques des 1000 simulations",
      escape = FALSE,
      col.names = c("Moyenne", "Variance", "$\\alpha$",
                    "$VaR(1-\\alpha)$", "$TVaR(1-\\alpha)$"))

q05 <- quantile(simUnif,0.05)

```

```

q95 <- quantile(simUnif,0.95)
q100 <- quantile(simUnif,1)
hist(simUnif, probability = TRUE, nclass = 20, col = "azure",
     main = "Distribution des 1000 simulations des pertes",
     xlab = "Perte actualisée (en $)", ylab = "Densité")

abline(v = mean(simUnif), col = "blue4", lty = 2, lwd = 3) # Moyenne
abline(v = q95, col = "green4", lty = 3, lwd = 3)
abline(v = q05, col = "green4", lty = 3, lwd = 3)
lines(density(simUnif), col = "red4", lwd = 2)

legend("topright",
     legend = c("Moyenne","Quantile à 5% et 95%"),
     col = c("blue4","green4"),
     lty = c(2,3),
     lwd = c(3,3),
     cex=0.70)

x1 <- min(which(density(simUnif)$x >= q95))
x2 <- max(which(density(simUnif)$x < q100))
with(density(simUnif), polygon(x=c(x1,x1:x2,x2)),
     y= c(0, y[x1:x2], 0), col="red3"))

## Fonction pour calculer le gain des options
valAction <- function(n,mu,sigma){
  S_t1 <- c()
  S_t2 <- c()
  optt1k1 <- c()
  optt1k2 <- c()
  optt2k1 <- c()
  optt2k2 <- c()
  optExo <- c()
  optExoACT <- c()
  i <- 1
  for(i in 1:n){
    firstyear <- 100 * exp(mu + sigma*rnorm(1,0,1))
    secondyear <- firstyear * exp(mu + sigma*rnorm(1,0,1))
    S_t1 <- append(S_t1,firstyear)
    S_t2 <- append(S_t2,secondyear)
    optt1k1 <- append(optt1k1,max(0,firstyear-115))
    optt1k2 <- append(optt1k2,max(0,firstyear-127))
    optt2k1 <- append(optt2k1,max(0,secondyear-115))
    optt2k2 <- append(optt2k2,max(0,secondyear-127))
    optExo <- append(optExo,max(0,firstyear-115)+max(0,secondyear-127))
    optExoACT <- append(optExoACT,
                       exp(-.02)*max(0,firstyear-115)+ exp(-.02*2)*max(0,secondyear-127))

    i <- i+1
  }
  ValAction <- data.frame("S_t1"=S_t1,
                          "S_t2"=S_t2,
                          "optt1k1"= optt1k1,
                          "optt1k2"=optt1k2,
                          "optt2k1"=optt2k1,

```

```

        "optt2k2"=optt2k2,
        "optExo"=optExo,
        "optExoACT"=optExoACT)

    ValAction
}

Gain <- valAction(1000,0.05,0.1)

# Prix de l'option en fonction de T et K.

# NOTE: J'ai fait la moyenne des simulation, afin d'obtenir le gain moyen du coût d'option
prixAchatEuroK1T1 <- round(exp(-0.02*1)*mean(Gain[,3]),2)
prixAchatEuroK2T1 <- round(exp(-0.02*1)*mean(Gain[,4]),2)
prixAchatEuroK1T2 <- round(exp(-0.02*2)*mean(Gain[,5]),2)
prixAchatEuroK2T2 <- round(exp(-0.02*2)*mean(Gain[,6]),2)
prixAchatExo <- round(mean(Gain[,8]),2)

Prix <- data.frame("T = 1, K = 115" = prixAchatEuroK1T1,
                  "T = 1, K = 127" = prixAchatEuroK2T1,
                  "T = 2, K = 115" = prixAchatEuroK1T2,
                  "T = 2, K = 127" = prixAchatEuroK2T2,
                  "Option exo" = prixAchatExo)

kable(Prix, caption = "Prix des options d'achat",
      escape = FALSE,
      align=c(rep('c',times=5)),
      col.names = c("$Opt Euro T_1 K_1$", "$Opt Euro T_1 K_2$",
                    "$Opt Euro T_2 K_1$", "$Opt Euro T_2 K_2$",
                    "Option Exotique"))

## Calcul du profit et VaR
profitEuro <- exp(-0.02*1)*Gain[,3] - prixAchatEuroK1T1

profitExo <- Gain[,8]-prixAchatExo

VaR95Euro <- tail(sort(profitEuro), 50)

VaR5Euro <- head(sort(profitEuro),50)

VaR95Exo <- tail(sort(profitExo), 50)

VaR5Exo <- head(sort(profitExo),50)

Tableau <- data.frame("Type d'option" = c("Européenne",
                                           "Exotique"),
                      "Profit Moyen" = c(round(mean(profitEuro),2),
                                           round(mean(profitExo),2)),
                      "VaR(5)" = c(round(max(VaR5Euro),2),
                                     round(max(VaR5Exo),2)),
                      "VaR(95)"= c(round(min(VaR95Euro),2),
                                     round(min(VaR95Exo),2)),

```

```

round(min(VaR95Exo),2)))

kable(Tableau, align=c("l",rep('c',times=3)),
      caption = "Statistiques des différentes options",
      col.names = c("Type d'option","Profit Moyen", "VaR(5)","VaR(95)"))

## Histogrammes

#Option d'achat européenne
hist(profitEuro,
     main = "Distribution des profits de l'option d'achat Européenne",
     nclass = 15 ,
     col = "#CCFFCC",
     xlim = c(min(profitEuro)-1,
               max(profitEuro)))

abline(v = mean(profitEuro),
       col= "blue4",
       lwd = 4,
       lty = 2)

abline(v = quantile(profitEuro, c(0.05, 0.95)),
       col = "#CC3366",
       lwd = 4,
       lty = 2)

legend("topright",
      legend = c("Moyenne des profits", "Var 5% et 95%"),
      col = c("blue4", "#CC3366"),
      lty = c(3,3),
      lwd = c(3,3))

#Option d'achat exotique
hist(profitExo,
     main = "Distribution des profits de l'option d'achat exotique",
     nclass = 20 ,
     col = "#CCFFCC",
     xlim = c(min(profitExo)-1,
               max(profitExo)))

abline(v = mean(profitExo),
       col= "blue4",
       lwd = 4,
       lty = 2)

abline(v = quantile(profitExo, c(0.05,0.95)),
       col = "#CC3366", lwd = 4, lty = 2)

legend("topright",
      legend = c("Moyenne des profits", "Var 5% et 95%"),
      col = c("blue4", "#CC3366"),
      lty = c(3,3), lwd = c(3,3))

```

```

### Fonction pour le rendement
SimRendement <- function(){
  t <- 0
  T_i <- c()
  temps <- 0
  rendement <- 1
  while(rendement<1.5){
    if(temps<=0){
      temps <- temps+rexp(1,0.1)
    }

    temps <- temps-1/365
    if(temps<=0){
      rendement <- rendement*exp(0.05*(1/365)+0.1*rnorm(1,0,sqrt(1/365))+runif(1,-0.215,0))
    }
    else{
      rendement <- rendement*exp(0.05*(1/365)+0.1*rnorm(1,0,sqrt(1/365)))
    }
    t <- t+1/365
  }

  t
}

SimTemps <- sort(replicate(1000,SimRendement()))

hist(SimTemps, nclass = 50, xlab = "Temps (en années)",ylab = "Fréquence",
     main = "Distribution des 1000 simulations pour voir un rendement de 50%",
     col = "#FFFFCC")
abline(v = mean(SimTemps),
      col= "blue4",
      lwd = 2,
      lty = 2)

abline(v = mean(SimTemps)+sd(SimTemps),
      col= "orange", lwd = 2, lty = 2)
abline(v = mean(SimTemps)-sd(SimTemps),
      col= "orange", lwd = 2, lty = 2)
abline(v = quantile(SimTemps,c(0.01,0.05,0.1,0.9,0.95,0.99)),
      col= "green4", lwd = 2, lty = 2)
legend("right", legend = c("Moyenne", "Écart-Type",
                          "Quantiles 1, 5, 10, 90, 95, 99"),
      col = c("blue4", "orange", "green 4"),
      lty = c(3,3), lwd = c(3,3))

# Valeurs des quantiles
quan <- quantile(SimTemps,c(0.01,0.05,0.1,0.9,0.95,0.99))
#Moyenne des simulations

```

```

Moy <- mean(SimTemps)
ecarttype <- sd(SimTemps)
Table2e <- data.frame("Statistiques"= c("$Moyenne$", "$Écart$-$type$",
                                         "$1^{er} ~ Percentile$",
                                         "$5^{e} ~ Percentile$",
                                         "$10^{e} ~ Percentile$",
                                         "$90^{e} ~ Percentile$",
                                         "$95^{e} ~ Percentile$",
                                         "$99^{e} ~ Percentile$"),
                      "Valeurs"= c(Moy, ecarttype,quan))

kable(Table2e, caption = "Statistiques de temps",
      align= c("l","c"), escape = FALSE)

```