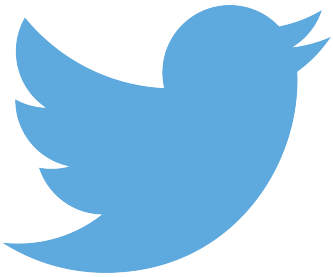
Simulation d'Evénements Rares pour Twitter

Ismail Lemhadri, Laurent Meunier

Ecole Polytechnique - juin 2016



Introduction

Twitter est devenu un réseau social d'envergure mondiale ces dernières années. Les utilisateurs de ce réseau peuvent envoyer des messages limités à 140 caractères, appelés "tweets". Chaque "tweet" peut être « retweeté », c'est-à-dire qu'un autre utilisateur renvoie le même "tweet" à ses contacts. Twitter compte certainement parmi les réseaux sociaux les plus populaires aujourd'hui, avec plus de 300 millions d'utilisateurs actifs début 2016. Il est caractérisé par sa nature duale puisqu'il peut être utilisé à la fois comme réseau social et comme moyen de disséminer l'information. La popularité de Twitter a attiré non seulement des particuliers, mais également des entreprises qui cherchent à promouvoir leur marque, et des robots, qui sont des programmes informatiques qui utilisent Twitter pour disséminer l'information, et parfois du contenu malveillant.

Dans ce travail, nous étudions l'auto-excitation du réseau Twitter, c'est-à-dire comment un message peut influencer le flux du réseau. Comment modéliser la dynamique des tweets "viraux" et la propagation pair-à-pair d'idées ou de tendances, par rapport à un flux exogène ? Peut-on quantifier une telle dynamique ? Est-elle prévisible ? Telles sont les questions auxquelles nous tâchons de répondre. Les retombées sont importantes, notamment pour la détection des dynamiques fortes avant qu'elles ne surviennent.

Nous nous intéressons plus particulièrement à la simulation de la dynamique de branchement sur Twitter et à sa quantification. En nous inspirant des travaux d'Ogata [3] et de Simma & Jordan [1, 4], nous utilisons des processus ponctuels pour modéliser le trafic de "tweets" sur une heure, une journée voire plus. Le

modèle choisi, de la classe des *processus de Hawkes,* a été utilisé pour modéliser de nombreuses situations où apparaît une dynamique d'auto-excitation. Ces processus ont été introduits en 1971 par Hawkes pour modéliser les répliques de tremblements de terre. Toutefois, ils sont également utilisées dans diverses autres disciplines. Des variantes de ce modèle ont été appliquées à la dynamique des marchés financiers, à la criminalité opportuniste, à la propagation des maladies épidémiques, et au marketing viral. La détection des phénomènes d'auto-excitation revêt un enjeu majeur dans ces domaines et peut améliorer grandement la confiance et la précision des prédictions.

Les processus de Hawkes sont bien adaptés à notre problème pour deux raisons principales. Tout d'abord, ces processus forment une extension très naturelle et commode des processus de Poisson. De plus, chacun de leurs paramètres peuvent être bien interprétés et expliqués pour la modélisation. A cet égard, la structure de branchement des processus de Hawkes se révèle très utile. Ainsi, un processus de Hawkes peut être vu comme un processus de population où les migrants arrivent à un taux de base λ_0. Puis chaque migrant donne naissance à des enfants suivant l'intensité $g(t)$, et ces enfants donnent eux-mêmes naissance suivant le même processus de Poisson non-homogène. Nous donnons ici une interprétation des "migrants" en tant que "tweets" indépendants, et des "enfants" comme des "retweets" ou bien des réponses au "tweet" parent.

Notre approche gagne progressivement en complexité. Dans un premier temps nous supposons que tous les "tweets" ont le même impact sur le flux du réseau. Ceci est l'occasion d'une première analyse d'événements rares, en l'occurence de l'explosion du nombre de tweets sur un court laps de temps. Des techniques de changement de probabilités sont utilisées afin d'estimer la probabilité d'une nouvelle "twitpocalypse", c'est-à-dire d'une rupture du réseau par dépassement des 2^{32} identifiants possibles pour les tweets. Nous introduisons ensuite différents modèles à "features", afin de caractériser les "tweets" susceptibles d'impacter le réseau de manière significative.[1] Enfin, nous adaptons les changements de probabilité aux modèles en présence de features.

1. Des exemples de features sont la présence d'une URL, de mots-clés, etc.

Table des matières

1 Processus de Hawkes

1.1 Définition et interprétation

Pour simuler le flux des tweets et retweets au cours d'une période de temps donnée de longueur $T > 0$, nous utilisons les processus de Hawkes unidimensionnels. Les processus de Hawkes sont des processus de comptage permettant d'étudier les phénomènes d'auto-excitations. Notons $(t_i)_i$ la suite des temps d'envois des "tweets". La valeur du processus de comptage N_t est donnée par :

$$N_t = \sum_{t_i<t} 1_{t_i<t}$$

L'intenité d'un processus de comptage est défini par :

$$\mathbb{P}(N_{t+h} - N_t = 1) = \lambda(t)h + o(h) \text{ lorsque } h \to 0.$$

Pour un processus de Hawkes, l'intensité est aléatoire et dépend des sauts précédents. Elle est donnée par :

$$\lambda(t) = \lambda_0 + \int_0^t g(t-s)dN_s = \lambda_0 + \sum_{t_i<t} g(t-t_i)$$

où λ_0 est un réel > 0, $N_s$est notre processus de Hawkes, les t_i sont les points du processus avant le temps t, et la fonction g, appelée *noyau de régression*, est nulle sur $]-\infty; 0[$ et positive et décroissante sur $[0; +\infty[$.

1.2 Un cas particulier : les processus de Poisson

Lorsque la fonction g ci-dessus est nulle, nous obtenons un processus de Poisson homogène. Ceci correspond à un processus sans auto-excitation, où les « tweets » se succèdent de manière complètement exogène :

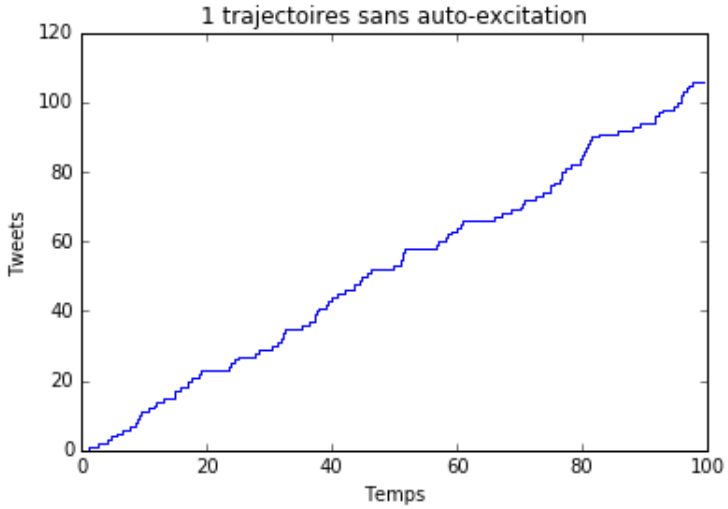


FIGURE 1.1 – Illustration d'une trajectoire d'un processus de Poisson homogène de paramètre $\lambda_0 = 1$.

1.3 Propriétés

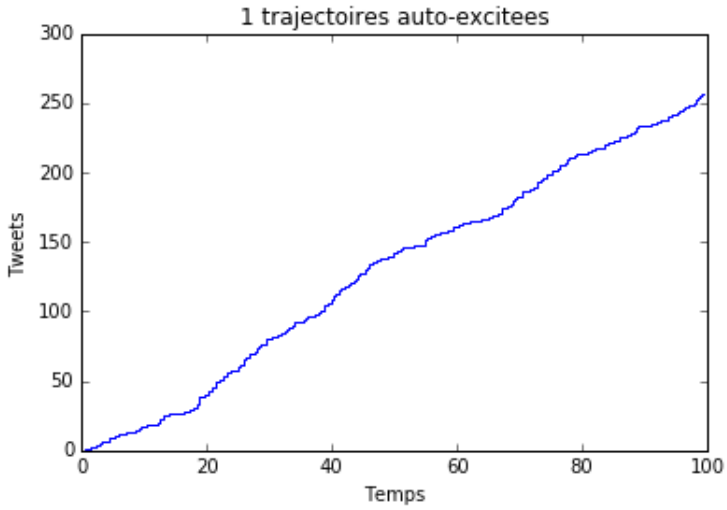


FIGURE 1.2 – Les processus de Hawkes permettent d'introduire l'auto-excitation. Ici, une trajectoire auto-excitée sur 100 secondes.

Nous étudions dans cette section quelques propriétés élémentaires des pro-

cessus de Hawkes.

Introduisons d'abord la notion de stationnarité d'un processus de Hawkes.

Définition. Un processus de Hawkes $(N_t)_{t\geq 0}$ est stationnaire si pour $t_0 > 0$, les processus $(N_t)_{t\geq 0}$ et $(N_{t+t_0})_{t\geq 0}$ suivent la même loi.

Nous nous intéressons principalement à la stationnarité asymptotique (que nous ne définissons pas précisément ici). L'idée est qu'à partir d'un t_0 suffisamment grand, la stationnarité est « approximativement vérifée ».

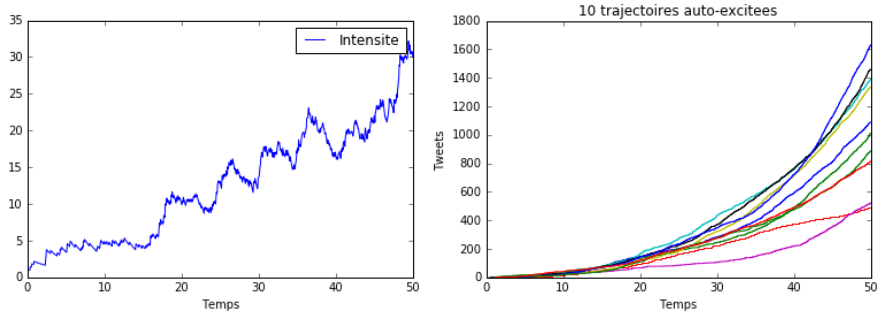


FIGURE 1.3 – Illustration de la non-stationnarité : l'intensité et le nombre de tweets explosent. Dans ce cas-ci, $\alpha = 1.1$. On constate que $\mathbb{E}[\lambda(t)]$ n'est pas constante ; l'intensité semble croître exponentiellement.

Proposition. *(Stationnarité du processus) Le processus est asymptotiquement stationnaire si et seulement si* $\int_0^{+\infty} g < 1$.

Le nombre $\|g\| = \int_0^{+\infty} g$, appelé *ratio de branchement,* a une interprétation importante en termes d'endogénéité et de dynamique de branchement. Intuitivement, $\|g\|$ représente le nombre moyen d'enfants (ou de "tweets" résultants) d'un individu (ou d'un "tweet"), $\|g\|^2$ le nombre moyen de petits-enfants... De sorte que le nombre moyen de descendants d'un individu (ou de descendants d'un "tweet") est $\sum_{k\geq 1} \|g\|^k = \frac{\|g\|}{1-\|g\|}$.

Le paramètre λ_0 représente l'intensité exogène de base du processus. Ainsi, à g fixé, le nombre total de tweets augmente proportionnellement à λ_0.

Maintenant étudions l'espérance du processus et de son intensité.

Proposition. *(Espérance de l'intensité du processus en régime stationnaire). Si* $\int_0^{+\infty} g < 1$, *l'espérance de l'intensité du processus est constante en régime stationnaire et vaut :*

$$\mathbb{E}(\lambda(t)) = \frac{\lambda_0}{1 - \int_0^{+\infty} g(s)ds}$$

On en déduit alors que $\mathbb{E}(N_t) = t.\mathbb{E}(\lambda(t)) = \frac{\lambda_0 t}{1-\int_0^{+\infty} g(s)ds}$.

La connaissance de ces espérances nous permettra plus tard d'estimer des évènements rares par changements de probabilité. En revanche, la variance de ce processus nous semble difficilement calculable.

Démonstration. (de la stationnarité et de l'espérance)
L'ésperance de l'intensité vaut :

$$\begin{aligned}
\mathbb{E}(\lambda(t)) &= \lambda_0 + \mathbb{E}(\int_{-\infty}^{t} g(t-s)dN_s) \\
&= \lambda_0 + \mathbb{E}(\int_{-\infty}^{t} g(t-s)\lambda(s)ds) \\
&= \lambda_0 + \int_{-\infty}^{t} g(t-s)\mathbb{E}(\lambda(s))ds \\
&= \lambda_0 + \mathbb{E}(\lambda(t)) \int_{-\infty}^{t} g(t-s)ds \text{ par stationnarité} \\
&= \lambda_0 + \mathbb{E}(\lambda(t)) \int_{0}^{+\infty} g(t)dt
\end{aligned}$$

Donc le processus est asymptotiquement stationnaire si et seulement si $\int_0^{+\infty} g(t)dt < 1$, et on a :

$$\mathbb{E}(\lambda(t)) = \frac{\lambda_0}{1 - \int_0^{+\infty} g(t)dt}$$

On en déduit également que :

$$\begin{aligned}
\mathbb{E}(N_t) &= \mathbb{E}(\int_0^t dN_s) \\
&= \mathbb{E}(\int_0^t \lambda(s)ds) \\
&= \int_0^t \mathbb{E}(\lambda(s))ds \\
&= \frac{\lambda_0 t}{1 - \int_0^{+\infty} g(t)dt}
\end{aligned}$$

□

2 Application à la simulation de flux Twitter

2.1 Paramètres de base

Plusieurs choix de fonction g peuvent être envisagés pour modéliser des flux Twitter.

En pratique, le choix le plus adapté devrait s'appuyer sur des données de flux réelles, en comparant les vraisemblances des différents modèles.

Dans la suite nous utilisons la fonction d'intensité suivante :

$$g(t) = 1_{t \geq 0} \alpha \beta e^{-\beta t}$$

β peut être vu comme l'inverse d'un temps de relaxation, c'est-à-dire la « durée de vie » du tweet dans le temps. α représente quant à lui l'impact d'un seul tweet sur l'intensité.

La fonction choisie doit traduire la diminution de l'impact d'un tweet au fil du temps. En d'autres termes, g doit être décroissante.

De plus, nous pouvons supposer qu'un tweet a un impact maximal peu après son émission, mais qu'il peut continuer à engendrer des tweets quelque temps au-delà. Il s'agit donc de choisir g de sorte que la queue de la distribution ne soit pas "écrasée". Ceci revient à bien choisir le temps de relaxation et donc β .

Par ailleurs, nous souhaitons représenter la causalité temporelle, à savoir qu'un tweet à la date t peut uniquement engendrer des tweets à une date $> t$. En l'occurence, g est décroissante sur $[0, +\infty[$. La contraite temporelle est renforcée par les observations suivantes :

1. Une prédiction sur le futur nécessite de connaître uniquement les événements jusqu'à une date donnée.
2. Nous souhaitons éviter les situations où un tweet A engendre un tweet B et le tweet B engendre le tweet A.

Enfin, il convient de s'assurer qu'un tweet "moyen" n'engendre pas plus d'un tweet, afin d'éviter un phénomène d'explosion du nombre de tweets. Les résultats précédents montrent que ceci est vérifié ssi $\alpha < 1$.

2.2 L'algorithme fondamental

La génération par principe de superposition s'appuie sur une représentation du processus comme un arbre aléatoire. En effet, un tweet est soit généré par le processus de Poisson de base λ_0, soit par un événement précédent. Il faut d'abord simuler le processus de base Π_0 qui correspond à un processus de Poisson homogène. Ensuite, pour chaque tweet de Π_0, simuler les tweets qui en découlent pour obtenir Π_1, et ainsi de suite jusqu'à obtenir un processus vide Π_K. Comme le nombre de tweets produits par le processus sur un intervalle borné est presque-sûrement fini, nous sommes assurés de la terminaison de cet algorithme. Si nous étendons cette représentation en incluant la source de chaque tweet, l'objet obtenu est un arbre aléatoire, où chaque tweet est un noeud de l'arbre, muni d'un timestamp correspondant à la date d'émission.

Cette méthode s'avère moins efficace du point de vue numérique. Nous lui préférons une simulation basée sur l'algorithme 2 de [3] . En effet, une autre méthode possible est de générer les tweets dans l'ordre chronologique d'apparition, en exploitant le caractère exponentiel du délai inter-tweets. La connaissance des temps d'apparition des tweets antérieurs à t suffit pour déterminer entièrement

l'intensité du processus à l'instant t, d'où la génération d'un nouveau tweet, qui modifie lui-même le reste de l'intensité.

Cette méthode exploite le principe du *thinning* qui est le suivant. Considérons un échantillon d'un processus de Poisson de moyenne λ. Si nous conservons un point quelconque de ce processus avec probabilité p et le rejetons avec probabilité $1-p$, l'objet obtenu est un échantillon d'un processus de Poisson de moyenne $p\lambda$. Ainsi, la simulation d'un processus de Poisson d'intensité non-homogène $\lambda(t)$ se ramène au cas homogène en majorant $\lambda(t)$ par une fonction constante par morceaux, notée λ^* ci-dessous.

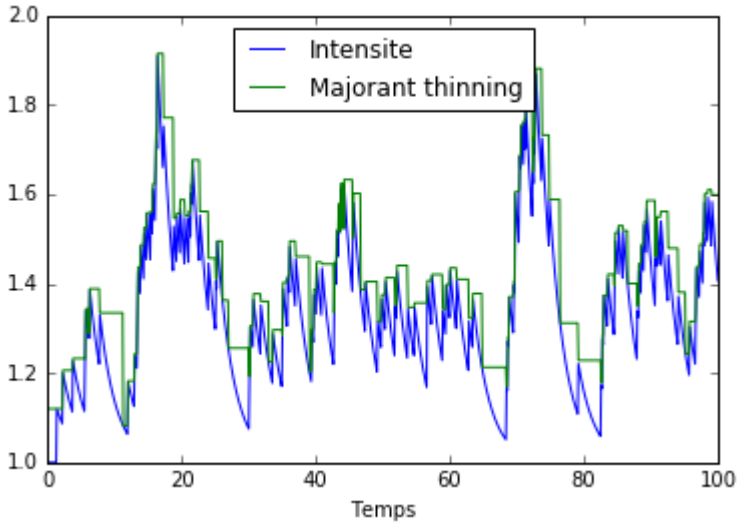


FIGURE 2.1 – Pour simuler un processus non homogène, la méthode du thinning se ramène à une intensité constante par morceaux. Grâceà la majoration par morceaux de l'intensité, les temps simulés sont rejetés moins souvent qu'avec une borne constante $\lambda^* = \sup_{0\leq t\leq T} \lambda(t)$. L'efficacité de l'algorithme en est améliorée et la complexité en temps diminuée.

Etant donné l'importance de cet algorithme dans notre étude, rappelons-le ci-dessous :

1)Poser $\lambda_0^* = \lambda_0$, $s_0 = 0$, $i = 0$, $j = 0$, $n = 1$
2)Générer $U_0 \sim Exp(\lambda_0)$
3)Si $U_0 \leq T.\lambda_0^*$ poser $t_1 = U$, sinon terminer
4)Augmenter i et poser $\lambda_i^* = \lambda(t_n|t_1, ..., t_{n-1}) + \alpha$
5)Augmenter j, générer U_j uniformément dans $[0, 1]$ et poser $s_j = s_{j-1} - \log(U_j)/\lambda_i^*$
6)Si $s_j > T$ terminer
7)Augmenter j et générer U_j uniformément dans $[0, 1]$

8)Si $U_j \leq \lambda(s_j|t_1,...,t_{n-1})/\lambda_j^*$, augmenter n , poser $t_n = s_j$ et revenir à 5)
9)Augmenter i , poser $\lambda_i^* = \lambda(s_j|t_1,...,t_{n-1}$ et revenir à 6)

2.3 Estimation des paramètres de l'intensité

Dans cette partie, nous nous intéressons au problème d'estimer l'intensité $\lambda(t)$, c'est-à-dire le jeu de paramètres $(\alpha, \beta, \lambda_0)$. Ce problème revêt une grande importance pratique car il permet de quantifier concrètement la dynamique de branchement et l'endogéneité du réseau.

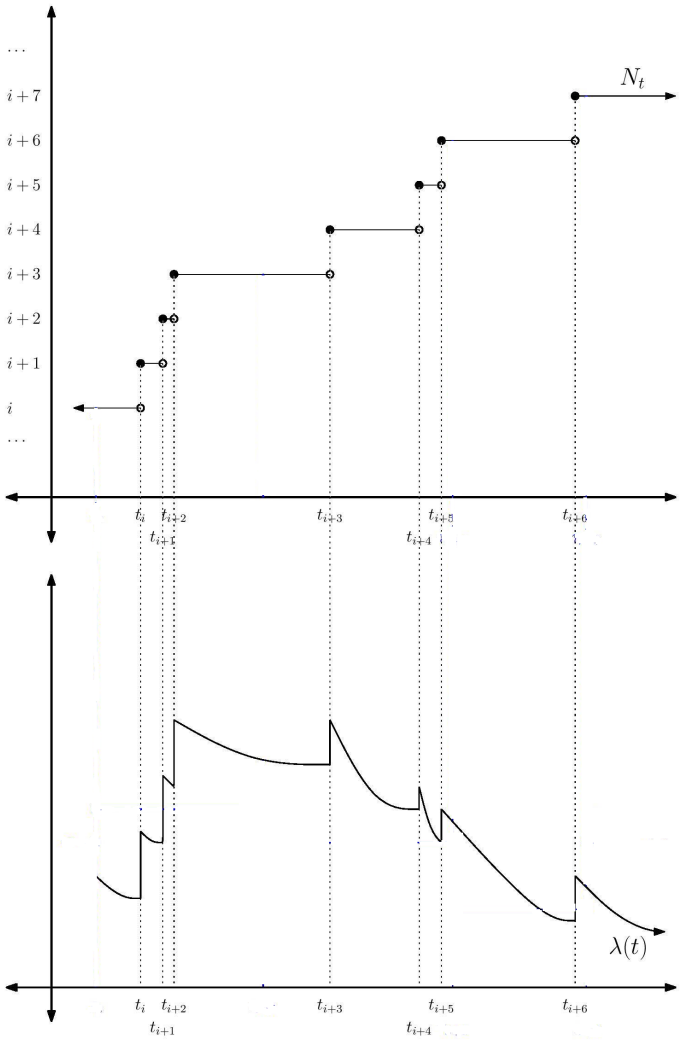


FIGURE 2.2 – Le problème de l'estimation : retrouver l'intensité non observée $\lambda(t)$ à partir des temps d'apparition des tweets uniquement.

Pour cela, et suivant [6], rappelons la valeur de la fonction de vraisemblance :

$$L(\alpha, \beta, \lambda_0) = exp(\int_0^T (1 - \lambda(s))ds + \int_0^T ln(\lambda(s))dN_s)$$

$$= exp(T - T\lambda_0 - \sum_{i=1}^{n} \alpha(1 - e^{-\beta(T-t_i)}) + \sum_{i=1}^{n} ln(\lambda_0 + \sum_{j<i} \alpha\beta e^{-\beta(t_i-t_j)})$$

Nous utilisons le principe de l'*estimation du maximum de vraisemblance.* Il s'agit donc de déterminer

$$\text{argmax} L(\alpha, \beta, \lambda_0)$$

sous la contrainte $\alpha, \beta, \lambda_0 \geq 0$. Deux questions se posent alors :

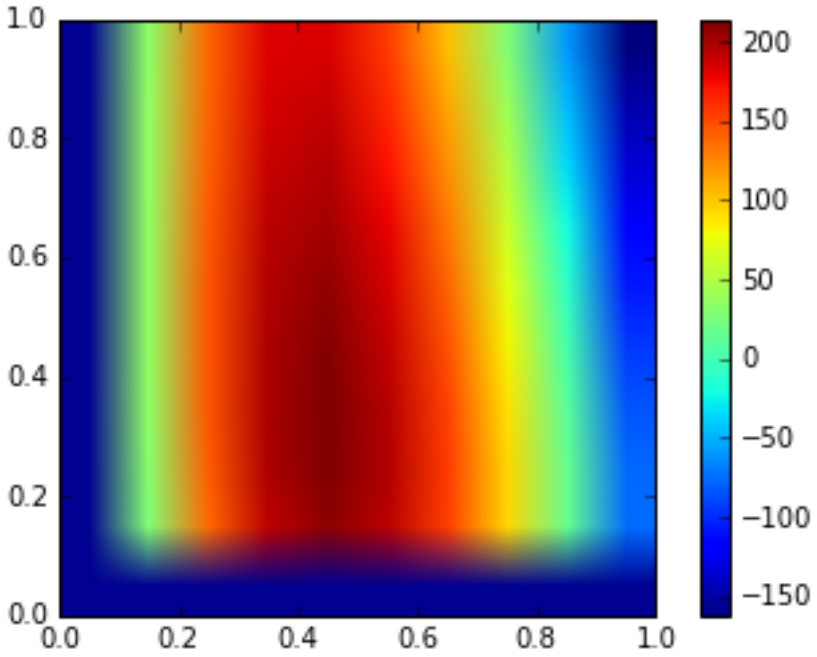
— D'un point de vue numérique, comment maximiser efficacement cette fonction de 3 variables ?

— Quelle confiance peut-on accorder à ces estimations ?

Afin d'estimer les paramètres sur des données représentatives, nous nous assurons au préalable d'être en régime stationnaire. En pratique, cela revient à simuler le processus sur l'intervalle $[0, T]$ et à l'estimer sur l'intervalle $[T - C, T]$ pour un réel $C \in [0, T]$ bien choisi. On constate que $C = 100$ fournit de bons résultats. De plus, si nous supposons connaître déjà λ_0, nous pouvons ramener le problème à l'estimation de α et β et en améliorer la précision. Comparons ces deux estimations :

```
T = 1550
Estimation avec deux inconnues A et B
Valeur réelle de A : 0.55 ;valeur estimée : 0.563782502702
Valeur réelle de B : 0.22 ;valeur estimée : 0.270749153214
Estimation avec trois inconnues Lambda_0, A et B
Valeur réelle de A : 0.55 ;valeur estimée : 0.586555755339
Valeur réelle de B : 0.22 ;valeur estimée : 0.259704133732
Valeur réelle de Lambda_0 : 0.85 ;valeur estimée : 0.802286895551
```

Nous traçons la *heatmap* ou « carte thermique » de la log-vraisemblance en fonction de (α, β) (à $\lambda_0 = 1$ fixé) ci-dessous :



Nous constatons que lorsque le point de départ est très proche du maximum estimé, la carte affiche un gradient très faible par rapport à la vraisemblance. Ceci est problématique pour calculer le maximum, et la précision n'est généralement pas très bonne. Le résultat dépend également de l'algorithme de maximization utilisé. Un algorithme de Nelder-Mead donne généralement de bons résultats lorsque le polygone a suffisamment de points.

Ainsi, avec une bonne technique de maximization, l'estimateur du maximum de vraisemblance peut donner de bons résultats tant que le nombre de paramètres à estimer est faible. Dès que le nombre de composantes du processus ponctuel augmente, ou s'il y a trop de paramètres, la précision de l'estimation n'est pas satisfaisante. En particulier, l'estimation de l'effet des features est délicate. Bien que les fréquences empiriques permettent d'estimer aisément leur distribution, ce n'est pas le cas pour leurs effets.

3 Simulation d'évènement rare

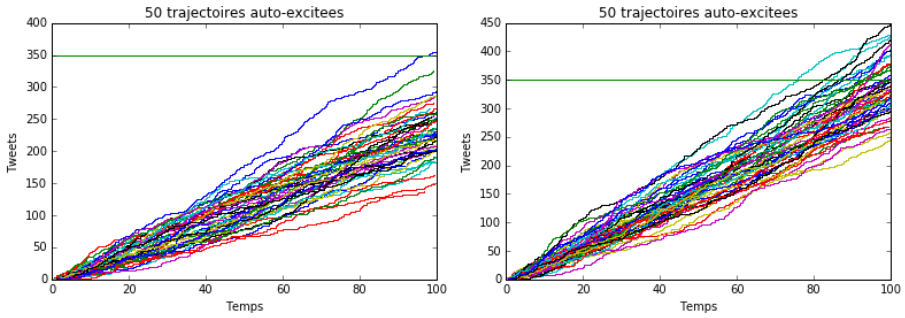


FIGURE 3.1 – A gauche, nous souhaitons simuler une « twitpocalypse », mais nous remarquons que nous franchissons trop rarement le seuil nécessaire. Il faudrait donc un nombre trop élevé de simulations pour obtenir une estimation correcte de la probabilité d'une « twitpocalypse ». A droite, nous simulons le même processus sous une nouvelle probabibilité où nous avons changé la valeur du paramètre λ_0, de telle sorte que la moyenne du nombre de tweets au temps T soit égal au seuil nécessaire. Ainsi l'événement devient commun et nous obtenons une meilleure estimation de la probabilité cherchée.

Jusqu'en 2010, chaque tweet était codé sur un un entier de 32 bits. Ceci limitait le nombre d'identifiants possibles des tweets à $2,147,483,647$. La "Twitpocalypse", survenue en juin 2009, correspond au dépassement de ce nombre. Cet évènement a engendré un dysfonctionnement partiel de certains serveurs de Twitter. Le référencement s'effectue depuis sur 64 bits. Dans cette partie, nous cherchons à estimer la probabilité d'une nouvelle "Twitpocalypse" en fonction de la période considérée. Une méthode de Monte Carlo classique s'avère inopérante, car pour obtenir un intervalle de confiance raisonnable, il faudrait considérer un nombre trop élevé de simulations. Nous adoptons donc une technique de réduction de variance par changement de probabilité.

3.1 Changement de probabilité d'un processus de Hawkes

Notons n le nombre d'identifiants possibles et $[0,T]$ la période de temps. L'événement "twitpocalypse" est $\{N_T > S\}$ pour $S = 2^{32}$. Tachons d'exprimer sa probabilité et d'en donner un intervalle de confiance. Pour obtenir un résultat probant, il faut changer la probabilité dans le but de rendre l'événement moins rare.

Rappelons la formule de changement de probabilité. Si X est une variable aléatoire simulée sous la loi $\mathbb{P}$ et $\tilde{X}$ désigne la même variable simulée sous la loi $\mathbb{Q}$, alors pour toute fonction borélienne f,

$$\mathbb{E}_{\mathbb{P}}(f(X)) = \mathbb{E}_{\mathbb{Q}}(f(\tilde{X})\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}))$$

où $\frac{d\mathbb{P}}{d\mathbb{Q}}$ désigne la vraisemblance (ou densité) de la loi $\mathbb{P}$ par rapport à la loi $\mathbb{Q}$.

Ainsi, au lieu d'utiliser une méthode de Monte Carlo classique pour estimer $\mathbb{P}(N_T > n)$, nous l'utilisons sous la nouvelle loi. La loi des grands nombres donne alors

$$\frac{1}{N}\sum_{k=1}^{N} 1_{\{\tilde{N}_T^{(k)} > S\}} \frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}^{(k)}) \to \mathbb{P}(N_T > S)$$

où les $\tilde{X}^{(k)}$ sont des copies i.i.d des processus de Hawkes simulés sous la nouvelle loi $\mathbb{Q}$.

Intéressons-nous au calcul de la vraisemblance. La nouveauté par rapport à la transformation d'Escher décrite dans le cours réside dans le caractère aléatoire de l'intensité de notre processus. Nous allons revenir aux sources du changement de probabilité en utilisant les vraisemblances des processus de Hawkes. Suivant [2], la vraisemblance d'un processus de Hawkes d'intensité $\lambda(t)$ et de durée T est donnée par :

$$L_\lambda = exp(\int_0^T (1-\lambda(s))ds + \int_0^T ln(\lambda(s))dN_s)$$

$$= exp(T - T\lambda_0 - \sum_{i=1}^{n} \alpha(1 - e^{-\beta(T-t_i)}) + \sum_{i=1}^{n} ln(\lambda(t_i)))$$

où $(t_i)_{1\leq i\leq n}$ sont les temps des sauts du processus N_T. t_n est défini par $max\{t_i/t_i < T\}$, c'est à dire $t_n = t_{N_T}$.

Le rapport de vraisemblance $\frac{d\mathbb{P}}{d\mathbb{Q}}$ voulu est alors donné par

$$\frac{d\mathbb{P}}{d\mathbb{Q}} = \frac{L_\lambda}{L_{\tilde{\lambda}}}$$

où λ et $\widetilde{\lambda}$ sont les intensités du processus N_T respectivement sous la loi $\mathbb{P}$ et la loi $\mathbb{Q}$.

Reste à choisir comment changer l'intensité du processus. Le premier changement envisageable est de multiplier l'intensité λ_0 du processus non excité par une constante. L'intérêt de cette transformation est qu'elle ne modifie pas la stationnarité du processus, puisque la valeur de α ne change pas dans la nouvelle probabilité. Dans ce cas on a :

$$\frac{d\mathbb{P}}{d\mathbb{Q}} = \prod_{t_i<T} (\frac{\lambda(t_i)}{\widetilde{\lambda}(t_i)}) exp((e^c - 1)\lambda_0 T)$$

Un autre changement possible est de multiplier toute l'intensité $\lambda(t)$ du processus par un constante e^c. On obtient alors :

$$\log \frac{d\mathbb{P}}{d\mathbb{Q}} = (e^c - 1)(T\lambda_0 - \sum_{i=1}^{n} \alpha(1 - e^{-\beta(T-t_i)})) - ne^c$$

3.2 Mise en œuvre

L'idée du changement de probabilité est de rendre l'évènement "Twitpocalypse" plus fréquent sous la nouvelle probabilité. Pour cela, nous faisons appel à un changement de probabilité de manière à avoir :

$$\mathbb{E}_{\mathbb{Q}}(\tilde{N}_T) = S$$

où S est, on le rappelle, le seuil de la "twitpocalypse".

Examinons les changements de probabilité possibles. Rappelons que pour un processus de Hawkes,

$$\mathbb{E}(N_T) = \frac{\lambda_0 T}{1-\alpha}$$

Dans le cas de la transformation $\lambda_0 \leftarrow e^c \lambda_0$, le choix $e^c = \frac{S(1-\alpha)}{\lambda_0 T}$ est immédiat.

Dans le cas où $\lambda(t) \leftarrow e^c \lambda(t)$, le choix de $e^c = \frac{S}{\lambda_0 T + \alpha S}$ est le seul changement possible. Mais ici une difficulté supplémentaire survient : nous devons nous assurer que la stationnarité asymptotique est conservée. Notons $f_{\mathbb{Q}}$ la fonction telle que $f_{\mathbb{Q}}(e^c) = \mathbb{E}_{\mathbb{Q}}(\tilde{N}_T)$. $f_{\mathbb{Q}}$ est une fonction continue, strictement croissante sur $[0; \log(\frac{1}{\alpha})[$, nulle en 0 et infinie en $\log(\frac{1}{\alpha})$. $f_{\mathbb{Q}}$ est donc bijective de $[0; \log(\frac{1}{\alpha})[$ sur $[0; +\infty[$, ce qui assure l'existence et l'unicité de e^c tel que $\mathbb{E}_{\mathbb{Q}}(\tilde{N}_T) = S$. Si $e^c \in [0; \log(\frac{1}{\alpha})[$ alors $e^c \alpha < 1$. Nous en déduisons que le changement de probabilité conserve la stationnarité asymptotique.

3.3 Résultats de la simulation

Nous simulons donc sous Python le changement de variable pour estimer la probabilité de la « twitpocalypse ». Nous comparons ici les résultats obtenus avec une méthode de Monte Carlo naïve et avec nos deux changement de probabilité. Nous nous restreignons à des intervalles de temps faibles afin de limiter le temps d'exécution. De même, nous choisissons des seuils relativement faibles comparativement à celui de la « twitpocalypse » réelle.

Nous effectuons 500 simulations avec le jeu de paramètres $T = 100, \alpha = 0.5, \beta = 0.3, \lambda_0 = 1$. Entre parenthèses est donnée la largeur de l'intervalle de confiance.

Seuil	Monte Carlo naïf	Changement $\lambda_0 \leftarrow \lambda_0 e^c$	Changement $\lambda(t) \leftarrow \lambda(t) e^c$
150	0.31 (0.04)	0.31 (0.03)	0.31 (0.03)
175	0.03 (0.014)	0.027 (0.0038)	0.029 (0.004)
200	0.0 (0.0)	0.00082 (0.00016)	0.00099 (0.00016)
250	0.0 (0.0)	2.09e-08 (6.56e-09)	2.91e-08 (6.94e-09)

Nous constatons comme attendu une réduction de la variance pour ces événements rares. Dans le cas où le seuil est trop élevé, la méthode naïve se révèle complètement inefficace.

4 Modèles avec features

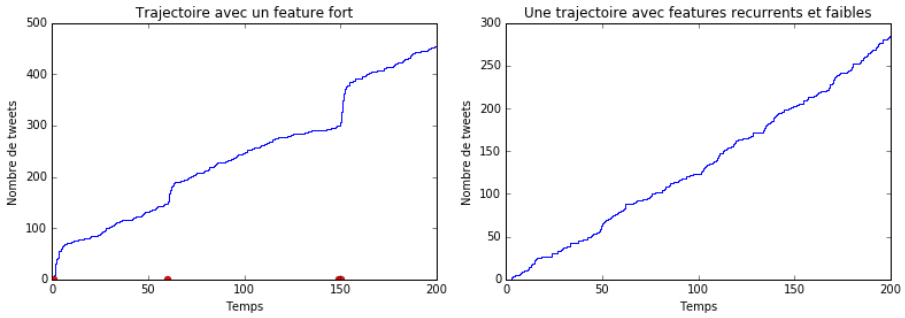


FIGURE 4.1 – Un exemple de trajectoires avec des tweets à « features ». Les tweets peuvent posséder une dizaine de features différentes. En rouge à gauche sont représentés quelques tweets « rares », responsables des « sauts » dans le processus de comptage.

Les features sont des "mots" présents dans les tweets qui impactent le flux selon leur présence. Nous considérons dans cette partie un ensemble de K features possibles. On prend $K = 10$ dans nos simulations.

Notons $(w_i)_{i \leq K}$ l'impact des features sur le flux de tweets. L'idée est d'associer une faible probabilité d'apparition aux tweets ayant un fort impact et une forte probabilité aux tweets ayant un faible impact. Dans un premier temps, nous présentons un modèle simple sans mémoire. Bien que simpliste, ce modèle a un avantage : le calcul de la log-likelihood (indispensable pour mener à bien les changements de probabilité) y est particulièrement aisé.

Notons $h_{(t,x)}(t', x')$ l'impact du tweet (t, x) sur le tweet (t', x') (en supposant $t < t'$). Dans le modèle sans mémoire, il n'y a pas de dépendance entre les tweets et cette fonction devient simplement $h(t', x')$.

4.1 Modèle sans mémoire

Le principe de ce modèle est de considérer que tous les features contenus dans les tweets (ou retweets) sont indépendants. Ce modèle simpliste permet d'observer des premiers résultats. Nous supposerons que chacune des K features apparait avec une probabilité p_i. Pour cela nous choisissons comme fonction g :

$$g(t, x) = 1_{t \geq 0}(\sum_{i \leq K} w_i x_i) \alpha e^{-\beta t}$$

où $X = (x_i)_{1 \leq i \leq K}$ est un vecteur binaire représentant l'ensemble des features présentes dans le tweet émis à l'instant t. Chaque x_i suit une loi de Bernoulli de paramètre p_i. Nous notons $h(x) = x^T \cdot \vec{w} = (\sum_{i \leq K} w_i x_i)$.

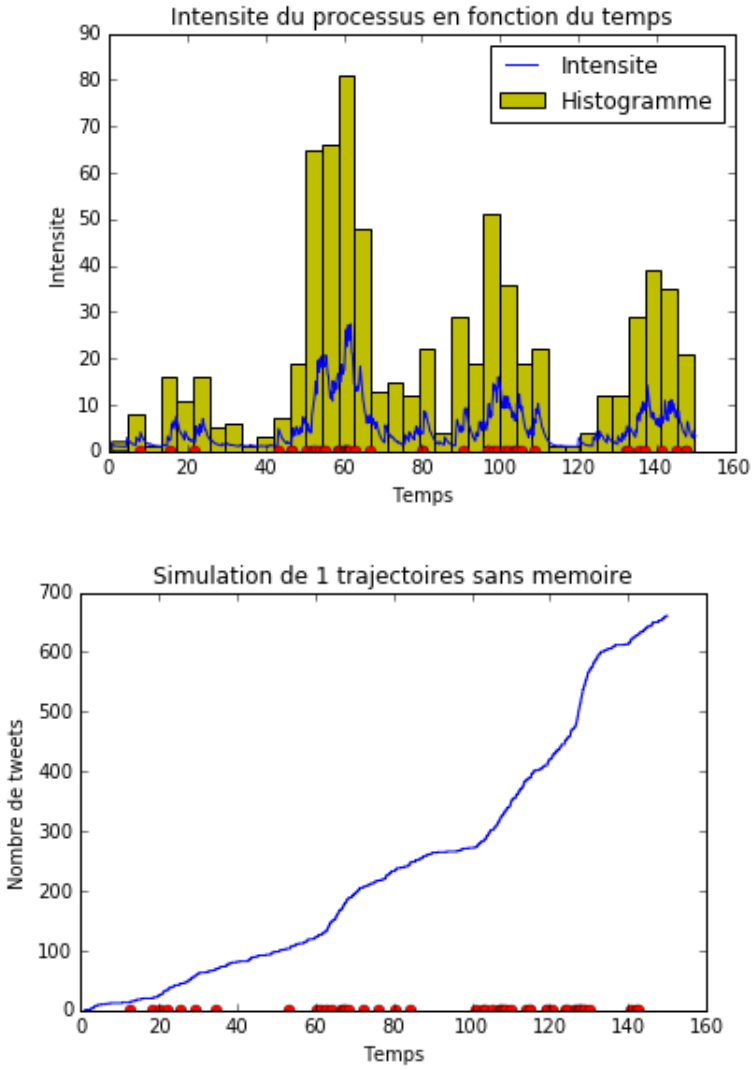


FIGURE 4.2 – En haut : histogramme d'une trajectoire et de l'intensité correspondante. En rouge sur l'axe des abcisses sont représentés les tweets à « fort impact ». Chaque colonne verte représente le nombre de tweets apparus sur l'intervalle de temps délimité par la largeur de la colonne. En bleu est représentée l'intensité $\lambda(t)$. On constate bien des « sauts » d'intensité lors de l'apparition de tweets « rares ». En bas : simulation de la trajectoire correspondante.

Intéressons-nous à la stationnarité du nouveau processus. L'espérance du nombre de tweets résultant d'un tweet (t, x) dépend désormais des features x , et il s'agit de vérifier que $N(x) < +\infty$ pour tout $x \in \{0,1\}^K$, où $N(x) = 1 + \int_{\mathbb{R}\times F} h_{(0,x)}(t', x')d(t', x')$. Cette condition est contraignante car elle limite l'impact possible pour les tweets à fort impact. Il s'avère que des écarts à cette condition de stationnarité sont tolérables dans le sens suivant.

Proposition. *(Stationnarité ; espérance de l'intensité) La condition de stationnarité est :*

$$\alpha\mathbb{E}(h(x)) = \alpha(\sum_{i\leq K} p_i w_i) < 1$$

De plus, nous avons :

$$\mathbb{E}(\lambda(t)) = \frac{\lambda_0}{1 - \alpha\mathbb{E}(h(x))}$$

Démonstration. Il suffit de reprendre la preuve dans le cas sans features.

Ainsi, nous pouvons autoriser certains tweets à afficher un impact fort tant qu'ils restent "suffisamment rares". □

Simulation d'événements rares dans le modèle avec features

Nous devons désormais prendre en compte dans notre changement de probabilité la présence des features et leur caractère aléatoire.

Notons θ l'ensemble des paramètres, c'est-à-dire $\lambda(t)$ et les valeurs des $(p_i)_{i\leq K}$. Notons également $x_i^{(j)}$ l'indicatrice de la présence de la i-ème feature dans le j-ième tweet.

Ainsi la vraisemblance L_θ de ce processus vaut :

$$L_\theta = L_\theta((N_t)_{t\leq T}|((x^{(j)})_j) \cdot L_\theta((x^{(j)})_j)$$

avec

$$L_\theta((N_t)_{t\leq T}|(x^{(j)})_j) = exp(\int_0^T (1 - \lambda(s))ds + \int_0^T ln(\lambda(s))dN_s)$$

la vraisemblance conditionnellement aux nouveaux paramètres introduits dans cette partie. De plus, l'absence de mémoire implique que

$$L_\theta((x_j)_j) = \prod_{i=1}^K p_i^{\sum_{j\leq n} x_i^{(j)}} (1 - p_i)^{n - \sum_{j\leq n} x_i^{(j)}}$$

est la vraisemblance de la présence des features dans les tweets.

Donnons également une propriété importante pour le calcul de l'espérance conditionnelle, dite formule de Bayes abstraite avec conditionnement, utile ici pour l'estimation d'événements rares.

Proposition. *(Espérance conditionnelle avec changement de probabilité)*
Soit f une fonction mesurable, X une variable aléatoire telle que $f(X) \in \mathbb{L}^1$. Soit F une algèbre.
Nous avons :

$$\mathbb{E}_{\mathbb{P}}(f(X)|F) = \frac{\mathbb{E}_{\mathbb{Q}}(f(\tilde{X})\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X})|F)}{\mathbb{E}_{\mathbb{Q}}(\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X})|F)}$$

Pour estimer $\mathbb{P}(N_T > S)$, nous utilisons ainsi l'estimateur suivant :

$$\mathbb{P}(N_T > S) \approx \frac{\frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F} 1_{\tilde{X}_i>S}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i)}{\frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i)}$$

Deux problèmes se présentent alors :
— L'événement $\tilde{X}_i \in F$ est l'événement d'apparition d'une feature de fort impact. Il faut donc rendre cet évènement peu rare, voire commun pour éviter d'avoir une somme vide. Pour cela, nous changeons les probabilité d'apparition des features à fort impact tout en conservant la stationnarité asymptotique.
— Il faut aussi donner un intervalle de confiance à cette estimateur. Nous appliquons pour cela une *delta-méthode* au couple

$$(\frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F} 1_{\tilde{X}_i>S}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i), \frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i))_M$$

Théorème. *(Delta-méthode)*
Soit $X_n$une suite de variables aléatoire vérifiant $\sqrt{n}(X_n - m) \to N(0, \Sigma)$ en loi avec Σ la matrice de covariance. Soit g une fonction dérivable en m vérifiant $g'(m)$ non nulle. On a alors :

$$\sqrt{n}(g(X_n) - g(m)) \to N(0, g'(m)^T\Sigma g'(m)) \text{ en loi}$$

Nous utilisons la delta-méthode pour le couple

$$(X_n, Y_n) = (\frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F} 1_{\tilde{X}_i>S}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i), \frac{1}{M}\sum_{i\leq M/\tilde{X}_i\in F}\frac{d\mathbb{P}}{d\mathbb{Q}}(\tilde{X}_i))_M$$

et $g(x, y) = \frac{x}{y}$. Nous ferons attention à ne ce que les variables ne soient pas corrélées pour éviter les termes non-diagonaux dans la matrice de covariance. Pour cela nous simulerons ces variables avec des copies indépendantes $(X_m)_m$et $(Y_m)_m$.

Dans notre cas, si $(X_n, Y_n) \to (x, y)$ alors, par le théorème centrale limite appliquée à ce couple de variables indépendantes :

$$\sqrt{n}((X_n, Y_n) - (x, y)) \to N(0, \begin{pmatrix}\sigma_1^2 & 0\\ 0 & \sigma_2^2\end{pmatrix}) \text{ en loi.}$$

De plus g est différentiable et vérifie :

$$g'(x,y) = \begin{pmatrix} \frac{1}{y} & -\frac{x}{y^2} \end{pmatrix}$$

Ainsi nous en déduisons par la delta-méthode :

$$\sqrt{n}(\frac{X_n}{Y_n} - \frac{x}{y}) \to N(0, \frac{\sigma_1^2}{y^2} + \frac{x^2}{y^4}\sigma_2^2)$$

Nous sommes ainsi en mesure de donner un intervalle de confiance asymptotique pour l'estimation de $\mathbb{P}(N_T > S)$.

Nous simulons sous Python cette nouvelle méthode pour estimer la probabilité de la « twitpocalypse » conditionnellement à la présence d'une feature de fort impact. Nous présentons ci-dessous les résultats obtenus avec le changement de probabilité. Le Monte Carlo naïf se révèle immédiatement inefficace au vu des seuils demandés.

Nous effectuons 500 simulations avec le jeu de paramètres $T = 100, \alpha = 0.2, \beta = 0.2, \lambda_0 = 1$. La probabibilité d'apparition du features rares est de 0.001. Entre parenthèses est donnée la largeur de l'intervalle de confiance.

La simulation de ce modèle sans mémoire ne change pas les grandes lignes des simulations précédentes. On constate toutefois que l'impact des features rend la contrainte de stationnarité très restrictive : le temps limite est souvent atteint en un temps trop long. En pratique, ceci implique que l'estimateur obtenu a un intervalle de confiance médiocre. Il convient d'augmenter l'horizon de temps T parallèlement au nombre de simulations afin d'améliorer l'estimateur.

Seuil	Changement $\lambda_0 \leftarrow \lambda_0 e^c$
200	0.0035 (0.002)
175	0.00015 (0.0003)
200	2.2e-07 (4.6e-07)
250	2.5e-08 (4.4e-08)

4.2 Modèles avec mémoire

Afin de tendre vers plus de réalisme dans la modélisation de l'impact des features, nous proposons des modèles avec mémoire qui introduisent une dépendance entre les tweets précédents et le tweet actuel. Les tweets peuvent désormais être appariés à un parent, tirés parmi les tweets à fort impact. On conçoit en effet que seuls ces derniers sont intéressants pour le caractère d'auto-excitation, car un tweet à faibles features a peu de chances d'être retweeté. Nous ignorons donc ces derniers et considérons uniquement les tweets dont l'impact dépasse un certain seuil noté seuilImpact (à fixer comme paramètre du modèle), comme candidats potentiels pour être parents.

Nous distinguons trois types possibles de tweets :

— Le type NEW correspond à un tweet indépendant de tous les tweets du passé. Ceci revient à le simuler selon le modèle sans mémoire précédent.

— Le type RETWEET, noté RT, correspond à une recopie exacte d'un tweet passé. Il s'agit de reprendre les mêmes features que celle du "tweet-père". En pratique, ceci revient à affecter au nouveau tweet exactement le même impact que celui de son père (puisque l'effet des features est entièrement résumé dans leur fonction d'impact).
— Le type RESPONSE, noté RSP, correspond à un tweet-réponse. Celui-ci recopie une partie des features de son tweet-père, et génère l'autre partie indépendemment. Ceci revient à affecter au nouveau tweet (t_n, x_n) dont le père est le tweet (t_i, x_i) un impact $h(t_n, x_n) = ah(t_i, x_i) + (1-a)h$, où l'impact h, aléatoirement choisi selon la distribution des features, correspond à celui d'un tweet de type NEW. La modification du poids $a \in [0, 1]$ permet de faire varier la "corrélation" entre le "tweet-fils" et son "tweet-père". Le cas $a = 1$ correspond au type RETWEET.

On néglige la possibilité d'appariement des "retweets" et des tweets-réponses, quand bien même ces derniers seraient à fort impact (c'est-à-dire qu'ils contiennent des features à fort impact). Ceci traduit la réalité de la diffusion des messages sur Twitter, où un retweet est moins populaire que le tweet original, et a moins de chances d'être rediffusé.

4.2.1 Modèle à mémoire markovienne

Dans ce premier modèle, le type d'un tweet est choisi uniformément parmi NEW/RT/RSP selon une variable indépendante du temps. De plus, dans les cas RT/RSP, le parent du tweet est le dernier tweet fort qui le précède. Nous appellerons *"père"* ce tweet fort.

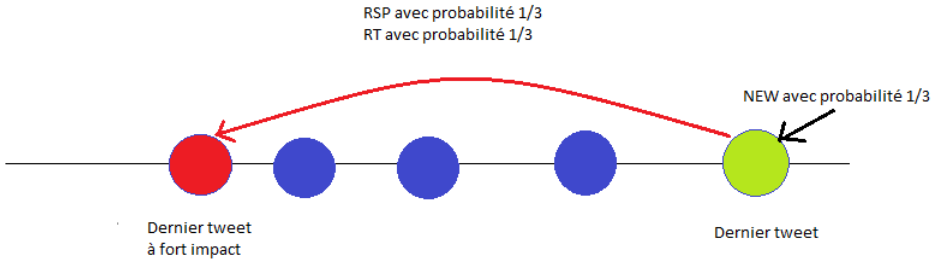


FIGURE 4.3 – Illustration du modèle markovien. Les boules représentent les tweets en fonction du temps.En vert est représenté le dernier tweet à être simulé. Les tweets à faible impact, ici en bleu, ne sont pas considérés comme parents éligibles. En rouge est représenté le processus d'appariement avec le dernier parent fort.

La condition de stationnarité nous semble difficile à calculer ici, et dépend (entre autres) du seuil fixé pour choisir les tweets éligibles à être parents.

En revanche, nous pouvons expliciter la vraisemblance :

$$L_\theta = L_\theta((N_t)_{t\leq T}|(h(x^{(j)}))_j) \cdot L_\theta(h((x^{(j)}))_j)$$

où $h(x^j)$ est, on le rappelle, l'impact du tweet j.
Reste à calculer le deuxième terme de droite :

$$L_\theta((h(x^{(j)}))_j) = \prod_j L_\theta(h(x^{(j)})|h(x^{(j-1)}), ..., h(x^{(1)}))$$

Par construction du modèle :

$$L_\theta(h(x^{(j)})|h(x^{(j-1)}), ..., (x^{(1)})) = \frac{1}{3}L_\theta(h(x^{(j)})|NEW, h(x^{(j-1)}), ..., h(x^{(1)}))$$

$$+\frac{1}{3}L_\theta(h(x^{(j)})|RT, h(x^{(j-1)}), ..., h(x^{(1)}))+\frac{1}{3}L_\theta(h(x^{(j)})|RSP, h(x^{(j-1)}), ..., h(x^{(1)}))$$

Or nous avons :

$$L_\theta(h(x^{(j)})|RT, h(x^{(j-1)}), ..., (x^{(1)})) = 1_{x^{(j)}=x^{père}}$$

$$L_\theta(h(x^{(j)})|NEW, h(x^{(j-1)}), ..., (x^{(1)})) = \frac{1}{2^K}\sum_{J\subset[1;K]} 1_{\{\sum_{j\in J} w_j = h(x^{(j)})\}}$$

$$L_\theta(h(x^{(j)})|RT, h(x^{(j-1)}), ..., (x^{(1)})) = \frac{1}{2^K}\sum_{J\subset[1;K]} 1_{\{\sum_{j\in J} w_j = \frac{h(x^{(j)})-ah(x^{(père)})}{1-a}\}}$$

Cette estimation théorique nous paraît inapplicable en pratique, et nous ne simulons pas d'évènements aléatoires sur ce modèle.

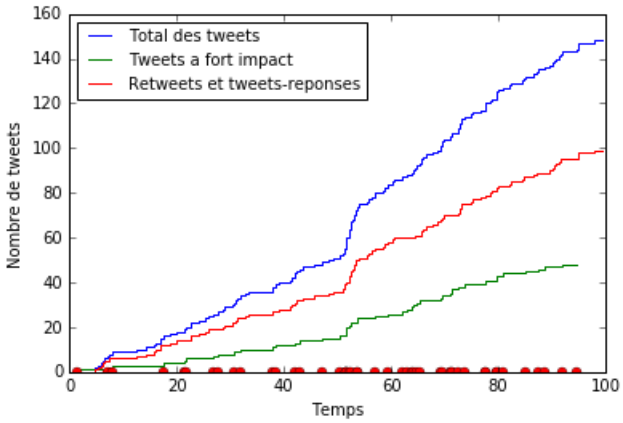


FIGURE 4.4 – Un exemple d'auto-excitation markovienne : la proportion est maintenue constante entre les trois types NEW,RSP,RT. Les points rouges représentent les tweets à forts impacts éligibles pour être « parents ».

4.2.2 Modèle à mémoire conditionnelle

Ce dernier modèle raffine en deux points du précédent :

— D'abord, on constate que la proportion de tweets du type NEW reste constante, proche de $\frac{1}{3}$, dans le modèle markovien. En pratique, l'explosion du nombre de tweets devrait principalement être due aux retweets et aux réponses. Il s'agit donc de tirer le type du tweet actuel plus ou moins fréquemmement selon le nombre de tweets à fort impact le précédant. Nous proposons la modélisation suivante :

$$\mathbb{P}(\text{type}(t_n, x_n) = \text{NEW}) = \frac{1}{3}(1 - a(\frac{|\{(t_i, x_i)/\text{impact(i)}>\text{seuilImpact}\}|}{n})^b$$

avec les paramètres $a, b > 0$. En pratique, le choix $a = 1, b = \frac{1}{2}$ donne des résultats satisfaisants : la proportion de tweets NEW diminue progressivement.

— On souhaite désormais choisir librement le père du tweet (t_n, x_n) parmi tous les tweets dont l'impact est $>$ seuilImpact. Nous proposons la formule suivante :

$$\mathbb{P}(\text{parent}(t_n, x_n) = (t_i, x_i)) = \frac{\text{impact}(x_i)}{t_n - t_i}$$

Ainsi, nous favorisons d'autant plus un tweet-père éventuel que son impact est élevé, et le pénalisons d'autant plus qu'il est éloigné dans le temps :

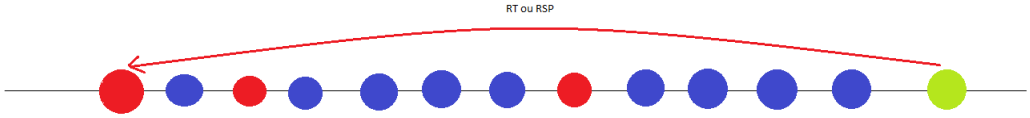


FIGURE 4.5 – Illustration du modèle à mémoire conditionnelle. Le tweet parent n'est plus nécessairement le dernier tweet « rare », mais est choisi parmi tous les parents forts, et ce d'autant plus que son impact est élevé.

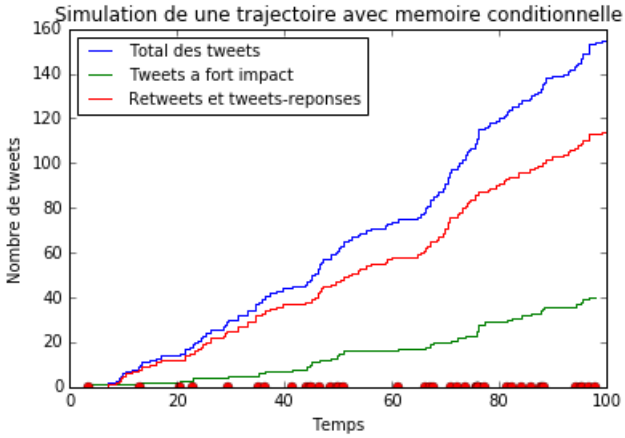


FIGURE 4.6 – Simulation d'une trajectoire avec mémoire conditionnelle : dans ce modèle plus réaliste, la proportion des tweets de type NEW diminue face aux « retweets » et tweets-réponses.

Références

[1] A. Simma & M. Jordan. Modeling Events with Cascades of Poisson Processes. *arXiv :* 1203.3516, 2012.

[2] I.M. Toke. An Introduction to Hawkes Processes with Applications to Finance. 4 février 2011.

[3] Y. Ogata. On Lewis's Simulation Method for Point Processes. *IEEE Transactions on Information Theory*, 1981.

[4] A. Simma, Modeling Events in Times using Cascades of Poisson Processes. *UC Berkeley Electronic Theses and Dissertations.* `http://escholarship.org/uc/item/2np020kh`.

[5] D. MacKinlay. Estimating self-excitation effects for social media using the Hawkesprocess. Master Thesis, Swiss Federal Institute of Technology Zurich, 11 mai 2015.

[6] T. Ozaki. Maximum Likelihood Estimation of Hawkes' Self-Exciting Point Processes. *Ann. Inst. Statist. Math 31.* 1979