Détection de ruptures dans des processus de Poisson - Stéphane Robin
Compte-rendu du séminaire

Louis Lacoste

30 Jan. 2024

Ce compte-rendu concerne la présentation Robin, 2019.

1 Introduction

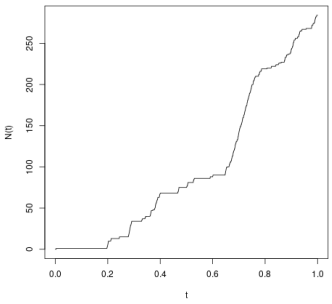


Figure 1 – Comptage de cris de chauve-souris (nuit du 17 juillet 2019)

Stéphane Robin nous présente une méthode qu'ils ont développé avec Emilie Lebarbier et Charlotte Dion-Blanc. Cette méthode est présentée dans l'article Dion-Blanc et al., 2023 sur HAL.

La méthode considère des données de comptages au cours du temps. Un exemple de telles données est celui du nombre de cris de chauve-souris au cours de la nuit. Ces données sont présentées dans sur la figure 1. Ou encore les données d'éruption du volcan Kilauea présentée sur la figure 2.

L'intervalle de temps est normalisé, $t \in [0,1]$ et les instants d'évènements sont les $0 < T_1 < \dots T_i < \dots T_n < 1$. Étant donné qu'il s'agit d'un comptage aléatoire, le processus *naturel*[1] est le processus de comptage, $N(t) = \sum_{i=1}^{n} \mathbb{1}_{T_i \leq t}$ et parmi les processus de comptage, le processus de Poisson défini par sa fonction d'intensité $\lambda(t)$.

2 Méthode

La méthode fait l'hypothèse que la fonction d'intensité est constante par morceaux et qu'il existe des *points de ruptures* les $(\tau_k)_{0 \leq k \leq K}$. Et alors pour $t \in I_k =]\tau_{k-1}; \tau_k], \lambda(t) = \lambda_k$. Ainsi l'objectif de la méthode est d'estimer les paramètres $\theta = ((\tau_k)_{0 \leq k \leq K}, (\lambda_k)_{0 \leq k \leq K})$ et de réaliser une *sélection de modèle* pour obtenir le nombre de segments K.

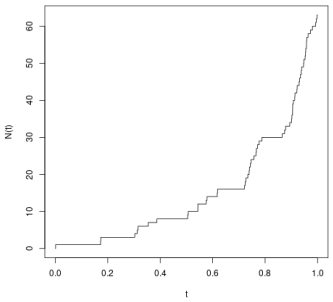


Figure 2 – Données d'éruption du Kilauea, 1750 - 1984

2.1 Segmentation

Un rappel sur la segmentation en temps discret, nous montre que la programmation dynamique permet ainsi de résoudre le problème d'estimation des paramètres dans ce cas qui semblait apparemment computationnellement complexe.

Dans le cas de la méthode, c'est à dire le temps *continu*, le problème d'optimisation est

$$(\widehat{\boldsymbol{\tau}}, \widehat{\boldsymbol{\lambda}}) = \underset{\boldsymbol{\tau} \in \mathcal{T}_K, \boldsymbol{\lambda} \in (\mathbb{R}^+)^K}{\operatorname{argmin}} \gamma(\boldsymbol{\tau}, \boldsymbol{\lambda})$$

L'additivité du contraste : $\gamma(\boldsymbol{\tau}, \boldsymbol{\lambda}) = \sum_{k=1}^{K} C(\Delta N_k, \Delta\tau_k, \lambda_k)$ en tant que somme sur les segments aide à la résolution du problème d'optimisation. En effet le $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_K)$ optimal peut être estimé grâce à la propriété d'additivité en résolvant $\widehat{\lambda}_k = \lambda_k(\tau) = \operatorname{argmin}_{\lambda_k \in \mathbb{R}^+} C(\Delta N_k, \Delta\tau_k)$. Et si la fonction de contraste est la log-vraisemblance négative, on a : $\widehat{\lambda}_k = \Delta N_k / \Delta\tau_k$.

Mais le problème difficile est celui de trouver le $\boldsymbol{\tau} = (\tau_0, \dots, \tau_K)$ optimal, car le problème d'optimisation est alors $\widehat{\tau} = \operatorname{argmin}_{\tau \in \mathcal{T}_K} \widehat{\gamma}(\tau), \widehat{\gamma}(\tau) = \gamma(\tau, \widehat{\lambda}(\tau))$ où $\mathcal{T}_K$ est l'espace de segmentation **continu**,

$$\mathcal{T}_K = \left\{\tau \in [0,1]^{K+1} : 0 = \tau_0 < \tau_1 \cdots < \tau_{K-1} < \tau_K = 1\right\}$$

Car le contraste n'est ni convexe ni continu par rapport à τ.

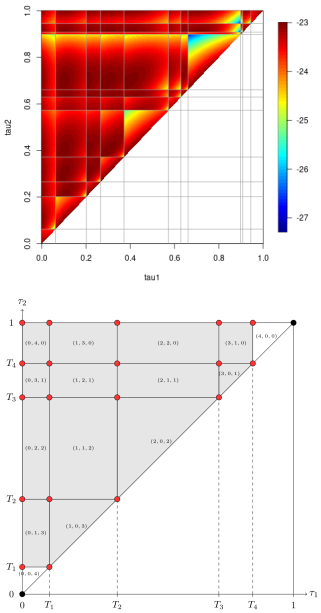
La figure 3 présente les valeurs de la fonction de contraste pour les paramètres donnés. Sur la figure, chaque "bloc" correspond à une partition des nombre d'évènements $\Delta N = (\Delta N_1, \Delta N_2, \Delta N_3)$.

1. Au sens du premier qui vienne à l'esprit.

L'idée est alors de partitionner le nombre d'évènements $\mathcal{N} = \left\{\nu \in \mathbb{N}^K : \sum_{k=1}^K \nu_k = n\right\}$ où ν_k est le nombre d'évènements dans le segment k. Puis à partir de cette partition, on peut partitionner l'espace de segmentation $\mathcal{T}(\nu) = \left\{\boldsymbol{\tau} \in \mathcal{T}_K : \Delta N = \nu\right\}$ qui satisfait la partition ν. Ce qui permet de réécrire le problème de minimisation :

$$\min_{\boldsymbol{\tau}\in\mathcal{T}_K} \widehat{\gamma}(\boldsymbol{\tau}) = \min_{\nu\in\mathcal{N}^K} \min_{\boldsymbol{\tau}\in\mathcal{T}(\nu)} \widehat{\gamma}(\boldsymbol{\tau})$$

FIGURE 3 – Fonction de contraste et partitionnement de l'espace pour $n = 10$, $K = 3$ et $\tau = (\tau_1, \tau_2)$



Et alors grâce aux propriétés de stricte concavité des fonctions de coût C par rapport aux $\Delta\tau_k$ les auteurs montrent que $\hat{\tau} = \text{argmin}_{\tau\in\mathcal{T}_K} \hat{\gamma}(\boldsymbol{\tau}) \subset \left\{T_1^-, T_1, \ldots, T_n^-, T_n\right\}$ [2]. Et ainsi, la programmation dynamique permet de trouver les points de ruptures avec une complexité au plus $\mathcal{O}(n^2)$.

Un contraste concave par rapport à $\Delta\tau$ est dit admissible (par exemple les contrastes de Poisson et Poisson-Gamma). Voici le contraste de Poisson :

$$C_P(\nu_k, \Delta\tau_k) = \nu_k(1 - \log\nu_k + \log\Delta\tau_k)$$

Et un contraste est dit désirable s'il n'autorise pas les segments de longueur nulle (et c'est le cas du Poisson-Gamma).

Voici donc la formule du contraste Poisson-Gamma :

$$C_{PG}(\nu_k, \Delta\tau_k) = -\log\Gamma(a + \nu_k) + (a + \nu_k)\log(b + \nu_k)$$

qui satisfait les deux propriétés, d'admissibilité et de désirabilité.

Celle-ci dérive du modèle Poisson-Gamma qui dit que pour chaque segment $1 \leq k \leq K$:

$$\lambda_k \overset{iid}{\sim} \mathcal{G}am(a, b), (N(t))_{t\in I_k} \sim PP(\lambda_k)$$

2.2 Sélection de modèle

Pour la sélection de modèle la propriété de *thinning* du processus de Poisson permet d'obtenir deux processus de Poisson indépendants et donc de pouvoir faire de la *cross-validation*.

En effet, en sélectionnant chaque point avec une probabilité f pour faire partie du jeu d'entraînement et en sélectionnant donc avec probabilité $1-f$ ceux faisant partie du jeu de test on obtient deux processus de Poisson indépendants.

$$N^L(t) \sim PP(f\lambda(t)) \perp\!\!\!\perp N^T(t) \sim PP((1-f)\lambda(t))$$

Et alors on peut répéter cette procédure M fois, estimer les paramètres sur le jeu d'apprentissage, puis moyenner les contrastes obtenus. Pour chaque m on a $\gamma_K^{T,m} = \gamma\left(N^T(t); \hat{\tau}^{L,m}, \frac{1-f}{f}\hat{\lambda}^{L,m}\right)$

On a alors accès à $\bar{\gamma}_K = \frac{1}{K}\sum_{m=1}^M \gamma_K^{T,m}$ et on peut alors sélectionner :

$$\hat{K} = \underset{K}{\text{argmin}}\, \bar{\gamma}_K$$

2.3 Extension : Processus de Poisson marqué

Le processus de Poisson marqué est une extension et la méthode de détection de rupture peut gérer ces données supplémentaires.

Un processus de Poisson est dit marqué si à chaque temps de saut T_i est associé une "marque". Pour l'exemple des volcans, il s'agit de la durée de l' éruption et pour les cris de chauve-souris, il peut s'agir soit de l'espèce ou encore de la durée du cri.

Mathématiquement, cela peut se noter :

$$(Y(t))_{0\leq t\leq 1} \sim MPP(\lambda(t), \mu(t)),$$

qui consiste en deux composantes :

$$(N(t))_{0\leq t\leq 1} \sim PP(\lambda(t)), X_i \sim \underbrace{\mathcal{F}}_{\text{loi de probabilité}} (\mu(T_i))$$

Et dans la présentation, on peut ainsi voir que sur un exemple d'éruption de l'Etna, la prise en compte des marques modifie l'estimation des paramètres. Ce qui dans le cas du volcan pourrait indiquer des modifications de son activité sous-jacente.

2. T_i^- représente l'instant juste avant le saut au temps T_i.

3 Apport personnel

Nous allons tout d'abord simuler un processus de Poisson inhomogène selon les conditions du modèle et voir ce que donne l'estimation avec le package `CptPointProcess`. Le code pour ce rapport est disponible sur https://github.com/Polarolouis/msv-seminaire-detction-rupture-processus-poisson.

La figure 4 présente les données simulées selon les paramètres suivants :

$$\lambda = (1, 3, 10, 3),\ \Delta N = (20, 20, 20, 20) \text{ et } N(1) = 80$$

Application de la méthode à données simulées

En donnant le nombre de segments

En fixant $K_{max} = 4$ et avec `selection = FALSE`. La méthode trouve donc les résultats suivant présentés dans la table 1.

begin	end	Dt	DN	lambda	code.end
0.00	0.58	0.58	20	35.28	-
0.58	0.74	0.15	19	120.64	-
0.74	0.79	0.05	20	310.92	.
0.79	1.00	0.21	20	94.55	.

TABLE 1 – Résultats de la méthode en donnant le nombre de segment

begin	end	Dt	DN	lambda	code.end
0.00	0.58	0.58	20	35.28	-
0.58	0.66	0.07	16	194.93	.
0.66	0.74	0.08	3	43.84	-
0.74	0.79	0.05	20	310.92	.
0.79	1.00	0.21	20	94.55	.

TABLE 2 – Résultats de la méthode en *cross-validation*

Soit pour $\lambda = (0.96, 3.27, 8.43, 2.56)$ en **re-divisant par le temps maximal utilisé pour normaliser**. Et on peut donc constater que la méthode parvient à retrouver les valeurs des différents paramètres.

En sélectionnant $\hat{K}$ avec la *cross-validation*

En fixant seulement `selection = TRUE`. La méthode trouve donc les résultats suivant présentés dans la table 2.

Soit pour $\lambda = (0.96, 5.29, 1.19, 8.43, 2.56)$.

Remarque : pour $K = 4$ et $N(1) = 80$, l'estimation de $\hat{K}$ est un peu hors des clous, mais nous avons testé en augmentant le nombre d'évènements. On peut imaginer que le *thinning* peut augmenter la variabilité, avec l'effet attendu en diminuant le nombre données.

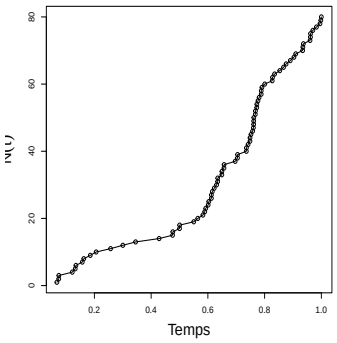


FIGURE 4 – Processus de Poisson inhomogène simulé

On peut donc voir que si la méthode sélectionne correctement le nombre de segment l'estimation de paramètres obtenue est alors bonne, même pour un faible nombre de données.

Analyse de données de SOUBEYRAND, 2024

Ces données sont extraite du travail de thèse toujours en cours de Lola Soubeyrand. Il s'agit de l'enregistrement d'une expérience de type bandit consistant à choisir un bras selon les récompenses passées obtenues.

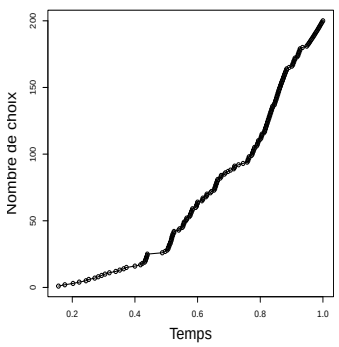


FIGURE 5 – Affichage des données de l'expérience

Nous allons seulement considérer les temps entre les choix pour réaliser notre analyse. Ces données sont présentées dans la figure 5.

Aux vues de l'expérience, on peut s'attendre à une phase exploratoire des différents bras du bandit (phase d'*exploration*) puis une fois le bras dont la stratégie convient au sujet trouvé à une concentration sur celui-ci (phase d'*exploitation*). À noter qu'avec les évènements rares et extrême des perturbations des convictions du sujet peuvent amener à des modifications du comportement. Ces analyses *a priori* nous invite à penser que l'on peut être dans le cas d'un processus de Poisson avec ruptures.

Et ainsi on trouve $\lambda = (0.1, 1.06, 9.28, 0.3, 4.85, 1.77, 6.76)$

En interprétant ces paramètres on peut voir deux premiers segments qui semblent correspondre à la phase exploratoire initiale, puis une intensité de 9

begin	end	Dt	DN	lambda	code.end
0.00	0.16	0.16	0	6.21	-
0.16	0.43	0.28	18	67.83	-
0.43	0.44	0.01	7	595.14	.
0.44	0.49	0.05	0	18.97	-
0.49	0.68	0.19	59	310.93	.
0.68	0.76	0.08	9	113.51	-
0.76	1.00	0.24	106	433.51	.

Table 3 – Résultats pour les données de Soubeyrand, 2024

qui correspond à la première phase d'exploitation. Et par la suite une cassure dûe à un évènement extrême négatif avant une reprise d'exploitation mixée à de l'exploration.

La détection de rupture pourra donc permettre d'analyser davantage les données d'autres sujets afin de voir si ce genre d'analyse peut fournir de plus amples information sur la prise de décision en situation d'incertitude.

Références

Dion-Blanc, C., Lebarbier, E., & Robin, S. S. (2023). Détection de Ruptures Multiples Pour Les Processus de Poisson. *54ème Journées De Statistique De Société Française De Statistique*. Récupérée 31 janvier 2024, à partir de https://hal.science/hal-04403138

Robin, S. (2019). Change-point detection in a Poisson process.

Soubeyrand, L. (2024, février 27). *Données de Thèse, Expérience Prise de Décisions Sous Incertitude* (csv). csv.