Processus de branchement et loi de Yule

ROCHER Alexandra, GAUTHIER François, MAGNY François

April 28, 2017

1 Processus de branchement

1.1

Soit $n > 1, n \in \mathbb{N}$,

$$\frac{d}{dt}p_{n+1} = \frac{\mathbb{P}(N(t+h) = n+1) - p_{n+1}}{h}$$

Or

$$\mathbb{P}(N(t+h) = n+1) = \sum \mathbb{P}(N(t+h) = n+1)\mathbb{P}(N(t) = k)$$

(formule des probabilités totales)
Ainsi

$$\mathbb{P}(N(t+h) = n+1) = \mathbb{P}(N(t+h) = n+1|N(t) = n+1)p_{n+1} + \mathbb{P}(N(t+h) = n+1|N(t) = n)p_n + O(h)$$

$$= \mathbb{P}(N(t+h) = n+1|N(t) = n+1)p_{n+1} + n\lambda h p_n$$

Mais

$$\sum_k \mathbb{P}(N(t+h) = k|N(t) = n+1) = 1$$

Donc

$$\mathbb{P}(N(t+h) = n+1|N(t) = n+1)p_{n+1} = 1 - \mathbb{P}(N(t+h) = n+2|N(t) = n+1)p_{n+1} = 1 - (n+1)\lambda h$$

Ainsi

$$\forall n \in \mathbb{N}, n > 2, \frac{d}{dt}p_n = \frac{(1 - (n+1)\lambda h)p_{n+1} + n\lambda h p_n - p_{n+1}}{h}$$

$$\frac{d}{dt}p_n = -(n+1)\lambda p_{n+1} + n\lambda p_n$$

Par un raisonnement similaire, pour $n = 1$, on a $\frac{d}{dt}p_1 = -\lambda p_1$

1.2

$\forall n \in \mathbb{N} \backslash \{1\}$

$$p_n(t) = e^{-\lambda t}(1 - e^{-\lambda t})^{n-1},$$

p_n vérifie l'équation différentielle du systeme précédent.
En effet :

$$\frac{d}{dt}p_{n+1} = -\lambda e^{-\lambda t}(1 - e^{-\lambda t})^n + n\lambda e^{-2\lambda t}(1 - e^{-\lambda t})^{n-1}$$

Ensuite,

$$-(n+1)\lambda p_{n+1} + n\lambda p_n = -\lambda e^{-\lambda t}(1 - e^{-\lambda t})^n + n\lambda(1 - e^{-\lambda t})^{n-1}[e^{-\lambda t} - e^{-\lambda t}(1 - e^{-\lambda t})]$$

Qui s'écrit aussi sous la forme :

$$-\lambda e^{-\lambda t}(1-e^{-\lambda t})^n + n\lambda e^{-2\lambda t}(1-e^{-\lambda t})^{n-1}$$

Ainsi $\forall n \in \mathbb{N}\backslash\{1\}$, p_n vérifie bien l'équation différentielle.
De même, $p_1(t)$ verifie aussi son équation différentielle.
De plus, avec l'ajout de la condition initiale le système d'équations différentielle devient un problème de Cauchy et n'admet qu'une unique solution.
Donc, la famille p_n est l'unique solution du système d'équation avec la condition initiale $p_1(0) = 1$

1.3

Soit $t \in \mathbb{R}^+$

$$\begin{aligned}
\mathbb{E}(t) &= \sum_{n=1}^{\infty} ne^{-\lambda t}(1-e^{-\lambda t})^{n-1} \\
&= e^{-\lambda t}\sum_{n=1}^{\infty} n(1-e^{-\lambda t})^{n-1} \\
&= e^{-\lambda t}\frac{1}{(1-(1-e^{-\lambda t}))^2} \\
\mathbb{E}(t) &= e^{\lambda t}
\end{aligned}$$

Donc le nombre d'especes croît en moyenne de maniere exponentielle avec le temps.

2 Loi de Yule

2.1

Par définition, on a :

$$E[p_n(t)] = \int_0^{+\infty} e^{-\lambda t}(1-e^{-\lambda t})^{n-1}\mu e^{-\mu t}dt$$

On fait le changement de variable suivant :

$$u = e^{-\lambda t}$$

, ce qui donne :

$$f_n = \frac{\mu}{\lambda}\int_0 1u^\rho(1-u)^{n-1} = \rho beta(\rho+1, n)$$

2.2

On pose $f(\rho, n-1) = \rho\int_0^1 u^\rho(1-u)^{n-1}$. Par une intégration par parties, on trouve que :

$$f(\rho, n-1) = \frac{n-1}{\rho+1}f(\rho+1, n-2) = ... = \frac{(n-1)(n-2)...1}{(\rho+1)(\rho+2)...(\rho+n)}$$

2.3

Pour simuler f_n, on simule d'abord T selon une loi exponentielle de paramètre μ, puis une loi géométrique de paramètre $exp(-\lambda t)$

Voici les résultats regroupés de la même manière que dans le tableau de la première partie :

Résultats algorithme	129	51	17	11	9	14	8	5	11	5	8	4	21
Résultats partie I	131	47	25	16	11	8	6	5	11	7	9	9	6

On remarque que les 2 estimations sont proches.

Voici un plot des résultats :

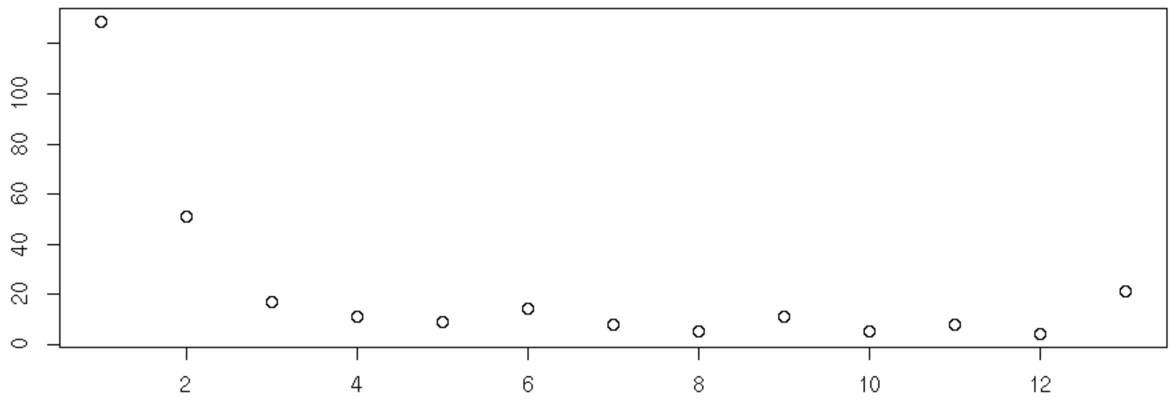


Figure 1: Estimation de la loi de Yule

2.4

Nous avons estimé informatiquement l'espérance de cette loi par un calcul de moyenne. Nous avons ainsi trouvé m=13.38.

Cela correspond à une estimation de la somme d'intégrales :

$$E[f_n] = \sum_{n=0}^{+\infty} n\rho\beta(\rho+1,n)$$

2.5

La probabilité de tirer une boule est proportionelle à son poids, donc la probabilité de ne pas tirer la boule blanche le premier coup est de $\frac{1}{1+\rho}$ et ainsi de suite. La probabilité de ne pas tirer la boule blanche au n ième coup sachant qu'on ne l'a toujours pas tirée est donc de $\frac{n}{n+\rho}$.

Par conséquent, nous avons, en utilisant la loi des probabilités conditionnelles, la formule :

$$P(N > n) = \prod_{i=1}^{n} \frac{i}{i+\rho}$$

Donc:

$$
\begin{aligned}
P(N=n) &= P(N>n-1) - P(N>n) && (1)\\
&= \prod_{i=1}^{n-1} \frac{i}{i+\rho} - \frac{n}{n+\rho} \prod_{i=1}^{n-1} \frac{i}{i+\rho} && (2)\\
&= (1 - \frac{n}{n+\rho}) \prod_{i=1}^{n-1} \frac{i}{i+\rho} && (3)\\
&= \frac{\rho}{n+\rho} \prod_{i=1}^{n-1} \frac{i}{i+\rho} && (4)\\
&= \frac{\rho \prod_{i=1}^{n-1} i}{\prod_{i=1}^{n} i+\rho} && (5)\\
& && (6)
\end{aligned}
$$

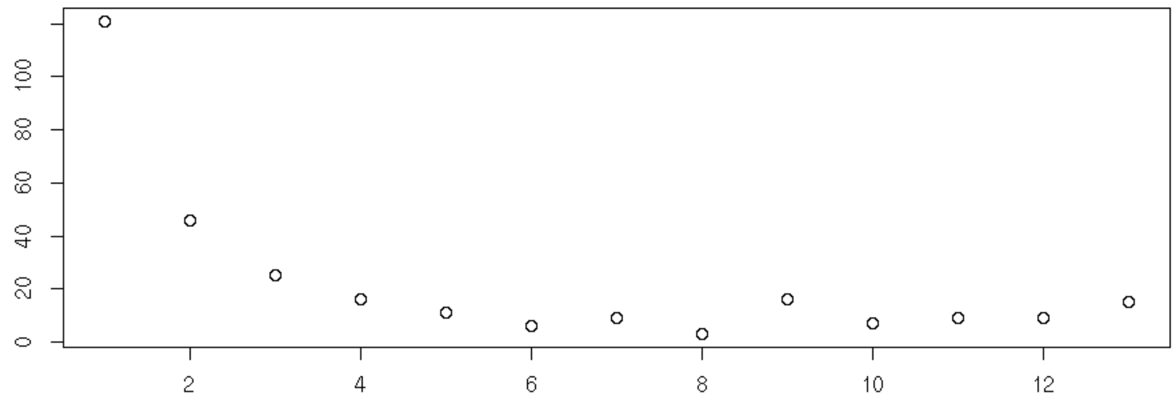
On retrouve bien la formule de la loi de YULE de la question 2.

On a donc pu simuler assez facilement la loi de Yule par une simple succession de simulations de lois de Bernouilli. On trouve ainsi, pour n=293, les résulats suivants, comparés avec les résultats théoriques trouvés dans la partie I :

Résultats algorithme	127	52	27	13	10	6	4	5	14	2	7	9	17
Résulas partie I	131	47	25	16	11	8	6	5	11	7	9	9	6

Comme pour la simulation précédente, les résultats sont très proches du résulat attendu.

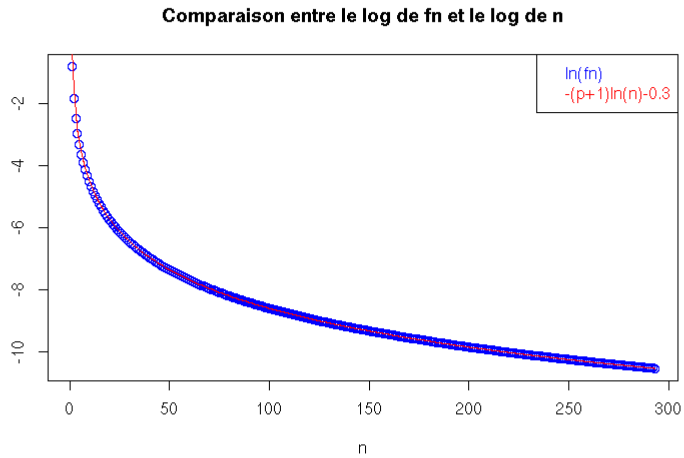
Voici un plot des résultats :



3 Ajustement du modèle aux données

3.1

En faisant un plot



On remarque qu'il existe bien une relation entre le logarithme de f_n et celui de n

3.2 Question 2

L'estimation précedente utilise la méthode des moindres carrés et donne une erreur gaussienne, cela ne correspond pas aux données qui suivent la loi de YULE.
Nous cherchons un meilleur estimateur, nous allons utiliser la méthode du maximum de vraisemblance, nous disposons d'un histogramme de données donc la loi à utiliser est la loi Multinomiale, généralisant la loi Binomiale que l'on aurait utilisé dans le cas d'un histogramme à deux classes.
Nous allons donc chercher le ρ qui maximise la loi multinomiale

$$\frac{n!}{n_1!...n_{35+}!}p_1^{n_1}...p_{35+}^{n_{35+}}$$

Pour les n et n_i venant de la table de données et les p_i calculés avec la loi de YULE de paramètre ρ.
En calculant sur R on trouve $\rho = 1.24552$

3.3 Question 3

Avec $\rho = 1.24552$ on calcule les probabilités des classes et donc l'effectif prévisionnel.

La simulation de 1000 tirages de la loi de densité $p(\theta/y)$:

```
> eff
 [1] 0.554668852 0.170902922 0.080509771 0.046044875 0.029489858 0.020350409 0.014808339 0.011211740 0.021482369 0.012167225 0.013062274 0.011871611 0.009831919
> eff = eff * 293
> eff
 [1] 162.517974  50.074556  23.589363  13.491148   8.640529   5.962670   4.338843   3.285040   6.294334   3.564997   3.827246   3.478382   2.880752
>
```

On trouve des résultats cohérents avec le tableau de données.