

Seeds - Random effect logistic regression

BRUNO LOPES Matheus, MOURDI Elias, SELAMNIA Najib, TRIOMPHE Amaury

15/04/2024

Lien vers notre Github : <https://github.com/azer1230/-BAYES-Projet-1>

Données étudiées

Le contexte de notre projet est l'étude de la germination de graines respectant certaines propriétés. Dans notre exemple, $N = 21$ plaques sont disposées pour accueillir 2 types de graines issues de 2 types de racines. Les tableaux ci-dessous recensent les résultats pour ces 4 types de population. $\forall i \in \{1, \dots, 21\}$, r_i correspond au nombre de graines germées et n_i correspond au nombre total de graines sur la i -ème plaque. Le rapport entre ces deux grandeurs est donc la proportion de graines ayant germé sur la dite plaque.

Bean			Cucumber		
r	n	r/n	r	n	r/n
10	39	0.26	5	6	0.83
23	62	0.37	53	74	0.72
23	81	0.28	55	72	0.76
26	51	0.51	32	51	0.63
17	39	0.44	46	79	0.58
			10	13	0.77

Table 1: Données récupérées pour la graine seed O. aegyptiaco 75

Bean			Cucumber		
r	n	r/n	r	n	r/n
8	16	0.5	3	12	0.25
10	30	0.33	22	41	0.54
8	28	0.29	15	30	0.5
23	45	0.51	32	51	0.63
0	4	0	3	7	0.43

Table 2: Données récupérées pour la graine seed O. aegyptiaco 73

Cadre mathématique

Hypothèses sur nos données (Matheus)

Si p_i est la probabilité de germination sur la plaque i , alors supposons que le nombre de graines germées r_i suit une loi binomiale :

$$r_i \sim \text{Binomial}(p_i, n_i)$$

De plus, supposons que le modèle est essentiellement une régression logistique à effets aléatoires, ce qui permet de traiter la surdispersion. Autrement dit :

$$\text{logit}(p_i) = \alpha_0 + \alpha_1 x_{1i} + \alpha_2 x_{2i} + \alpha_{12} x_{1i} x_{2i} + b_i \text{ où } b_i \sim \mathcal{N}(0, \tau)$$

Où x_{1i} , x_{2i} sont le type de graine et l'extrait de racine de la i -ème plaque, et un terme d'interaction $\alpha_{12} x_{1i} x_{2i}$ est inclus. $\alpha_0, \alpha_1, \alpha_2, \alpha_{12}, \tau$ ont des priors indépendants "non informatifs" fournis, qui seront supposés comme suit :

$$\alpha_i \sim \mathcal{N}(0, 10^6), \text{ pour } i \in \{0, 1, 2\} \alpha_{12} \sim \mathcal{N}(0, 10^6) \tau \sim \text{gamma}(10^{-3}, 10^{-3})$$

Une dernière hypothèse que nous ferons également est que r_i sont indépendants.

Graphe acyclique orienté (Najib)

Lois conditionnelles (Matheus)

Comme nous allons appliquer Hastings-within-Gibbs, nous devons avoir les lois conditionnelles de tous les paramètres de l'expression de $\text{logit}(p_i)$, c'est-à-dire que nous devons obtenir toutes les lois postérieures. Pour α_0 , nous aurons :

$$\pi(\alpha_0 | \alpha_1, \alpha_{12}, \alpha_2, r, b, \tau) \propto \pi(\alpha_1, \alpha_{12}, \alpha_2, r, b, \tau | \alpha_0) \pi(\alpha_0)$$

Dans le contexte de H-W-Gibbs, comme nous allons mettre à jour les paramètres séparément en considérant les autres comme des valeurs fixes, nous aurons :

$$\pi(\alpha_0 | \alpha_1, \alpha_{12}, \alpha_2, r, b, \tau) \propto \pi(r | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, b, \tau) \pi(\alpha_0) = \pi(\alpha_0) \prod_{i=1}^N \pi(r_i | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, b, \tau) = \pi(\alpha_0) \prod_{i=1}^N p_i^{r_i} (1-p)^{n_i-r_i}$$

Comme tous suivent la même loi a priori, nous aurons des expressions similaires pour α_1, α_{12} et α_2 . Pour τ , nous devons, comme τ dépend de b qui suit une loi normale, qui dans ce cas est conjuguée par la loi gamma (loi a priori de τ), obtenir directement la loi a posteriori de τ :

$$\tau | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, b, r \sim \text{gamma}(10^{-3} + \frac{N}{2}, 10^{-3} + \frac{\sum_{i=1}^N b_i^2}{2})$$

Une fois τ mis à jour dans l'algorithme, nous pourrions mettre à jour chaque b_i , pour $i \in \{1, \dots, N\}$, où chacun aura la loi a posteriori suivante :

$$\pi(b_i | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, r_i, \tau) \propto \pi(\alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, r_i, \tau | b_i) \pi(b_i)$$

En considérant que $\alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, \tau$ sont des paramètres déjà fixes et que $b_i \sim N(0, \frac{1}{\tau})$, nous pouvons écrire :

$$\pi(b_i | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, r_i, \tau) \propto \pi(r_i | \alpha_0, \alpha_1, \alpha_{12}, \alpha_2, i, \tau, b_i) \exp(-\frac{b_i^2 \tau}{2}) = p_i^{r_i} (1-p_i)^{n_i-r_i} \exp(-\frac{b_i^2 \tau}{2})$$

Maintenant, ayant toutes les lois conditionnelles, nous pouvons appliquer notre algorithme Hastings-within-Gibbs.

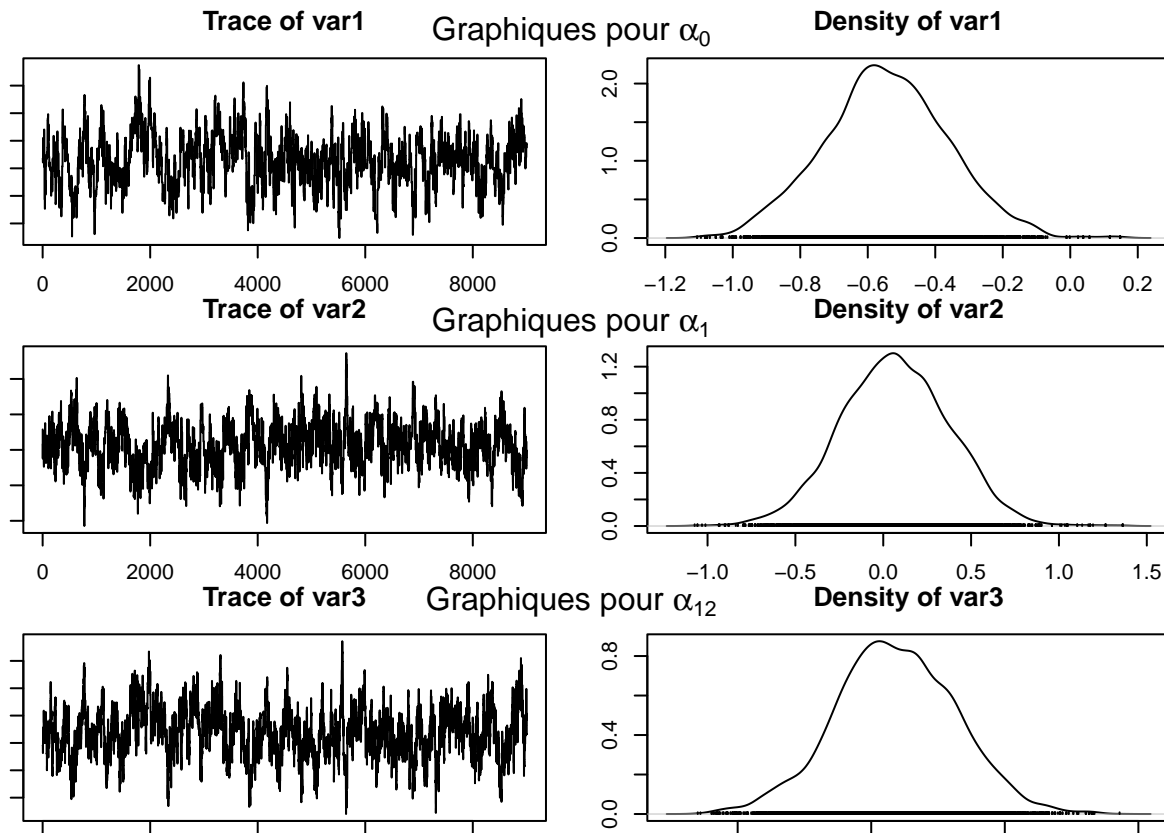
Résultats de l'implémentation algorithmique

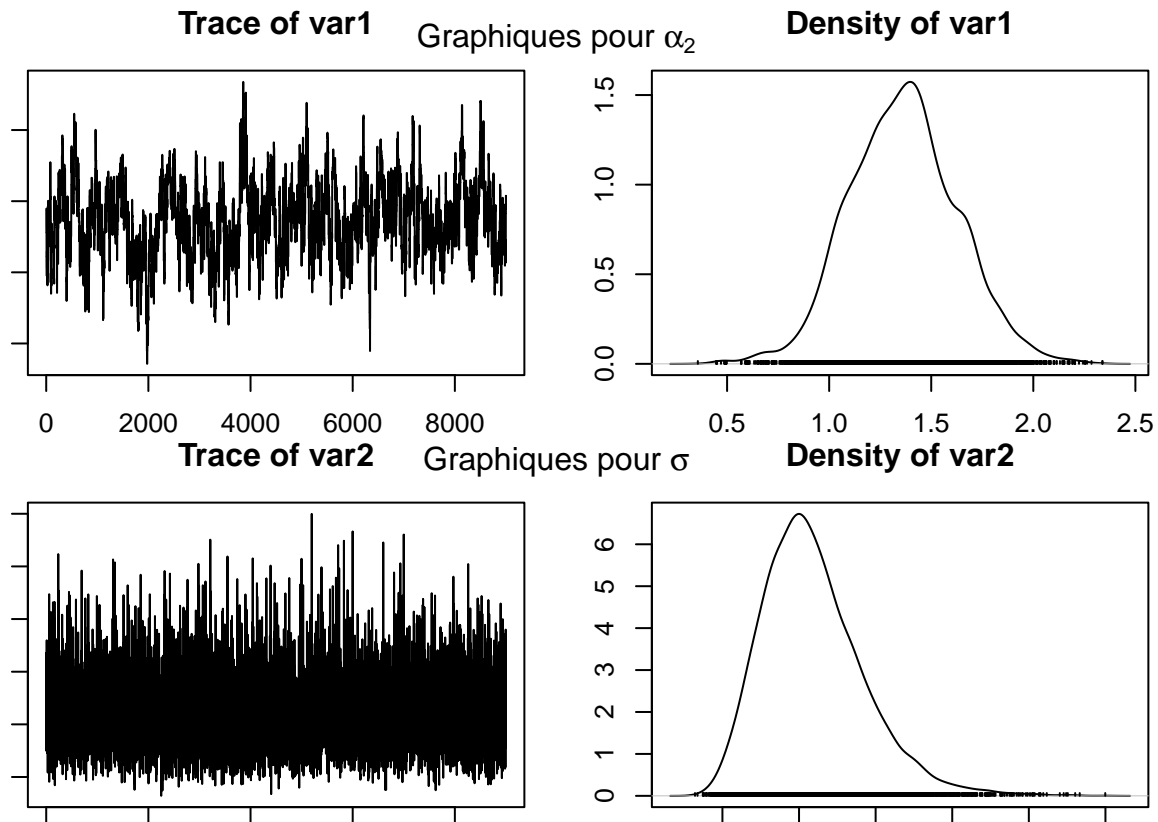
Grâce au calcul précédent des lois conditionnelles, nous avons pu implémenter un algorithme de Hasting Within Gibbs pour estimer nos paramètres. De la même manière que ce qui est donné dans l'énoncé, nous avons généré 10^4 réalisations auxquelles nous avons retiré les 1000 premières, correspondant à la burnin period. Les résultats obtenus, ainsi qu'une comparaison avec ce qui est donné dans l'énoncé, sont mentionnés dans le tableau ci-dessous.

Paramètres	Moyenne		Écart-type	
	Résultat	Énoncé	Résultat	Énoncé
α_0	-0.5562	-0.5525	0.1865	0.1852
α_1	0.0706	0.08382	0.3252	0.3031
α_{12}	-0.8021	-0.8165	0.4564	0.4109
α_2	1.3511	1.346	0.2745	0.2564
σ	0.3198	0.267	0.0661	0.1471

Table 3: Résultats de notre algorithme Hastings within Gibbs

On rend également compte des résultats des chaînes de Markov générées (à gauche), ainsi que de leurs distributions (à droite) dans les graphiques ci-dessous:





Les résultats obtenues sont cohérents avec l'objectif initial visé dans l'énoncé. Les valeurs sont distribuées autour des moyennes attendues et les écart-types sont également en accord avec les attentes, en observant des variations inférieurs à 20%.

(Voir pour les sigma)

Analyse des résultats (Amaury)

Nom	x1	x2	logit(p)
Bean 75	0	0	-0.5517254
Cucumber 75	0	1	0.8027722
Bean 73	1	0	-0.4682697
Cucumber 73	1	1	0.04229868

Table 4: Logit(p) pour différentes valeurs de x1 et x2