

Algorithme VBEM pour l'estimation  
du processus de Cox log gaussien  
Application à la cartographie des adventices

J. Radoszycki, N. Peyrard, R. Sabbadin

**Unité de Biométrie et Intelligence artificielle - INRA Toulouse**

Vendredi 23 Novembre 2012

# Plan

- 1 Contexte : comparaison de deux modèles pour la cartographie des adventices - algorithme VBEM
- 2 Contribution : deux algorithmes VBEM pour l'estimation du processus de Cox log gaussien
- 3 Évaluation de ces deux algorithmes
- 4 Conclusion et perspectives

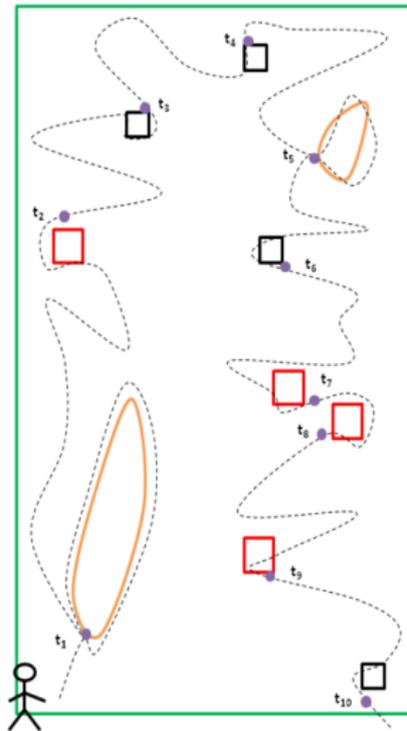
# Question écologique



Vs



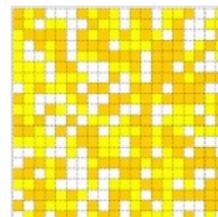
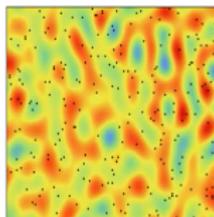
# Données adventices



- Comptages (quadrats 0.36m<sup>2</sup>)
- Notes d'abondances (stations 16m<sup>2</sup>)
- Taches
- Parcours du notateur

# Comparaison de deux modèles pour la cartographie des adventices

<b>Modèle</b>	Processus de Cox log gaussien	Champ de Markov
<b>Point de vue</b>	bayésien	fréquentiste
<b>Support</b>	continu	discret
<b>Variable modélisée</b>	comptage	note d'abondance
<b>Pattern spatial</b>	lisse	lisse ou abrupt
<b>Estimation et sélection de modèles approchées</b>	MCMC	Méthodes variationnelles



# Principe des méthodes variationnelles

Soit  $m$  un modèle de distribution complexe  $P_m$ .

- 1 Choix d'une famille  $\mathcal{Q}$  de distributions pour approcher  $P_m$
- 2 Choix du meilleur représentant  $q^{var} = \min_{q \in \mathcal{Q}} KL(q|P_m)$
- 3 Estimation des paramètres dans un cadre fréquentiste (VEM) ou bayésien (VBEM)  $\rightarrow \hat{\theta}_m^{var}$
- 4 Sélection de modèle :  $BIC(P_m, \hat{\theta}_m^{MV}) = \ln P_m(y|\hat{\theta}_m^{MV}, m) - \frac{1}{2}d_m \ln n$   
 BIC variationnel :  $BIC(q^{var}, \hat{\theta}_m^{var})$

# Objet du stage

**But** : sélectionner le modèle le mieux adapté aux  
adventices (critère BIC)



besoin de méthodes  
comparables



**Objet du stage** : spécialisation et implémentation d'un algorithme  
VBEM (*Variational Bayesian EM*)  
pour l'estimation du processus de Cox log gaussien

# Processus de Cox log gaussien

**processus de Cox...** = processus de Poisson non homogène dont l'intensité est elle-même un processus stochastique

- 1 **...log gaussien** :  $\lambda_x = \exp(\beta + S_x)$  où  $S$  champ gaussien centré de covariance :  $\Sigma(x, x') = \text{Cov}(S_x, S_{x'}) = \sigma^2 \exp(-\alpha \|x - x'\|)$
- 2 comptages  $Y_j$  sur quadrats  $A_j$  :  $Y_j | \lambda \sim \mathcal{P}(\Lambda_j)$  où  $\Lambda_j = \int_{A_j} \lambda_x dx$
- 3 conditionnellement à  $Y_j = n_j$ , les  $n_j$  points sont disposés dans  $A_j$  de manière indépendante selon la densité  $f(x) = \frac{\lambda_x}{\Lambda_j}$

# Processus de Cox log gaussien

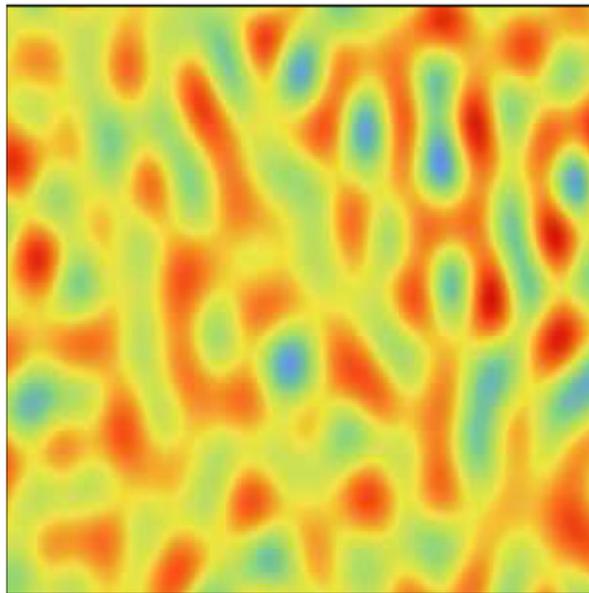
**processus de Cox...** = processus de Poisson non homogène dont l'intensité est elle-même un processus stochastique

- ① **...log gaussien** :  $\lambda_x = \exp(\beta + S_x)$  où  $S$  champ gaussien centré de covariance :  $\Sigma(x, x') = \text{Cov}(S_x, S_{x'}) = \sigma^2 \exp(-\alpha \|x - x'\|)$
- ② comptages  $Y_j$  sur quadrats  $A_j$  :  $Y_j | \lambda \sim \mathcal{P}(\Lambda_j)$  où  $\Lambda_j = \int_{A_j} \lambda_x dx$
- ③ conditionnellement à  $Y_j = n_j$ , les  $n_j$  points sont disposés dans  $A_j$  de manière indépendante selon la densité  $f(x) = \frac{\lambda_x}{\Lambda_j}$

## MODELE BAYESIEN

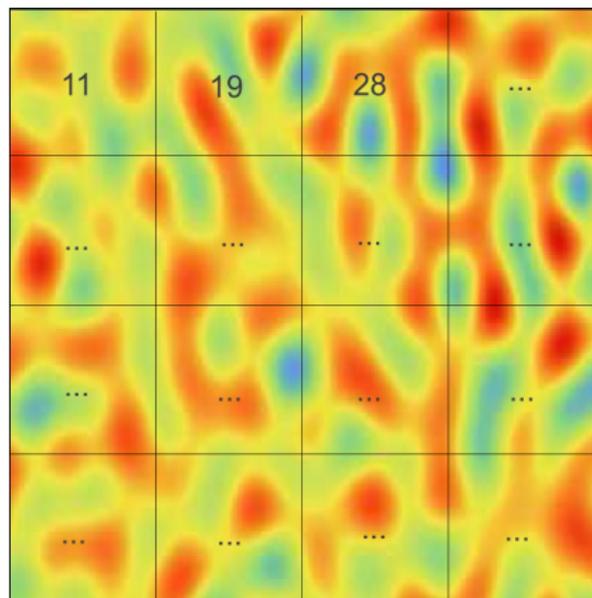
# Processus de Cox

## 1. Intensité $\lambda$



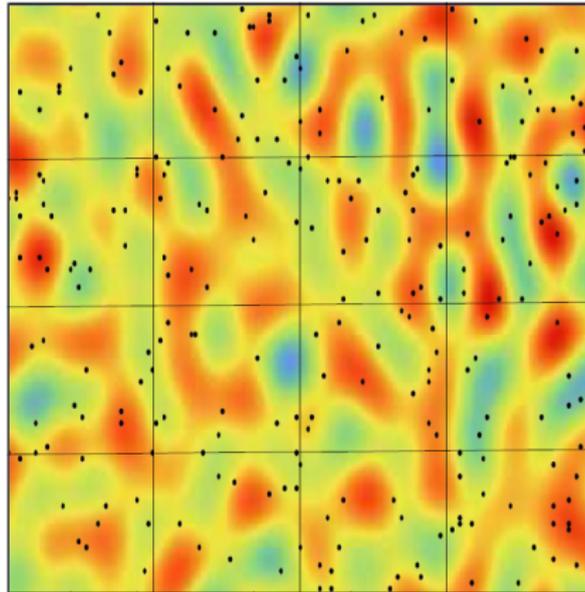
# Processus de Cox

2.  $Y_j | \lambda \sim \mathcal{P}(\Lambda_j)$  où  $\Lambda_j = \int_{A_j} \lambda_x dx$



## Processus de Cox

3. conditionnellement à  $Y_j = n_j$ , les  $n_j$  points sont disposés dans  $A_j$  de manière indépendante selon la densité  $f(x) = \frac{\lambda_x}{\Lambda_j}$



## L'algorithme VBEM (*Variational Bayesian EM*)

M.J. Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, University College London, 2003.

**VBEM** : pour estimer  $S$  et  $\theta = (\beta, \sigma, \alpha)$  de manière approchée dans un cadre bayésien

$$\begin{aligned} \ln p(y) &\geq \int \int q_{S,\theta}(S, \theta) \ln \frac{p(S, y, \theta)}{q_{S,\theta}(S, \theta)} dS d\theta \quad (\text{Jensen}) \\ &= \ln p(y) - KL(q_{S,\theta}(\cdot) | p_{S,\theta}(\cdot | y)) \end{aligned}$$

→ maximal pour  $q_{S,\theta}(\cdot) = p_{S,\theta}(\cdot | y)$ , difficile à évaluer

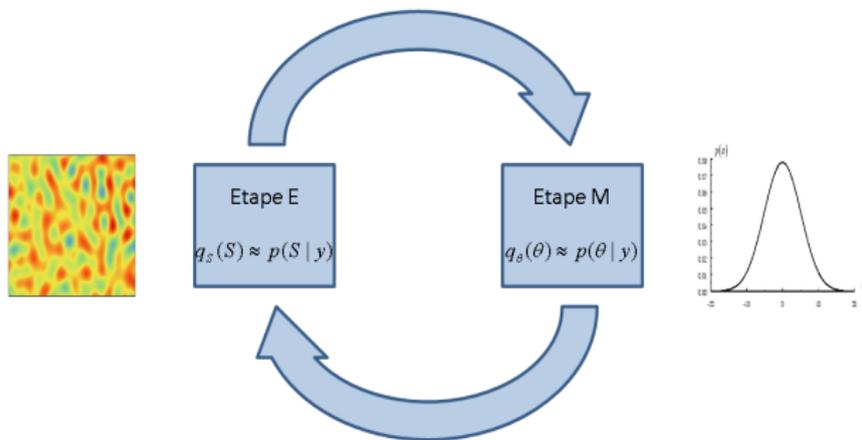
→ approximation de  $q_{S,\theta}(S, \theta)$  par  $q_S(S)q_\theta(\theta)$  :

$$\begin{aligned} \mathcal{F}(q_S(S), q_\theta(\theta)) &= \int \int q_S(S)q_\theta(\theta) \ln \frac{p(S, y, \theta)}{q_S(S)q_\theta(\theta)} dS d\theta \\ &= \ln p(y) - KL(q_S(\cdot)q_\theta(\cdot) | p_{S,\theta}(\cdot | y)) \end{aligned}$$

# L'algorithme VBEM (*Variational Bayesian EM*)

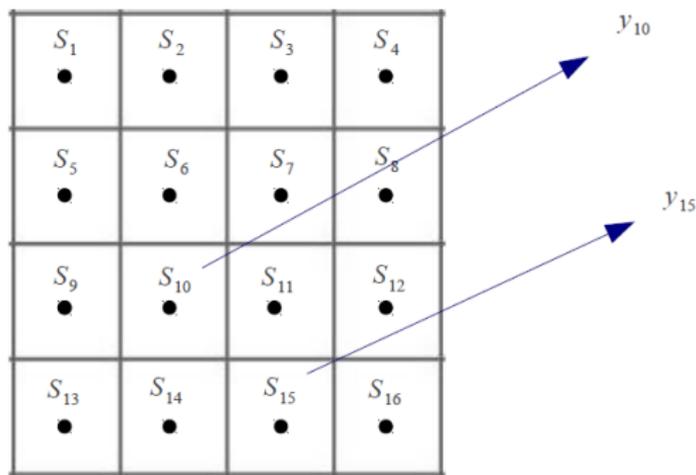
**Etape E** :  $q_S^{(t+1)} = \operatorname{argmax}_{q_S \in \mathcal{Q}} \mathcal{F}(q_S(S), q_\theta^{(t)}(\theta))$

**Etape M** :  $q_\theta^{(t+1)} = \operatorname{argmax}_{q_\theta \in \mathcal{Q}' } \mathcal{F}(q_S^{(t+1)}(S), q_\theta(\theta))$



$$\hat{\theta}^{\text{MAP}} = \operatorname{argmax}_\theta q_\theta^{\text{VBEM}}(\theta)$$

# Notations



$$\mathcal{V} = \{1, 2, \dots, N\}, N = 16$$

$$\mathcal{O} = \{10, 15\}, n = 2$$

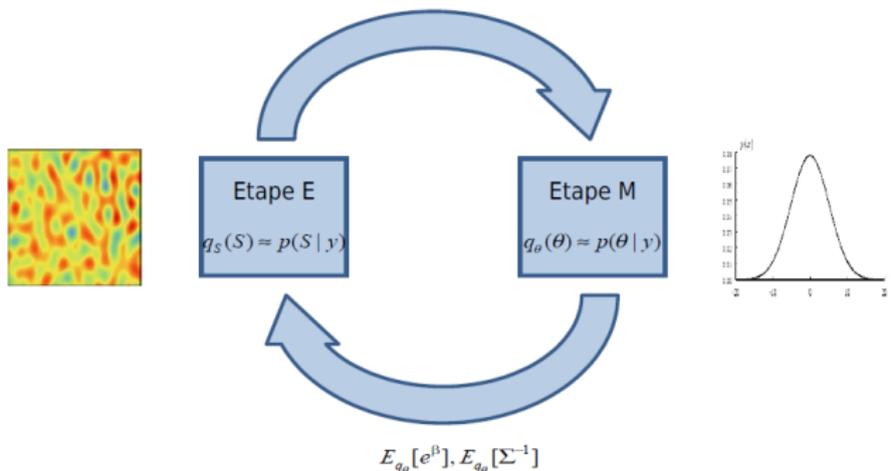
$n = N \Leftrightarrow$  échantillonnage complet

# Algorithme VBEM pour le processus de Cox log gaussien

estimation grossière de  $\Lambda_j = \int_{A_j} \lambda_x dx$  par  $|A_j| \lambda_j$

$$E_{q_{S_j}}[S_j], E_{q_{S_j}}[S_j^2], j \in \mathcal{V}$$

$$E_{q_{S_j}}[e^{S_j}], j \in \mathcal{O}$$



→ compliqué car le modèle n'est pas exponentiel conjugué

## Etape E

Hypothèse :  $q_S(S) = \prod_{j \in \mathcal{V}} q_{S_j}(S_j)$

Résultat : formule explicite mais loi non classique pour les  $q_{S_j}(S_j), j \in \mathcal{O}$  :

$$q_{S_j}(S_j) \propto \exp\left(-\frac{(S_j - m_j)^2}{2\sigma_j^2}\right) = \mathcal{N}(m_j, \sigma_j^2) \text{ si } j \notin \mathcal{O}$$

$$q_{S_j}(S_j) \propto \exp\left(-\mathbb{E}_{q_\theta}[\Lambda_j]\right) \times \mathbb{E}_{q_\theta}[\Lambda_j]^{y_j} \times \exp\left(-\frac{(S_j - m_j)^2}{2\sigma_j^2}\right) \text{ si } j \in \mathcal{O}$$

où  $\forall j \in \mathcal{V}, \sigma_j^2 = \frac{1}{\mathbb{E}_{q_\theta}[(\Sigma^{-1})_{jj}]}$

et  $\forall j \in \mathcal{V}, m_j = -\sigma_j^2 \sum_{l \in \mathcal{V}, l \neq j} \mathbb{E}_{q_\theta}[(\Sigma^{-1})_{jl}] \mathbb{E}_{q_{S_l}}[S_l]$

Problème : calcul de  $\mathbb{E}_{q_{S_j}}[S_j], \mathbb{E}_{q_{S_j}}[S_j^2]$  ( $j \in \mathcal{V}$ ) et  $\mathbb{E}_{q_{S_j}}[\exp(S_j)]$  ( $j \in \mathcal{O}$ ) nécessaires pour l'étape M

## Etape E : version 1

on approche  $q_{S_j}(S_j), j \in \mathcal{O}$  par une gaussienne grâce à un DL d'ordre 2 de  $e^{S_j}$  et une identification de la moyenne et de la variance :

$$\forall j \in \mathcal{O}, q_{S_j}(S_j) \approx \mathcal{N}(m'_j, \sigma_j'^2)$$

### Etape E version 1

- 1 résolution d'un système à  $N + n$  équations et inconnues (les  $m_j$  et  $m'_j$ )
- 2 calcul des espérances nécessaires pour l'étape M :  
 $\mathbb{E}[S_j]$  et  $\mathbb{E}[S_j^2]$  : direct car loi gaussienne  
 $\mathbb{E}[e^{S_j}]$  : direct car loi log gaussienne

## Etape E : version 2

$$\begin{aligned}\forall j \in \mathcal{O}, \mathbb{E}_{q_{S_j}}[S_j] &= \int K \exp(-\mathbb{E}_{q_\theta}[\Lambda_j]) \mathbb{E}_{q_\theta}[\Lambda_j]^{y_j} \exp\left(-\frac{(S_j - m_j)^2}{2\sigma_j^2}\right) S_j dS_j \\ &= \mathbb{E}_{\mathcal{N}(m_j, \sigma_j^2)}[K' \exp(-\mathbb{E}_{q_\theta}[\Lambda_j]) \mathbb{E}_{q_\theta}[\Lambda_j]^{y_j} S_j]\end{aligned}$$

→ méthode de Monte-Carlo

### Etape E version 2

itérer

- 1 calcul des  $E_{q_{S_j}}[S_j], j \in \mathcal{O}$  par Monte-Carlo (simulation selon  $\mathcal{N}(m_j, \sigma_j^2)$ )
- 2 résolution d'un système linéaire à  $N$  équations et inconnues (les  $m_j$ ) grâce aux  $E_{q_{S_j}}[S_j], j \in \mathcal{O}$

calcul des autres espérances ( $\mathbb{E}_{q_{S_j}}[S_j^2]$  et  $\mathbb{E}_{q_{S_j}}[\exp(S_j)]$ ) par Monte-Carlo (simulation selon  $\mathcal{N}(m_j, \sigma_j^2)$ )

## Etape M

cas de lois *a priori* uniformes bornées et indépendantes  
 (généralisable à toutes lois *a priori* indépendantes)

Résultat :  $q_\theta(\theta) \propto q_\beta(\beta)q_{\sigma,\alpha}(\sigma, \alpha)$  où

$$q_\beta(\beta) \propto \exp \left( \sum_{k \in \mathcal{O}} \left( -e^\beta |A_k| \mathbb{E}_{q_{S_k}} [e^{S_k}] + \beta y_k \right) \right) \mathbb{1}_{\beta \in [\beta_{min}; \beta_{max}]}$$

$$q_{\sigma,\alpha}(\sigma, \alpha) \propto \frac{1}{|\Sigma|^{1/2}} \exp \left( -\frac{1}{2} \sum_{k \in \mathcal{V}} \left( (\Sigma^{-1})_{kk} \mathbb{E}_{q_{S_k}} [S_k^2] + \mathbb{E}_{q_{S_k}} [S_k] \right. \right. \\ \left. \left. \sum_{l \neq k} (\Sigma^{-1})_{kl} \mathbb{E}_{q_{S_l}} [S_l] \right) \right) \mathbb{1}_{\sigma \in [\sigma_{min}; \sigma_{max}]} \mathbb{1}_{\alpha \in [\alpha_{min}; \alpha_{max}]}$$

→ lois non classiques

→ calcul de  $\mathbb{E}_{q_\beta} [e^\beta]$  et  $\mathbb{E}_{q_{\sigma,\alpha}} [\Sigma^{-1}]$  pour l'étape E par la méthode de Monte-Carlo  
 (simulation selon la loi *a priori*)

# Initialisation

① pour  $\beta$  et  $\sigma$  : 
$$\begin{cases} \mathbb{E}[\lambda_i] = e^{\beta + \sigma^2/2} \\ \mathbb{V}[\lambda_i] = e^{2\beta + \sigma^2} (e^{\sigma^2} - 1) \end{cases}$$

on estime à partir des données :

$$\mathbb{E}[\lambda_i] \approx \frac{\hat{E}[Y]}{|A|} \text{ et } \mathbb{V}[\lambda_i] \approx \frac{\hat{V}[Y] - \hat{E}[Y]^2}{|A|^2}$$

→ résolution d'un système 2x2

- ② pour  $\alpha$  : méthode basée sur le variogramme pas assez précise  
 → mode de la loi *a priori* ou tirage dans la loi *a priori*

# Protocole d'évaluation

- dimension de la grille :  $10 \times 40$  ( $N = 400$ )
- taille de la maille :  $0.6m$  (quadrats de  $0.36m^2$ )

## Lois a priori

- cas de lois *a priori* uniformes bornées :  
 $\beta_{min} = -2, \beta_{max} = 2, \sigma_{min} = 0.2, \sigma_{max} = 2, \alpha_{min} = 0.3, \alpha_{max} = 1$
- cas de lois *a priori* normales et lognormales :  
 $\beta \sim \mathcal{N}(0, 0.1), \ln \sigma \sim \mathcal{N}(0, 0.05), \ln \alpha \sim \mathcal{N}(0, 0.2)$

## Simulation du processus de Cox ( $\times N_5$ )

- 1 tirage des paramètres  $\alpha, \beta$  et  $\sigma$  dans la loi *a priori*
- 2 simulation du champ  $S$  et de l'intensité  $\lambda$  en tout point  $x_j$  de la grille
- 3 simulation des comptages  $Y_j$  en tout quadrat  $A_j$
- 4 retrait de certains comptages dans le cas d'un échantillon incomplet

# Protocole d'évaluation

## Algorithme VBEM

- nombre d'itérations algo VBEM : 100
- nombre d'itérations étape E version 2 : 5
- nombre de simulations Monte-Carlo : 50
- estimation des paramètres par le mode de  $q_\theta$  moyenné sur les 20 dernières itérations
- estimation du champ  $S$  par  $\mathbb{E}_{q_S}[S]$  moyennée sur les 20 dernières itérations

# Protocole d'évaluation

## Etapas d'évaluation

- 1 comparaison des deux versions de VBEM développées
- 2 comparaison VBEM/MCMC
- 3 VBEM avec un échantillonnage incomplet

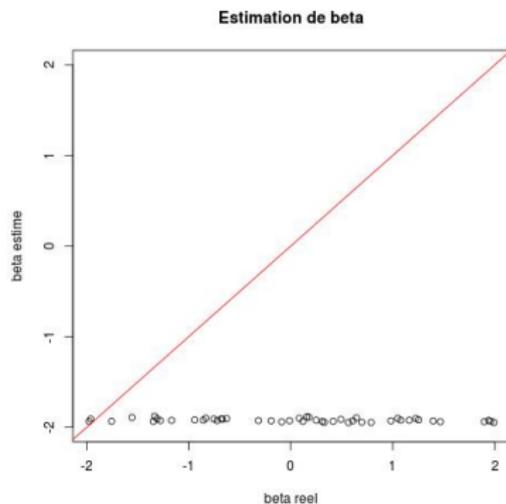
## Critère d'évaluation

$$\text{EQM}(\hat{\beta}) = \frac{1}{N_S} \sum_{i=1}^{N_S} (\hat{\beta}_i - \beta_i)^2$$

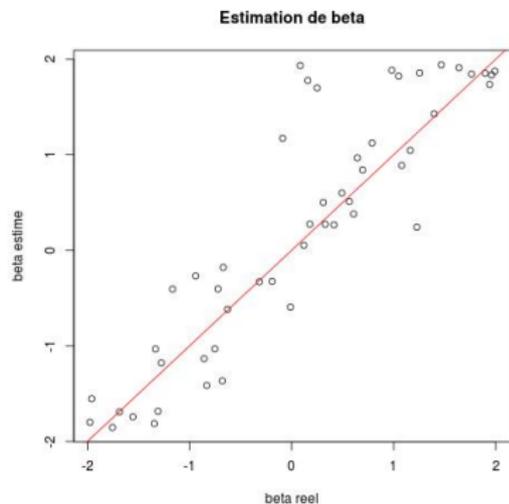
$$\text{EQM}(\hat{S}) = \frac{1}{N \times N_S} \sum_{i=1}^{N_S} \sum_{j=1}^N (\hat{S}_j^i - S_j^i)^2$$

# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E ( $\beta$ ) échantillonnage complet - lois *a priori* uniformes



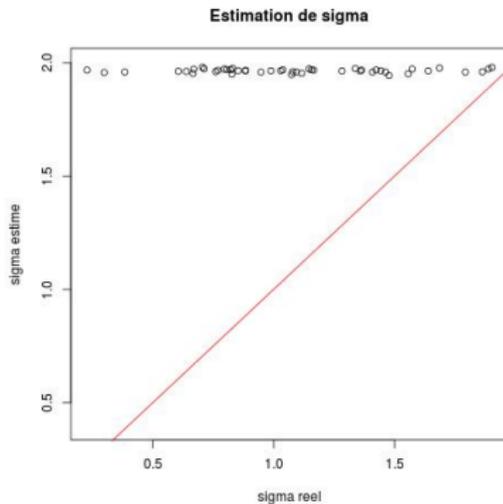
(a) Etape E version 1



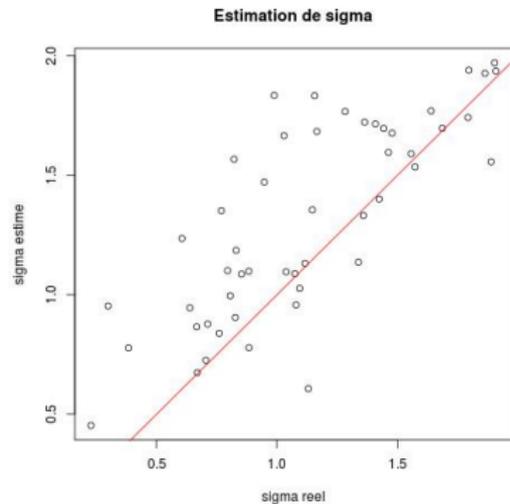
(b) Etape E version 2

# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E ( $\sigma$ ) échantillonnage complet - lois *a priori* uniformes



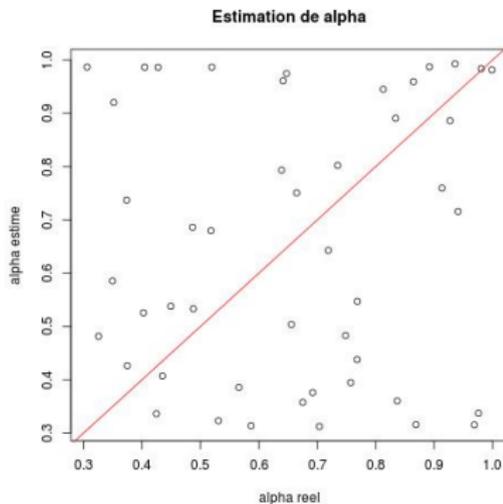
(c) Etape E version 1



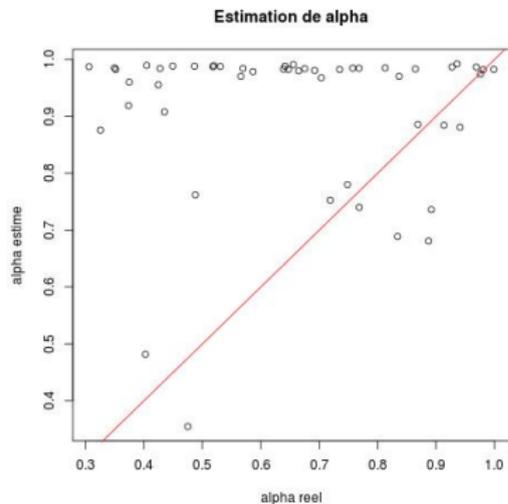
(d) Etape E version 2

# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E ( $\alpha$ ) échantillonnage complet - lois *a priori* uniformes



(e) Etape E version 1

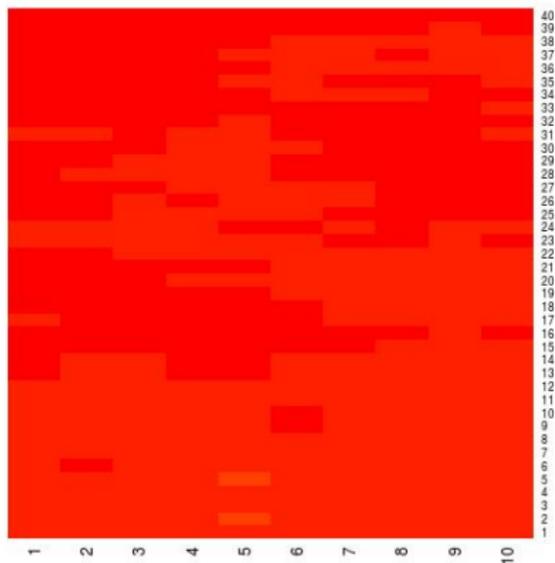


(f) Etape E version 2

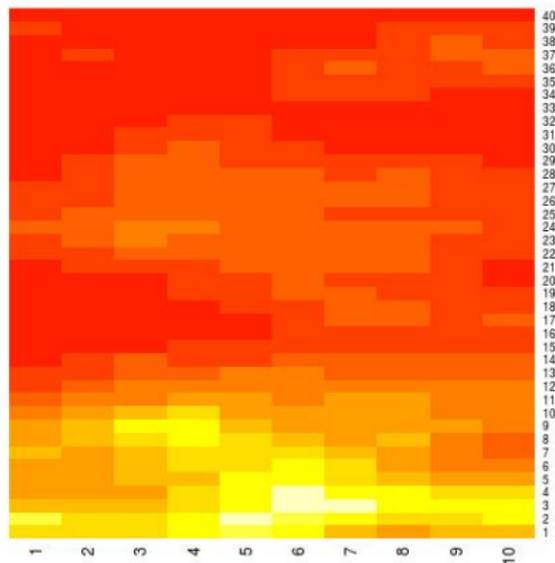
# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E échantillonnage complet - lois *a priori* uniformes

champ S simule



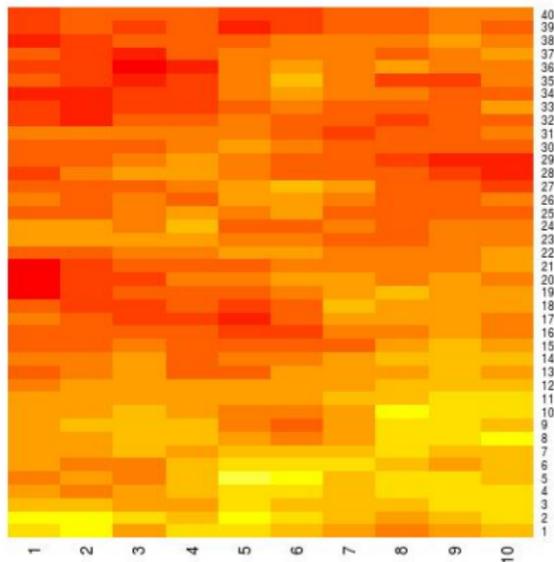
champ S estime (VBEM version 1)



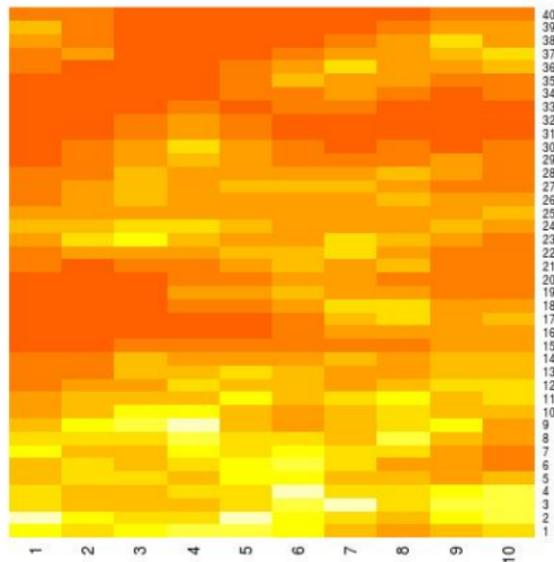
# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E échantillonnage complet - lois *a priori* uniformes

champ S simulé



champ S estimé (VBEM version 2)



# Evaluation (1/3)

## 1. Comparaison des deux versions de l'étape E échantillonnage complet - lois *a priori* uniformes

	version 1	version 2
$\beta$	5.11	0.35
$\sigma$	0.93	0.12
$\alpha$	0.10	0.13
S	1384.5	0.82

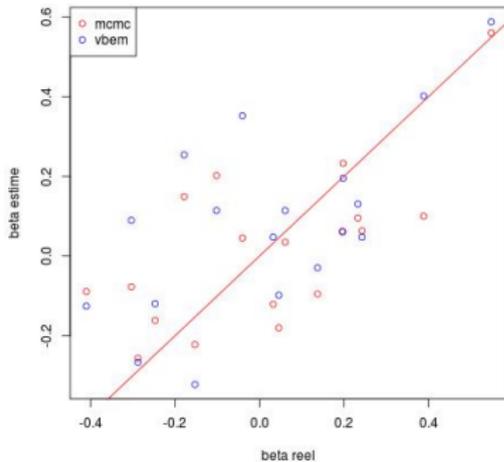
TABLE: EQM algorithme VBEM - 50 simulations

- version 1 : pas satisfaisant
- version 2 : satisfaisant sauf pour  $\alpha$

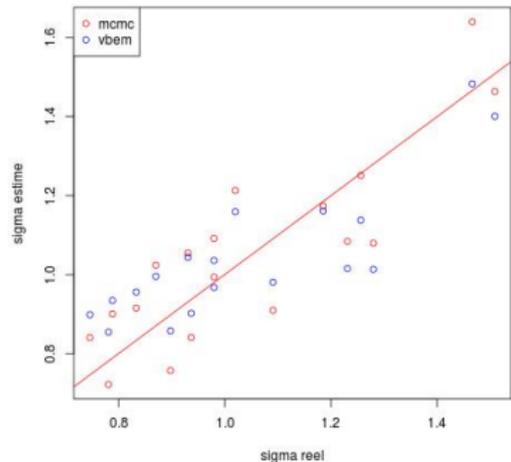
# Evaluation (2/3)

## 2. Comparaison de VBEM (version 2) avec MCMC échantillonnage complet - lois *a priori* normale et lognormales

Estimation de beta

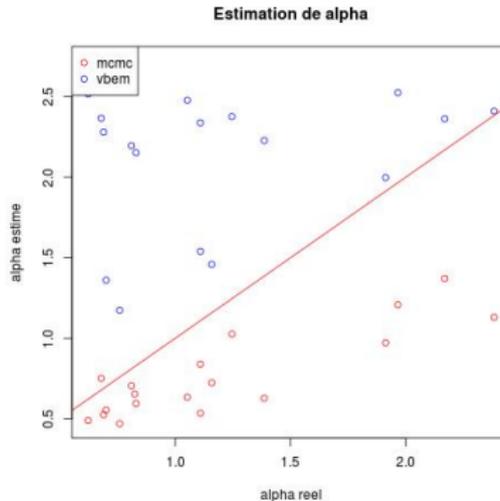


Estimation de sigma



## Evaluation (2/3)

### 2. Comparaison de VBEM (version 2) avec MCMC échantillonnage complet - lois *a priori* normale et lognormales



## Evaluation (2/3)

### 2. Comparaison de VBEM (version 2) avec MCMC échantillonnage complet - lois *a priori* normale et lognormales

	VBEM v2	MCMC
$\beta$	0.044	0.036
$\sigma$	0.015	0.015
$\alpha$	1.28	0.29
S	0.63	0.59

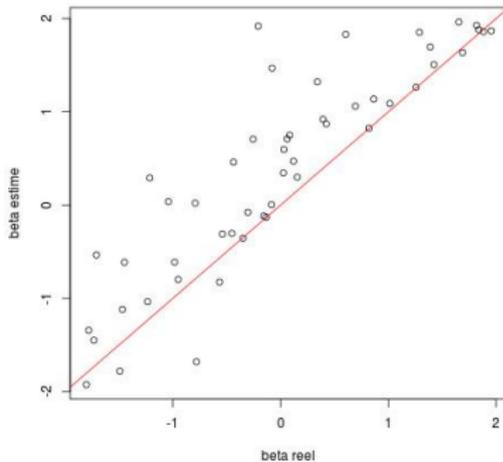
TABLE: EQM algorithme VBEM v2 vs MCMC - 18 simulations

→ EQMs du même ordre sauf pour  $\alpha$

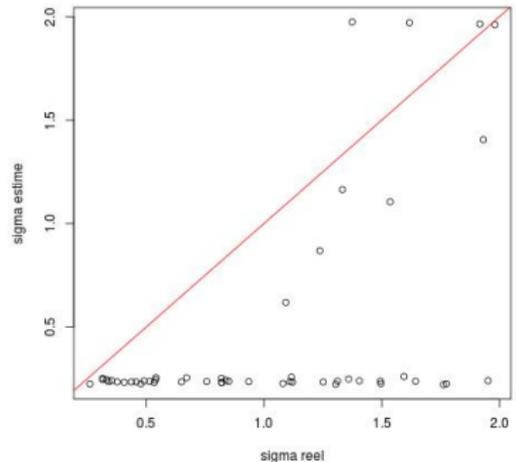
## Evaluation (3/3)

### 3. VBEM (version 2) avec un échantillonnage de 70% lois *a priori* uniformes

Estimation de beta

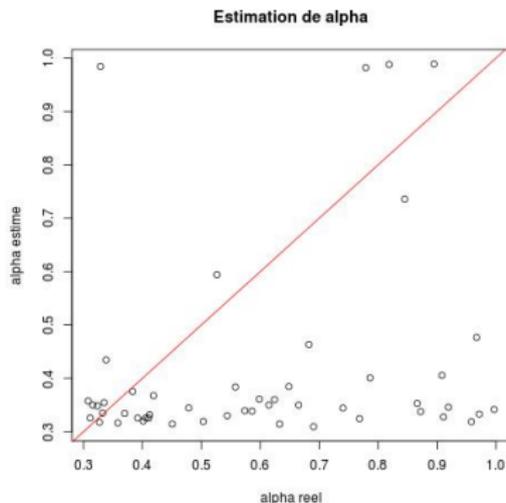


Estimation de sigma



## Evaluation (3/3)

### 3. VBEM (version 2) avec un échantillonnage de 70% lois *a priori* uniformes



# Conclusion

- algorithme d'estimation original et plus rapide que MCMC pour le processus de Cox log gaussien (24h vs 12 min)
- avec la version 2, performances comparables avec MCMC sauf pour  $\alpha$
- premier pas pour la comparaison processus de Cox/champ de Markov

# Perspectives

- poursuite des expérimentations (comparaison avec MCMC+VBEM avec échantillonnage incomplet)
- amélioration de l'estimation de la densité sur les quadrats
- amélioration de l'étape E : levée de l'hypothèse champ moyen  
exemple : algorithme EM-EP (EP : Minka 2001, EM-EP : Kim et Ghahramani 2003)
- prise en compte des données "abondance" et "tache"
- comparaison avec champ de Markov sur données adventices (critère BIC variationnel)

# Merci de votre attention !



## Initialisation de $S$ et $\theta$

### Etape 1 : Mise à jour des composantes de $\theta = (\sigma^2, \alpha)$

- tirage de  $\theta'$  selon  $\mathcal{N}(\theta, v)$
- acceptation de  $\theta'$  avec probabilité  $\Delta(\theta, \theta') = \min\left(\frac{p(S|\theta')q(\theta',\theta)}{p(S|\theta)q(\theta,\theta')}, 1\right)$  où  $q(\theta', \theta)$  est la densité de la loi normale de moyenne  $\theta'$  et de variance  $v$  appliquée à  $\theta$

### Etape 2 : mise à jour des valeurs du champ $S$

- tirage des  $S'_i$  selon  $\mathcal{N}(S_i, v_2)$
- acceptation des  $S'_i$  avec probabilité  $\Delta(S, S') = \min\left(\frac{p(y_i|S'_i, \beta)p(S'_i|\theta)q(S', S)}{p(y_i|S_i, \beta)p(S_i|\theta)q(S, S')}, 1\right)$  où  $q(S', S)$  est la densité de la loi normale de moyenne  $S'$  et de variance  $v_2$  appliquée à  $S$

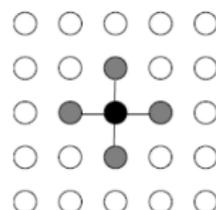
## Etape 3 : Mise à jour de $\beta$

- tirage de  $\beta'$  selon  $\mathcal{N}(\beta, v_3)$
- acceptation de  $\beta'$  avec probabilité

$\Delta(\beta, \beta') = \min \left( \frac{\prod_{i=1}^N p(y_i | S_i, \beta') q(\beta', \beta)}{\prod_{i=1}^N p(y_i | S_i, \beta) q(\beta, \beta')}, 1 \right)$  où  $q(\beta', \beta)$  est la densité de la loi normale de moyenne  $\beta'$  et de variance  $v_3$  appliquée à  $\beta$

# Champ de Markov

$Y_i \in \Omega$  : note d'abondance sur  
le quadrat  $A_i$   
 $\Omega = \{0, 1, \dots, 6\}$



**Champ de Markov : modèle graphique  $G = (V, E)$**

① Propriété markovienne :

$$\forall y \in \Omega, P(Y_i = y_i | Y_{V \setminus \{i\}} = y_{V \setminus \{i\}}, \theta) = P(Y_i = y_i | Y_{N(i)} = y_{N(i)}, \theta)$$

② Positivité :  $\forall y \in \Omega, P(y|\theta) > 0$

théorème de Hammersley-Clifford : champ de Markov  $\Leftrightarrow$  **distribution de Gibbs** :

$$\forall y \in \Omega, P(y|\theta) = \frac{1}{Z(\theta)} \prod_{i \in V} \psi_i(y_i, \theta) \prod_{(i,j) \in E} \psi_{(i,j)}(y_i, y_j, \theta)$$

# Champ de Markov

