Simulation et Monte Carlo: projets 2018-19

Nicolas Chopin

Vous devez former un groupe de trois étudiants au sein de votre groupe de TD, choisir un des projets suivants, et le traiter d'ici la dernière séance de TD, où vous ferez une présentation orale de 15 minutes devant les autres étudiants. Pas besoin de rendre un rapport rédigé: vous pouvez cependant rendre le jour de la soutenance un document contenant certains graphiques et résultats, mais surtout, vous devez envoyer vos programmes à votre chargé de TD, qui vérifiera que ce programme fonctionne bien.

Vous avez tout à fait le droit et même je vous encourage à chercher sur internet ou dans la littérature scientifique des inspirations pour effectuer votre projet. Une seule obligation: citez vos sources!

Si vous bloquez, contactez votre chargé de TD (qui me contactera si nécessaire).

Point Essentiel: toujours évaluer (d'une façon ou d'une autre: intervalles de confiance, box-plots, etc.) l'erreur de Monte Carlo de vos résultats. Dans le cas du MCMC, pensez aussi aux ACF (graphe de la fonction d'autocorrélation pour chaque composantes) et aux "traces" (valeur de chaque composante en fonction du temps, notamment pour déterminer le burn-in). Faites preuve d'un esprit scientifique!

Les questions bonus sont facultatives: elles sont réservées aux étudiants qui veulent s'investir plus dans leur projet. Leur bonne résolution sera récompensée par une meilleure note, mais uniquement si le reste du projet a été bien traité.

The social network

On représente un réseau de n individus par un graphe à n sommets, et des arrêtes aléatoires ($X_{ij} = 1$ si les individus i et j sont reliés par une arrête). On modélise la loi jointe des X_{ij} de la façon suivante:

$$p(x|\theta) = \frac{1}{Z(\theta)} \exp\{\theta S(x)\}$$

où la statistique résume certaines caractéristiques du réseau; par exemple, $S(x) = (\sum_{i<j} x_{ij}, \sum_{i<j<k} x_{ij}x_{jk}x_{ik})$ (nombre d'arrêtes, et nombre de triangles).

1. Proposer et mettre en oeuvre un algorithme pour simuler selon $p(x|\theta)$ pour θ fixé.
2. On cherche maintenant à estimer θ par maximum de vraisemblance, à partir d'un x (unique) observé. (On pourra par exemple considérer le jeu de données des mariages entre familles florentines: http://www.casos.cs.cmu.edu/computational_tools/datasets/sets/padgett/) Exprimer la vraisemblance du modèle, et montrer que la constante de normalisation, non calculable, peut être approchée (à une constante près) par une méthode d'importance sampling, où la loi de proposition est la loi du modèle pour un certain θ_0. (Il faut donc utiliser le Gibbs sampler de la question précédente pour simuler cette loi.) Mettre en oeuvre une méthode de maximisation de la fonction ainsi obtenue (où la constante de normalisation est remplacée par une approximation de Monte Carlo; noter que l'approximation varie selon θ, mais que les simulations du Gibbs sampler sont générées une fois pour toute.) On pourra par exemple utiliser une méthode de type gradient. (L'approche décrite dans cette section est communément appelée MC-MLE ou MCMC-MLE.)
3. Bonus: chercher dans la littérature un meilleur algorithme pour simuler selon $p(x|\theta)$ (par exemple l'algorithme "tie-no-tie" dans "Specification of exponential-family random graph models: terms and computational aspects" de Morris et al)