MADI

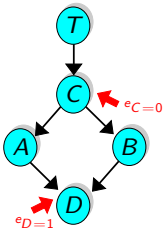
Inférence approchée dans les réseaux Bayésiens

Pierre-Henri WUILLEMIN

DESIR
LIP6
pierre-henri.wuillemin@lip6.fr

Limites de l'inférence exacte : calcul de $P(X \mid e)$

Les réseaux bayésiens sont un outil qui permet d'aggrandir de manière considérable la famille des loi jointes d'un grand nombre de variables aléatoires que l'on peut traiter (informatiquement).



Question : Y a-t-il une limite ?

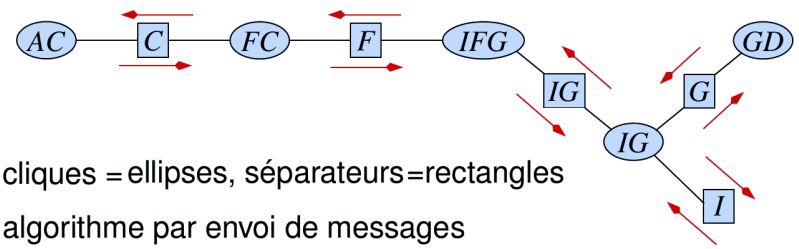
- Occupation mémoire : un nœud avec beaucoup de parents nécessite beaucoup de paramètres pour $P(X \mid \Pi_X)$.
- Temps de traitement : Peut-on toujours mener les calculs sur un BN ?

Complexité de l'inférence dans les BNs.

- L'inférence exacte dans un réseau bayésien est NP-difficile .
- L'inférence approchée dans un réseau bayésien est NP-difficile.

1. Cooper, G. F., 1990. *The computational complexity of probabilistic inference using Bayesian belief networks*. Artif. Intell. 42, 393-405.
2. Dagum, P. & Luby, M., 1993. *Approximating probabilistic inference in Bayesian belief networks is NP-hard*. Artif. Intell. 60, 141–153.

Limites de l'inférence exacte : calcul de $P(X \mid e)$



- cliques = ellipses, séparateurs=rectangles
- algorithme par envoi de messages

$$P(X) = \frac{1}{Z} \prod_{C \in JT} \Phi_C(C)$$

Complexité de l'inférence dans les BNs.

D'une manière générale, la complexité en espace et en temps d'un algorithme d'inférence exact est exponentielle en la *treewidth* du graphe : $O\left(n^2 \cdot e^{maxW}\right)$.

treewidth $\approx$ taille de la plus grande clique.

Limites de l'inférence exacte (3)

En pratique, il est extrêmement facile de construire des réseaux bayésiens qui ne seront pas traitables :

- Un nœud avec un grand nombre de parents.
- Un BN de très grande taille.
  - dynamic BN,
  - Enrichissement du language des BNs pour la représentation de la répétition de motifs : OOBN,
  - etc.
- Des variables aléatoires discrètes avec un grand nombre de valeurs possibles.
- etc.

Dans les applications, on rencontre facilement ce genre de BN. D'où la nécessité d'une inférence approchée.

Inférence approchée dans les BNs

Un algorithme de calcul approché fournit un solution raisonnable dans un temps raisonnable.

Inférences approchées dans les BNs

Pour les BNs, on peut distinguer deux familles principales de méthodes approchées :

- Simplification ou relaxation des algorithmes d'inférence exacte,
- Inférences basées sur la simulation.

Inférence approchée par simulation

Inférence approchée basée sur la simulation

Les inférences basées sur la simulation approchent la loi jointe recherchée par l'inférence grâce à la génération d'un grand nombre d'instances de cette loi. On appelle parfois ces instances des **particules**.

D'où deux grandes questions :

- Qu'est ce qu'on met exactement dans une particule ?
- Comment générer ces particules ?

Méthode de Monte Carlo

Définition (Simulation)

"step by step the probabilities of separate events are merged into a composite picture which gives an approximate but workable answer to the problem"

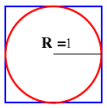
The Monte Carlo Method, D.D. McCracken, Scientific American, 1955

Monte Carlo *[1947 – von Neumann and Ulam (Los Alamos Scientific Laboratory)]*
Projet consistant à utiliser des nombres aléatoires pour simuler des séquences complexes d'évènements :
Simulation de la diffusion des neutrons dans un matériau fissile.

roulette : méthode bien connue de génération de nombres aléatoires.

Autres utilisations fréquentes : Aiguille de Buffon (1777), équations elliptiques ou paraboliques (diffusion), systèmes linéaires, optimisations (recuit), finance, go ...

Un exemple : approximation de π



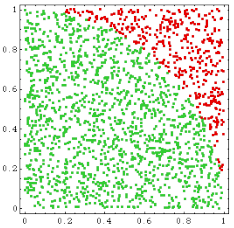
Méthode : on jette des cailloux dans le carré.
Hypothèse : $NbJets \propto Surface$

d'où

$$\frac{NbJets_{Cercle}}{NbJets_{Total}} = \frac{\pi \cdot R^2}{(2 \cdot R)^2} = \frac{\pi}{4}$$

6 jets dans le cercle sur 10 jets en tout $\Rightarrow \hat{\pi}_{10} = 2.4$
89 jets dans le cercle sur 100 jets en tout $\Rightarrow \hat{\pi}_{100} = 3.57$
750 jets dans le cercle sur 1000 jets en tout $\Rightarrow \hat{\pi}_{1000} = 3$
7852 jets dans le cercle sur 10000 jets en tout $\Rightarrow \hat{\pi}_{10000} = 3.1408$

Autrement dit, on choisit aléatoirement un point du carré, et on vérifie (par *distance* ≤ 1) si il est dans le cercle. Puis on itère.



simulation avec 4000 jets

Un exemple : approximation de π - formalisation

En notant $(x_i, y_i)_{i \leq N}$ les positions des N jets successifs.

Soit la fonction indicatrice du disque dans le carré : $\mathbf{1}_{\bigcirc}(x, y) = \begin{cases} 1 \text{ si } x^2 + y^2 \leq 1 \\ 0 \text{ sinon} \end{cases}$

Le calcul précédent revient donc à calculer :

$$\frac{\sum_{i \leq N} \mathbf{1}_{\bigcirc}(x_i, y_i)}{N}$$

Qu'est-on en train d'estimer par une telle méthode ?

La solution idéale du problème serait de tester tous les points du carré pour connaître exactement la fraction de ceux-ci appartenant au cercle :

- Le "nombre" de point du carré : $\int_{(x,y)\in\square} dx\, dy$
- Le "nombre" de point du cercle : $\int_{(x,y)\in\square} \mathbf{1}_{\bigcirc}(x, y) dx\, dy$
- En introduisant une loi p uniforme sur $\square$ (changement de mesure) :
  $\int_{(x,y)\in\square} p(x, y) dx\, dy = 1$ et $\int_{(x,y)\in\square} \mathbf{1}_{\bigcirc}(x, y) p(x, y) dx\, dy$

On estime donc $\int_{(x,y)\in\square} \mathbf{1}_{\bigcirc}(x, y) p(x, y) dx\, dy$ par $\frac{\sum_i \mathbf{1}_{\bigcirc}(x_i, y_i)}{N}$.

Monte Carlo en statistique bayésienne

- Les méthodes de Monte Carlo proposent donc une simulation stochastique pour le calcul d'intégrales (ou d'équations différentielles).
- Il s'avère que l'intégration (ou l'équivalent discret : **la somme**) est une opération fondamentale dans les statistiques (et particulièrement dans la statistique bayésienne) à partir de :

$$posterior \propto L(ikelihood) \times P(rior)$$

- Calculer la constante de normalisation : $\int L \times P$ car $posterior = \frac{L \times P}{\int L \times P}$
- Marginaliser une distribution jointe : $P(x_2) = \int P(x_1, x_2) dx_1$
- Statistiques sur une distribution : $E_P(f) = \int f(x) P(x) dx$
  - Moyenne de P : $f(x) = x$
  - Moment d'ordre 2 de P : $f(x) = x^2$
  - $P(A)$: $f(X) = \mathbf{1}_A$

Synthèse et résultats théoriques sur Monte Carlo

Supposons que nous voulions calculer $\mu = E_P(f) = \int f(x) P(x) dx$.
S'il n'y a pas de résultats analytiques, la méthode de Monte Carlo propose d'utiliser une suite $(X_i)_{i \leq N}$ d'observations de variables aléatoires, i.i.d, suivant la loi P et d'estimer μ par :

$$\widehat{\mu}_N = \frac{1}{N} \sum_{i \leq N} f(X_i)$$

Loi forte des grands nombres

Si $E(|X|) < \infty$ alors
(presque sûrement)

$$\lim_{N \to \infty} \frac{1}{N} \sum_{i \leq N} X_i = E(X)$$

Théorème de la Limite Centrale

Soit $S_n = \sum_{i=1}^{n} X_i$ et $\mu_n = \frac{S_n}{n}$, avec les X_i v.a. indépendante, à variance finie. Alors

$S_n \underset{n \to \infty}{\longrightarrow} \mathcal{N}\left(n \cdot \mu; n \cdot \sigma^2\right)$ ou $\frac{\mu_n - \mu}{\sigma / \sqrt{n}} \underset{n \to \infty}{\longrightarrow} \mathcal{N}(0; 1)$

Propriétés Monte Carlo

$\widehat{\mu}_N \underset{N \to \infty}{\longrightarrow} \mu = \int f(x) P(x) dx$ $\quad \widehat{\sigma}_N^2 = \frac{1}{N-1} \sum_{i \leq N} [f(X_i) - \widehat{\mu}_N]^2$ $\quad \frac{\widehat{\mu}_N - \mu}{\widehat{\sigma}_N / \sqrt{N}} \sim \mathcal{N}(0; 1)$

Monte Carlo : convergence

Estimation de la variance

$$s^2 = \frac{1}{M-1} \sum_{i=1}^{M} \left(f(x_i) - \hat{f}\right)^2 = \frac{\sigma^2}{M}$$

où σ^2 est la variance de P

Rappels : intervalles de confiance

- $[\hat{f} - s, \hat{f} + s]$: 66%
- $[\hat{f} - 2 \cdot s, \hat{f} + 2 \cdot s]$: 95%
- $[\hat{f} - 3 \cdot s, \hat{f} + 3 \cdot s]$: 99%

MonteCarlo dans les BNs : forward sampling

Dans un BN, on veut estimer $\forall i, P(X_i \mid e)$ à partir du BN et de e.

Les questions sont donc :

- Quelles particules pour calculer toutes ces distributions marginales ?
- Comment générer ces particules ?
- Combien en générer ?

Particules dans un BN

Afin de pouvoir calculer toutes les lois marginales dans un BN, une particule doit être une instance de l'ensemble des variables du BN.

On reconstruit une base de données à partir d'un BN : processus inverse de l'apprentissage.

Forward Sampling : génération des particules

Il est aisé de générer une particule (une valeur) d'une loi marginale :

Particule mono-dimensionnelle

Générer une particule pour une loi $P(X)$ consiste à tirer une valeur de X en suivant la distribution $P(X)$.

Comment faire pour un BN ?
Soit p la particule à générer et $p_{<X_i>}$ la valeur dans p de la variable X_i.

$p_{<X>}$ dans un BN

- pour les variables sans parent, la procédure ci-dessus permet de fournir certaines composantes de la particule.
- pour les variables avec parents, si $\forall X_j \in \Pi_X$, $p_{<X_j>}$ a déjà été tiré, il suffit de tirer $p_{<X>}$ suivant la loi $P(X \mid X_j = p_{<X_j>}, \forall X_j \in \Pi_X)$.

Génération d'une particule dans un BN

Générer une particule dans un BN revient donc à itérer la procédure ci-dessus sur l'ensemble des variables dans un ordre topologique.

Forward Sampling : calculer $P(X_i = x)$ par simulation

Inférence approché par forward sampling

Soit $(p_k)_{k \in D}$ l'ensemble des particules générés,

$$P(X_i = x) \approx \hat{P}_D(X_i = x) = \frac{1}{|D|} \sum_{k \in D} \mathbf{1}_{X_i = x}(p_k)$$

où $\mathbf{1}_{X_i=x}(p) = \begin{cases} 1 & \text{si} p_{<X_i>} = x \\ 0 & \text{sinon} \end{cases}$

Complexité de l'algorithme :

$$O(|D| \cdot |\text{BN}| \cdot (\max_{i \in |\text{BN}|} |\Pi_{X_i}| + \log \max_{i \in |\text{BN}|} |X_i|))$$

Forward Sampling : $|D|$ pour estimer $P(X_i = x)$?

Inégalité de Hoeffding [analyse de l'erreur absolue]

Soit $\mathcal{D} = \{X_1, \cdots, X_M\}$ M variables de Bernoulli indépendantes avec une même probabilité de succès p. Soit $T_{\mathcal{D}} = \frac{1}{M} \sum_m X_m$.

$$P(T_{\mathcal{D}} > p + \epsilon) \leq e^{-2M\epsilon^2}$$
$$P(T_{\mathcal{D}} < p - \epsilon) \leq e^{-2M\epsilon^2}$$

Pour le forward sampling,

$$P(|\hat{P}_D(X_i = x) - P(X_i = x)| > \epsilon) \leq 2e^{-2|M|\epsilon^2}$$

Estimation à ϵ près avec un degré de confiance $1 - \delta$

Il faut $2e^{-|M|\epsilon^2} < \delta$, soit :

$$M_a(\epsilon, \delta) \geq \frac{\ln \frac{2}{\delta}}{2\epsilon^2}$$

Forward Sampling : $|D|$ pour estimer $P(X_i = x)$? (2)

Inégalité de Chernoff [analyse de l'erreur relative]

Soit $\mathcal{D} = \{X_1, \cdots, X_M\}$ M variables de Bernoulli indépendantes avec une même probabilité de succès p. Soit $T_{\mathcal{D}} = \frac{1}{M} \sum_m X_m$.

$$P\left(T_{\mathcal{D}} > p(1 + \epsilon)\right) \leq e^{\frac{-Mp\epsilon^2}{3}}$$
$$P\left(T_{\mathcal{D}} < p(1 - \epsilon)\right) \leq e^{\frac{-Mp\epsilon^2}{3}}$$

Pour le forward sampling,

$$P\left(\hat{P}_D(X_i = x) \notin P(X_i = x) \cdot (1 \pm \epsilon)\right) \leq 2e^{-|M| \cdot P(X_i = x) \cdot \epsilon^2 / 3}$$

Estimation à ϵ près avec un degré de confiance $1 - \delta$

Il faut $2e^{-|M| \cdot P(X_i = x) \cdot \epsilon^2} < \delta$, soit :

$$M_r(\epsilon, \delta) \geq 3 \frac{\ln \frac{2}{\delta}}{P(X_i = x) \cdot \epsilon^2}$$

Forward Sampling : $P(X_i = x \mid e)$?

Première idée

$$P(X \mid e) = \frac{P(X, e)}{P(e)} \Rightarrow P(X \mid e) \approx \hat{P}_D(X \mid e) = \frac{\hat{P}_D(X, e)}{\hat{P}_D(e)}$$

Problème : $P(e)$ souvent petit : estimation difficile et demande une grande taille de D.

Seconde idée : *Rejection Sampling*

Générer des particules suivant $P(\cdot \mid e)$ plutôt que suivant $P(\cdot)$:

1. Générer une particule p comme dans le Forward Sampling
2. Si $p_{<e>}$ n'est pas compatible avec e, rejeter la particule

$$P(X_i = x \mid e) \approx \hat{P}_D(X_i = x \mid e) = \frac{1}{|D|} \sum_{k \in D} \mathbf{1}_{X_i = x}(p_k)$$

Problème : $P(e)$ souvent petit $\Rightarrow$ beaucoup de rejets : difficultés pour D de grande taille.

$P(X_i = x \mid e)$: *Likelihood Weighting*

Rejection sampling n'est vraiment pas efficace. Au lieu de rejeter tant de particules, pourquoi ne pas les pondérer par leur vraisemblance suivant e ?

Likelihood Weighting algorithm

1. Générer une particule p suivant Forward Sampling.
2. Forcer $p_{<e>} \leftarrow e$  *Les particules ne sont pas cohérentes !*
3. Associer à p le poids $w_p = \prod_{e_i \in e} P(e_i \mid \Pi_{e_i})$

 $w_p \neq P(e \mid (p \setminus e))$
w_p est la probabilité du tirage successif des e_i dans le processus de *sampling*.

Estimation par *Likelihood Weighting*

$$P(X_i = x \mid e) \approx \hat{P}_D(X_i = x \mid e) = \frac{\sum_{k \in D} w_{p_k} \cdot \mathbf{1}_{X_i = x}(p_k)}{\sum_{k \in D} w_{p_k}}$$

C'est bien une généralisation du Forward Sampling si $e = \emptyset$.
Généralisons encore : Importance Sampling

$P(X_i = x)$: *Importance Sampling*

Rappel : On veut estimer $E_P(f)$ par $\frac{1}{M} \sum_m f(p_m)$ où les p_m sont des particules générées suivant la loi P.
Que faire si la loi P contient des "zones de raretés" rendant difficile l'estimation ?

Sampling distribution

Une loi Q est une loi d'échantillonnage pour $P \iff \forall x, P(x) > 0 \Rightarrow Q(x) > 0$

Comment utiliser Q pour établir une approximation de $E_P(f)$?

Unnormalized (ou Unweighting) Importance Sampling

$$E_P(f) = E_Q\left(f \cdot \frac{P}{Q}\right)$$

$$\text{alors } E_P(f) \approx \frac{1}{M} \sum_m f(p_m) \cdot \frac{P(p_m)}{Q(p_m)} \text{ où les } p_m \text{ sont générées suivant } Q$$

preuve : $E_Q(f \frac{P}{Q}) = \sum_x Q(x) \cdot f(x) \cdot \frac{P(x)}{Q(x)} = \sum_x f(x) \cdot P(x) = E_P(f)$

$P(X_i = x \mid e)$: *Importance Sampling* (2)

On utilise $E_P(f) = E_Q\left(f \cdot \frac{P}{Q}\right)$ pour échantillonner.

Problème : Si la loi cherchée est $P(X \mid e)$, on ne connait pas la loi ! Il faudrait la calculer pour $\frac{P}{Q}$ ...

Supposons $\tilde{P}$ connue vérifiant $P \propto \tilde{P}$ ($\exists Z, P = \frac{1}{Z}\tilde{P}$, $Z = \sum_x \tilde{P}(x)$).

$$E_P(f) = \sum_x Q(x) \cdot f(x) \cdot \frac{P(x)}{Q(x)} = \frac{1}{Z}\sum_x Q(x) \cdot f(x) \cdot \frac{\tilde{P}(x)}{Q(x)} = \frac{\sum_x Q(x) \cdot f(x) \cdot \frac{\tilde{P}(x)}{Q(x)}}{\sum_x \tilde{P}(x)} = \frac{\sum_x Q(x) \cdot f(x) \cdot \frac{\tilde{P}(x)}{Q(x)}}{\sum_x Q(x) \cdot \frac{\tilde{P}(x)}{Q(x)}}$$

$$E_P(f) = \frac{E_Q\left(f\frac{\tilde{P}}{Q}\right)}{E_Q\left(\frac{\tilde{P}}{Q}\right)}$$

Importance Sampling

En nommant $\omega(x) = \frac{\tilde{P}(x)}{Q(x)}$,

$$E_P(f) \approx \frac{\sum_m f(p_m) \cdot \omega(p_m)}{\sum_m \omega(p_m)}$$ où les p_m sont générées suivant Q.

Importance sampling : calcul de petites probabilités

Soit $Z \sim \mathcal{N}(0,1)$ (loi normale centrée réduite,fonction de densité $\phi(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$), calculer :

$$p_a = P(Z \geq a) = \int_a^{+\infty} \phi(t)dt = E_Z(I_{(Z \geq a)})$$

Monte Carlo

Tirer $p_1, \cdots, p_M$ en suivant $\mathcal{N}(0,1)$ afin de calculer

$$\hat{p}_a = \frac{1}{M}\sum_i I_{(Z \geq a)}(p_i)$$

Pour $a = 5$, on a déjà : $p_a = 2.87 \cdot 10^{-7}$! Environ 1 sur 3.5 millions de $I_{(Z \geq a)}(p_i)$ vaut 1...

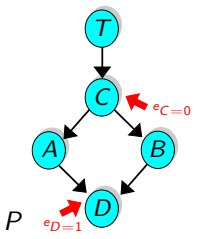
Importance sampling

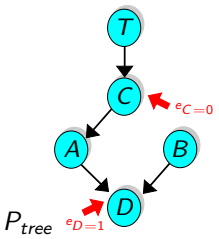
Tirer $p_1, \cdots, p_M$ en suivant $\mathcal{N}(a,1)$ afin de calculer

$$\hat{p}_a^{[IS]} = \frac{1}{M}\sum_i I_{(Z \geq a)}(p_i)\frac{\phi(p_i)}{\phi(p_i - a)}$$

$P(X_i = x \mid e)$: *Importance Sampling* (3)

$$E_P(f) \approx \frac{\sum_m f(p_m) \cdot \omega(p_m)}{\sum_m \omega(p_m)}$$ où $\begin{cases} \text{les } p_m \text{ sont générées suivant } Q. \\ \omega(x) = \frac{\tilde{P}(x)}{Q(x)} \end{cases}$



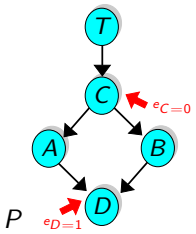


Importance sampling dans un réseau bayésien

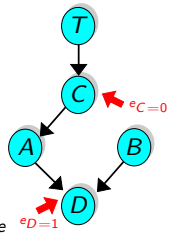
- $Q(X) = P_{tree}(X \mid e)$ calculable en temps polynomial (Pearl, 88)
- $\tilde{P}(X) = P(X, e)$ donc $\tilde{P}(p_m)$ calculable en temps polynomial.

$P(X_i = x \mid e)$: *Importance Sampling* (3)

$$E_P(f) \approx \frac{\sum_m f(p_m) \cdot \omega(p_m)}{\sum_m \omega(p_m)} \text{ où } \begin{cases} \text{les } p_m \text{ sont générées suivant } Q. \\ \omega(x) = \frac{\tilde{P}(x)}{Q(x)} \end{cases}$$



P

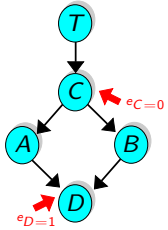


P_{tree}

Importance sampling dans un réseau bayésien

- $Q(X) = P_{tree}(X \mid e)$ calculable en temps polynomial (Pearl, 88)
- $\tilde{P}(X) = P(X, e)$ donc $\tilde{P}(p_m)$ calculable en temps polynomial.

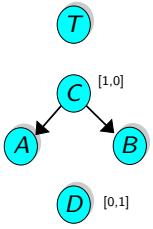
Likelihood Weighting vs. Importance Sampling



BN contextualisé

Soit un BN B et une information e, on appelle 'BN contextualisé' le BN B_e construit ainsi,

- $\forall X \in B, X \notin e, \Pi_{X[B_e]} = \Pi_{X[B]}$, de même loi.
- $\forall X \in B, X \in e, \Pi_{X[B_e]} = \emptyset$, $P_{B_e}(X) = \delta_e$.



Équivalence LW et IS

Pour calculer $P(X \mid e)$, l'algorithme LW correspond exactement à IS avec comme *Sampling Distribution* la loi du BN contextualisé par e.

Estimation de l'erreur

Pour estimer la probabilité $P(X_i = x \mid e)$ par LW pour une erreur relative ϵ avec un degré de confiance $1 - \delta$, il faut itérer tant que :

$$\sum_p w_p < \frac{4(1+\epsilon)}{\epsilon^2} \ln \frac{2}{\delta} \cdot u^{|e|} \text{ où } u \text{ est un majorant des paramètres du BN.}$$

Limites des méthodes de MonteCarlo

Une limitation assez évidente de ces algorithmes est leur dépendance à la position des observations : une observation placée à une racine va être facilement prise en compte alors qu'une observation à une feuille sera beaucoup plus mal gérée. Ceci est principalement dû à la nécessité du tirage d'un échantillon dans un ordre topologique ... Comment faire autrement ?

Pour estimer $\mu = \int f(x)P(x)dx$, la méthode de Monte Carlo propose d'utiliser une suite $(X_i)_{i \leq N}$ d'observations de v.a., i.i.d, suivant la loi π et d'estimer μ par $\widehat{\mu}_N = \frac{1}{N} \sum_{i \leq N} f(X_i)$.

Que faire si il n'est pas possible d'obtenir des variables aléatoires i.i.d suivant π ?

Principe de MCMC : Monte Carlo Markov Chain

Il s'agirait de construire une suite (X_i) de variables aléatoires qui seraient (presque) i.i.d , suivant π. Il serait alors possible d'utiliser la méthode de Monte Carlo pour estimer μ.

Chaîne de Markov

Plus fréquemment qu'une relation fonctionnelle entre les (X_n), on peut étudier des relations d'indépendances conditionnelles entre ces différentes variables aléatoires.

Propriété de Markov

Un processus stochastique vérifie la propriété de Markov (d'ordre 1) si et seulement si :

$$P(X_n \mid X_0, \cdots, X_{n-1}) = P(X_n \mid X_{n-1})$$

(lorsque cette probabilité a un sens : i.e. $P(X_0, \cdots, X_{n-1}) > 0$)

Définition (Chaîne de Markov)

Une chaîne de Markov est un processus stochastique vérifiant la propriété de Markov (d'ordre 1).

Chaîne de Markov homogène

Propriété (Homogénéité)

Une chaîne de Markov est dite homogène si

$$\forall n > 0, P(X_n \mid X_{n-1}) = P(X_1 \mid X_0)$$

Définition (Probabilité et matrice de transition)

Soit $(X_n)_{n\in\mathbb{N}}$ une chaîne de Markov homogène, alors

- *la probabilité de transition de i à j est $p_{ij} = P(X_n = j \mid X_{n-1} = i)$*
- *la matrice de transition P est la matrice des $(p_{ij})_{i,j\in S}$ (si S est fini).*

Définition (Graphe de transition)

Si $(X_n)_{n\in\mathbb{N}}$ une chaîne de Markov homogène (S fini), alors le graphe de transition est un graphe $G = (S, E)$ orienté qui vérifie :
$(i \to j) \in E \iff p_{ij} \neq 0$

Étude en régime permanent

Ce qui nous intéresse ici est le comportement de la chaîne de Markov si on laisse se dérouler le processus durant un temps très important.
Que peut-on dire de la position du système ? Suit-il une loi de probabilité particulière ?
En notant $\pi^{(n)}$ le vecteur de probabilité du système à l'instant n, on se rappelle que :

$$\pi^{(n+1)} = \pi^{(n)} \cdot P = \pi^{(0)} \cdot P^n$$

Définition (distribution de probabilité invariante)

Une distribution de probabilité est invariante pour la chaîne de Markov si et seulement si elle s'écrit comme le vecteur π et :

$$\pi = \pi \cdot P$$

i.e. : π est un vecteur propre de P^T pour la valeur propre 1

En supposant que $(\pi^{(n)})_{n\in\mathbb{N}}$ converge vers π^* alors :

$$\pi^* = lim_{n\to\infty}\pi^{(n)} = \pi^{(0)} \cdot lim_{n\to\infty}P^n = \pi^{(0)} \cdot P^*$$

Propriété

$(\pi^{(n)})_{n\in\mathbb{N}}$ converge vers π^* indépendamment de $\pi^{(0)}$ si et seulement si $lim_{n\to\infty}P^{(n)} = P^*$, matrice dont toutes les lignes sont égales entre elles (et égalent à π^*).

Ergodicité

Définition (Chaîne de Markov ergodique)

Une chaîne de Markov est ergodique si et seulement si elle est irréductible, apériodique et récurente positive.

Théorème (théorème ergodique)

Une chaîne de Markov ergodique est telle que $(\pi^{(n)})_{n\in\mathbb{N}}$ *converge, quelque soit* $\pi^{(0)}$, *vers* π^* *vérifiant :*

$$\begin{cases} \pi^* \cdot P = \pi^* \\ \pi^* \cdot 1 = 1 \end{cases}$$

De plus, $\pi^*_j = \frac{1}{M_j}$

Autrement dit, la proportion des instants où la chaîne se trouve dans l'état j *tend vers* π^*_j *avec probabilité 1. Pour presque toutes les trajectoires,* **la moyenne temporelle est identique à la moyenne spatiale**.

Limites de Monte Carlo et solution

Pour estimer $\mu = \int f(x)P(x)dx$, la méthode de Monte Carlo propose d'utiliser une suite $(X_i)_{i\leq N}$ d'observations de v.a., i.i.d, suivant la loi π et d'estimer μ par $\widehat{\mu}_N = \frac{1}{N}\sum_{i\leq N} f(X_i)$.

Que faire si il n'est pas possible d'obtenir des variables aléatoires i.i.d suivant π ?

Principe de MCMC : Monte Carlo Markov Chain

Il s'agirait de construire une suite (X_i) de variables aléatoires qui seraient (presque) i.i.d , suivant π. Il serait alors possible d'utiliser la méthode de Monte Carlo pour estimer μ.
Une Chaîne de Markov (à temps discret), ergodique, de loi stationnaire π ($\pi = \pi \cdot P$) est un processus stochastique qui permet de générer une telle suite (X_i).

PS : il faut un "certain nombre" d'itérations pour qu'une chaîne de Markov s'approche de la convergence (c'est à dire $P(X_t) \approx \pi$). Cette période (ou burn-in) passée, on peut considérer que $X_t \perp\!\!\!\perp X_{t+1}$, puisque les 2 v.a. suivent la même loi π.

MCMC : Monte Carlo Markov Chain

Changement de point de vue

- Quand on étudie les MC(TD), à partir d'une matrice de transition P, il s'agit de trouver la distribution stationnaire π.
- Pour les MCMC, étant donnée une loi π, il s'agit de construire une MC(TD) convergent vers cette loi π.

Comment construire cette chaîne de Markov ?

Définition (Algorithme de Metropolis-Hastings – 1953)

Soit les lois $q(X \mid Y)$ *lois candidates ou instrumentales, on construit alors* $\alpha(X,Y) = \min\left(1, \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)}\right)$

- *Soit* x_t *la position courante du processus stochastique*
- *Itérations :*
  1. *Proposer un candidat* y *suivant la loi* $q(. \mid x_t)$
  2. *Calculer* $\alpha(x_t, y)$
  3. *Avec la probabilité* $\alpha(x_t, y)$, $x_{t+1} = y$, *sinon* $x_{t+1} = x_t$
- *les* $(x_t)_{m\leq t\leq N}$ *forment une suite de v.a. i.i.d utilisables pour une approximation MC.*

MCMC - suite

1. Proposer un candidat y suivant la loi $q(. \mid x_t)$
2. Calculer $\alpha(x_t, y)$
3. Avec la probabilité $\alpha(x_t, y)$, $x_{t+1} = y$: acceptation, sinon $x_{t+1} = x_t$: rejet

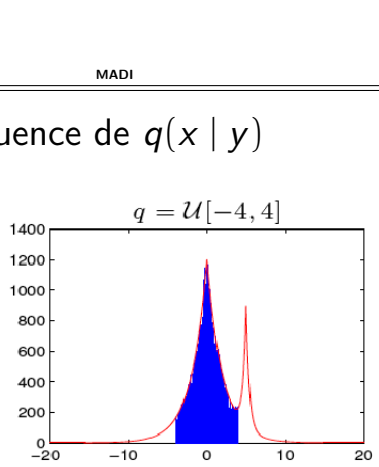
Les étapes 1 et 3 sont indépendantes, on peut donc calculer la probabilité de transition par :

$$\begin{cases} P(X_{t+1} = y \mid X_t = x) = q(y \mid x) \cdot \alpha(x, y) & \forall x \neq y \\ P(X_{t+1} = x \mid X_t = x) = 1 - \sum_{y \neq x} P(X_{t+1} = y \mid X_t = x) \end{cases}$$

Cette chaîne de Markov est irréductible et apériodique en fonction de $q(x \mid y)$ et $\alpha(x, y)$. Quelle est son point fixe ?

$\alpha(X, Y) = \min\left(1, \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)}\right)$
$\Rightarrow \pi(X_t)q(X_{t+1} \mid X_t)\alpha(X_t, X_{t+1}) = \pi(X_{t+1})q(X_t \mid X_{t+1})\alpha(X_{t+1}, X_t)$
$\Rightarrow \pi(X_t)P(X_{t+1} \mid X_t) = \pi(X_{t+1})P(X_t \mid X_{t+1})$
En sommant sur X_t : $\sum_x \pi(X_t = x)P(X_{t+1} \mid X_t = x) = \pi(X_{t+1}) \Rightarrow \pi \cdot P = \pi$

Si cette CM(TD) converge, c'est vers π ■



Influence de $q(x \mid y)$

Une méthode MCMC dans les BNs : Gibbs sampling

Le Gibbs sampling est une méthode MCMC qui prend une forme très simple dans les BNs :

Itération $t + 1$ du Gibbs Sampling

soit p^t la particule échantillonnée à t, on note $-X$ toutes les variables du BN sauf X,

$$\exists X \in \text{BN}, \begin{cases} p^{t+1}_{<X>} \text{ est tirée suivant la distribution } P(X \mid p^t_{<-X>}) \\ p^{t+1}_{<-X>} = p^t_{<-X>} \end{cases}$$

rappel : La couverture de Markov d'un nœud est constitué de ses parents, de ses enfants et des parents de ses enfants.

En fait, calculer $P(X \mid p^t_{<-X>})$ ne nécessite que la couverture de Markov (Markov Blanket) de X car, dans un BN, $X \perp\!\!\!\perp -X \mid MB(X)$.

Inférence approchée par simplification

Approximation de la distribution à calculer

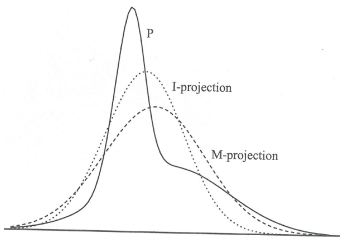
Constat : calculer $P(X \mid e)$ est trop difficile.
But : trouver une loi $Q(X)$ plus facile et "proche" de $P(X \mid e)$.

Soit $\mathfrak{Q}$ un ensemble (convexe) de distributions faciles,
il s'agit de trouver $\min_{Q \in \mathfrak{Q}}(\text{distance}(Q,P))$ en utilisant $D_{KL}(P\|Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$!

D_{KL} n'est pas symétrique ! !

I-projection et M-projection

Moment projection : $Q_M = \arg\min_{Q \in \mathfrak{Q}} D_{KL}(P\|Q)$
Information projection : $Q_I = \arg\min_{Q \in \mathfrak{Q}} D_{KL}(Q\|P)$



- M-projection préfère rendre probable tout x où $P(x) > 0$,
- I-projection privilégie les grandes probabilités de P

I-projection et M-projection

M-projection

Moment projection : $Q_M = \arg\min_{Q \in \mathfrak{Q}} D_{KL}(P\|Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$

- Supporté par P : la "vraie" loi,
- M-projection est certainement le calcul le plus correct,
- Mais nécessite $P(x)$: on ne sait pas calculer !

I-projection

Information projection : $Q_I = \arg\min_{Q \in \mathfrak{Q}} D_{KL}(Q\|P) = \sum_x Q(x) \log \frac{Q(x)}{P(x)}$

- Supporté par Q : la "fausse" loi simplifiée,
- I-projection a tendance à sur-évaluer les probas fortes,
- Mais nécessite $Q(x)$: on devrait savoir calculer !

Calcul de la plus proche distribution factorisée

On suppose $P(X, Y)$ connue. On veut trouver la plus proche distribution où les 2 variables sont indépendantes.
On cherche donc $Q(X, Y) = Q(X) \cdot Q(Y)$ la plus proche de P.

Paramètres à estimer :

Fonction à minimiser :

Sous les contraintes :

Calcul de la plus proche distribution factorisée (2)

Lagrangien :

Dérivées partielles w.r.t. θ_{x^k} :

Point selle = dérivées nulles :

Valeur de λ_x **(resp.** λ_y**)** :

Valeur de θ_x **(resp.** θ_y**)** :

Calcul du KL pour la I-projection

$$\begin{aligned} D_{KL}(Q||P) &= \sum_x Q(x) \log \frac{Q(x)}{P(x)} \\ &= \sum_x Q(x) \log Q(x) - \sum_x Q(x) \log P(x) \\ &= -H(Q) + H(Q, P) \end{aligned}$$

$$D_{KL}(Q||P) = H(Q, P) - H(Q)$$

- $H(Q)$: Entropy
- $H(Q, P) = E_Q(\log P)$: Cross Entropy

J. E. Shore and R. W. Johnson, *Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy*, Information Theory, IEEE Transactions on, vol. 26, no 1, pp. 26-37, Jan. 1980.

Mean Field Approximation

On veut calculer : $P(X) = \frac{1}{Z} \prod_{C \in JT} \Phi_C(C)$ les observations sont intégrées dans les $\Phi_C(C)$.

Mean Field : $Q(x)$ au plus simple

$$Q(x) = \prod_{X_i} Q_i(X_i)$$

Entropy : facile à calculer

$H(Q) = \sum_x Q(x) \log Q(x) = \sum_i \sum_{X_i} Q_i(X_i) \log Q_i(X_i)$

Cross Entropy : facile à calculer (à une translation de $-\log Z$ près)

$\mathrm{H}(Q, P) = \sum_x Q(x) \log P(x) = \sum_{C \in JT} \sum_C \prod_{X_i \in C} Q_i(X_i) \log \Phi_C(C) - \log Z$

Minimisation de $D_{KL}(Q||P)$

Dans un cas général, on a :
$$\begin{aligned} D_{KL}(Q||P) &= H(Q,P) - H(Q) \\ &= -\mathbb{E}_Q(\log P) + \mathbb{E}_Q(\log Q) \\ &= \mathbb{E}_Q(\log Q) - \sum_C \mathbb{E}_Q(\log \Phi_C) + \log Z \end{aligned}$$

On appelle fonction d'énergie :
$\mathcal{F}[P, Q] = \sum_C \mathbb{E}_Q(\log \Phi_C) - \mathbb{E}_Q(\log Q) = \sum_C \mathbb{E}_Q(\log \Phi_C) - H(Q)$

$\log Z = D_{KL}(Q||P) + \mathcal{F}[P, Q]$, constante dans l'inférence en cours.

Conséquences

- $\min D_{KL}(Q||P) \iff \max \mathcal{F}[P, Q]$
- Comme $D_{KL}(Q||P) \geq 0$ et tend (au mieux) vers 0, $\log Z \geq \mathcal{F}[P, Q]$ qui tend au mieux vers $\log Z$
- Trouver Q le plus proche de P correspond trouver Q pour que $\mathcal{F}[P, Q]$, borne inférieure de $\log Z$, en soit le plus proche possible $\Rightarrow$ Estimation de Z.

Inférence approchée comme une optimisation

Mean Field inference

$$\begin{aligned} \max \quad & \mathcal{F}[P, \quad (Q_1, \cdots, Q_n)] \\ \text{s.c.} \quad & \forall j, \sum_{X_j} Q_j(X_j) = 1 \\ & \forall j, Q_j(X_j) \geq 0 \end{aligned}$$

Optimisation sous contrainte d'une fonction dérivable $\Rightarrow$ multiplicateurs de Lagrange :$L(Q; \lambda) = \mathcal{F}[P, (Q_1, \cdots, Q_n)] + \sum_j \lambda_j \left(\sum_{X_j} Q_j(X_j) - 1 \right)$
La caractérisation du point selle donne :

Solution de l'approximation Mean Field

Q est un point stationnaire pour Mean Field $\iff$

$$Q(X_j = x_j) = \frac{1}{Z_j} \exp \sum_{C \ni X_j} \mathbb{E}_Q(\log \Phi_C(., x_j))$$

Il n'y a pas unicité du point stationnaire.

Algorithme Mean Field

MeanField(Φ, Q_0)

- $ATraiter = X$
- $TantQueATraiter \neq \emptyset$
  - choisir $X_j \in ATraiter$
  - Mise à jour de Q_j

$$Q^{t+1}(X_j = x_j) = \exp \sum_{C \ni X_j} \mathbb{E}_{Q^t}(\log \Phi_C(., x_j))$$

  - Normaliser Q_j^{t+1}
  - $ATraiter \leftarrow ATraiter \setminus \{X_j\}$
  - Si $Q^t \neq Q^{t+1}$ Alors $ATraiter \leftarrow ATraiter \cup voisins(X_j)$
- $FinTantQue$

Certitude de convergence pour cet algorithme, mais pas forcément vers la loi P recherchée.

Mean Field et méthode variationnelles

Variationnelle = généralisation de Mean Field

Idée : utiliser une famille $\mathfrak{Q}$ moins naïve que le produit de marginales.
Donc proposer pour Q une structure calculable simplement mais plus complexe. (arbre, ...).

For more :
Probabilistic Graphical Models : Principles and Techniques, Daphne Koller and Nir Friedman, MIT Press, 2009