

□ **Échantillonnage de Gibbs** – L'algorithme d'échantillonnage de Gibbs (en anglais *Gibbs sampling*) est une méthode itérative et approximative qui utilise un petit ensemble d'affectations (particules) pour représenter une loi de probabilité. Pour une affectation aléatoire x , l'échantillonnage de Gibbs effectue les étapes suivantes pour $i \in \{1, \dots, n\}$ jusqu'à convergence :

- Pour tout $u \in \text{Domain}_i$, on calcule le poids $w(u)$ de l'affectation x où $X_i = u$
- On échantillonne v de la loi de probabilité engendrée par $w : v \sim P(X_i = v | X_{-i} = x_{-i})$
- On pose $X_i = v$

Remarque : X_{-i} veut dire $X \setminus \{X_i\}$ et x_{-i} représente l'affectation correspondante.

□ **Filtrage particulaire** – L'algorithme de filtrage particulaire (en anglais *particle filtering*) approxime la densité postérieure de variables d'états à partir des variables observées en suivant K particules à la fois. En commençant avec un ensemble de particules C de taille K , on répète les 3 étapes suivantes :

- Étape 1 : proposition – Pour chaque particule $x_{t-1} \in C$, on échantillonne x avec loi de probabilité $p(x|x_{t-1})$ et on ajoute x à un ensemble C' .
- Étape 2 : pondération – On associe chaque x de l'ensemble C' au poids $w(x) = p(e_t|x)$, où e_t est l'observation vue à l'instant t .
- Étape 3 : é-échantillonnage – On échantillonne K éléments de l'ensemble C' en utilisant la loi de probabilité engendrée par w et on les met dans C : ce sont les particules courantes x_t .

Remarque : une version plus coûteuse de cet algorithme tient aussi compte des particules passées à l'étape de proposition.

□ **Maximum de vraisemblance** – Si l'on ne connaît pas les lois de probabilité locales, on peut les trouver en utilisant le maximum de vraisemblance.

$$\max_{\theta} \prod_{x \in \mathcal{D}_{\text{train}}} p(X = x; \theta)$$

□ **Lissage de Laplace** – Pour chaque loi de probabilité d et affectation partielle $(x_{\text{Parents}(i)}, x_i)$, on ajoute λ à $\text{count}_d(x_{\text{Parents}(i)}, x_i)$ et on normalise ensuite pour obtenir des probabilités.

□ **Espérance-maximisation** – L'algorithme d'espérance-maximisation (en anglais *expectation-maximization* ou *EM*) est une méthode efficace utilisée pour estimer le paramètre θ via l'estimation du maximum de vraisemblance en construisant de manière répétée une borne inférieure de la vraisemblance (étape E) et en optimisant cette borne inférieure (étape M) :

- Étape E : on évalue la probabilité postérieure $q(h)$ que chaque point e vienne d'une partition particulière h avec :

$$q(h) = P(H = h | E = e; \theta)$$

- Étape M : on utilise la probabilité postérieure $q(h)$ en tant que poids de la partition h sur les points e pour déterminer θ through via le maximum de vraisemblance.