L'apprentissage bayésien à l'heure des LLM

Pierre Alquier (ENSAE 2003), ESSEC Business School

1 La statistique bayésienne et le succès des méthodes de Monte Carlo

Le point de vue bayésien offre un paradigme extrêment élégant pour l'estimation statistique. Etant donné un modèle statistique indexé par un paramètre $\theta \in \Theta$ et une fonction de vraisemblance $p(x|\theta)$ pour un échantillon $x = (x_1, \ldots, x_n)$, la spécification d'une loi *a priori* π conduit à l'existence d'une loi *a posteriori* pour un échantillon x :

$$\pi(\mathrm{d}\theta|x) = \frac{p(x|\theta)\pi(\mathrm{d}\theta)}{\int_\vartheta p(x|\vartheta)\pi(\mathrm{d}\vartheta)}. \tag{1.1}$$

La loi *a posteriori* contient toute l'information nécessaire pour l'inférence sur le paramètre : estimation, intervalles de confiance, tests, etc. La principale difficulté rencontrée en pratique est que, mis à part dans un nombre limité de modèles (dit conjugués) dans lesquels la loi *a posteriori* admet une forme explicite, toutes les opérations sur $\pi(\mathrm{d}\theta|x)$ (échantillonnage, calcul d'espérance) sont très difficiles, en particulier à cause de l'intégrale au dénominateur (1.1).

Le développement des méthodes de Monte Carlo, et en particulier des algorithmes de Monte Carlo par chaînes de Markov (MCMC) au cours des années 1970 a été la clé qui a permis de populariser l'approche bayésienne dans les applications de la statistique, au cours des années 80 et 90. Par exemple, l'algorithme de Gibbs proposé en 1984 [5]. A l'ENSAE, ces méthodes sont arrivées très tôt avec les cours de Christian Robert, devenus depuis des monographies [9] mondialement connues.

Ces algorithmes restent depuis très utilisés dans les applications, et ont évidemment connu de nombreuses améliorations. Il était donc très tentant, lors de l'accélération de la recherche en *machine learning* (apprentissage statistique) au début des années 2000, de penser que les méthodes de Monte Carlo pourraient jouer un rôle central dans le développement d'une approche bayésienne de l'apprentissage. Malheureusement, la mise à jour des paramètres dans les algorithmes MCMC devient très lente lorsque la dimension des paramètres augmente, et le nombre d'itérations nécessaires pour arriver à la convergence évolue dans certains cas exponentiellement. Or, les modèles utilisés en apprentissage ont un nombre de paramètres énorme : réseaux de neurones profonds, modèles de languages (LLM), etc. A titre d'example, il y avait 175×10^9 paramètres dans GPT-3.5. L'application des méthodes MCMC s'est avérée tout simplement impossible dans de tels modèles.

Si on regarde de plus près, même dans le cas de l'apprentissage non bayésien, la taille des paramètres et le volume de données ne permettent pas l'utilisation d'algorithmes très sophistiqués. On est plutôt revenu à des méthodes très frustes. Les vraisemblances (ou plus généralement, les fonctions de coût) sont optimisées par des méthodes du premier ordre : descente de gradient, et le gradient lui-même est approché par sous-échantillonage. Tous les algorithmes utilisés pour entraîner des réseaux de neurones reposent sur (des variantes de) l'algorithme de gradient stochastique, développé dans les années... 1950[8].

En résumé, quitte à simplifier à outrance : l'intégration dans la relation (1.1) n'est plus faisable dans des régimes où l'optimisation l'est encore. De très nombreux travaux de recherche depuis les années 2000 ont donc tenté d'approcher la loi *a posteriori* par des algorithmes d'optimisation. La section suivante montre que ceci peut être justifié par un principe général.

2 La loi a posteriori comme solution d'un problème d'optimisation

On peut démontrer que la loi *a posteriori* $\pi(\cdot|x)$ est la solution (unique) d'un problème d'optimisation :

$$\pi(\cdot|x) = \arg\max_{q\in\mathcal{M}(\Theta)} \left\{ \int \log[p(x|\theta)]q(\mathrm{d}\theta) - \mathrm{KL}(q,\pi) \right\} \tag{2.1}$$

où $\mathcal{M}(\Theta)$ est l'ensemble de toutes les lois de probabilités sur Θ, et KL est la divergence de Kullback-Leibler, donnée par

$$\mathrm{KL}(q,\pi) = \int \log\left(\frac{\mathrm{d}q}{\mathrm{d}\pi}(\theta)\right) q(\mathrm{d}\theta)$$

si q admet une densité par rapport à π, et $\mathrm{KL}(q,\pi) = +\infty$ sinon. L'intuition menant à la relation (2.1) est venue d'une analogie formelle avec la thermodynamique statistique (dans l'appendice, je discuterai cette analogie et donnerai une dérivation informelle de (2.1)). Le lecteur habitué au *machine learning* aura reconnu une forme habituelle, avec un terme d'attache aux données : l'intégrale de la log-vraisemblance, et une pénalité, mesurant ici l'éloignement de q par rapport à la loi *a priori* π : $\mathrm{KL}(q,\pi)$.

Le programme d'optimisation (2.1) est sous-jacent à la plupart des approches pour l'apprentissage bayésien en grande dimension : les algorithmes proposés reposent sur une modification ou une relaxation de ce problème.

3 Trois relaxations de l'approche bayésienne

Une première relaxation consiste à restreindre le domaine d'optimisation : on ne maximise plus (2.1) sur l'ensemble de toutes les lois de probabilités, mais dans un sous-ensemble

$\mathcal{Q} \subset \mathcal{M}(\Theta)$. L'objectif devient donc :

$$\max_{q \in \mathcal{Q}} \left\{ \int \log[p(x|\theta)] q(\mathrm{d}\theta) - \mathrm{KL}(q, \pi) \right\}. \tag{3.1}$$

Historiquement, cette approche est apparue pour des paramètres de grande dimension $\theta = (\theta_1, \ldots, \theta_d)$, avec $\mathcal{Q}$ l'ensemble des lois sur Θ rendant toutes les coordonnées indépendantes les unes des autres. Autrement dit, $q(\mathrm{d}\theta) = q_1(\mathrm{d}\theta_1) \ldots q_d(\mathrm{d}\theta_d)$. Dans beaucoup de modèles intéressants en pratique, où il n'y a pas de conjugaison sur θ, on retrouve un effet de conjugaison sur chacun des θ_i individuellement, ce qui facilite grandement l'optimisation de la relation (3.1) coordonnée par coordonnée. Ceci est remarquablement expliqué dans un des articles originaux [6]. Cette première relaxation de l'approche bayésienne est connue sous le nom d'approximation variationnelle. Elle fait aujourd'hui pleinement partie de la boîte à outils bayésienne de tout spécialiste de l'apprentissage. Pour les développements plus récents, je renvoie le lecteur à l'article [4]. Une version plus drastique de l'approche variationnelle consiste à chercher une approximation de la loi *a posteriori* sous une forme paramétrique. Par exemple, $\mathcal{Q}$ est l'ensemble de toutes les lois gaussiennes sur θ : $q = \mathcal{N}(\mu, \Sigma)$. Le problème de maximisation est alors en dimension finie : on peut utiliser un simple algorithme de gradient stochastique pour maximiser (3.1). Plus la famille $\mathcal{Q}$ est petite, plus le programme d'optimisation devient facile, mais le prix à payer est évidemment une approximation de moins bonne qualité de la loi *a posteriori*. En pratique, cela semble être un prix inévitable à payer pour arriver à utiliser l'approche bayésienne afin d'entraîner des réseaux de neurones (vraiment) profonds ou des LLM. Les réseaux de neurones profonds bayésiens s'approchant de l'état de l'art en reconnaissance d'image reposent tous sur des approximations gaussiennes de la loi *a posteriori*. Parmi les examples marquants, l'équipe de Khan a réussi à entraîner GPT-2 (773×10^6 paramètres) en utilisant une approximation variationnelle gaussienne [10].

Dans les approches non-bayésiennes, les réseaux de neurones profonds sont souvent entraînés en optimisant des critères qui ne sont pas exactement une log-vraisemblance : par exemple, des mesures de performance prédictive. Une deuxième relaxation consiste donc à remplacer la log-vraisemblance dans (2.1) ou (3.1) par un tel critère :

$$\max_{q \in \mathcal{Q}} \left\{ \int C(x, \theta) q(\mathrm{d}\theta) - \mathrm{KL}(q, \pi) \right\}. \tag{3.2}$$

J'espère vous convaincre que c'est une idée naturelle souvent utilisée. Dans le cas de la régression linéaire avec un bruit gaussien (de variance 1, pour simplifier), $x_i = (y_i, z_i)$ et

$$p(x_1, \ldots, x_n|\theta) = \frac{1}{(\sqrt{2\pi})^n} \exp\left[-\frac{1}{2} \sum_{i=1}^{n} (y_i - \langle \theta, z_i \rangle)^2 \right].$$

Maximiser la log-vraisemblance est donc équivalent à minimiser le critère des moindres carrés :

$$C(x, \theta) := -\sum_{i=1}^{n} (y_i - \langle \theta, x_i \rangle)^2.$$

Bien entendu, l'estimateur des moindres carrés est utilisé même quand on ne suppose pas le bruit gaussien ! Cette deuxième relaxation de l'approche bayésienne est aussi devenue très populaire en apprentissage. En statistique bayésienne, elle a été plus controversée. Cependant, l'article [3] a établi qu'elle restait compatible avec la théorie de la décision lorsque la fonction de coût est additive par rapport aux observations dans l'échantillon : $C(x, \theta) = \sum_{i=1}^{n} c(x_i, \theta)$. Dans un article co-écrit avec Nicolas Chopin et James Ridgway, nous avons proposé une étude théorique de ces deux premières relaxations [2] : nous montrons que, même si la solution q du problème d'optimisation dans la relation (3.2) n'est pas forcément une bonne approximation de la loi *a posteriori*, elle fournit quand même une méthode d'estimation et de prédiction avec une erreur optimale au sens minimax. De façon générale, la théorie dite PAC-Bayésienne permet l'analyse statistique de (3.2), voir des notes de cours récemment publiées [1].

Une troisième relaxation est aujourd'hui considérée : il s'agit d'autoriser un changement de pénalité. En pratique, certains auteurs ont remplacé la divergence de Kullback-Leibler $\mathrm{KL}(q, \pi)$ par d'autres divergences comme la divergence de Rényi ou des f-divergences. On obtient donc une version encore plus générale de notre problème d'optimisation :

$$\max_{q \in \mathcal{Q}} \left\{ \int C(x, \theta) q(\mathrm{d}\theta) - \mathrm{D}(q, \pi) \right\} \tag{3.3}$$

où D est une divergence à spécifier. Plusieurs motivations ont été proposées : l'optimisation serait plus facile, les solutions plus stables, etc. Il s'agit d'un axe de recherche prometteur, mais cette relaxation ne fait pas encore consensus. Je renvoie cependant le lecteur à l'article [7], qui est un vrai manifeste du point de vue *Bayes as optimization*. Il contient en particulier des arguments intéressants en faveur de cette troisième relaxation.

4 Conclusion

Au sens strict du terme, chacune des trois relaxations considérées produit une loi q sur θ qui ne peut plus être interprétée comme un conditionnement, comme dans (1.1). Donc, même si elles sont apparues comme des tentatives pour approcher la loi *a posteriori*, *stricto sensu*, elles doivent être considérées comme non bayésiennes. Il y a d'ailleurs un débat sur comment désigner les algorithmes issus de la relation (3.3) : quasi-bayésiens ? bayésiens généralisés ? Le fait est que cette relation 1) contient l'inférence bayésienne comme cas particulier 2) a été inspiré de l'approche bayésienne et 3) permet de résoudre des problèmes non-accessibles à une approche strictement bayésienne, comme entraîner des LLM. La plupart des chercheurs en apprentissage semblent considérer qu'explorer les potentialités du programme dans la relation (3.3), en théorie et en pratique, est plus important que savoir dans quelle case le ranger.

Appendice : lien formel avec la thermodynamique et dérivation informelle de la relation (2.1)

En thermodynamique statistique, les lois de Gibbs lient la probabilité d'un état u à son énergie E_u :

$$p_u = \frac{\exp(-\beta E_u)}{Z}$$

où Z est la constante de normalisation $Z = \sum_u \exp(-\beta E_u)$ et β un paramètre physique (lié à la constante de Boltzmann). Notons au passage que ceci avait déjà conduit Geman et Geman [5] à la définition de vraisemblance sous la forme :

$$p(x|\theta) = \frac{\exp(-\beta E_\theta(x))}{Z_\theta}$$

et à un algorithme MCMC permettant de simuler suivant la loi *a posteriori* qui en résulte, que Geman et Geman ont nommé... algorithme de Gibbs.

Il s'avère que les lois de Gibbs étaient connues en physiques comme solutions de problèmes d'optimisation : elles maximisent l'entropie sous des contraintes de moments. C'est ce qui a conduit à trouver la relation (2.1). Il en est fourni ici une dérivation (qui n'est pas très loin d'une preuve rigoureuse si on rajoute les conditions d'intégrabilité nécessaires). Je rappelle d'abord que, quel que soient q et q', $\mathrm{KL}(q, q') \geq 0$, avec égalité si et seulement si $q = q'$ (ce résultat se démontre aisément avec l'inégalité de Jensen). On a donc :

$$\begin{aligned}
0 &\leq \mathrm{KL}\left(q, \pi(\cdot|x)\right) \\
&= \mathrm{KL}\left(q, \frac{p(x|\theta)\pi(\mathrm{d}\theta)}{\int_\vartheta p(x|\vartheta)\pi(\mathrm{d}\vartheta)}\right) \\
&= \int \log\left(\frac{\int_\vartheta p(x|\vartheta)\pi(\mathrm{d}\vartheta)}{p(x|\theta)}\frac{\mathrm{d}q}{\mathrm{d}\pi}(\theta)\right) q(\mathrm{d}\theta) \\
&= \log \int_\vartheta p(x|\vartheta)\pi(\mathrm{d}\vartheta) - \int \log p(x|\theta) q(\mathrm{d}\theta) + \int \log\left(\frac{\mathrm{d}q}{\mathrm{d}\pi}(\theta)\right) q(\mathrm{d}\theta) \\
&= \log \int_\vartheta p(x|\vartheta)\pi(\mathrm{d}\vartheta) - \int \log p(x|\theta) q(\mathrm{d}\theta) + \mathrm{KL}(q, \pi).
\end{aligned}$$

D'après la première ligne, la quantité à droite est minimisée exactement pour $q = \pi(\cdot|x)$, ce qui prouve (2.1).

Références

[1] P. Alquier. User-friendly introduction to PAC-Bayes bounds. *Foundations and Trends® in Machine Learning*, 17(2) :174–303, 2024.

[2] P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*, 17(236) :1–41, 2016.

[3] P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B : Statistical Methodology*, 78(5) :1103–1130, 2016.

[4] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference : A review for statisticians. *Journal of the American statistical Association*, 112(518) :859–877, 2017.

[5] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6) :721–741, 1984.

[6] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. In *Learning in graphical models*, pages 105–161. Springer, 1998.

[7] J. Knoblauch, J. Jewson, and T. Damoulas. An optimization-centric view on Bayes' rule : Reviewing and generalizing variational inference. *Journal of Machine Learning Research*, 23(132) :1–109, 2022.

[8] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[9] C. P. Robert and G. Casella. *Monte Carlo statistical methods (2nd edition)*. Springer, 2024.

[10] Y. Shen, N. Daheim, B. Cong, P. Nickl, G. M. Marconi, B. C. E. M. Raoul, R. Yokota, I. Gurevych, D. Cremers, M. E. Khan, and T. Möllenhoff. Variational learning is effective for large deep networks. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 44665–44686. PMLR, 21–27 Jul 2024.