Chapter 5, Part 3: Bayesian Inference

Advanced Topics in Statistical Machine Learning

Tom Rainforth

Hilary 2024

rainforth@stats.ox.ac.uk

Bayesian Inference

- Given a model, how can we actually characterize the posterior $p(\theta|\mathcal{D})$?
- This turns out to be surprisingly difficult and requires us to use methods for **Bayesian inference**
- Covering this topic properly is unfortunately beyond the scope of the course, but we will go through some key ideas that are necessary for putting Bayesian modeling into context

Bayesian Inference is Hard!

- It might at first seem like Bayesian inference is a straightforward problem
  - By Bayes' rule we have that $p(\theta|\mathcal{D}) \propto p(\mathcal{D}|\theta)p(\theta)$ and so we already know the relative probability of any one value of θ compared to another.
- In practice, this could hardly be further from the truth
  - In general it is an **NP–hard problem**
  - It is akin to calculating a **high–dimensional integral**

The Normalization Constant

If $p(\mathcal{D})$ is unknown, we lack scaling when evaluating a point

- We have no concept of how **relatively** significant that point is compared to the distribution as a whole
- We don't know how much mass is **missing**
- The larger the space of θ, the more difficult this becomes

Image Credit: www.theescapeartist.me

Example

Consider a model where $\theta \in \{1, 2, 3\}$ with a corresponding uniform prior $P(\theta) = 1/3$ for each θ.

Now presume that for some reason we are only able to evaluate the likelihood at $\theta = 1$ and $\theta = 2$, giving $p(\mathcal{D}|\theta = 1) = 0.001$ and $p(\mathcal{D}|\theta = 2) = 0.01$ respectively.

Depending on the marginal likelihood $p(\mathcal{D})$, the posterior probability of $P(\theta = 2|\mathcal{D})$ will vary wildly:

- $p(\mathcal{D}) = 0.004$ gives $P(\theta = 2|\mathcal{D}) = 5/6$
- $p(\mathcal{D}) = 1/3$ gives $P(\theta = 2|\mathcal{D}) = 1/100$

Characterizing the Posterior

- Knowing $p(\mathcal{D})$ is **not** sufficient (or necessary!) for estimating expectations with respect to the posterior such as the posterior predictive distribution
    - Most inference methods will actually sidestep the calculation of $p(\mathcal{D})$ (this is generally harder than the inference itself)
- At its heart, the problem of Bayesian inference is a problem of where to concentrate our finite computational resources so that we can effectively characterize the posterior; being able to evaluate it piecewise is not always enough for this

General Inference Strategies

Most strategies for Bayesian inference fall into one of three categories:

- **Heuristic** approximations (point estimates, Laplace approximation)
- **Sample** based approximations (importance sampling, rejection sampling, MCMC, sequential Monte Carlo, Hamiltonian Monte Carlo)
- **Surrogate** based approximations (variational inference, message passing, normalizing flows)

Maximum a Posteriori (MAP) Parameters

The **maximum a Posteriori (MAP)** parameters in a Bayesian model are the mode of the posterior:

$$\tilde{\theta}_{\mathsf{MAP}} = \arg\max_{\theta \in \vartheta} p(\theta|\mathcal{D}) = \arg\max_{\theta \in \vartheta} p(\mathcal{D}|\theta)p(\theta). \tag{1}$$

This is sometimes used as a point estimate to make predictions cheaply by taking $p(\mathcal{D}^*|\mathcal{D}) \approx p(\mathcal{D}^*|\tilde{\theta}_{\mathsf{MAP}})$.

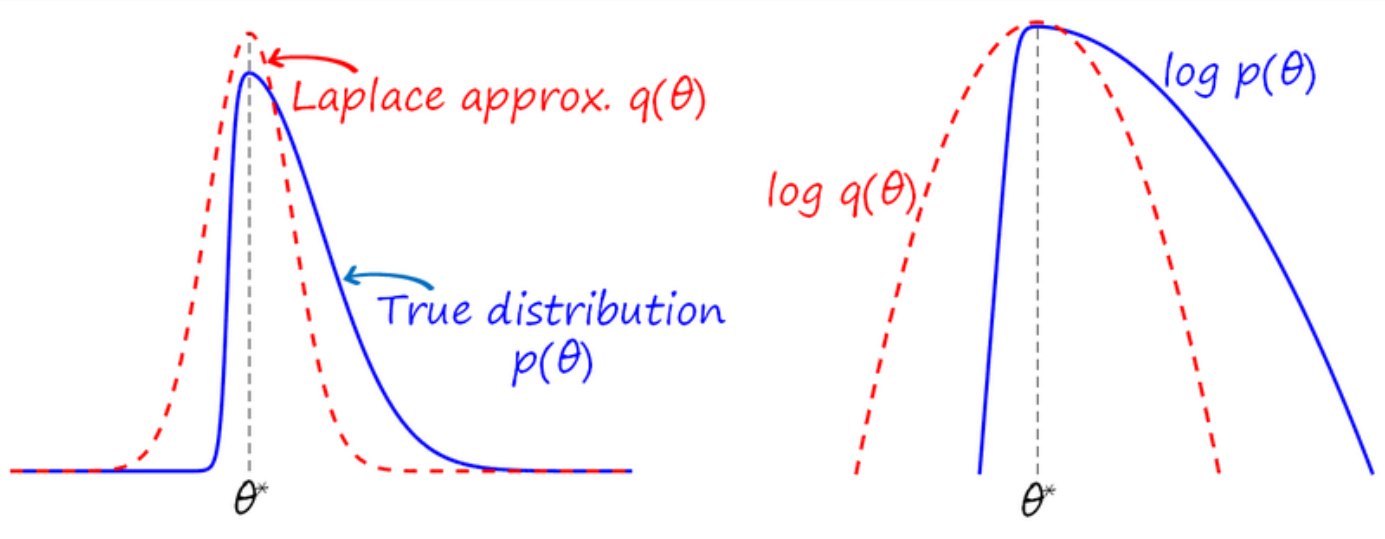
Though this if far cheaper than full inference, it has some significant drawbacks:

- It incorporates less information into the predictive distribution and can be a very crude approximation
- The position of the MAP estimate is dependent of the parametrization of the problem

Laplace Approximation

Based on a Taylor expansion and matching the local curvature (see notes), the **Laplace approximation** is a local **Gaussian** approximation around the MAP that takes the inverse of negative Hessian of the log joint density as the covariance function:

$$p(\theta|\mathcal{D}) \approx \mathcal{N}\left(\theta; \tilde{\theta}_{\text{MAP}}, \left(-\nabla^2_\theta \log\left(p(\theta, \mathcal{D})\right)|_{\theta=\tilde{\theta}_{\text{MAP}}}\right)^{-1}\right) \tag{2}$$



Images Credit: Luis Herranz

Monte Carlo Estimators

If we can draw **samples** from the posterior, we can form Monte Carlo estimates for any expectation we might wish to calculate:

$$\mathbb{E}_{p(\theta|\mathcal{D})}[f(\theta)] \approx \frac{1}{N}\sum_{n=1}^{N} f(\hat{\theta}_n) \quad \text{where} \quad \hat{\theta}_n \sim p(\theta|\mathcal{D}) \tag{3}$$

This produces an estimator whose mean squared error is $O(1/N)$

We cannot usually draw exact samples from the posterior, but instead construct methods which produce **approximate** samples.

Two main ways for doing this:

- Produce weighted samples that become equivalent to samples from $p(\theta|\mathcal{D})$ in expectation (importance sampling based)
- Constructing a Markov chain of samples whose distribution gets increasingly close to $p(\theta|\mathcal{D})$ (MCMC based)

Importance Sampling

- **Importance sampling** is a common sampling method that is also the cornerstone for many more advanced inference schemes
- It uses a proposal $q(\theta)$ to draw samples before applying corrective **importance weights** to account for the fact that our samples are drawn from the wrong distribution
- These weights are given by $p(\theta|\mathcal{D})/q(\theta)$, which comes from the fact that

$$\mathbb{E}_{p(\theta|\mathcal{D})}[f(\theta)] = \mathbb{E}_{q(\theta)}\left[\frac{p(\theta|\mathcal{D})}{q(\theta)}f(\theta)\right] = \mathbb{E}_{q(\theta)}\left[\tilde{w}(\theta)f(\theta)\right]$$

  where $\tilde{w}(\theta) = p(\theta|\mathcal{D})/q(\theta)$
- In practice, we cannot evaluate these weights exactly, so we instead use $w(\theta) = p(\theta, \mathcal{D})/q(\theta)$ followed by self-normalizing our weights

Self–Normalized Importance Sampling Algorithm

1. Draw N i.i.d. samples $\hat{\theta}_n \sim q(\theta) \quad n = 1, \ldots, N$
2. Assign weight $w_n = p(\hat{\theta}_n, \mathcal{D})/q(\hat{\theta}_n)$ to each sample
3. Self normalize the weights: $\bar{w}_n = w_n/(\sum_{m=1}^{N} w_m)$
4. Combine the samples to form the empirical measure

$$p(\theta|\mathcal{D}) \approx \sum_{n=1}^{N} \bar{w}_n \delta_{\hat{\theta}_n}(\theta) \tag{4}$$

5. This can used to be estimate $\mathbb{E}_{p(\theta|\mathcal{D})}[f(\theta)]$ for any f using

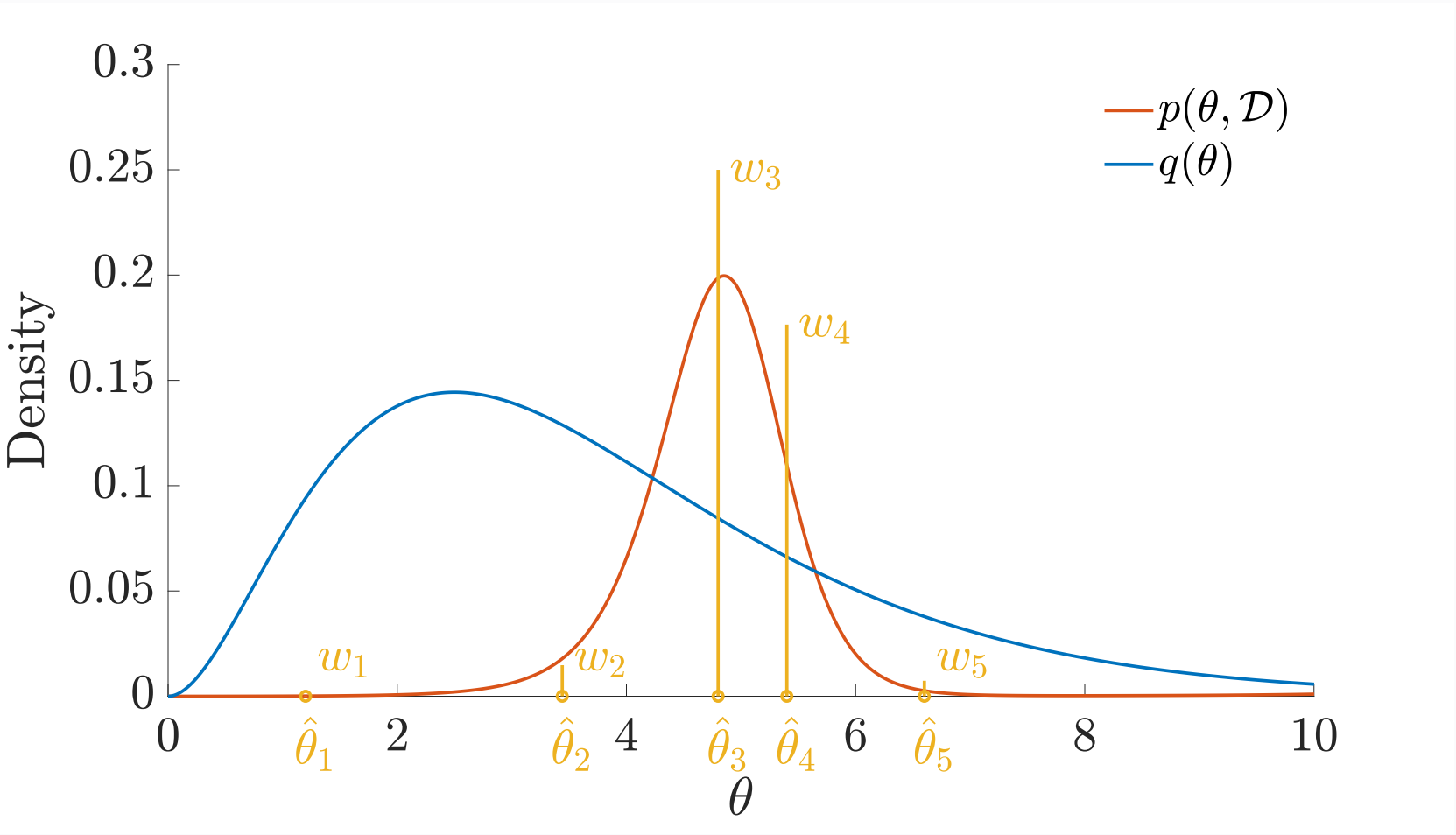
$$\mathbb{E}_{p(\theta|\mathcal{D})}\left[f(\theta)\right] \approx \sum_{n=1}^{N} \bar{w}_n f(\hat{\theta}_n) \tag{5}$$

Note that the average of the unnormalized weights is an unbiased estimator of the marginal likelihood: $\mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N} w_n\right] = p(\mathcal{D})$

Why Self-Normalization?

$$
\begin{aligned}
\mathbb{E}_{p(\theta|\mathcal{D})}[f(\theta)] &= \mathbb{E}_{q(\theta)}\left[\frac{p(\theta|\mathcal{D})}{q(\theta)} f(\theta)\right] \\
&= \frac{1}{p(\mathcal{D})}\,\mathbb{E}_{q(\theta)}\left[\frac{p(\theta,\mathcal{D})}{q(\theta)} f(\theta)\right] \\
&= \mathbb{E}_{q(\theta)}\left[\frac{p(\theta,\mathcal{D})}{q(\theta)} f(\theta)\right] \Big/ \mathbb{E}_{p(\theta)}\left[p(\mathcal{D}|\theta)\right] \\
&= \mathbb{E}_{q(\theta)}\left[\frac{p(\theta,\mathcal{D})}{q(\theta)} f(\theta)\right] \Big/ \mathbb{E}_{q(\theta)}\left[\frac{p(\theta,\mathcal{D})}{q(\theta)}\right] \\
&= \frac{\mathbb{E}\left[w_1 f(\theta_1)\right]}{\mathbb{E}[w_1]}
\end{aligned}
$$

Importance Sampling



$$w(\theta) = p(\theta, \mathcal{D})/q(\theta)$$

Surrogate Based Approximations

- Surrogate approaches directly learn an **approximate distribution** $q(\theta) \approx p(\theta|\mathcal{D})$ that we use as a replacement once learned (e.g. drawing approximate samples $\hat{\theta} \sim q(\theta)$)
- For example, we can introduce a parameterized approximation $q(\theta; \phi)$ and then minimize some divergence $\mathbb{D}$ between the approximation and the posterior

$$\phi^* = \arg\min_{\phi} \mathbb{D}(q(\theta; \phi)||p(\theta|\mathcal{D}))$$

- This allows us to convert the inference problem into an **optimization**
  - For certain choices of divergence, this optimization only requires evaluations of the joint $p(\theta, \mathcal{D})$
- The most common such approach is **variational inference** which uses $\mathrm{KL}(q(\theta; \phi)||p(\theta|\mathcal{D}))$; we will return to it later

Recap

- Bayesian inference is hard; it is often the main bottleneck in using Bayesian approaches
- Even if we can directly evaluate the posterior (which is rare), this may not be enough to characterize it and estimate expectations
- Monte Carlo methods give us a mechanism of representing distributions through samples
- We can alternatively try to approximate the posterior with a surrogate

Further Reading

The following are more for those interested in reading around on the subject than material that will actually be helpful for the course itself

- Chapters 6 and 7 of the notes for a previous course I taught on Bayesian Machine Learning: `https://www.cs.ox.ac.uk/files/11549/main.pdf`
- Chapters 1, 2, 7, and 9 of Art Owen's online book on Monte Carlo: `https://statweb.stanford.edu/~owen/mc/`
- David MacKay on Monte Carlo methods `http://videolectures.net/mackay_course_12/`