EPSE 581C: Bayesian Methods

Ed Kroc

University of British Columbia

ed.kroc@ubc.ca

November 4, 2019

Computational issues

All of Bayesian inference is based on the *posterior distribution*, but for complex, real-world situations, the posterior is often analytically untractable. Several potential problems:

- How to calculate the *normalizing factor* in the denominator of Bayes' theorem?
- How to calculate the *expectation* (e.g. mean, standard deviation) of a complicated posterior density?
- How to calculate the *mode* of anything?
- How to do all this flexibly; i.e. without having to restrict ourselves to very special classes of data and priors (e.g. conjugate priors)?

Computational issues

There are *many* techniques for addressing these issues; we will only briefly survey some of the most common/important:

- Numerical integration (deterministic)
- EM algorithm (deterministic)
- Monte Carlo sampling (stochastic)
- Markov chain Monte Carlo methods (stochastic)
  - Metropolis-Hastings algorithm
  - Gibbs sampler

Numerical integration

Numerical integration techniques actually pre-date integration itself!
I.e. how do we approximate an area under a curve?

- (finite) Riemann sum
- Quadratic approximation
- Laplace approximation
- etc.

All based on the idea that you can approximate an area of a complicated region by splitting it up into a bunch of simple regions and then summing up all their individual areas.

The EM algorithm

- In statistics, one of the most commonly employed methods to calculate (approximate) the *maximum* (or minimum) of a function is the *expectation-maximization (EM) algorithm*
- The algorithm iterates these two steps to find the (local) maximum (minimum) of a probability density function:

(1) Calculate the *expectation* of the posterior predictive distribution, call it

$$z_1 = \mathbb{E}(\widetilde{y} \mid \boldsymbol{y}) = \int \widetilde{y} \cdot f(\widetilde{y} \mid \boldsymbol{y}) \, d\widetilde{y}$$

(2) Augment the dataset to $\{\boldsymbol{y}, z_1\}$ and then find the new posterior density for this augmented dataset. This new posterior has a corresponding posterior predictive density, so we can return to step (1) and define z_2 as before. Iterating this procedure, z_n will converge to the *maximum* (mode) of the original posterior density.

Monte Carlo sampling

- Rather than the deterministic methods already discussed, there is a whole class of *stochastic* (i.e. random sampling based) methods for calculation/approximation.
- Most of these techniques fall under the general term of *Monte Carlo sampling*, loosely defined as any technique where you use a theoretical probability sample to approximate a target quantity.
- For example, if we want to calculate the *posterior mean*, then we can simply generate a bunch of random draws from the posterior distribution and compute their sample mean. Do this for a large enough sample size, and the approximation will be very good. (Why?)
- Could approximate *posterior standard deviation* the same way, using the sample standard deviation of a bunch of random draws from the posterior.

Markov chain Monte Carlo (MCMC) methods

- Markov chain Monte Carlo (MCMC) methods simply extend this basic idea by relaxing the requirement that each draw comes from the same distribution.
- For example, it is often very difficult to work directly with the posterior distribution! So how could we "sample" from it?
- Instead, we generate a random sample from a *sequence* of simpler probability distributions, chosen so that in the long run, these distributions converge to the target (posterior) distribution. This sequence of distributions forms a *Markov chain*.

Markov chain Monte Carlo (MCMC) methods

You could easily spend an entire term talking about these mathematical ideas (see MATH 303, 545), but just to fix terminology:

- A *Markov chain* is a random sequence (stochastic process), $X_1, X_2, \ldots$, that obeys the *Markov property*:

$$\Pr(X_{t+1} \mid X_t, \ldots, X_2, X_1) = \Pr(X_{t+1} \mid X_t), \text{ for all time points } t.$$

- Put another way, given the present value of the Markov chain, its future value is independent of the past.

Markov chain Monte Carlo (MCMC) methods

- MCMC methods all setup a sequence of random variables $X_1, X_2, \ldots$ obeying the Markov property so that this random sequence converges to what we want (e.g. the posterior distribution).
- Thus, we can "sample from the posterior" by sampling sequentially from the Markov chain.
- Eventually, our samples will be close enough to the target distribution, so we can approximate whatever we want by the usual sample analogues (i.e. we can use Monte Carlo approximation).'
- Important question: *How long is long enough?*
- Important note: Regardless of that answer, notice that we would *never* want to use the early sample draws from an MCMC method, as there is not enough time for the Markov chain to converge. This generates what is called a *burn-in* period/window.

Metropolis-Hastings algorithm

- The Metropolis-Hastings algorithm is one of the most common MCMC methods around.
- Let $\pi(x)$ be our target density (e.g. joint posterior density), and let x_0 be an arbitrary starting value. Iteratively, proceed as follows:
  - Simulate a candidate value y from the distribution given by $\Pr(X_n \mid X_{n-1} = x_{n-1})$.
  - Define the *acceptance ratio*

$$\alpha(y \mid x_{n-1}) = \min\left\{\frac{\pi(y)\Pr(X_n = x_{n-1} \mid X_{n-1} = y)}{\pi(x_{n-1})\Pr(X_n = y \mid X_{n-1} = x_{n-1})},\ 1\right\}$$

  - Simulate $u \sim \mathit{Unif}(0,1)$. If $u \leqslant \alpha(y \mid x_{n-1})$, then the next state of the Markov chain is equal to y (i.e. we set $X_n = y$); otherwise, the Markov chain stays in the same state (i.e. we set $X_n = x_{n-1}$).

Metropolis-Hastings algorithm

- The MH algorithm is extremely flexible and can be used to approximately simulate data from all kinds of complicated distribution using only simple probability distributions.
- Notice a key feature of MH: although the acceptance ratio α depends on the target (posterior) density), it does *not* depend on the *normalizing constant*, because it is a ratio of the target density at two different values. Greatly simplifies computation!
  - Use MH to simulate normal data using a uniform distribution: `https://bookdown.org/rdpeng/advstatcomp/metropolis-hastings.html`
  - Use MH to simulate exponential data using a normal distribution: `https://stephens999.github.io/fiveMinuteStats/MH-examples1.html`

Gibbs sampler

Gibbs sampling is a particularly effecient kind of MH procedure, especially for high dimensional problems:

- Given a target density $\pi(y_1, \ldots, y_p)$, define the *full conditional* densities

$$\pi(y_i \mid y_{-i}) = \pi(y_i \mid y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_p).$$

- Our Markov chain will now sample from each of these full conditional densities cyclically at each time step.
- Also, rather than worrying about the accept/reject stage, we will always accept and move onto the next full conditional density.

Can be shown to very computationally efficient. Lots of ways to make it even better too.

Weak Law of Large Numbers

But why the heck do all these stochastic approximations even work?

Weak Law of Large Numbers

Let $X_1, X_2, \ldots$ be a sequence of independent and identically distributed (i.i.d.) random variables, each having finite mean $\mathbb{E}(X_i) = \mu$ and finite variance $\text{Var}(X_i) = \sigma^2$. Then for any $\varepsilon > 0$,

$$\lim_{n \to \infty} \Pr\left(\left| \frac{X_1 + \cdots + X_n}{n} - \mu \right| > \varepsilon \right) = 0.$$

We say that the sample mean of the X_i's *converge in probability*, or *converge in measure*, to the population (true) mean of X_i. That is, the sample averages, $\overline{X}_n$, converge in probability to the expectation.

In statistics, we say that $\overline{X}_n$ is a *consistent estimator* of μ.

Weak Law of Large Numbers

- This law holds in greater generality. In particular, it still holds even if $\text{Var}(X_i) = \sigma_i^2$ are all different, as long as $\sigma_i^2 \leqslant C$ for all i and some C finite.
- Note that the WLLN does *not* apply to random variables with infinite mean or variance, like Cauchy random variables.
- The WLLN law justifies the intuition that after a large number of i.i.d experiments, the sample average of the r.v. should be close, to the true expectation.
- The WLLN (or its generalization for Markov chains) guarantees that our sample estimates will converge to the target value, given enough time.

Visualizing the WLLN

http://digitalfirst.bfwpub.com/stats_applet/stats_applet_10_prob.html

Weak Law of Large Numbers

- The WLLN does *not* say that $|\overline{X}_n - \mu| > \varepsilon$ cannot happen upon further experimentation. In fact, the WLLN leaves open the possibility that we observe such a discrepancy infinitely often upon further experimentation.
- Thankfully, there is a Strong Law of Large Numbers that says this (usually) cannot happen. The SLLN says that for any $\varepsilon > 0$ and for all n large enough, $\Pr(|\overline{X}_n - \mu| > \varepsilon) = 0$.
- For Markov chains, this is the difference between *weak* and *strong mixing*. Lots of fascinating and important math here, and the real reason there are so many different approximation techniques that have been developed in practice. Not all techniques work equally well all of the time.

Weak vs. Strong Law of Large Numbers

Weak Law of Large Numbers

Let $X_1, X_2, \ldots$ be a sequence of independent and identically distributed (i.i.d.) random variables, each having finite mean $\mathbb{E}(X_i) = \mu$ and finite variance $\text{Var}(X_i) = \sigma^2$. Let $\bar{X}_n$ denote the sample average of $X_1, \ldots, X_n$. Then for any $\varepsilon > 0$,

$$\lim_{n \to \infty} \Pr\left(\left|\bar{X}_n - \mu\right| > \varepsilon\right) = 0.$$

Strong Law of Large Numbers

Under the same conditions as the WLLN, for any $\varepsilon > 0$,

$$\Pr\left(\lim_{n \to \infty} \left|\bar{X}_n - \mu\right| > \varepsilon\right) = 0.$$

We say that $\bar{X}$ converges *almost surely* or *almost everywhere* to μ.

Weak vs. Strong Law of Large Numbers

- There are certain random variables for which the WLLN holds, but the SLLN fails. This is rare, but explains the terminology "weak" vs. "strong".
- The proof of the SLLN requires notions from measure theory.
- For most well-behaved random variables, an i.i.d. sample will obey both the WLLN and the SLLN.
- Note: the Central Limit Theorem is often used to create measures of uncertainty about our sample estimates (e.g. confidence intervals), but it is the Law of Large Numbers that assures us that our sample estimates are "good" to begin with. In general, we have the following logical implications:

$$\text{CLT} \longrightarrow \text{SLLN} \longrightarrow \text{WLLN}$$

Return to model building and validation

- Cross-validation techniques
- Information criteria for model selection

Akaike (An) Information Criterion (AIC)

- Every statistical model has an associated AIC.
- Let k be the number of model parameters and let $\widehat{L}$ be the maximum value of the likelihood function (given those parameters); then define the AIC of the model as follows:

$$AIC = 2k - 2\ln(\widehat{L}).$$

- Among a set of candidate models, the one with the *lowest* AIC is the "best", according to Akaike's information criterion.
- Informally, AIC penalizes a model for overfit (too many parameters), and rewards a model for optimizing the likelihood.

Quantifying information content

- Formally, AIC is a *relative comparison* of *information loss*.
- Define the *Kullback-Leibler divergence* (*relative entropy*) between two probability density functions f and g as:

$$KL(f, g) = -\int_{-\infty}^{\infty} f(x) \cdot \log\left(\frac{g(x)}{f(x)}\right) \, dx,$$

  and analogously for probability mass functions p and q as

$$KL(p, q) = -\sum_{x} p(x) \cdot \log\left(\frac{q(x)}{p(x)}\right)$$

- This is the amount of information lost when we use g (or q) to estimate f (or p).
- Notice: if $p = q$, then $KL(p, q) = 0$.

Quantifying information content

- Formally, AIC is a *relative comparison* of *information loss*.
- Suppose the data come from some *true* (unknown) data-generating process given by the function f.
- We could then calculate the information lost when estimating this reality by some other model, with associated likelihood g.
- Of course, we never know f, but Akaike showed that the AIC is a good estimate (under some conditions) of the information lost when using g instead of f.
- Less information loss is good, so we want usually want models that have the smallest AIC (could be a large negative number!)

Akaike (An) Information Criterion (AIC)

- The formal AIC justification is only valid asymptotically (as sample size grows without bound).
- For small sample sizes, one usually uses the modified AIC instead:

$$AICc = AIC + \frac{2k^2 + 2k}{n - k - 1},$$

  where n is sample size and k is the number of model parameters.
- Using information theory, one can show that the AIC will settle on the "best" *predictive* model, under certain conditions.
- AIC (AICc) is always a *relative* measure of model fit; i.e. the exact values of the AIC are always *totally meaningless* on their own.

Bayesian Information Criterion (BIC)

- Every statistical model has an associated BIC (also called SIC for Schwarz).
- Let k be the number of model parameters, n be the sample size, and let $\widehat{L}$ be the maximum value of the likelihood function (given those parameters); then define the AIC of the model as follows:

$$BIC = \ln(n)k - 2\ln(\widehat{L}).$$

- Among a set of candidate models, the one with the *lowest* BIC is the "best", according to Bayesian information criterion.
- Informally, BIC penalizes a model for overfit (too many parameters) proportionally to a function of sample size, and rewards a model for optimizing the likelihood.

Bayesian Information Criterion (BIC)

- Similar information content interpretations/justifications exist for the BIC.
- Using these, one can show that the BIC will settle on the "best" explanatory model, under certain conditions.
- As with the AIC, BIC is always a *relative* measure of model fit.
- BIC makes sense *even if you are performing a frequentist analysis.* Why?
- Informally, one can show that the BIC assumes a uniform prior on all candidate models while the AIC assumes a different prior on the set of candidate models.

Deviance Information Criterion (DIC)

- The DIC is commonly used as a model selection index/techniques when models are being fit via MCMC algorithms.
- The exact formula for DIC is similar to AIC/BIC, but relies on the *effective number of parameters* of a model. This is a theoretical quantity that cannot be directly calculated, and it reflects the MCMC (hypothetical sampling) uncertainty in our model estimates.
- In a certain sense, the DIC generalizes the AIC to situations where we can only approximately estimate our model via MCMC techniques.
- As with all ICs, smaller numbers represent "better" fits, and the exact values of the DIC are meaningless except *relative* to each other for a particular analysis.

Cross-validation

- Cross-validation is yet another extensively used techniques for model selection and validation.
- Tools can be applied in either a frequentist of Bayesian context, but particularly common in the Bayesian framework.
- The simplest kind of cross-validation is *leave-one-out cross validation*

Leave-one-out cross-validation

Proceed as follows:

(1) Uniformly at random, exclude *one* data point from your sample of size n. This is the *validation* set.

(2) Fit your proposed model on the remaining $n - 1$ data points. This is the *training* set.

(3) Quantify how well the fitted model "predicts" the originally excluded data point (perhaps by computing its posterior predictive probability)

(4) Repeat steps (1)–(3) many times, arriving at measure of average predictive fit (average posterior predictive probability).

Can compare different models according to this "cross-validation error" then, finding which one does the best job at predicting the artificially missing data.

Cross-validation

- Obviously not feasible to do cross-validation by-hand.
- Many ways to generalize or alter the cross-validation approach.
  - Could put $p > 1$ points in the validation set.
  - Could exclude, say, 10% of your data points for validation.
- Asymptotically, cross-validation will settle on the "best" predictive model; i.e. cross-validation will settle on the same model as the AIC, with large enough sample sizes.

Model selection/building/validation

- If you have *large* datasets (say, $n > 1,000$) with many potential parametes (say, $k > 50$), then the above tools can be very useful.
- If you are in "small data" situations, then these tools can still be used, but they are very likely to be *not* as useful as simply examining your model residuals.
- The above tools have rich and beautiful mathematical theories that motivate and justify their use; in practice, however, people tend to just use the tools as a way to not have to think critically about which model *they should actually choose*.