6/12/19

W1a-1

# Sampling - I

Announcements:

- Christian gives his first lecture today on "Why Bayes is Better!", which will be very cool. [We frontloaded Daniel for this week because he is off-shell the rest of the week (but back for the last two).]

look at visualizations & posterior

- Social dinner tonight at Bella Italia at 7pm.

show questions in notebook

- We will return to recap the signal and background notebook on Thursday morning. Before then, if you can jot down (it doesn't have to be in the notebook) answers for the questions before the four cases at the end, we can compare notes and decide on conclusions. Please ask questions before then!

- Mini-project I: Toy model for EFT parameter estimation. This is a less-guided task than the ones we've done so far, which will have you put together ideas and tools we've discussed.
  - You'll work on it Thursday and Friday afternoon in the exercise session and we'll do a recap early next week. Nothing to hand in but we'll be happy to review what you come up with.
  - Overall idea is to reproduce some results in a paper co-authored by Daniel and me (and others): "Bayesian parameter estimation for effective field theories," arXiv: 1511.03618. It's a very long paper, so don't try to read all of it! We'll guide you to the relevant parts.
  - The paper uses toy models for effective field theories, namely Taylor series of some specified functions, to present guidelines for parameter estimation. This will also be a check of whether you can follow Bayesian statistics discussions in the literature (or give them practice)

display paper

6/12/19

- You'll find summaries in section II that touch on topics we have discussed and will discuss.
- Function: $g(x) = \left(\frac{1}{2} + \tan\left(\frac{\pi}{2}x\right)\right)^2$ represents the true theory (cf. BCD) $= 0.25 + 1.57x + 2.47x^2 + 1.29x^3 + \ldots$
  Our model for an EFT is
  $$g_{th}(x) \equiv \sum_{i=0}^{k} a_i x^i$$
  and we want to fit $i = 0, 1, 2, 3, \ldots$ of the constants $a_i$.
- The primary goal is to reproduce and interpret Table III (on page 12 of the arXiv preprint).
- We'll have other tasks (eg., reproduce Figure 1), which you can do as time permits. We'll give hints and guidance.
- * We're not expecting to do anything before tomorrow afternoon – this is just a heads-up.
- Learning goals
  - Apply and extend the Bayesian parameter estimation from the course
  - Explore the impact of central features:
    - how much data and how precise it is
    - applying an informative prior
  - Learn about some diagnostics for Bayesian parameter estimation.
  - Try out sampling on a controlled problem

6/12/19

W1a-3

# Why MCMC? (Based on Gregory, Chp 12)

We have been emphasizing that in the Bayesian approach, everything is a pdf. One type of pdf is for the parameters of theory, which we'll denote by the vector $\vec{\theta}$, given data $D$:

$$p(\vec{\theta} \mid D, I)$$

and information $I$. Suppose we have a theoretical model for this.

- Maybe these are the LECs for an effective field theory Hamiltonian.

And now we want to calculate the expectation value of a function of $\vec{\theta}$: $\langle f(\vec{\theta}) \rangle$. Or $\vec{\theta}$ characterizes a signal and background.

- As we discussed in doing the central limit theorem:

$$\langle f(\vec{\theta}) \rangle = \int f(\vec{\theta})\, p(\vec{\theta} \mid D, I)\, d\vec{\theta} \equiv \int g(\vec{\theta})\, d\vec{\theta}$$

Note that this is more than traditional calculations, in which we would have single values of $\vec{\theta}$, e.g. denoted $\hat{\vec{\theta}}$, that we might have found by minimizing a $\chi^2$. E.g. we identified the particular values of $\theta_1, \theta_2, \ldots \theta_n$ that best reproduced scattering data. Then we would calculate $f(\hat{\vec{\theta}})$, which might be the binding energy of a nucleus.

- But $\langle f(\vec{\theta}) \rangle$ means we do a multidimensional integral over the full range of possible $\vec{\theta}$ values, weighted by the probability density function $p(\vec{\theta} \mid D, I)$, which we have worked out.
- This is a lot more work!
- We frequently also have a situation where we want to integrate (marginalize) over a subset of parameters $\vec{\theta}_B$ to find a probability for the rest $\vec{\theta}_A$. E.g. over parameters for the width of a signal and other parameters characterizing our model for the Higgs mass.
- These multidimensional integrals then become a necessity to do, but conventional methods for low dimension (e.g. Gaussian quadrature or Simpson's rule) become inadequate rapidly with the increase of dimension.

6/12/19

- The integrals are particularly problematic because the posteriors are very small in much of the integration volume, which will typically be a challenging shape.

- To approximate such integrals one turns to Monte Carlo methods. The straightforward (naive) MC integration evaluates the integral by randomly distributing $n$ points in the multidimensional volume $V$ of possible $\vec{\theta}$'s. $V$ has to be large enough to cover where $p(\vec{\theta}|D,I)$ is significantly different from zero.

- Then $\langle f(\vec{\theta})\rangle = \int_V g(\vec{\theta})\,d\vec{\theta} \doteq V \times \langle g(\vec{\theta})\rangle \pm V\sqrt{\frac{\langle g^2(\vec{\theta})\rangle - \langle g(\vec{\theta})\rangle^2}{n}}$

where $\langle g(\vec{\theta})\rangle = \frac{1}{n}\sum_{i=1}^{n} g(\vec{\theta}_i)$ ← $n$ not components of $\vec{\theta}$, but separate vectors $\langle g^2(\vec{\theta})\rangle = \frac{1}{n}\sum_{i=1}^{n} g^2(\vec{\theta}_i)$

[eg in one-D, the average of a function $\overline{g(\theta)} = \frac{1}{b-a}\int_a^b g(\theta)\,d\theta$ from calculus. But we can estimate $\overline{g(\theta)}$ by averaging over a set $n$ of random samples $\overline{g(\theta)} \doteq \frac{1}{n}\sum_{i=1}^{n} g(\theta_i)$
Then $\langle f(\theta)\rangle \doteq \int_a^b g(\theta)\,d\theta \doteq \frac{b-a}{n}\sum_{i=1}^{n} g(\theta_i)$. Here $b-a$ is the "volume" $V$.

Does this always work? No, we just saw a failure yesterday! But it usually works.]

- The uncertainty is assuming a Gaussian approximation is valid. Note the dependence on $\frac{1}{\sqrt{n}}$, which means you can get a more precise answer by increasing $n$. Slowly better, each additional decimal point accuracy costs you a factor of 100 in $n$!

- Key problem: too much time is wasted sampling regions where $p(\vec{\theta}|D,I)$ is very small. If for one parameter the fraction of significant is $10^{-1}$, in $M$-parameter problem the fraction of the volume is $10^{-M}$. This necessitates importance sampling which reweights the integrand to more appropriately distribute points (eg. VEGAS) but this is difficult to accomplish.

6/12/19

Bottom line: its not feasible to draw a series of independent random samples from $p(\vec{\theta}|D,I)$ for larger $\vec{\theta}$.
- remember, independent means if $\vec{\theta}_1, \vec{\theta}_2, \ldots$ is the series, knowing $\vec{\theta}_1$ doesn't tell us anything about $\vec{\theta}_2$

*** But the samples don't need to be independent, they just need to generate $p(\vec{\theta}|D,I)$ in the correct proportions (e.g. as indicated by histogramming the samples, it approximates $p(\vec{\theta}|D,I)$.

$\Rightarrow$ Do a random walk in the parameter space of $\vec{\theta}$ so that the probability for being in a region is proportional to $p(\vec{\theta}|D,I)$ for that region.
- $\vec{\theta}_{i+1}$ follows from $\vec{\theta}_i$ by a transition probability (kernel) $\Rightarrow p(\vec{\theta}_{i+1}|\vec{\theta}_i)$
- assumed to be "time independent", so same $p(\vec{\theta}_{i+1}|\vec{\theta}_i)$ no matter when you do it.

$\Rightarrow$ Markov chain and method is Markov chain Monte Carlo.

Basic structure of algorithm:

(1) Given $\vec{\theta}_i$, propose a value for $\vec{\theta}_{i+1}$, call it $\vec{\phi}$ ("candidate"), sampled from $q(\vec{\phi}|\vec{\theta}_i)$. This $q$ could take many forms, so for concreteness imagine it as a multivariate normal with mean given by $\vec{\theta}_i$ and variance $\vec{\sigma}^2$.
- decreased probability as you get away from current sample
- $\vec{\sigma}$ determines the step size.

(2) Decide whether or not to accept candidate $\vec{\phi}$ for $\vec{\theta}_{i+1}$. Here we'll use a Metropolis condition (later we'll see other ways that may be better).
- This dates from the 1950's in physics but didn't become widespread in statistics until almost 1980.
- Enabled Bayesian methods to take off.

6/12/19

Calculate Metropolis ratio!

$$r = \frac{p(\vec{\phi}|D,I)\, q(\vec{\theta}_i|\vec{\phi})}{p(\vec{\theta}_i|D,I)\, q(\vec{\phi}|\vec{\theta}_i)}$$

(proposed: $\vec{\phi}$; current: $\vec{\theta}_i$)

q may be symmetric $q(\vec{\theta}_1|\vec{\theta}_2) = q(\vec{\theta}_2|\vec{\theta}_1)$

if so ⇒ Metropolis

If not, Run Metropolis-Hastings.

Decision!

if $r \geq 1$, set $\vec{\theta}_{i+1} = \vec{\phi}$ accept

if $r < 1$, so less probable, don't always reject!
accept with probability $r$ (remember $0 \leq r \leq 1$)
by sampling a uniform $(0,1)$ distribution.
If $U \sim \text{Unif}(0,1)$ is $U \leq r$, then $\vec{\theta}_{i+1} = \vec{\phi}$
else $\vec{\theta}_{i+1} = \vec{\theta}_i$

Note that the last case means you do have a $\vec{\theta}_{i+1}$, but it is the same as $\vec{\theta}_i$ (so the chain continues to grow.

Acceptance probability is the minimum of $1, r$

Algorithm pseudo code:

1. initialize $\vec{\theta}_i$, set $i=0$
2. Repeat { Obtain new candidate $\vec{\phi}$ from $q(\vec{\phi}, \vec{\theta}_i)$
   Sample $U \sim \text{uniform}(0,1)$
   If $U \leq r$ set $\vec{\theta}_{i+1} = \vec{\phi}$, else set $\vec{\theta}_{i+1} = \vec{\theta}_i$
   i++
   }

Plan:
1. look at visualizations
2. look at a basic example for Poisson distribution
3. consider afternoon exercise and extra features
4. look at emcee example from intro

W1a-7

6/12/19

# Visualization of mcmc Sampling

- There are excellent javascript & mcmc sampling out there.
- A particularly effective set of interactive demos was created by Chi Feng, available at https://chi-feng.github.io/mcmc-demo/
- These demos range from random walk Metropolis Hastings to Adaptive MH to Hamiltonian Monte Carlo to No-U-Turn Sampler (NUTS) to Metropolis-adjusted Langevin Algorithm (MALA) to Hessian-tmc (H2MC), to Stein Variational Gradient Descent (SVGD) to Nested Sampling with RadFriends (RadFriends-NS).

- An accessible introduction to mcmc with simplified versions of Feng's visualization by Richard McElreath. Let's look at the first part of his blog entry at http://elevanth.org/blog/2017/11/28/build-a-better-markov-chain/
- Recall basic structure of Metropolis-Hastings
  1) make a random proposal for new parameter values
  2) accept or reject the proposal based on a Metropolis criterion

- First simulation is Random Walk Metropolis-Hasting
  - Target distribution is two-dimensional Gaussian (just the product)

* If the distribution correlated? How do you know?

- An arrow indicates a proposal, which is accepted (green) or rejected (red)
- notice that the direction and a length of the proposal arrow varies.
- seems to do ok on such a simple distribution, as indicated by how well the projected posteriors get filled in.
- but it is diffusing - a random walk - which is not so efficient.

A more complicated shape can cause problems:

- MH can spend a lot of time exploring over again same regions
- If not specially tuned, can reject many proposals (red arrows)

W1a-8

6/12/19

- Donut shape is much trickier:
  - Notice that the projected 1d posteriors don't seem to be so complex, but this is a difficult topology.
  - Is it realistic? The claim is that when there are many parameters (high dimensional space), this is analogous to a common target distribution.

- Problems: constantly looking for right step size that is big enough to explore the space, but small enough to not get rejected too much.
  - High dimensions is a big space! Hard to stay in a region of high probability while also exploring enough (in reasonable time).

- Note on donuts in high dimensions
  - see bayes_talk.028.png
  - look at average radius of points sampled from multivariate Gaussians as a function of the dimension
  - blue is 1d, green is 2d, ...; yellow is 6d.
  - imagine yellow as 6 dimensional shell ⇒ analog is two dimensional donut.

- Take a look at Feroz site
  - banana distribution — difficult
  - multimodal — very, very tough (see Christian's talks)
  - try adjusting proposal $\sigma$ (Gaussian proposal with sd = $\sigma$.)
    ⇒ try this on donut: to get green you need excellent step size tuning.

- Back to McElreath page. What is the answer? "Better living through physics."
  - This means Hamiltonian Monte Carlo → show examples
  - We'll come back to that next week and stick with MH this week.

W1a-9

6/12/19

# Metropolis_Poisson-example from Gregory, section 12.2

- See: mcmc-sampling/Metropolis_Poisson_example.ipynb
- We've already seen the Poisson distribution $p(k|\mu) = \frac{\mu^k e^{-\mu}}{k!}$ $k \geq 0$ integer and we've sampled it through a scipy.stats function. Here we'll do it via MCMC.
- Markov chain: starts with some initial value, then each successive one is generated from previous.
- Step through the procedure for Poisson. (do separately on board.)
  - Then step through the code.
- Look at the two graphs produced
  - mcmc trace: value at successive MC steps. Notice the fluctuations: it stays reasonably close to 3 but still can jump high.
  - Histogram shows how well we're doing.

  ⇒ use ctrl-enter to run many times.
  - Note the outliers at the beginning: needs to equilibrate. This is called the warm-up (or "burn-in") time.
  - How do you expect it to behave for different $\mu$?
  - Do the questions.
- Note: the proposal pdf is asymmetric
  - symmetric means that probability to jump to $\theta_{new}$ from $\theta^t$ is same as likelihood of jumping back to $\theta^t$ from $\theta_{new}$: $q(\theta_{new}|\theta^t)$
    - Typically $N(\theta^t, \sigma)$ with fixed $\sigma$.
    - Symmetric because difference of $\theta_{new}, \theta^t$ appears squared.

W1a-D

6/12/19

Afternoon exercise: MCMC-random-walk-and-sampling

- play with random walks
  ⇒ answer questions
- autocorrelation

- MCMC sampling of a Lorentzian pdf using the random walk Metropolis algorithm.

. Work through as much as you can: ask questions!

Packaged solvers:
emcee
pymc3
pystan
and more.

Look at emcee example ⇒ parameter_estimation_in_bayesTALENT_intro.ipynb
- Gaussian noise example
- emcee set up
- Go through parameters ndim, nwalkers, nburn, nsteps
- You can use this as a template!

W1a-1

6/12/19

How does emcee work?

emcee is an algorithm based on Goodman and Weare's paper called "Ensemble Samplers with Affine Invariance"

- Examples of affine transformations: translation, scaling, reflection, rotation, shear mapping. Preserves points, straight lines, planes.
- explanation at iacs-courses.seas.harvard.edu/courses/am207/blog/lecture-16.html