6.882 Presentation: Visualization in Bayesian Workflow

Tarek Mansour, Edgar Minasyan, Chandler Squires

Part 1 (Chandler): Overview and Model Choice

Throughout this course, we've developed a powerful toolbox. We've looked at modeling techniques like topic models and nonparametrics, and inference algorithms like ADVI and Hamiltonian Monte Carlo. While these are great in theory, as we've probably seen throughout the course of working on our projects, the hard part is putting these techniques into practice. To help us out with that, we've looked at a set of tools for evaluation.

Last week, we talked about Simulation-Based Calibration, which provides a fundamental snanity check of whether or not our model is implemented correctly. Then on Thursday, we talked about some problems particular to MCMC samplers, like getting decent mixing, and we talked about some diagnostics we can do to fix any problems with mixing. Today, we come to the last topic in this section, and this class, which is a super useful one now that we're going out into the world to actually use these techniques, and that's actually coming up with a model and improving it until we think it explains our data. This paper outlines a workflow for this and gives us some visual tools we can use to look for places to improve our model.

The approach the authors advocate is an iterative one of model extension, where we start with a simple model and make it more and more sophisticated until we're happy with how it explains the data. The

- Choosing a model (data to use, dependencies between variables, distributions)
- Choosing prior (vague vs. weakly informative)
- Checking algorithm (even if implemented correctly, could have some trouble, e.g. the funnel of death)
- Checking posterior (make sure it kinda looks like the data)

The example that they use throughout the paper is this problem of predicting pollution level in areas without the sensors that can measure it, using this technique called regression calibration, which in general takes some variable that's related to the variable of interest but more widely available and finds the relationship between them given the data available. In this case, the variable that they use for calibration is satellite measurements of aerosol optical depth, which measures how much aerosols in the atmosphere block light by absorption or scattering.

The first model they start out with is a simple linear regression between the log of the particulate matter concentration and the log of the aerosol optical depth. Even here a lot of choices are already being made: we're ignoring all of the information about location and picking linear regression on their log scales, rather than linear regression on the original scale or a more sophisticated regression like GPR.

Show plots using other regressions and final regression

When they think about ways to expand this model, they make an offhand comment about Simpson's paradox. This is actually a really important point because its connected to one of the main reasons that we would want to expand a model: there are some confounding variables we're not taking into account. The example of Simpson's paradox that I've always seen is in the context of medical treatments and I thought it would be nice to walk through that and give some perspective then link it back to the current problem.

* draw medical treatment example on board*

* draw regression example on board*

Here's where they start considering the location information from the data. I guess partially because it's easier to show, but also because they don't want to divide up the data too much, they use the super-region information rather than country or city. Now when they do the regression, they see some variation in the slopes around the slope they got when using all the data together. So the next thing they come up with is a model where there's a baseline slope and intercept and there's some variation around that.

* draw original model and hierarchical model on board*

And finally they give a third model that gives us an idea for an alternative when we don't have labels, and actually ends up performing better when we get to the section on posterior predictive checks. Instead of using labels, we can essentially assign them ourselves by running some sort of clustering like K-means, which would work really well for the Simpson's paradox example. What they actually do is group all the ground monitor measurements from a country together into a unit, then cluster the countries based on their average particulate matter concentration. They choose to break the data up into 6 clusters, I think mostly to compare with the model 2.

Part 2 (Tarek): Prior and Posterior Predictive Checks

As Chandler discussed, the preliminary analysis coupled with the sort of visualization techniques mentioned in the paper helped find three models: a simple liner regression one, and two multilevel linear regression ones. Thus, so far we've seen how visualization can help choose the model, aka, the likelihood. In my talk, I will be discussing how we can use these sort of visualization tools to choose a prior but also how we can use them to "check" our posterior after inference. One of the techniques that will be discussed is how we can use our prior and posterior to genereate *fake data*, then use this generated data to evaluate the prior and the posterior.

Weakly Informative Joint Prior Data Generating Process and Prior Preditive Checks

There's a debate on how prior should be chosen. There's a couple of types of priors. I will just mention them here then talk more in depth about them:

- non informative priors: this is seen as just a step in the Bayesian workflow, it does not incorporate much information
- fully informative prior: in a way, it strongly captures the information we have about the problem.
- weakly informative prior: a weaker version of strongly, will discuss more in a bit
- conjugate prior: just makes comutation easier
- regularizing prior: in genre

In general, in the Bayesian framework, we specify a prior on all of the parameters. This leads to us having a joint prior distribution over data and parameters, which can be marginalized over the parameters to obtained a marginal prior over the data.

*** BOARD WORK***
Prior: $p(\theta)$
Joint Prior: $p(\theta, y) = p(y|\theta) \cdot p(\theta)$
Marginal Prior: $p(y) = \int p(y|\theta)p(\theta)d\theta$

So we basically have a generative model and we can already generate some y's, which are considered fake or synthetic data. Now, we can compare these fakes y's to the obserations to see "how" well our prior is doing and choose what is called a **'weakly informed joint prior data generating process"**. In a way, we are evaluating the prior in the context of likelihood: prior an dliklehood have to interact reasonably to generate reasonable data.
There's a couple of things to mention here. First, "how well" should be defined carefully since it is not always clear what a good prior is. Second, I want to dicuss the motivation of doing this.

In general, it is usually recommended in Bayesian statistics to choose a vague prior or noninformative priors. The key is that these priors don't place too much probability in any particular interval, so they are priors that somewhat do not heavily favor some values, a "perfect" example of such priors is a uniform distribution, where all the values have equal weight. This way we sort of let the data choose where the mode is. We're giving more "power" to the data by making the prior hold very little or no information.

There's a small caveat here: often we give probability to implausible values. The paper argues that is better to use our domain knowledge to give probability to extreme but plausible values, without giving probability to implausible values. We still want to give some weight to extremes, so we don't use a fully informative prior but a weakly informative prior. The idea is to use the marginal prior over the data to choose a more "weakly informative". "Weakly" sort of mislead me at first, but what it seems like they mean in the paper, is that we use the marginal prior to make sure that we're not getting implausible values, so in a way they argue that we should use a prior that is more restrained. In the paper they propose an example to show how such priors can fail pretty poorly:

Sex-attractivness example
*** BOARD WORK***
There's a paper called "Beautiful parents have more daughters" by Kanazawa (2017) were they measured in some way adolescents' attractiveness on a scale from 1-5 then recoreded the sex of these people's children. Sample size was ~ 3000. In the sample there was a positive correlation betwen attractivness and sex ratios. The paper reports a comparison between the most attractive group (5) and the others (1-4) sex ratios: the difference was 8% more girls for group 5, with standard error 3.3%. We have wo things here the prbability of girl birth for group 5, p_1 and the others, p_2, the difference is $\delta = p_1 - p_2$.

Uniform prior:
post $\propto$ likelihood. We get a normal $\mathcal{N}(0.08, 0.033^2)$ which implies that there's a 99.2% chance that beautiful parents are more likely to have girls, and an implied 50% chance that the difference is higher than 8%.
Danger: uniform prior really contradicts what we know about the human sex ratio $\sim 48.5\%$ and this is like a very stable estimate in general.
Fully informative prior:
$\mathcal{N}(0, 0.001^2)$. gives $\mathcal{N}(0.00007, 0.001^2)$, so we clearly don't see the previous results.
Weakly informative prior: between the two extremes, allows around 0.5 to 1 percentage point $\mathcal{N}(0, 0.005^2)$.
gives $\mathcal{N}(0.002, 0.005^2)$ ———————

This is what they do in the paper. They mentioned generating a "flip book" of datasets to sort of see what our prior output distriubtion looks like. They use two different types of priors on parameters. The first one is the vague recommended one, so this is just a prior that allows parameters to highly vary
. From Chandler's notation:
** BOARD WORK***
For vague:
$\beta_k \sim \mathcal{N}(0, 100)$
$\tau_k^2 \sim InvGamma(1, 100)$
For weakly informative:
$\beta_0 \sim \mathcal{N}(0, 1)$
$\beta_1 \sim \mathcal{N}(1, 1)$ $\tau_k^2 \sim \mathcal{N}_+(0, 1)$

The inverse Gamma is basically a random variable that is the reciprocal of a Gamma, so it would have been equivalent to parametrizing the Gaussian with mean and precision, and draw the presicion from a Gamma distribution.

SHOW PLOTS ON Gamma disitribution

So we see that we have high variability on the slope and intercept which can be really large.

SHOW PLOTS ON PRIOR PREDICTIVE CHECKS + talk about them

I mentioned tht "weakly" as misleading but basically here they sort of have to mention it to make sure that the information is still weak: we can still generate very extreme but plausible data, which we can see in the second plot. We can get really high values of the order of 10^4.

Posterior Preditive Checks

Now we switch to posterior predictive checks. This is very similar to the idea of generating data from the prior to see how it's doing except that now our generation process is data-informed as they call it in the paper.
** BOARD WORK***
Now we use the posterior to generate data
$p(\hat{y}|y) = \int p(\hat{y}|\theta)p(\theta|y)d\theta$

Mainly, we can check if our posterior overall sort of looks like the data.

SHOW PLOTS ON Y GENERATION WITH DENSITY ESTIMATION + talk about them

---

Also, we are using the dataset twice: for fitting and for checking. So it is better to use some metrics that are somewhat orthongal to the parameters included in the model. Here they used a Gaussian, which is parametrized by mean and variance, so it made sense to evaluate the results using skewness. Using the mean wouldn't tell us much because the we are probably fitting the Gaussian's mean to the mean of the dataset.

SHOW PLOTS ON SKEW + talk about them

IF TIME PERMITS:

SHOW PLOTS ON INTERREGION+ talk about them

Finally, I wanted to mention some interesting analogy here. It seems like we're encoding the y into the parameter space θ,then decode back, in the generation process, to get some new observations $\hat{y}$ and then compare that data to the real data to evalute our encoding. It is sort of similar to what we do with Variational Autoencoders except that we do not update parameters here we just sort of evaluate the model. Another interesting thing to mention is GANs in this context because we can draw an interesting analogy. This is very similar to what we do with GANs in general. We get some generated "distribution" on images and we looks at it to see if it looks like the real data distribution.

Overfitting Considerations

There's some concern here. It seems like we are using the data to choose the model, so we might have generalization problems here. For the priors, they made sure to mention that we should have generation

mechanism that are broad and that gave a wide range of possble datasets (more broad than what is observed). They emphasize that the the prior should focus on **plausible** datasets and not necessarily datasets similar to the observed dataset. For the posterior predictive checks, Edgar will touch on this in his discussion about influential measurements.

Part 3 (Edgar): HMC diagnostics and outlier/influential measurement detection

We've already talked about most of the components of the Bayesian workflow: choosing the model, choosing a prior, checking the posterior. Now let us see how one can use visualization tools properly in order to verify the correctness of an inference algorithm. Specifically, we look at HMC. As we know, Hamiltonian dynamics and its discretizations (like leapfrog integration) basically repeat the following updates for L steps

$H(q,p) = U(q) + K(p)$
$q = q + \epsilon p$
$p = p - \epsilon \nabla_q U(q)$

Before going into the visualization techniques that help us check the performance of HMC, as well as get information about the posterior landscape, I want to show an example of how things can go wrong. When working on my research project I encountered a peculiar problem with HMC that both did and didn't make sense to me

show the two plots while starting to explain

Basically, the first nonsense plot is standard HMC while the second one that seems to work much better is a hack-modified version where the correction probability ignores the kinetic energy K. The issue is that this is a steep landscape when initialized not at the top region, so for one leapfrog integration the momentum keeps piling up in the same direction, trying to get the state to the top. When reaching there, $K(p')$ is already huge and the state gets rejected. This keeps happening, but when we only look at U, the algorithm seems to work because it simply prefers high-probability regions – suits the problem. The actual answer to this question is choosing a mass matrix M for the kinetic energy, but that's not relevant. We just saw how a standard HMC implemented from the paper can terribly fail and we want to have tools to detect such things.

Clearly, we saw that there were many ways of checking MCMC performance, from convergence to autocorrelation and mixing. This is more or less what one can do for a general MCMC method, but HMC itself provides us more tools to analyze the geometrical structure of the posterior landscape to exploit. Basically, for HMC to work well one needs the geometry of the typical set of the posterior to be mostly smooth. Moreover, when this geometry is not smooth one can expect rapid divergence in the Hamiltonian when using leapfrog integration. Fortunately, we can use the error in the Hamiltonian with a large threshold as a heuristic to identify such non-smooth points. Here is where the visual part comes: with proper visualization, one can detect whether the heuristic alarms were false negatives or there actually is a non-smooth region in the typical set. This not only gives us an idea about what's happneing with HMC but, in general, what the structure of the posterior landscape is like. With such non-smoothness, no MCMC using local information would be able to overcome this obstruction of exploration.

show all four plots, explain what each means and what they are

Two types of plots: bivariate scatterplots, parallel coordinate plots. If there's structure for the green points in bivariate scatterplots or the green lines in parallel coordinate plots, then something is actually wrong – have an issue to handle about steepness and non-smoothness. Otherwise, it's a false alarm and everything is going fine.

Another point I want to talk about is identifying outliers or influential measurements using visualization tools – this also helps us with model selection or even model modification. For this we would be using *leave-one-out corss validation* (LOO), particularly the predictive distribution $p(y_i|y_{-i})$ or usually its log. This is an intuitive outlier detection since the ones with very small probabilities are probably outliers. We clearly want $p(y_i|y_{-i})$ to be large for most of y so one can extend this idea to model selection.

To do this, we will compute the expected log predictive desnities (ELPD) for the individual data points, we use Pareto smoothed importance sampling (PSIS). This method basically stabilizes importance sampling by changing u of the largest weights according to the generalized Pareto distribution. It's not important to know the details, but this provides an estimate of the shape parameter of the distribution $\hat{k}$. Moreover, PSIS can also be used to find the LOO apprximation – it can give estimates for ELPD for individual data points. Using this as a visualization tool, we plot $ELPD_3 - ELPD_2$ for models 2 and 3 vs each individual data point in our dataset. Whenever this value is positive, it's good for model 3 and vice versa.

* show only the plot with ELPDs *

The already mentioned shape parameter $\hat{k}$, which when positive indicates the number of finite moments of the Pareto distribution, turns out to be useful in deciding points' influence. It's an estimate of the following value

$$k = \inf\{k' > 0 : D_{\frac{1}{k'}}(p\|q) < \infty\}, \quad D_\alpha(p\|q) = \frac{1}{\alpha - 1} \log \int_X p(x)^\alpha q(x)^{1-\alpha} dx$$

This is Renyi divergence and note that k being closer to 0 means p, q being closer to each other – Renyi divergence is still a divergence. PSIS provides us with the estimate $\hat{k}$ between the full log predictive distribution and LOO log predictive distribution for all the individual points. Thus, we get a sense of the more influential points.

* show the plot, explain stuff on it, why lower than zero *

We can see that model 3 is doing better than model 2 from the first plot. The second one is the influential points for model 2. Looking at Mongolia we see that model 3 really outperforms model 2 on that individual point. Then looking at model 2's influential points, we see Mongolia is by far the most influential so model 3 is able to contain this outlier of a point much better than model 2.

Conluding remarks: all the tools for visualization discussed are very important. But as authors claim it's more important to have a concrete principled approach to the Bayesian workflow and not take any part of it, such as choosing priors or variables or checking if the result actually works, etc., lightly.