Likelihood-Free Variational Inference

Alexander Moreno, work done with Edward Meeds and Max Welling

Background

- Scientists often want to infer some parameters of observed data
- Examples:
  - Given an image of a brain, does this brain have cancerous tumors?
  - Given counts of populations at different times, can we infer fitness parameters of each population?
- In some domains, we can generate simulated data with the parameters as inputs. That is, given fitness parameters and initial population counts, generate trajectory.

Problem Statement

- Often, exact inference, where we use data to answer these questions, can't be done (definition of exact inference to come)
- How can we use the simulators to help us?

Problem Statement

- We want to estimate the probability of some parameters , given the data $P(\theta \mid X)$
- We assume that we have simulators, where we can plug in θ and get synthetic data $f(\theta, u)$, where u is a randomness term
  - Example: we set the fitness parameters of different species, and get a simulation of species counts
- How is $P(\theta \mid X)$ usually calculated?

Bayes Theorem: Exact Inference

$$P(\theta \mid X) = \frac{P(X \mid \theta)P(\theta)}{P(X)}$$

Bayes Theorem (Cont.)

- Idea:
  - Start with initial belief about the distribution of parameters (prior)
  - Multiply that by how likely the observed data is, given parameters (likelihood)
  - Normalize so that you still have a probability distribution (normalizing constant)
- You now have an updated belief, given evidence (posterior)
- Problem: often normalizing constant can't be calculated
- In our case, even the likelihood can't
- How can we use simulators to help us solve this?

Current Solution, Monte Carlo Estimation

- In this context, Monte Carlo methods are a family of techniques for sampling from a probability distribution when doing so directly is difficult
- That is, we can't directly sample the fitness parameters, given the observed data, but these techniques can help us do so

ABC Rejection Algorithm

- Simplest method: ABC rejection algorithm
- Idea: Draw parameters from prior distribution, simulate using those parameters. If the simulated data is close to the real data, keep it. Otherwise, throw it out.
- Problem: when posterior narrow compared to prior
- Problem: does not scale well to high dimensional data, where we have many features (characteristics) describing our data.
  - We will spend too much time throwing out data

Approximate MCMC

- Idea: “build a Markov chain on θ and correlate successive observations so that more time is spent in regions of high posterior probability” –Richard Wilkinson

- Problem: you end up making too many calls to simulator
  - A call to simulator at every time step

- Need a surrogate model

Solution: Variational Inference

- Use variational inference: much faster
- Idea: have a simpler distribution q with parameters, find parameters that minimize disrepancy (KL-divergence) between the simple distribution and the true posterior distribution
- This scales well to high dimensions and requires far fewer calls to the simulator than MCMC

Issue: the Likelihood

- Recall: likelihood is $P(X \mid \theta)$. How likely is our observed data, given the parameters?
- Variational Inference requires computation of the likelihood. In our case, this is intractable
- We can rewrite the objective function to be in terms of a pseudo-likelihood $P(X \mid f(\theta, u))$, which depends on simulator output
- We can now use variational inference!

One more issue!

- When you want to minimize your objective function, you have to take derivatives. In this case, we have to take derivatives with respect to simulator output
- A simulator might be a very complicated code-base.
  - If it’s 1000 lines of Python code, does it really make sense to differentiate it by hand?
- Solution: automatic differentiation
  - This applies the chain rule automatically to every single line of code

Example: AD

- We have a function functionToDifferentiate, and variables to differentiate with respect to gradvariables
- Derivatives = T.grad(functionToDifferentiate, gradvariables)
- Source: https://github.com/y0ast/Variational-Autoencoder

What do we have so far?

- It’s implemented for a simple test case, linear regression
- Now we have to get it working on a real problem
  - Find a usable simulator and dataset
  - Current target: use Fisher Wright model for population genetics
  - If this works, probably find one more use case

Productivity in the Program

Last Year's Program

- The output for the program was:
  - Two posters: Ben and Jason on visualization, Cody, Chris, and Miroslav on changepoint detection
  - Two workshop papers: one where I was first author, one where I was fifth author
  - All at SC14
  - Both papers extended to journal papers, decisions not yet available

On the successful submissions

- Cody and Ben were the most experienced researchers in the program, Chris, while not as experienced in independent research, is very strong technically.
- I continued some of the work from the Spring, and was lucky enough to get to work on a project that had the infrastructure built and only needed the experiments to be run, which I was responsible for.

Goals

- It's a short program, and there are two reasonable goals:
  - If your research over the summer is not related to your research back in the US, do the topic of your host and get your name on a poster or workshop paper at SC15
    - In some cases, based on topic, another venue may be better: for machine learning, ICML/NIPS/AISTATS, for visualization, KDD/IUI
  - Alternatively, do your own topic, continue the collaboration after the summer and get conference/journal papers with the group you work with

Working with your Hosts

- If they have a topic and you don’t see collaboration post-PIRE in the cards, do their topic
- If you’re doing their topic, they will generally know very well what it takes to get a poster/workshop paper/conference paper accepted, so you should really follow their guidance closely
  - If you don’t agree with them. Ask yourself: how much do I really know about this area and what the community wants?
- If you are doing your own topic, ask yourself honestly: am I an experienced enough researcher to do this, and does it make sense given the time constraints?

Travel and Vacation

- You’re in an interesting place: Japan, Brazil, Scotland, Holland
  - You usually won’t be ‘forced’ to do much
  - It’s easy to find yourself turning this into primarily a vacation
- Why shouldn’t you do that?

Sustainability

- Academia has a ton of opportunities to see a lot of cool places and do a lot of cool things if you're productive
  - Get a paper
  - Present it at SC15
  - From that paper, leverage it to get into another program/extend to another conference
  - Repeat
- You can milk one trip too much, or do several

A few things that I’ve observed

- Your team may lack an important, key skill needed for the project
    - Mention this to your PI and push to find collaborators
- People who choose their own topic without sufficient experience
    - Getting something handed to you is a luxury. Don’t give it up simply because “I want to do my own thing.”
- People who argue with their PI about the approach when they have no background in the topic

Don’t Lose Sight

- A lot of the time, you have a lot of things to do, and there is one thing that is the least pleasant possible thing, but it’s required to move forward
- Do it!
- Aim for something reasonable, but keep pushing until it happens