Surrogate Likelihoods

Roman Kouznetsov

October 2023

1 Introduction

model learning: learning likelihood parameters in conjunction with an (approximate) posterior

data subsampling: update inference using only a subsample of data at a time

Why are these 2 definitions crucial? Because the authors highly motivate a Bayesian analysis that can support both features. Most variational inference algorithms can support these regimes, but MCMC methods are asymptotically unbiased which makes them attractive.

Goal: Make the best of both worlds!

2 The Problem

Given a dataset $\mathcal{D}$, we have a model of the form:

$$p_\theta(\mathcal{D}, \mathbf{z}) = p_\theta(\mathbf{z}) \prod_{n=1}^{N} p_\theta\left(y_n \mid \mathbf{z}, \mathbf{x}_n\right)$$

Note that the authors do comment about the local latent variable regime but primarily focus on global latents in this paper.
Converting everything into log form, we get that the log likelihood is:

$$\log_\theta p_\theta(\mathcal{D}, \mathbf{z}) = \log_\theta p_\theta(\mathbf{z}) + \sum_n \log_\theta p_\theta\left(y_n \mid \mathbf{z}, \mathbf{x}_n\right) \equiv \Psi_0(z) + \Psi_L(\mathcal{D}, z)$$

The joint likelihood $p_\theta(\mathcal{D}, \mathbf{z})$ is the unnormalized kernel for the posterior $p(z|\mathcal{D})$.

- MCMC methods yield consistent estimators.
- But MCMC methods don't naturally support mini-batching.
- MCMC may not be that efficient when N is high.

This paper, **Surrogate Likelihoods for Variational Inference**, addresses the fact that this might not be very efficient if N is very large or if $\Psi_L(\mathcal{D}, z)$ and/or its gradient is tough to compute.

3 Previous Work & Setup

3.1 Variational Inference

We have covered variational inference ad nauseam, so I will not re-derive the ELBO here. However, we should remember what the ELBO is and what it does for us.

$$\textbf{ELBO} = E_{q_\phi(z)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) - \log q_{\boldsymbol{\psi}}(\boldsymbol{z})\right] = \mathbb{E}_{q_{\boldsymbol{\psi}}(\boldsymbol{z})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})\right] - D_{\mathbb{KL}}\left(q_{\boldsymbol{\psi}}(\boldsymbol{z}) \| p_{\boldsymbol{\theta}}(\boldsymbol{z})\right)$$

The ELBO is an objective we can maximize by learning variational parameters that represent a tractable approximate posterior. In situations where exact posterior inference is intractable, the ELBO saves the day!

3.2 Annealed Importance Sampling (AIS)

3.2.1 Introduction

- AIS dates all the way back to 2001.
- It's quite funny how similar it feels to the modern day stable diffusion models that also make iterative updates to the end goal.
- AIS is also consistent as $K \to \infty$,
- This means that choosing a large enough K can work very well in practice.

3.2.2 Formalization

The overall idea at player here is:

"Intuitively given two distributions, which might be disjoint in their support, we create intermediate distributions that are "bridging" from one to another. Then we do MCMC to move around these distributions and hope that we end up in our target distribution."

In the context of Bayesian analysis, we start with an initial proposal distribution of the prior: $q_0(z) = p_\theta(z)$ $(\beta_0 = 0)$. We choose the unnormalized posterior as the target: $f_K(z) = p_\theta(\mathcal{D}, z)$ $(\beta_K = 1)$. So, naturally, you want to **bridge** your way from the prior to the posterior.

You could go forward from the initial proposal to the target as follows:

$$q_{\text{fwd}}\left(\mathbf{z}_{0:K}\right) = q_0\left(\mathbf{z}_0\right) \mathcal{T}_1\left(\mathbf{z}_1 \mid \mathbf{z}_0\right) \cdots \mathcal{T}_K\left(\mathbf{z}_K \mid \mathbf{z}_{K-1}\right)$$

You could also go backward from the target to the proposal as follows:

$$q_{\text{bwd}}\left(\mathbf{z}_{0:K}\right) = p_\theta\left(\mathcal{D}, \mathbf{z}_K\right) \tilde{\mathcal{T}}_K\left(\mathbf{z}_{K-1} \mid \mathbf{z}_K\right) \cdots \tilde{\mathcal{T}}_1\left(\mathbf{z}_0 \mid \mathbf{z}_1\right)$$

Here each $\mathcal{T}_k$ is a MCMC kernel that leaves the bridging density $f_k(\mathbf{z})$ invariant. It makes sense that the bridging densities are $f_k(\mathbf{z}) \propto q_0(\mathbf{z})^{1-\beta_k} p_\theta(\mathcal{D}, \mathbf{z})^{\beta_k}$ where $\{\beta_k\}$ are inverse temperatures that satisfy $0 < \beta_1 < \beta_2 < \ldots < \beta_K = 1$.

$$p_\theta(\mathcal{D}) = \int d\mathbf{z}\, p_\theta(\mathcal{D}, \mathbf{z}) = \int d\mathbf{z_{0:K}}\, q_{\text{bwd}}\left(\mathbf{z}_{0:K}\right) = \mathbb{E}_{q_{\text{fwd}}(\mathbf{z}_{0:K})}\left[\frac{q_{\text{bwd}}\left(\mathbf{z}_{0:K}\right)}{q_{\text{fwd}}\left(\mathbf{z}_{0:K}\right)}\right]$$

where the reverse transitions $(\tilde{\mathcal{T}}_k)$ are simply:

$$\tilde{\mathcal{T}}_k\left(\mathbf{z}_{k-1} \mid \mathbf{z}_k\right) = \mathcal{T}_k\left(\mathbf{z}_k \mid \mathbf{z}_{k-1}\right) f_k\left(\mathbf{z}_{k-1}\right) / f_k\left(\mathbf{z}_k\right)$$

Note that plugging in the forward and backward chains, we get that we can approximate the evidence via importance sampling:

$$p_\theta(\mathcal{D}) = \mathbb{E}_{q_{\text{fwd}}(\mathbf{z}_{0:K})}\left[\frac{q_{\text{bwd}}\left(\mathbf{z}_{0:K}\right)}{q_{\text{fwd}}\left(\mathbf{z}_{0:K}\right)}\right] \approx \mathbb{E}_{p(x)}[x] = \frac{1}{\sum_i^N w_i} \sum_i^N x_i w_i$$

where w_i can be algebraically mathed out to be

$$\prod_{k=1}^{K} \frac{f_k(z_{k-1})}{f_{k-1}(z_{k-1})}$$

Finally, notice that Jensen's inequality allows us to place a variational lower bound on the log evidence.

3.3 HMC

3.3.1 Intuitive Explanation

- "HMC generates a hypothetical physical system: imagine a ball with a certain kinetic energy rolling around a landscape with valleys and hills (the analogy breaks down with more than 2 dimensions) defined by the posterior you want to sample from. Every time you want to take a new MCMC sample, you randomly pick the kinetic energy and start the ball rolling from where you are. You simulate in discrete time steps, and to make sure you explore the parameter space properly you simulate steps in one direction and then twice as many in the other direction, turn around again etc."

- The above analogy breaks down for higher dimensions, but the idea is the same. The kinetic energy and momentum help make proposal based on the shape of the posterior (which leads to high acceptance probabilities even in high dimensions). Other proposals typically move along a small amount of dimensions at a time, which can lead to poor convergence.

3.3.2 Slightly More Formal Explanation

- So how does it work? Given that most of the novel work leaves most of HMC intact, I won't get into **all** the details, but the general gist is the following:
  - Suppose you are given a momentum $\mathbf{v}$.
  - Also, suppose you have a physical system that has potential energy $V(\mathbf{z})$ and kinetic energy $T(\mathbf{v})$.
  - Kinetic energy has a recognizable formula from phsyics $T(\mathbf{v}) = \frac{1}{2}\mathbf{v}^{\mathbf{T}}\mathbf{M}\mathbf{v}$.
  - We can then define the potential energy as $V(\mathbf{z}) = -\log p_\theta(\mathcal{D}, \mathbf{z})$. This makes sense because in regions of high potential area, we are in high regions of can minimize the NLL (and therefore maximize LL).
  - Hamiltonian dynamics can yield an MCMC by defining the Hamiltonian as $H(\mathbf{z}) = T(\mathbf{z}) + V(\mathbf{z})$
  - These systems in theory operate on a continuum, but if we were to discretize these steps, we can take a leap along the trajectory and then refresh our momentum.

The discretized update looks like the following for a move from $(z_{k-1}, y_{k-1}) \rightarrow (z_k, y_k)$:

$$\begin{array}{ll} \hat{\mathbf{z}}_k \leftarrow \mathbf{z}_{k-1} + \frac{\eta}{2}\mathbf{M}^{-1}\mathbf{v}_{k-1} & \hat{\mathbf{v}}_k \leftarrow \mathbf{v}_{k-1} - \eta\nabla V\left(\hat{\mathbf{z}}_k\right) \\ \mathbf{z}_k \leftarrow \hat{\mathbf{z}}_k + \frac{\eta}{2}\mathbf{M}^{-1}\hat{\mathbf{v}}_k & \mathbf{v}_k \sim \mathcal{N}\left(\gamma\hat{\mathbf{v}}_k, \left(1-\gamma^2\right)\mathbf{M}\right) \end{array}$$

Note that while 2 updates to the position z_k are made, the step size is divided in half across a changing momentum value and so the magniture of each update is that of a single step.
Worthy of note **since it gets taken for granted right away**, the momentum updates are called **refresh (transition) kernels** where as updates to latents and momentum as a result of moving along kinetic and potential energy curves are called **leap (transition) kernels**. The proverbial "leap" is because this is discrete step even though these movements happen in the continuous space.

These 2 updates get combined to from the **HMC kernel**: $\mathcal{T}_k(z_k, v_k | z_{k-1}, v_{k-1})$

3.4 Differentiable AIS

- DAIS is a prominent attempt at combining AIS and HMC together!

- DAIS removes the accept/reject step in HMC to make the objective differentiable.
- Removing this step doesn't provide all theoretical guarantees, but maintains AIS and the variational lower bound!

Leveraging the aforementioned HMC Kernel and plugging the negative log annealed bridge density $(1-\beta_k)q_0(z)+\beta_k \log p_\theta(\mathcal{D}, z)$ for the potential energy, we get the HMC infused AIS equations as follows:

$$\begin{aligned} q_{\text{fwd}}\left(\mathbf{z}_{0:K}, \mathbf{v}_{0:K}\right) &= q_0\left(\mathbf{z}_0\right) q_0\left(\mathbf{v}_0\right) \times \textstyle\prod_{k=1}^{K} \mathcal{T}_k\left(\mathbf{z}_k, \mathbf{v}_k \mid \mathbf{z}_{k-1}, \mathbf{v}_{k-1}\right) \\ q_{\text{bwd}}\left(\mathbf{z}_{0:K}, \mathbf{v}_{0:K}\right) &= p_\theta\left(\mathcal{D}, \mathbf{z}_K\right) \times \textstyle\prod_{k=1}^{K} \tilde{\mathcal{T}}_k\left(\mathbf{z}_{k-1}, \mathbf{v}_{k-1} \mid \mathbf{z}_k, \mathbf{v}_k\right) \end{aligned}$$

where $q_0\left(\mathbf{v}_0\right)=\mathcal{N}\left(\mathbf{v}_0 \mid \mathbf{0}, \mathbf{M}\right)$ is the momentum distribution. Here each kernel $\mathcal{T}_k$ performs a single leapfrog step as in Eqn. 8 using the annealed potential energy

$$\begin{aligned} V_k(\mathbf{z}) &= -\left(1-\beta_k\right) \log q_0(\mathbf{z}) - \beta_k \log p_\theta(\mathcal{D}, \mathbf{z}) \\ &= -\left(1-\beta_k\right) \log q_0(\mathbf{z}) - \beta_k \left(\Psi_0(\mathbf{z}) + \Psi_{\mathrm{L}}(\mathcal{D}, \mathbf{z})\right) \end{aligned}$$

Notice that DAIS only performs a leapfrog step and does NOT have an accept/reject technique.

The term $\Psi_L(\mathcal{D}, z)$ is literally the ONLY term that the authors attempt to change in this whole regime. Obviously, this is because it leads to the most expensive computation, but it's important to take note of this so we don't lose the big picture.

It can be shown that the variational objective of DAIS is:

$$\begin{aligned} \mathcal{L}_{\text{DAIS}} &\equiv \mathbb{E}[\, \log q_{\text{bwd}}\left(\mathbf{z}_{0:K}, \mathbf{v}_{0:K}\right) - \log q_{\text{fwd}}\left(\mathbf{z}_{0:K}, \mathbf{v}_{0:K}\right)] \\ &= \mathbb{E}\Bigg[\log p_\theta\left(\mathcal{D}, \mathbf{z}_K\right) - \log q_0\left(\mathbf{z}_0\right) + \\ &\qquad \sum_{k=1}^{K} \left\{\log \mathcal{N}\left(\hat{\mathbf{v}}_k, \mathbf{M}\right) - \log \mathcal{N}\left(\mathbf{v}_{k-1}, \mathbf{M}\right)\right\}\Bigg] \end{aligned}$$

The sum over log normals is actually the result of ratios of refresh steps.

4 Proposition

$\mathcal{L}_{\text{DAIS}}$ can certainly lead to tight bounds on the log evidence $p_\theta(\mathcal{D})$. However, looking back at the formula for $\mathcal{L}_{\text{DAIS}}$, this ELBO requires us to take a gradient for each point in the dataset (N in total), K times due to the iterations of latents, making this whole process $\mathcal{O}(NK)$.

If N is huge, this is very bad. This motivates approaches that can reduce the number of computations needed.

While I won't go into detail here since the proof is short and included in A.4, both the DAIS objectives generated by our 2 approaches yield a valid variational objective s.t.

$$\log p_\theta(\mathcal{D}) - \mathcal{L} \geq \mathrm{KL}(q_{\mathrm{fwd}}(\mathbf{z}_k)||p_\theta(\mathbf{z}_k|\mathcal{D})) \geq 0$$

As $\mathcal{L}$ increases, the KL divergence must go to 0 which means that the forward chain is a better approximation of the posterior.

4.1 Naive Subsampling DAIS: NS-DAIS

GOAL: Allow for DAIS to work on minibatches.

NS-DAIS tries to accomplish this by computing B likelihood terms instead of N likelihood terms. These terms get averaged (multiplied by $\frac{1}{B}$), and then multiplied by N to maintain the scale of the sum of all log likelihoods, yielding a new estimator: $\frac{N}{B}\Psi_L(\mathcal{D}_J, z)$

By only having to calculate B likelihoods, sampling and optimization is $\mathcal{O}(N) \rightarrow \mathcal{O}(B)$.

However, note that because the batch assignment can be random, the estimator $\frac{N}{B}\Psi_L(\mathcal{D}_J, z)$ is a stochastic one. The authors of this paper attempt to take a similar approach that reduces stochasticity.
Note that the authors partially thought this was the way since the authors of NS-DAIS stated that lowering the step size in the gradient doesn't help alleviate error and that gradient noise needs to be reduced.

4.2 Surrogate Likelihood DAIS: SL-DAIS

So NS-DAIS has introduced stochasticity since the random selection for the batch influences what we plug in for $\frac{N}{B}\Psi_L(\mathcal{D}_J, z)$. SL-DAIS proposes choosing a surrogate likelihood that is significantly cheaper to evaluate than $\Psi_L(\mathcal{D}, z)$.
However, the question remains on how exactly to construct such a surrogate likelihood. The paper presents 4 parameterization regimes; I have saved the best one for last.

4.3 Surrogate Likelihood Parameterizations

4.3.1 CS-INIT

Algorithm 1 CS-INIT Parameterization

Require: Dataset Size: N

$N_{\text{surr}} << N$

Surrogate Data: $(\tilde{y}_i, \tilde{\mathbf{x}}_i)|_{i=1}^{N_{\text{surr}}} \leftarrow$ BAYES_CORESET(1, N, N_{surr})

Surrogate Likelihood: $\Psi_L(z) = \sum_n w_n \log p_\theta(\tilde{y}_n|\mathbf{z}, \tilde{x}_n)$

$\phi, \mathbf{w} \leftarrow \text{NN}_\phi(\text{x})$

4.3.2 CS-FIX

Algorithm 2 CS-FIX Parameterization

Require: Dataset Size: N

$N_{\text{surr}} << N$

Surrogate Data: $(\tilde{y}_i, \tilde{\mathbf{x}}_i)|_{i=1}^{N_{\text{surr}}}, \mathbf{w} \leftarrow$ BAYES_CORESET(1, N, N_{surr})

Surrogate Likelihood: $\hat{\Psi}_L(z) = \sum_n w_n \log p_\theta$

4.3.3 NN

Algorithm 3 NN Parameterization

$\hat{\Psi}_L(z) = \text{NN}_\phi(z)$

4.3.4 RAND

Algorithm 4 RAND Parameterization

Require: Dataset Size: N

$N_{\text{surr}} << N$

Surrogate Data: $(\tilde{y}_i, \tilde{\mathbf{x}}_i)|_{i=1}^{N_{\text{surr}}} \leftarrow$ RAND(1, N, N_{surr})

Surrogate Likelihood: $\hat{\Psi}_L(z) = \sum_n w_n \log p_\theta(\tilde{y}_n|\mathbf{z}, \tilde{x}_n)$

$\phi, \mathbf{w} \leftarrow \text{NN}_\phi(\text{x})$

Worthy to note since this confused me a bit. The choice to randomly select N_{surr} points from the dataset doesn't *magically* remove the stochasticity. The reduction is stochasticity is that the weights in the surrogate likelihood are learned by a NN and hopefully recover the same likelihood regardless of what data points are chosen. NS-DAIS on the other hand, **cannot** recover the same distribution and is doomed to fail from the start. The sampling noise for the batch cannot be reduced.

Also note that this is not the same as the NN parameterization since the NN doesn't have the guidance of the true likelihood function $p_\theta(y|z,x)$.

5 Results from Experiments

The main point that this paper wants to communicate is that SL-DAIS is a better continuation of the groundwork laid in NS-DAIS.

- All SL-DAIS to NS-DAIS (the 2 methods that keep DAIS but introduce a way to minibatch) comparisons lead to SL-DAIS coming out on top.
- However, the results when compared to DAIS or even standard variational methods are mixed and very problem dependent.
- DAIS can still outperform SL-DAIS in some areas, but may not be computationally viable.

5.1 Logistic Regression

5.1.1 Parameterization Regimes

On 2 data sets (Higgs and SUSY), 4 regimes are evaluated with large $N_{\text{surr}} = 1024$ and small $N_{\text{surr}} = 64$. The ELBO for each settings is compared to the mean field Normal ELBO baseline and the result is presented as $\text{ELBO}_{\text{Setting}} - \text{ELBO}_{\text{Mean Field}}$.

- **RAND > CS-INIT > CS-FIX > NN**
- Bayesian coresets don't perfectly capture tail information since they prioritize representing the posterior mode.
- NN performs poorly since it is not guided at all by the true likelihood function.
- CS-INIT and RAND perform similarly, which is great because we can likely get many of the benefits that CS-INIT provides but without a complicated coreset algorithm to collect N_{surr} points.

5.1.2 Model Comparisons

- **5** logistic regression datasets.
- **10** methods compared: 8 Variational, 2 MCMC.
- D is the latent space dimension in Figure 2.
- Out of the variational methods SL-DAIS with MVN setup and $K = 8$ performed the best on the CovType datasets (though they lose to HMCECS on likelihood).

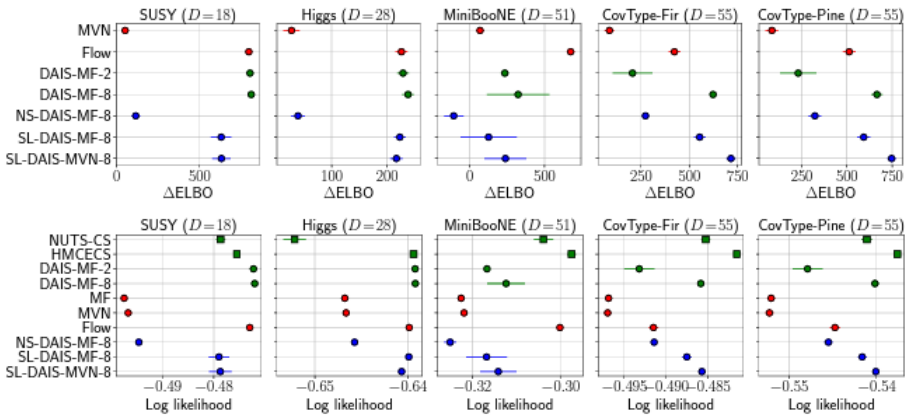

Figure 2: We report ELBO improvements and test log likelihoods for the logistic regression experiment in Sec. 7.3. ELBO improvements are with respect to the mean-field (MF) Normal baseline. Circles denote variational methods and squares denote MCMC methods. Blue methods are ours, red methods are mini-batchable variational methods, and green methods are everything else. Metrics are averaged over 7 independent replications and error bars denote standard errors (we do 3 independent replications for the most expensive methods, namely DAIS and HMCECS). See Sec. A.7.3 in the supplement for test accuracies. Note that numerals in method names indicate the number of HMC steps K used.

	MF	MVN	Flow	NS-DAIS-MF	SL-DAIS-MF	NS-DAIS-MVN	SL-DAIS-MVN
ELBO	6.60	5.36	**2.08**	5.00	2.60	4.16	2.20
Log likelihood	6.16	5.88	**2.16**	4.60	2.44	4.36	2.40
Opt. time	**38.8**	58.1	3420.3	113.1	83.1	152.1	131.6

Table 2: We report performance ranks w.r.t. ELBO and test log likelihood across 5 train/test splits and 5 datasets for the logistic regression experiment in Sec. 7.3. We also report time per optimization step in milliseconds as in Fig. 3. Lower is better for all metrics. The rank satisfies $1 \leq \text{rank} \leq 7$, since we compare 7 scalable variational methods.

Figure 1: Enter Caption

5.2 Class Imbalance

- All DAIS frameworks are evaluated on a class imbalance problem where the ratio of common to rare class gradually increases.
- DAIS > SL-DAIS > NS-DAIS
- This experiment paints SL-DAIS as superior to NS-DAIS, but that DAIS can still be better to use when N is low or the full log likelihood is easy to compute.

5.3 Gaussian Process Regression/Classification

- SL-DAIS beats standard MVN VI.

- Just validates the approach somewhat, nothing really to write home about with the final 2 experiments.

6 Major Takeaways

- NS-DAIS was proposed to create a mini-batching scheme for DAIS that's more computationally feasible.
- SL-DAIS introduces a way to estimate posteriors using a subset of the data via surrogate likelihoods.
- The performance benefits of mini-batching
- DAIS has better performance in low data regimes, but struggles to keep up in the higher data regime.
- In high data regimes, SL-DAIS may not have the same theoretical guarantees as SL-DAIS but is able to do MANY more bridge steps at a similar computational cost which can help performance in practice.