6.882: Bayesian Modeling and Inference **Lecture Date: February 22, 2018**

Automatic Differentiation Variational Inference

Lecturer: Anna Sinelnikova and Brian Trippe *Scribe: Helian Feng and Gregory Young*

1 Automating Variational Inference

1.1 Motivation

In class, we have discussed methods for performing Bayesian inference, in particular, variational inference (VI). However, even from the first reading on LDA by Biel et al. (2003), there have been questions and uncertainties surrounding the performance of these methods.

For example, the computations surrounding LDA were quite intense, and attempts to simplify calculations using a technique like MFVB were illustrated by Turner & Sahani (2017) to have problems in terms of bias. Thus, attempts to define a methodology that speeds up (or even automates) Bayesian inference computations while not sacrificing accuracy would be extremely useful.

1.2 Kucukelbir et al. (2011)

In this paper, the authors introduce an algorithm for automated differentiation variational inference (ADVI), which through some tricks and transformations, they claim can automate variational inference by reducing the computational complexity of gradients of the ELBO, a concept first introduced to us back in the Hoffman et al. (2013) paper on SVI. Before diving into some of the finer details, it is important to enumerate some of the desired characteristics of an automated algorithm like this one:

- Fast
- Accurate
- Correlation-aware
- Black-box to user

Speed and accuracy are important for obvious reasons. Correlation-awareness, while a part of the accuracy aspect, deserves a separate mention because it is not necessarily straightfoward to capture it (e.g. performance suffers) as this paper discusses when comparing the mean-field and full-rank versions of ADVI.

Finally, black-box is important because these algorithms can be complicated! However, if they are to be widely used, they should present an interface that is friendly to someone who is not well-versed in VI. Coming up with a good model to use for Bayesian inference is hard enough as it is, so if users can offload the inference computations to the automated method, that would give them more time to focus on developing and trying out models instead of tweaking lots of parameters to perform inference.

1.3 Example: Bayesian Logistic Regression

Consider the following setup:

$$D = \{(x_{i1}, x_{i2}), y_i\}_{i=1}^{N}$$

$$y_i \sim Bern(p = \frac{1}{1 + exp(-\beta^T X)})$$

$$\beta \sim p(\beta|\alpha) = N(\beta|0, I)$$

This example is pretty simple with only two variables, and it might seem unnecessary to unleash the full power of Bayesian inference on an example like this. Couldn't we just use a MAP estimate and call it a day? Perhaps, but suppose x_1 and x_2 were very correlated. Suddenly, we could end up finding many "good" MAP estimates, all of which could be incorrect because they don't properly account for the correlated nature of the inputs. VI, however, allows us to account for that subtlety. If we choose the VI route, we end up with the following posterior:

$$P(\beta|\alpha, D) = \frac{P(\beta|\alpha)P(y|\beta, x)}{\int P(\beta|\alpha)P(y|\beta, x)d\beta}$$

Assuming that our integral is intractable (as it would be in most cases), we then turn to VI to perform the necessary approximations to facilitate computations (more detail can be found in the Hoffman paper and in the first couple of lectures). We summarize it here:

VI approach:

1. Propose $q_\phi(\beta)$

2. Find $\phi^* = \text{argmax}_q \; E\left[\log \frac{p(\beta, y|\alpha, x)}{q_\phi(\beta)}\right]$

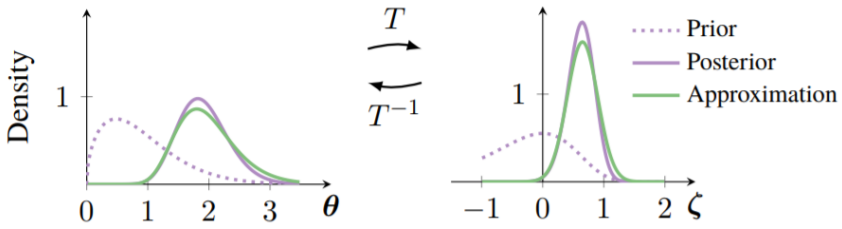
Maximizing this expectation (the ELBO) is not straightforward though, as the authors point out. Although existing processes for ELBO maximization can involve some automation through software, "[e]ach step requires expert thought and analysis in the service of a single algorithm for a single model" (pg. 6), an imposition that most certainly violates our black-box expectation for automation.

1.4 Enabling Automation

The general idea of automation is standardization, so to that end, the authors introduce two transformations:

1. Transformation of latent variable support onto the real space $\mathbb{R}^K$.

2. Elliptical Standardization



(a) Latent variable space **(b)** Real coordinate space

Figure 1: Transforming the latent variable to real coordinate space. The purple line is the posterior. The green line is the approximation. **(a)** The latent variable space is $\mathbb{R}_{>0}$. **(a→b)** T transforms the latent variable space to $\mathbb{R}$. **(b)** The variational approximation is a Gaussian in real coordinate space.

The transformation in step 1 standardizes the support so that variational approximations can be chosen independent of the model (Figure 1 above from the paper shows an example). The second transformation facilitates the computation of gradients over the ELBO by allowing us to push the gradient inside the expectation, as it is the expression inside the expectation that has a more tractable gradient.

Of course, there still remains the necessity of picking the variational approximation to use. In lecture (and in the paper), we chose to impose a Gaussian distribution. This leads to the following algorithm for ADVI:

ADVI approach:

1. Transform the latent variables into the R^k coordinate space.

2. Posit a Gaussian distribution on the transformed coordinate space, that is propose $q_\phi(\beta)$ is MVN i.e. $q_\phi(\beta) = N(\beta|\mu, \Sigma)$, where $\Sigma = LL^T$, and L is the Cholesky factor.

3. Apply automatic differentiation (via elliptical standardization) to maximize the objective function and calculate the posterior in the transformed coordinate space.

4.Apply the inverse of the transform performed in step 1 to transform the posterior back into the original latent variable coordinate space.

1.5 ADVI in Action

Let us revisit the ELBO, defined in different but equivalent terms, below:

$\mathcal{L} = \int \log P(y, \beta|x, \alpha) q_\phi(\beta) d\beta + H(q_\phi(\beta))$, where $H(q_\phi(\beta))$ is an entropy term.

Here, we impose our Gaussian distribution on $q_\phi(\beta) = \mathcal{N}(\beta|\mu, \Sigma = LL^T)$ and take advantage of that to perform the following elliptical standardization:

$\epsilon = L^{-1}(\beta - \mu)$

Note that $q_\phi(\epsilon) = \mathcal{N}(\epsilon|0, I)$. Consequently, we can rewrite our ELBO as follows:

$$\begin{aligned}
\mathcal{L} &= \int \log P(y, \beta|x, \alpha) q_\phi(\beta) d\beta + H(q, \beta) \\
&= \int \log p(y, \beta = L\epsilon + \mu) q_\phi(\beta = L\epsilon + \mu)\ d(L\epsilon + \mu) + C \\
&= \int \log p(y, \beta = L\epsilon + \mu) q_\phi(\epsilon) d\epsilon + C \\
&= E_{q_\phi(\epsilon)}[\log p(y, \beta = L\epsilon + \mu)] + C
\end{aligned}$$

We make the jump from the second to the third line due to this equality:

$$\begin{aligned}
q_\phi(\beta)d\beta = q_\phi(\beta = L\epsilon + \mu)d(L\epsilon + \mu) \\
= q_\phi(\epsilon) \cdot |L| \cdot |L|^{-1} d\epsilon \\
= q_\phi(\epsilon) d\epsilon
\end{aligned}$$

It is now thanks to elliptical standardization that we can approximately compute the gradient (via Monte Carlo integration) as follows:

$$\begin{aligned}
\nabla_\phi(E_{q(\phi)}[\log p(B = L\epsilon + \mu, y|x, \alpha)]) &= E_{q(\phi)}[\nabla_\phi \log p(B = L\epsilon + \mu, y|x, \alpha)] \\
&\approxeq \frac{1}{M} \sum_i^M \nabla_\phi \log p(B = L\epsilon_i + \mu, y|x, \alpha)
\end{aligned}$$

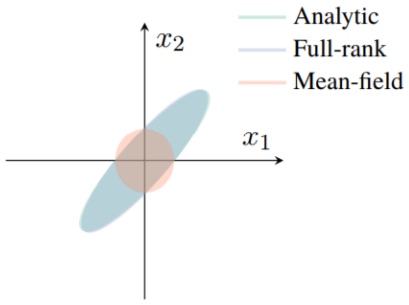
Because the Monte Carlo integration can be automated, and the gradient of the log is tractable, it is evident that the approximation can be automated, thereby allowing us to automate ELBO-maximization.

2 How is Good is the Automation?

This ultimate question of algorithm quality is something that loomed over, in varying degrees, many of the other algorithms that have been discussed so far. In this paper, the authors make a good attempt to illustrate

how performant ADVI is using some more real-life examples, so let's see how convincing their evidence was.

2.1 Multivariate Gaussian



	Analytic	Full-rank	Mean-field
Variance along x_1	0.28	0.28	0.13
Variance along x_2	0.31	0.31	0.14

Figure 4: Comparison of mean-field and full-rank ADVI on a two-dimensional Gaussian model. The figure shows the accuracy of the full-rank approximation. Ellipses correspond to two-sigma level sets of the Gaussian. The table quantifies the underestimation of marginal variances by the mean-field approximation.

One of the first experiments on which the authors tested ADVI was on a two-dimensional Gaussian model. In this experiment, they compared both the mean-field and full-rank ADVI algorithms. Similar to the Turner & Sahani paper, we can observe in the **Figure 4** above that the mean-field version underestimates the marginal variances of the individual variables due its failure to account for correlations between the two variables, which the full-rank version does account for.

2.2 Logistic Regression

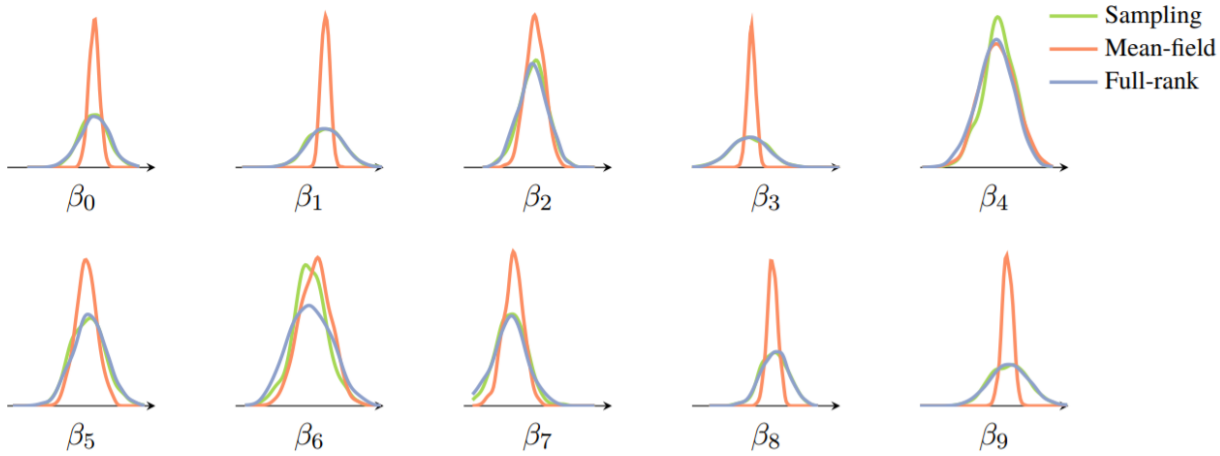


Figure 5: Comparison of marginal posterior densities for a logistic regression model. Each plot shows kernel density estimates for the posterior of each coefficient using 1000 samples. Mean-field ADVI underestimates variances for most of the coefficients.

The authors perform a similar experiment with multiple-logistic regression. The results in the **Figure 5**

above are similar to those of the multivariate-Gaussian experiment. The mean-field version underestimates the marginal variances, while the full-rank does not.

2.3 Stochastic Volatility Time Series

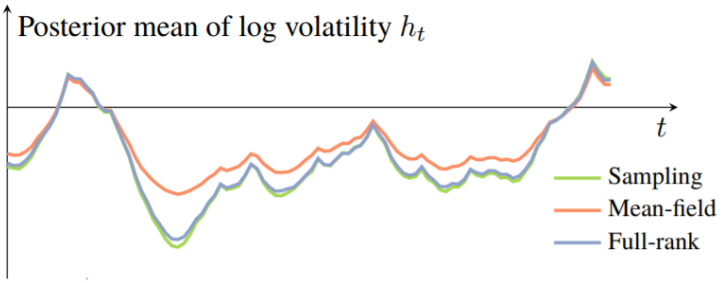
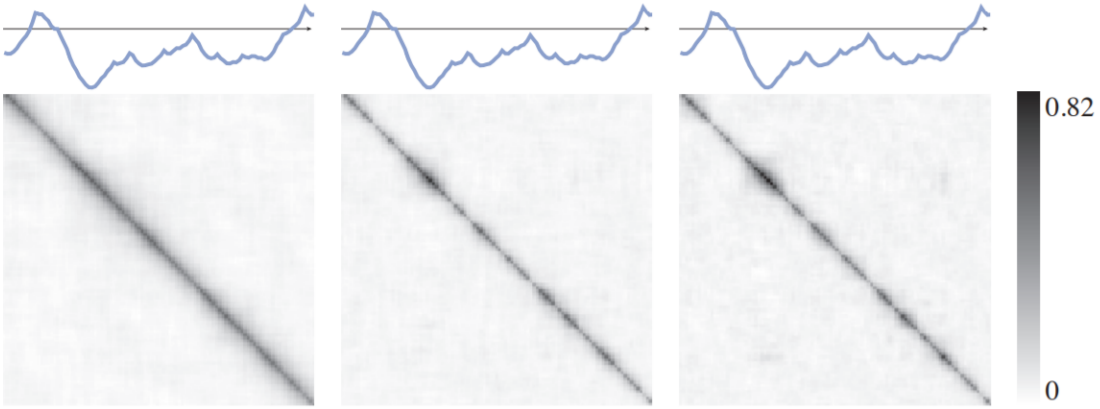


Figure 6: Comparison of posterior mean estimates of volatility h_t. Mean-field ADVI underestimates h_t, especially when it moves far away from its mean μ. Full-rank ADVI matches the accuracy of sampling.



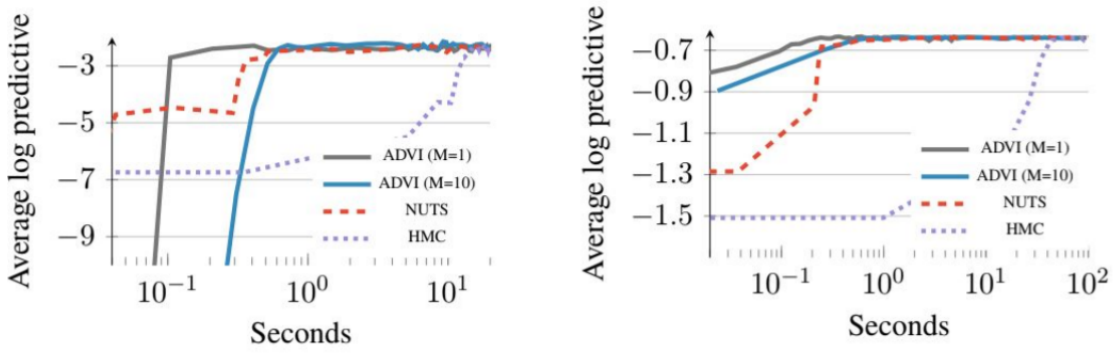
(a) Mean-field **(b)** Full-rank **(c)** Sampling

Figure 7: Comparison of empirical posterior covariance matrices. The mean-field ADVI covariance matrix fails to capture the local correlation structure seen in the full-rank ADVI and sampling results. All covariance matrices exhibit a blurry spread due to finite sample size.

In this experiment, the authors compare the performance of mean-field and full-rank on time series data, which is non-exchangeable. Although full-rank does a better job than mean-field in terms of estimating the volatility, the superiority is not as clear, especially in their **Figure 7**. Part of the blurriness can be attributed to noisy (finite) sampling, and otherwise, the gray lines in all three graphs are relatively close.

A similar statement could be made about the time series graph in their **Figure 6**. Except for a brief period where the mean-field method greatly underestimated the variance, its measurements are pretty close to those of sampling.

2.4 Non-Conjugate Regression



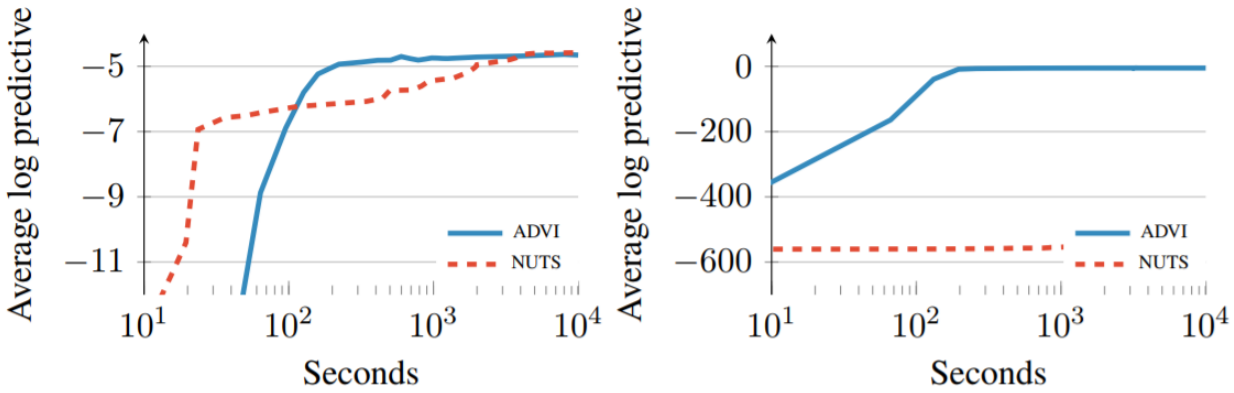
(a) Linear regression with ARD **(b)** Hierarchical logistic regression

Figure 10: Held-out predictive accuracy results | hierarchical generalized linear models on simulated and real data.

In this experiment, the authors examine the performance of ADVI with respect to benchmark algorithms (HMC and NUTS) on two non-conjugate regression models. The results in the figure above show that ADVI is able to converge to the same level of predictive accuracy as both benchmark algorithms in a significantly shorter amount of time. What's interesting to note is that $M = 1$ case (for ADVI) is performing better than the $M = 10$ case. This seems a little unintuitive because more data should generally help the model, not make it less helpful to use! Unfortunately, the authors don't expand on that observation.

In addition, it would have been useful to know what the variances (or certainties) were surrounding these average predictive accuracies. As mentioned in class, the mean alone does not tell the complete picture of how effective ADVI is. Furthermore, they don't actually mention which version of ADVI is used in those experiments, though given the complexity problems surrounding full-rank ADVI, it is likely that both ADVI graphs are for mean-field. If so, that goes to demonstrate how inefficient full-rank ADVI can be even in a moderate-dimension situation.

2.5 Non-negative Matrix Factorization

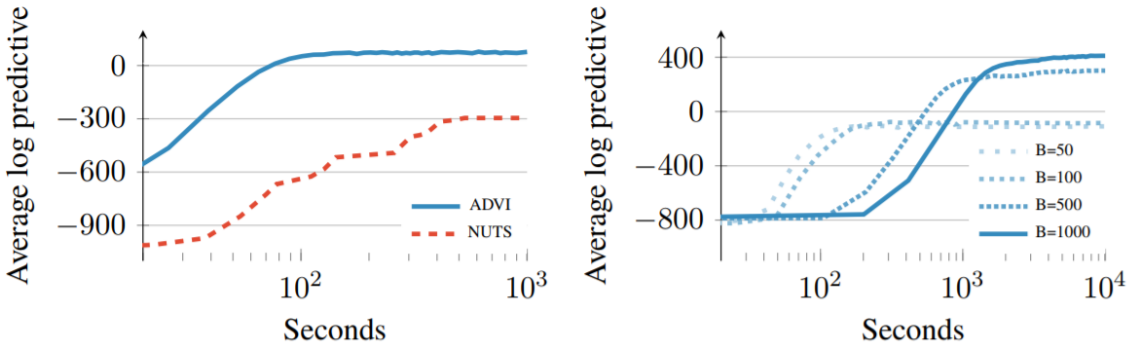


(a) Gamma Poisson predictive likelihood **(b)** Dirichlet exponential predictive likelihood

In this experiment, the authors examine the performance of ADVI with respect to benchmark algorithms (HMC and NUTS) on two non-conjugate matrix factorization models. The results in the figure above show that ADVI is able to converge to the same, if not better, level of predictive accuracy as the NUTS algorithm in a significantly shorter amount of time (HMC can't even produce good results on the same timescale).

Similar to the non-conjugate regression graphs, they did not share information regarding the variances in these averages nor did they clarify which version of ADVI they were running, both of which would have been useful in terms of evaluating ADVI's performance.

2.6 Gaussian Mixture Model



(a) Subset of 1000 images **(b)** Full dataset of 250 000 images

Figure 12: Held-out predictive accuracy results | GMM of the imageCLEF image histogram dataset. **(a)** ADVI outperforms NUTS (Hoffman and Gelman, 2014). **(b)** ADVI scales to large datasets by subsampling minibatches of size B from the dataset at each iteration (Hoffman et al., 2013).

In this experiment, the authors examine the performance of ADVI with respect to benchmark algorithms in the context of non-conjugate Gaussian mixture models analyzing the **imageclef** dataset. The results in the figure above show that ADVI is able to converge to a better level of predictive accuracy than the NUTS algorithm (HMC can't even produce good results on the same timescale). It also shows that when applied to the entire dataset, ADVI can use mini-batching to scale better than NUTS can (it isn't even plotted in the "b" figure).

Similar to the non-conjugate regression graphs, they did not share information regarding the variances in these averages nor did they clarify which version of ADVI they were running, both of which would have been useful in terms of evaluating ADVI's performance.

2.7 Taxi Trajectories

Figure 14: A visualization of fifty thousand randomly sampled taxi trajectories. The colors represent thirty Gaussian mixtures and the trajectories associated with each.

(a) Trajectories that take the inner bridges.

(b) Trajectories that take the outer bridges.

Figure 15: Two clusters using SUP-PPCA subspace clustering.

In this experiment, the authors examine taxi trajectories and use ADVI to extract a multi-dimensional subspace that can be used to represent all trajectories, after which mixture models can be used to cluster the them together. In this section, the authors tout the performance of ADVI on the dataset relative to that of HMC and NUTS but fail to mention which version is used (likely mean-field) as well as discuss the usefulness / correctness of these clusters beyond calling them "informative."

2.8 Takeaways

Although ADVI shows promise to provide more efficient variational inference algorithms, the results are not entirely convincing. First, there is still a distinct trade-off between correctness (full-rank) and performance (mean-field) that makes satisfying ALL of our specifications from **Section 1.2** difficult. Second, we don't know how consistent the algorithm is in terms of helping compute the optimal latent parameters. It could be that there is a lot of variance in the performance (or prediction accuracy), meaning that the algorithm could be unstable in some way.

References

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer, New York, NY.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). *Latent dirichlet allocation.* Journal of machine Learning research, 3 (Jan) : 993 − 1022

Hoffman, M. D., Blei, D. M., Wang C., and Paisley, J. *Stochastic Variational Inference.* Journal of Machine Learning Research, 13 (May) : 1303 − 1347

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., Blei, D. M. (2017) *Automatic*

Differentiation Variational Inference. Journal of Machine Learning Research, 17 (Jan) : 1 − 45

Turner, R. E. and Sahani, M. (2011). *Two problems with variational expectation maximisation for time series models*, pages 104 − 124. Cambridge University Press.