Variational Bayesian Methods For Learning Autoencoders

Heta S. Desai

Abstract

How can we efficiently perform inference and learning in directed probabilistic models when dealing with continuous latent variables and intractable posterior distributions, especially with large datasets? We propose a stochastic variational inference and learning algorithm that can handle large datasets and, under certain mild differentiability conditions, can even address intractable cases. Our contributions are twofold. First, we demonstrate that by reparameterizing the variational lower bound, we obtain a lower bound estimator that can be easily optimized using standard stochastic gradient methods. Second, we show that for independent and identically distributed (i.i.d.) datasets with continuous latent variables for each data point, posterior inference becomes particularly efficient by fitting an approximate inference model (or recognition model) to the intractable posterior using the proposed lower bound estimator. The theoretical benefits are supported by experimental results.

1. Introduction

How can we efficiently perform approximate inference and learning in directed probabilistic models with continuous latent variables and/or parameters, when their posterior distributions are intractable? The variational Bayesian (VB) approach optimizes an approximation to the intractable posterior, but the common mean-field method requires analytical solutions to expectations with respect to the approximate posterior, which are generally not feasible. We propose a reparameterization of the variational lower bound that leads to a simple, differentiable, unbiased estimator of the lower bound. This SGVB (Stochastic Gradient Variational Bayes) estimator allows for efficient approximate posterior inference in nearly any model with continuous latent variables and/or parameters, and can be optimized using standard stochastic gradient ascent techniques. For i.i.d. datasets with continuous latent variables per data point, we introduce the Auto-Encoding VB (AEVB) algorithm. The AEVB algorithm enhances inference and learning efficiency by utilizing the SGVB estimator to optimize a recognition model. This approach enables efficient approximate posterior inference through simple ancestral sampling, eliminating the need for costly iterative inference methods like MCMC per data point. The learned recognition model can also be applied to various tasks, such as recognition, denoising, representation learning, and visualization. When a neural network is used as the recognition model, this leads to the variational auto-encoder.

2. Method

This approach can be used to derive a lower bound estimator (a stochastic objective function) for various directed graphical models with continuous latent variables. In this section, we focus on the common scenario where we have an i.i.d. dataset with latent variables for each data point. The goal is to perform maximum likelihood (ML) or maximum a posteriori (MAP) inference on the global parameters, along with variational inference on the latent variables.

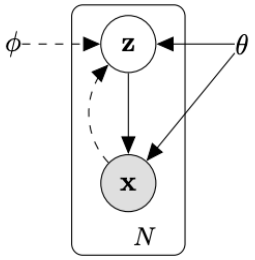


Figure 1: The type of directed graphical model under consideration. Solid lines denote the generative model $p_{\boldsymbol{\theta}}(\mathbf{z})p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, dashed lines denote the variational approximation $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ to the intractable posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$. The variational parameters $\boldsymbol{\phi}$ are learned jointly with the generative model parameters $\boldsymbol{\theta}$.

It's also possible to extend this to perform variational inference on the global parameters as well; this extended algorithm is provided in the appendix, though experiments on this are left for future research. While our method is applicable to online, non-stationary settings such as streaming data, we simplify the discussion here by assuming a fixed dataset.

2.1 Problem Scenario

Consider a dataset X={x(i)}i=1NX = \{x^{(i)}\}_{i=1}^{N}, consisting of NN i.i.d. samples from a continuous or discrete variable xx. We assume that the data is generated by a random process involving an unobserved continuous latent variable zz. The process unfolds in two steps: (1) a value z(i)z^{(i)} is drawn from a prior distribution pθ∗(z)p_{\theta^*}(z), and (2) a value x(i)x^{(i)} is drawn from a conditional distribution pθ∗(x|z)p_{\theta^*}(x|z). We assume that both the prior pθ∗(z)p_{\theta^*}(z) and likelihood pθ∗(x|z)p_{\theta^*}(x|z) belong to parametric families of distributions pθ(z)p_{\theta}(z) and pθ(x|z)p_{\theta}(x|z), and their probability density functions (PDFs) are differentiable with respect to both θ\theta and zz. However, a key challenge is that many aspects of this process remain hidden from us, including the true parameters θ∗\theta^* and the latent variables z(i)z^{(i)}.

Crucially, we do not make the common simplifying assumptions about marginal or posterior probabilities. Instead, our goal is to devise a general algorithm that remains efficient even in the case of:

1. **Intractability**: In situations where the marginal likelihood pθ(x)=∫pθ(z)pθ(x|z)dzp_{\theta}(x) = \int p_{\theta}(z) p_{\theta}(x|z) dz is intractable (making it impossible to evaluate or differentiate), where the true posterior density pθ(z|x)=pθ(x|z)pθ(z)pθ(x)p_{\theta}(z|x) = \frac{p_{\theta}(x|z) p_{\theta}(z)}{p_{\theta}(x)} is also intractable (thus excluding the use of algorithms like EM), and where the required integrals for mean-field variational Bayesian algorithms are likewise intractable. This is common in cases involving complex likelihood functions, such as a neural network with nonlinear hidden layers.
2. **Large Datasets**: When the dataset is large enough that batch optimization becomes too expensive, we aim to update parameters using small mini-batches or even single data points. Sampling-based methods, such as Monte Carlo EM, are typically too slow in this context, as they require an expensive sampling loop for each data point.

In this scenario, we address three related challenges:

1. **Efficient Approximate Maximum Likelihood (ML) or Maximum A Posteriori (MAP) Estimation** for the parameters θ\theta: The parameters themselves may be of interest, especially if we are analyzing a natural process. They can also help simulate the hidden random process and generate synthetic data that resembles the observed data.
2. **Efficient Approximate Posterior Inference** for the latent variable zz, given an observed value xx and a fixed set of parameters θ\theta: This is useful for tasks such as coding or data representation.
3. **Efficient Approximate Marginal Inference** for the variable xx: This is important for performing various inference tasks that require a prior over xx. Common applications in computer vision include tasks like image denoising, inpainting, and super-resolution.

To tackle the previously described challenges, we introduce a *recognition model* qϕ(z|x)q_\phi(z|x), which serves as an approximation to the intractable true posterior pθ(z|x)p_\theta(z|x). Unlike traditional mean-field variational inference, this approximation doesn't need to be factorial, and its parameters ϕ\phi are not derived from closed-form solutions of expectations. Instead, we present a method for learning the recognition model's parameters ϕ\phi alongside the generative model's parameters θ\theta.

From the viewpoint of coding theory, the latent variables zz can be interpreted as hidden representations or *codes*. Therefore, we often refer to the recognition model qϕ(z|x)q_\phi(z|x) as a *probabilistic encoder*, since it maps a given data point xx to a probability distribution (such as a Gaussian) over potential codes zz that could have generated xx. Similarly, the model pθ(x|z)p_\theta(x|z) is referred to as a *probabilistic decoder*, as it maps a code zz back to a distribution over possible values of the original input xx.

2.2 The variational bound

The marginal likelihood for the dataset can be broken down into a sum over the marginal likelihoods of individual data points:

$$\log p_\theta(x(1),\ldots,x(N)) = \sum_{i=1}^{N} \log p_\theta(x(i))$$

Each individual term can be expressed as:

$$\log p_\theta(x(i)) = D_{KL}(q_\varphi(z|x(i)) \| p_\theta(z|x(i))) + L(\theta, \varphi; x(i))$$

Here, the first term on the right is the Kullback-Leibler (KL) divergence between the approximate posterior and the true posterior, which is always non-negative. This makes the second term, $L(\theta, \varphi; x(i))$, a *lower bound* on the marginal likelihood, often called the **variational lower bound** or **ELBO** (Evidence Lower Bound). This can be rewritten as:

$$\log p_\theta(x(i)) \geq L(\theta, \varphi; x(i)) = E_{q_\varphi(z|x)}[-\log q_\varphi(z|x) + \log p_\theta(x, z)]$$

Or alternatively:

$$L(\theta, \varphi; x(i)) = -D_{KL}(q_\varphi(z|x(i)) \| p_\theta(z)) + E_{q_\varphi(z|x(i))} h \log p_\theta(x(i)|z)$$

The goal is to compute the gradients of this lower bound with respect to both the generative model parameters θ\thetaθ and the variational parameters ϕ for optimization. However, differentiating with respect to ϕ is particularly challenging. The typical Monte Carlo estimator for gradients in this scenario is:

$\nabla_\varphi E_{q_\varphi(z)}[f(z)] = E_{q_\varphi(z)}[f(z)\nabla_\varphi \log q_\varphi(z)] \simeq \frac{1}{L}\sum_{l=1}^{L} f(z)\nabla_\varphi \log q_\varphi(z^{(l)})$

This is often approximated as:

$z^{(l)} \sim q_\varphi(z|x^{(i)})$.

However, this estimator tends to have *very high variance*, making it impractical for use in most real-world scenarios.

2.3 The SGVB estimator and AEVB algorithm

Paraphrased:

In this section, we present a practical way to estimate the variational lower bound and compute its gradients with respect to the model parameters. We assume that the approximate posterior has the form $q_\varphi(z|x)$, though it's worth noting that the method is also applicable when the approximate posterior does not depend on the input x, i.e., $q_\varphi(z)$.

The fully variational Bayesian approach for learning a posterior distribution over the parameters is described in the appendix.

Given certain mild conditions (discussed in Section 2.4), we can reparameterize the sampled latent variable $z \sim q_\varphi(z|x)$ using a differentiable function $g_\varphi(\epsilon, x)$, where ϵ is a separate random noise variable. This transformation allows us to express the sampling process in a way that makes gradient-based optimization possible.

$z = g_\varphi(\epsilon, x)$ with $\epsilon \sim p(\epsilon)$

See section 2.4 for general strategies for choosing such an appropriate distribution $p(\epsilon)$ and function $g_\varphi(\epsilon, x)$. We can now form Monte Carlo estimates of expectations of some function $f(z)$ w.r.t. $q_\varphi(z|x)$ as follows:

$E_{q_\varphi(z|x^{(i)})}[f(z)] = E_{p(\epsilon)}[f(g_\varphi(\epsilon, x^{(i)}))] \simeq \frac{1}{L}\sum_{l=1}^{L} f(g_\varphi(\epsilon^{(l)}, x^{(i)}))$ where $\epsilon^{(l)} \sim p(\epsilon)$

We apply this technique to the variational lower bound (eq. (2)), yielding our generic Stochastic Gradient Variational Bayes (SGVB) estimator $\tilde{\mathcal{L}}^A(\theta, \varphi; x^{(i)}) \simeq \mathcal{L}(\theta, \varphi; x^{(i)})$:

$\tilde{\mathcal{L}}^A(\theta, \varphi; x^{(i)}) = \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x^{(i)}, z^{(i,l)}) - \log q_\varphi(z^{(i,l)}|x^{(i)})$

where $z^{(i,l)} = g_\varphi(\epsilon^{(i,l)}, x^{(i)})$ and $\epsilon^{(l)} \sim p(\epsilon)$

4 Related work

The wake-sleep algorithm [HDFN95] is, to the best of our knowledge, the only other on-line learning method in the literature that is applicable to the same general class of continuous latent variable models. Like our method, the wake-sleep algorithm employs a recognition model that approximates the true posterior. A drawback of the wake-sleep algorithm is that it requires a concurrent optimization of two objective functions, which together do not correspond to optimization of (a bound of) the marginal likelihood. An advantage of wake-sleep is that it also applies to models with discrete latent variables. Wake-Sleep has the same computational complexity as AEVB per datapoint.

Stochastic variational inference [HBWP13] has recently received increasing interest. Recently, [BJP12] introduced a control variate schemes to reduce the high variance of the na¨ıve gradient estimator discussed in section 2.1, and applied to exponential family approximations of the posterior. In [RGB13] some general methods, i.e. a control variate scheme, were introduced for reducing the variance of the original gradient estimator. In [SK13], a similar reparameterization as in this paper was used in an efficient version of a stochastic variational inference algorithm for learning the natural parameters of exponential-family approximating distributions.

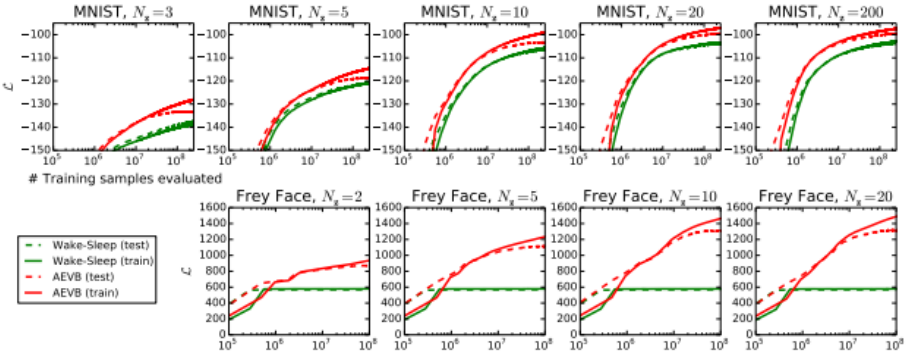
The AEVB algorithm exposes a connection between directed probabilistic models (trained with a variational objective) and auto-encoders. A connection between linear auto-encoders and a certain class of generative linear-Gaussian models has long been known. In [Row98] it was shown that PCA corresponds to the maximum-likelihood (ML) solution of a special case of the linear-Gaussian model with a prior $p(z) = N(0, I)$ and a conditional distribution $p(x|z) = N(x;Wz, I)$, specifically the case with infinitesimally small .

In relevant recent work on autoencoders [VLL+10] it was shown that the training criterion of unregularized autoencoders corresponds to maximization of a lower bound (see the infomax principle [Lin89]) of the mutual information between input X and latent representation Z. Maximizing (w.r.t. parameters) of the mutual information is equivalent to maximizing the conditional entropy, which is lower bounded by the expected loglikelihood of the data under the autoencoding model [VLL+10], i.e. the negative reconstrution error. However, it is well known that this reconstruction criterion is in itself not sufficient for learning useful representations [BCV13]. Regularization techniques have been proposed to make autoencoders learn useful representations, such as denoising, contractive and sparse autoencoder variants [BCV13]. The SGVB objective contains a regularization term dictated by the variational bound (e.g. eq. (10)), lacking the usual nuisance regularization hyperparameter required to learn useful representations. Related are also encoder-decoder architectures such as the predictive sparse decomposition (PSD) [KRL08], from which we drew some inspiration. Also relevant are the recently introduced Generative Stochastic Networks [BTL13] where noisy auto-encoders learn the transition operator of a Markov chain that samples from the data distribution. In [SL10] a recognition model was employed for efficient learning with Deep Boltzmann Machines.

These methods are targeted at either unnormalized models (i.e. undirected models like Boltzmann machines) or limited to sparse coding models, in contrast to our proposed algorithm for learning a general class of directed probabilistic models. The recently proposed DARN method [GMW13], also learns a directed probabilistic model using an auto-encoding structure, however their method applies to binary latent variables. Even more recently, [RMW14] also make the connection between auto-encoders, directed proabilistic models and stochastic variational inference using the reparameterization trick we describe in this paper. Their work was developed independently of ours and provides an additional perspective on AEVB.

5 Experiments

We trained generative models of images from the MNIST and Frey Face datasets3 and compared learning algorithms in terms of the variational lower bound, and the estimated marginal likelihood. The generative model (encoder) and variational approximation (decoder) from section 3 were used, where the described encoder and decoder have an equal number of hidden units. Since the Frey Face data are continuous, we used a decoder with Gaussian outputs, identical to the encoder, except that the means were constrained to the interval (0, 1) using a sigmoidal activation function at the



: Comparison of our AEVB method to the wake-sleep algorithm, in terms of optimizing the lower bound, for different dimensionality of latent space (Nz). Our method converged considerably faster and reached a better solution in all experiments. Interestingly enough, more latent variables does not result in more overfitting, which is explained by the regularizing effect of the lower bound. Vertical axis: the estimated average variational lower bound per datapoint. The estimator variance was small (< 1) and omitted. Horizontal axis: amount of training points evaluated. Computation took around 20-40 minutes per million training samples with a Intel Xeon CPU running at an effective 40 GFLOPS.

decoder output. Note that with hidden units we refer to the hidden layer of the neural networks of the encoder and decoder. Parameters are updated using stochastic gradient ascent where gradients are computed by differentiating the lower bound estimator $\nabla_{\theta,\varphi} L(\theta, \varphi; X)$ (see algorithm 1), plus a small weight decay term corresponding to a prior $p(\theta) = N(0, I)$. Optimization of this objective is equivalent to approximate MAP estimation, where the likelihood gradient is approximated by the gradient of the lower bound. We compared performance of AEVB to the wake-sleep algorithm [HDFN95]. We employed the same encoder (also called recognition model) for the wake-sleep algorithm and the variational autoencoder. All parameters, both variational and generative, were initialized by random sampling from N (0, 0.01), and were jointly stochastically optimized using the MAP criterion. Stepsizes were adapted with Adagrad [DHS10]; the Adagrad global stepsize parameters were chosen from {0.01, 0.02, 0.1} based on performance on the training set in the first few iterations. Minibatches of size M = 100 were used, with L = 1 samples per datapoint.

Likelihood lower bound We trained generative models (decoders) and corresponding encoders (a.k.a. recognition models) having 500 hidden units in case of MNIST, and 200 hidden units in case of the Frey Face dataset (to prevent overfitting, since it is a considerably smaller dataset). The chosen number of hidden units is based on prior literature on auto-encoders, and the relative performance of different algorithms was not very sensitive to these choices. Figure 2 shows the results when comparing the lower bounds. Interestingly, superfluous latent variables did not result in overfitting, which is explained by the regularizing nature of the variational bound. Marginal likelihood For very low-dimensional latent space it is possible to estimate the marginal likelihood of the learned generative models using an MCMC estimator. More information about the marginal likelihood estimator is available in the appendix. For the encoder and decoder we again used neural networks, this time with 100 hidden units, and 3 latent variables; for higher dimensional latent space the estimates became unreliable. Again, the MNIST dataset was used. The AEVB and Wake-Sleep methods were compared to Monte Carlo EM (MCEM) with a Hybrid Monte Carlo (HMC) [DKPR87] sampler; details are in the appendix. We compared the convergence speed for the three algorithms, for a small and large training set size. Results are in figure 3.

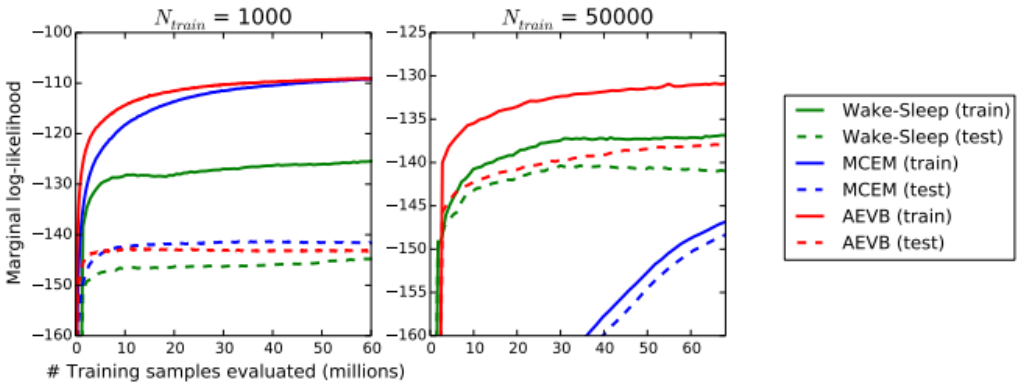


Figure 3: Comparison of AEVB to the wake-sleep algorithm and Monte Carlo EM, in terms of the estimated marginal likelihood, for a different number of training points. Monte Carlo EM is not an on-line algorithm, and (unlike AEVB and the wake-sleep method) can't be applied efficiently for the full MNIST dataset. Visualisation of high-dimensional data If we choose a low-dimensional latent space (e.g. 2D), we can use the learned encoders (recognition model) to project high-dimensional data to a lowdimensional manifold. See appendix A for visualisations of the 2D latent manifolds for the MNIST and Frey Face datasets.

6 Conclusion

We have introduced a novel estimator of the variational lower bound, Stochastic Gradient VB (SGVB), for efficient approximate inference with continuous latent variables. The proposed estimator can be straightforwardly differentiated and optimized using standard stochastic gradient methods. For the case of i.i.d. datasets and continuous latent variables per datapoint we introduce an efficient algorithm for efficient inference and learning, Auto-Encoding VB (AEVB), that learns an approximate inference model using the SGVB estimator. The theoretical advantages are reflected in experimental results.

7 Future

work Since the SGVB estimator and the AEVB algorithm can be applied to almost any inference and learning problem with continuous latent variables, there are plenty of future directions: (i) learning hierarchical generative architectures with deep neural networks (e.g. convolutional networks) used for the encoders and decoders, trained jointly with AEVB; (ii) time-series models (i.e. dynamic Bayesian networks); (iii) application of SGVB to the global parameters; (iv) supervised models with latent variables, useful for learning complicated noise distributions.

References

[BCV13] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. 2013.

[BJP12] David M Blei, Michael I Jordan, and John W Paisley. Variational Bayesian inference

with Stochastic Search. In Proceedings of the 29th International Conference on Machine Learning (ICML-12), pages 1367–1374, 2012.

[BTL13] Yoshua Bengio and Eric Thibodeau-Laufer. Deep generative stochastic networks train- ´able by backprop. arXiv preprint arXiv:1306.1091, 2013.

[Dev86] Luc Devroye. Sample-based non-uniform random variate generation. In Proceedings of the 18th conference on Winter simulation, pages 260–265. ACM, 1986.

[DHS10] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12:2121–2159, 2010.

[DKPR87] Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid monte carlo. Physics letters B, 195(2):216–222, 1987.

[GMW13] Karol Gregor, Andriy Mnih, and Daan Wierstra. Deep autoregressive networks. arXiv preprint arXiv:1310.8499, 2013.

[HBWP13] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. The Journal of Machine Learning Research, 14(1):1303–1347, 2013.

[HDFN95] Geoffrey E Hinton, Peter Dayan, Brendan J Frey, and Radford M Neal. The" wakesleep" algorithm for unsupervised neural networks. SCIENCE, pages 1158–1158, 1995.

[KRL08] Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. Fast inference in sparse

coding algorithms with applications to object recognition. Technical Report CBLLTR-2008-12-01, Computational and Biological Learning Lab, Courant Institute, NYU,2008.

[Lin89] Ralph Linsker. An application of the principle of maximum information preservation to linear systems. Morgan Kaufmann Publishers Inc., 1989.

[RGB13] Rajesh Ranganath, Sean Gerrish, and David M Blei. Black Box Variational Inference.arXiv preprint arXiv:1401.0118, 2013.

[RMW14] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and variational inference in deep latent gaussian models. arXiv preprint arXiv:1401.4082, 2014.

[Row98] Sam Roweis. EM algorithms for PCA and SPCA. Advances in neural information processing systems, pages 626–632, 1998.

[SK13] Tim Salimans and David A Knowles. Fixed-form variational posterior approximation through stochastic linear regression. Bayesian Analysis, 8(4), 2013.

[SL10] Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep boltzmann machines. In International Conference on Artificial Intelligence and Statistics, pages 693–700, 2010.

[VLL+10] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine

Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. The Journal of Machine Learning Research