
 Infer.java project	TIN2015-74368-JIN 01/02/2017	
--	--	---

Reference: **TIN2015-74368-JIN**

Project full title: **INFER.JAVA: UN LENGUAJE DE PROGRAMACION. PROBABILISTICO PARA EL DESARROLLO DE APLICACIONES**

Deliverable no.: **E1**

Title of the deliverable: **State of the Art and Functional Requirements**

Author(s):	Andrés R. Masegosa, Rafael Cabañas de Paz
Version:	0.1

An Introduction to Probabilistic Modeling with Deep Neural Networks

Abstract

Recent advances in variational inference are significantly expanding the toolbox of probabilistic modeling. Historically, variational inference (and probabilistic modeling) has been restricted to small or medium data sets which fit within the main memory of the computer, and to distributions families belonging to the conjugate exponential family, where variational updating equations can be computed in closed-form. Two main advances are helping variational methods to overcome these restrictions: (i) scalable variational methods based on stochastic gradient descent and distributed computation engines allow to train probabilistic models on massive data sets, and (ii) novel Monte-Carlo based gradient estimation techniques allow to perform inference over general probabilistic models far beyond the exponential family. The main practical consequence is the possibility to include deep neural networks within a probabilistic model (and make inference) to model complex non-linear stochastic relationships between random variables. This greatly expands the scope of application of probabilistic models and allows to integrate many of the successful advances obtained by the deep learning community in recent years.

1 Introduction

The seminal works of Judea Pearl and Stephen Larutizen about probabilistic graphical models (PGMs) placed probabilistic modeling as a indispensable tool for dealing with many problems involving any form of uncertainty within many different fields such as artificial intelligence, statistics, data mining, machine learning, etc. PGMs has been present for the last 30 years becoming a well established and highly influential body of research.

At the same time, the inference problem, as the problem of computing the posterior probability over hidden quantities given the known evidence, has been the corner-stone (and the bottleneck) of the feasibility and applicability of PGMs. There have been many advances in the inference part during this time.

At the beginning, the first proposed inference algorithms were able to compute this posterior in a exact way by exploiting the conditional independence relationships encoded by the graphical structure of the model. Even though, model's probability distributions were strongly restricted (i.e. multinomial and conditional linear Gaussian distributions). Quickly researchers realized these exact inferences schemes were not

powerful enough to deal with complex stochastic dependency structures that arise in different fields due to the high computational costs of these inference algorithms. Then, approximated inference methods were started to be applied.

Monte-Carlo methods were one of the first approximate methods employed to make inference over complex PGMs. They are extremely powerful and able to approximate complex posterior distributions. However, they have serious issues like problems of convergence of the underlying Markov chain, poor mixing, etc. when having to approximate highly dimensional posteriors. And computing these highly-dimensional posteriors started to be relevant in many domains, specially when researchers seek to apply a Bayesian approach to learn the parameters of their PGMs from data. In this case, the learning problem reduces to compute the posterior probability over the parameters of the model. For models with a large number of parameters, the application of Monte-carlo methods become infeasible. And these issues gave rise to the development of alternative approximate inference schemes.

Belief propagation (BP), and the close scheme called Expectation propagation (EP), has been successfully used in many applications of PGMs helping to overcome many of the limitations of Monte-carlo methods. They are approximate deterministic inference techniques which can be implemented using message-passing scheme which exploits the graph structure of the PGM and, hence, the underlying conditional independence relationships among variables. In terms of distributional assumptions, BP was mainly restricted to multinomial and Gaussian distributions, while EP allows for a more general family of distributions although restricted by the need to define a non-trivial quotient operation between the involved densities. As already commented, these techniques (an many variations also published later) are deterministic and overcame some of the difficulties of Monte-carlo methods. However they presented two main issues: they did not guarantee the convergence to an approximate and meaningful solution; and did not scale to the kind of models that appear in the context of Bayesian learning (i.e. plateau like models). Again, these issues motivated researchers to look into alternative approximate inference schemes.

Variational methods (Wainwright et al., 2008) were firstly explored within the Michael Jordan's lab in the late 90s, inspired by their successful application in inference problems encountered in statistical physics. They are deterministic approximate inference techniques like BP and EP methods. Their main innovation comes from casting the inference problem as the problem of maximizing a well defined loss function (i.e. the ELBO function) acting as an inference proxy. In general, variational methods guarantee convergence to a local minimum of this ELBO function, and, then, to a meaningful solution. By transforming the inference problem in a continuous optimization problem, variational methods could take advantage of recent advances in continuous optimization theory. That was the case of the widely adopted stochastic gradient descent algorithm, which was successfully used by the machine learning community to scale their learning algorithms to big data sets. This same learning algorithm was adapted to the variational inference problem by Blei et al. (), giving the opportunity to apply probabilistic modeling approaches to problems involving massive data sets. But, in terms of distributional assumptions, VI methods were tightly restricted to the conjugate exponential family, where ELBO's gradients can be computed in closed-form. Ad-hoc approaches were developed over the years for specific models outside the exponential

family, but no general approaches were available until recently.

In this paper we review the recent developments in variational inference methods that are allowing to apply probabilistic modeling far beyond the conjugate exponential family and over massive data sets. The main practical advance coming these developments is the possibility of introducing deep neural networks to define complex highly non-linear dependency relationships among random variables. This is greatly expanding the scope of application of probabilistic modeling.

The rest of paper is structured in three main sections. In the first section, we start by briefly revising conjugate exponential family models and highlighting that these models are essentially able to model only linear dependency relations among random variables. We then revise recent works that show how deep neural networks can be used within a probabilistic model to capture non-linear relationships, which are frequent in many real world problems like image processing, natural language processing, etc. We also review a new family of probabilistic programming languages which allow to express universal computable probability distributions using DNNs and which rely on recently release deep learning libraries like TensorFlow and PyTorch. In the second section, we review the new variational inference methods that power the inference engine of this powerful class of probabilistic models. We start by revising the main techniques used to compute the gradient of the ELBO function for this class of general models. We then discussed recent works that show how to scale these inference techniques to massive data sets. Recent works improving the quality of the approximations of this new family of variational inference methods are also review. We finished summarizing and pointing to future research directions within this active field of research.

2 Probabilistic Modeling within the Conjugate Exponential Family

In the first subsection we introduce notation and present conjugate exponential family and latent variable models (LVMs). In the second subsection, we detail how to apply variational inference methods to fit LVMs from data. Finally, we show how to scale variational methods to learn LVMs over massive data sets.

2.1 Latent Variable Models

The conjugate exponential family (Barndorff-Nielsen, 2014) has been largely studied in the statistics field and cover a very wide and widely used range of probability distributions and density functions such as Multinomial, Normal, Gamma, Dirichlet, Beta, etc. They have been largely used by the machine learning community (Bishop, 2006; Koller & Friedman, 2009; Murphy, 2012) to exploit many of their nice properties for parameter learning and inference tasks.

In our case, we focus on probabilistic models with the structure shown in Figure 1 belonging to the conjugate exponential family. These kind of models are also known as latent variable models (LVMs) (Bishop, 1998; Blei, 2014). LVMs are widely used probabilistic models which tries to uncover hidden patterns in our data set. These

hidden patterns are modeled by means of a set of global, denoted by β , and local stochastic random variables, denoted by z , which can not be observed. The observed data, denoted by x , is assumed to be generated from stochastic random variables whose distribution is conditioned to both the local and global hidden variables. A vector of fixed (hyper) parameters denoted by α is also included in this kind of models.

The joint distribution of this probabilistic model factorizes into a product of local terms and a global term,

$$p(x, z, \beta | \alpha) = p(\beta | \alpha) \prod_{i=1}^N p(x_i, z_i | \beta).$$

As the model is assumed to belongs to the conjugate exponential family, the functional form of the conditional distributions of the model are specified as follows,

$$\begin{aligned} \ln p(\beta | \alpha) &= \ln h(\beta) + \alpha^T t(\beta) - a_g(\alpha) \\ \ln p(z_i | \beta) &= \ln h(z_i) + \eta_z(\beta)^T t(z_i) - a_z(\eta_z(\beta)) \\ \ln p(x_i | z_i, \beta) &= \ln h(x_i) + \eta_x(z_i, \beta)^T t(x_i) - \eta_x(z_i, \beta), \end{aligned} \quad (1)$$

where the scalar functions $h(\cdot)$ and $a_l(\cdot)$ are the base measure and the log-normalizer, respectively; the vector function $t(\cdot)$ is the *sufficient statistics* vector.

By using properties of the conjugate exponential family, we can also derived the functional form of the joint distributions over the local variables (z_i, x_i) given the global parameters β ,

$$\ln p(x_i, z_i | \beta) = \ln h(x_i, z_i) + \beta^T t(x_i, z_i) - a_l(\beta), \quad (2)$$

Another standard assumption (Hoffman et al., 2013) in this kind of models is that the complete conditional forms of the latent variables given the observations and the other latent variables can also be expressed in exponential family form,

$$\begin{aligned} \ln p(\beta | x, z) &= h(\beta) + \eta_g(x, z)^T t(\beta) - a_g(\eta_g(x, z)) \\ \ln p(z_i | x_i, \beta) &= h(z_i) + \eta_l(x_i, \beta)^T t(z_i) - a_l(\eta_l(x_i, \beta)). \end{aligned} \quad (3)$$

By conjugacy properties, the natural parameter of the global posterior $\eta_g(x, z)$ can be expressed as,

$$\eta_g(x, z) = \alpha + \sum_{i=1}^N t(x_i, z_i) \quad (4)$$

Example 1 Principal Component Analysis (PCA) is a classic statistical technique for dimensionality reduction. It maps a D dimensional point x to a K dimensional latent representation z through a affine matrix of dimensions $D \times K$, β . A simplified probabilistic view of PCA (Tipping & Bishop, 1999) can be describe as follows,

$$\begin{aligned} \beta &\sim \mathcal{N}_{D \times K}(0, \mathbf{I}\sigma_\beta) \\ z_i &\sim \mathcal{N}_K(0, \mathbf{I}\sigma_z) \\ x_i &\sim \mathcal{N}_D(\beta^T z_i, \mathbf{I}\sigma_x), \end{aligned}$$

where $\alpha = (\sigma_\beta, \sigma_z, \sigma_x)$ are the hyperparameters of the model.

This model is a LVM where β acts as the global hidden variable and z_i is the local hidden variables associated to the sample x_i . It belongs to the conjugate exponential family because all the conditionals satisfy Equation (??) with the following natural parameters,

$$\eta_\beta(\alpha) = \begin{bmatrix} \frac{1}{\sigma_\beta} \mathbf{0} \\ \frac{-1}{2\sigma_\beta} \mathbf{I} \end{bmatrix} \quad \eta_{x_i}(\beta, z_i) = \begin{bmatrix} \frac{1}{\sigma_x} \beta^T z_i \\ \frac{-1}{2\sigma_x} \mathbf{I} \end{bmatrix}$$

The main limitation of this model is the assumption of a linear relationship between the hidden and the observed variables.

LVMs include popular models like LDA (Blei et al., 2003) models to uncover the hidden topics in a text corpora, mixture of Gaussian models to discover hidden clusters in our data (Bishop, 2006), probabilistic principal component analysis for revealing a low-dimensional representation of the data (Tipping & Bishop, 1999), models with hierarchical latent variables to capture the drift in a data stream (Borchani et al., 2015; Masegosa et al., 2017a), etc. Many machine-learning books contain entire sections devoted to them (Bishop, 2006; Koller & Friedman, 2009; Murphy, 2012).

2.2 Mean-Field Variational Inference

The problem of Bayesian inference reduces to compute the posterior over the unknown quantities given the observations,

$$p(\beta, z|x) = \frac{p(x|z, \beta)p(z|\beta)p(\beta)}{\int p(x|z, \beta)p(z|\beta)p(\beta)dzd\beta}. \quad (5)$$

Computing the above posterior is usually intractable for many interesting models because it requires to solve a highly-multidimensional integral. As commented in the introduction, VI methods are one of the best performing options to address this problem. In this section we revise the main ideas behind this approach.

Variational inference is a deterministic technique for finding tractable posterior distributions, denoted by q , which approximates the Bayesian posterior, $p(\beta, z|x)$, that is often intractable to compute. More specifically, by letting \mathcal{Q} be a set of possible approximations of this posterior, variational inference solves the following optimization problem for any model in the conjugate exponential family:

$$\min_{q(\beta, z) \in \mathcal{Q}} KL(q(\beta, z) || p(\beta, z|x)), \quad (6)$$

where KL denotes the Kullback-Leibler divergence between two probability distributions.

In the *mean field variational* approach the approximation family \mathcal{Q} is assumed to fully factorize. Extending the notation of Hoffman et al. (2013), we have that

$$q(\beta, z|\lambda, \phi) = q(\beta|\lambda) \prod_{i=1}^N q(z_i|\phi_i).$$

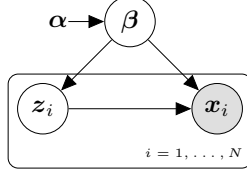


Figure 1: Core of the probabilistic model examined in this paper. See Section ?? for details.

Furthermore, each factor variational distribution is assumed to belong the same family as the model’s complete conditionals,

$$\begin{aligned}\ln q(\beta|\lambda) &= h(\beta) + \lambda^T t(\beta) - a_g(\lambda) \\ \ln q(z_i|\phi_i) &= h(z_i) + \phi_i^T t(z_i) - a_l(\phi_i).\end{aligned}$$

As can be seen, λ parameterizes the variational distribution of β , while ϕ has the same role for the variational distribution of z .

To solve the minimization problem in Equation (6), the variational approach exploits the transformation

$$\ln P(x) = \mathcal{L}(\lambda, \phi) + KL(q(\beta, z|\lambda, \phi)||p(\beta, z|x)), \quad (7)$$

where $\mathcal{L}(\cdot|\cdot)$ is a *lower bound* of $\ln P(x)$ since KL is non-negative. As $\ln P(x)$ is constant, minimizing the KL term is equivalent to maximizing the lower bound. Variational methods maximize this lower bound by using gradient based methods.

The functional form of the \mathcal{L} function can expressed as follows,

$$\mathcal{L}(\lambda, \phi) = \mathbb{E}_q[\ln p(x, z, \beta)] - \mathbb{E}_q[\ln q(\beta, z|\lambda, \phi)] \quad (8)$$

The key advantage of having a conjugate exponential model is that the gradients of the \mathcal{L} function can be always computed in closed form (Winn & Bishop, 2005). The natural gradients¹ with respect to the variational parameters λ and ϕ can be computed as follows,

$$\begin{aligned}\nabla_{\lambda}^{nat} \mathcal{L} &= \alpha + \sum_{i=1}^N \mathbb{E}_{\phi_i}[t(x_i, z_i)] - \lambda \\ \nabla_{\phi_i}^{nat} \mathcal{L} &= \mathbb{E}_{\lambda}[\eta_l(x_i, \beta)] - \phi_i,\end{aligned} \quad (9)$$

where $\mathbb{E}_{\phi_i}[\cdot]$ and $\mathbb{E}_{\lambda}[\cdot]$ denotes expectations with respect to $q(z_i|\phi_i)$ and $q(\beta|\lambda)$, respectively.

From the above gradients we can derive a coordinate ascent algorithm to optimize the ELBO function with the following coordinate ascent rules,

$$\begin{aligned}\lambda^* &= \alpha + \sum_{i=1}^N \mathbb{E}_{\phi_i}[t(x_i, z_i)] \\ \phi_i^* &= \mathbb{E}_{\lambda}[\eta_l(x_i, \beta)].\end{aligned} \quad (10)$$

¹Natural gradient is ...

Example 2 For the PCA model depicted in Example 1, the variational distributions would be $q(\beta|\lambda) = \prod_{i=1}^D \mathcal{N}_K(\mu_{\beta,i}, \Sigma_{\beta,i})$ and $q(z|\phi) = \prod_{i=1}^N \mathcal{N}_K(\mu_{z,i}, \Sigma_{z,i})$. And the gradient wrt to the variational parameters shown in Equation (14) can be computed by using the following equality,

2.3 Scalable Varitional Inference

Performing variational inference in big data sets (i.e. when N is a very large number) raises many challenges. Firstly, the model itself may not fit in memory, and, secondly, computing the ELBO's gradient wrt λ depends linearly on the size of the data set, which can be prohibitively expensive in this case. The most popular method (Hoffman et al., 2013) for addressing these issues, and which are able to scale VI to massive data sets, relies on stochastic optimization techniques (Bottou, 2010; Robbins & Monro, 1951). The method is called *stochastic variational inference*. The key idea behind this method is to compute noise and unbiased estimates of the ELBO's gradient, denoted by $\hat{\nabla}_{\lambda}^{nat} \mathcal{L}$, by randomly selecting a mini-batch of M data samples,

$$\hat{\nabla}_{\lambda}^{nat} \mathcal{L} = \alpha + \frac{N}{M} \sum_{m=1}^M \mathbb{E}_{\phi_i^*} [t(\mathbf{x}_{i_m}, \mathbf{z}_{i_m})] - \lambda, \quad (11)$$

where i_m is the variable index from the subsampled mini-batch. This is an unbiased estimate because $\mathbb{E}[\hat{\nabla}_{\lambda}^{nat} \mathcal{L}] = \nabla_{\lambda}^{nat} \mathcal{L}$. Then, the ELBO is maximized using an stochastic gradient ascent method,

$$\lambda^{t+1} = \lambda^t + \rho_t \hat{\nabla}_{\lambda}^{nat} \mathcal{L}(\lambda^t). \quad (12)$$

If the learning rate ρ_t satisfies the Robbins-Monro conditions (i.e. $\sum_{t=1}^{\infty} \rho_t = \infty$ and $\sum_{t=1}^{\infty} \rho_t^2 < \infty$), the above updating equation is guaranteed to converge to a stationary point of the ELBO function.

The size of the mini-batch is chosen to be $S \ll M$ to reduce the computational complexity of computing the gradient, and with $S > 1$ in order to reduce the variance in the estimate of the gradient. The optimal value is used to be problem dependent.

Alternative ways to scale up variational inference in conjugate exponential models involve the use of distributed computing clusters. For example, in (Masegosa et al., 2017b) the data set is assumed to be stored among different machines. Then the problem of computing the ELBO's gradient given in Equation (14) is scaled up by distributing the computation of the $\mathbb{E}_q[\eta_{\beta}(\mathbf{x}_i, \mathbf{z}_i)]$ terms. So each machine computes this term for those samples that are locally stored. Finally, all the terms are sent to a master node which aggregates them and compute the gradient.

3 Probabilistic Models with Deep Neural networks

Specify the range of LVMS covered by these approach.

LVMS with DNNs can be found in the literature with different names. Deep generative models refers to the capacity of these models to generate data samples using

probabilistic constructs that include DNNs, although under this label can be found models which does not fall inside this category.

If LVMs were usually restricted to be inside the conjugate exponential family, because only in this case inference was feasible, the introduction of VAE (and the inference techniques that we will review in Section 4), has inspired many recent work about extending LVMs with DNNs. (Johnson et al., 2016) contains different examples of this approach an extends Gaussian mixture models, latent linear dynamical systems and latent switching linear dynamical systems with non-linear relationships modeled by DNNs. Other approaches has been proposed along these lines like Gaussian Mixture Variational Autoencoders (Dilokthanakul et al., 2016) which a variant of VAE with a Gaussian mixture as a prior distribution. Recurrent Hidden Semi-Markov Model (Linderman et al., 2016) extends hidden semi-Markov models with recurrent neural networks. LDA models for uncovering topics in text data were also recently extended in (Card et al., 2017) following similar ideas.

We consider here a deep neural network (Goodfellow et al., 2016) as a deterministic non-linear function, denoted by $f(\cdot; \theta)$, and which is parameterized by a vector θ .

A *latent variable models with DNNs* has the general structure provided in Figure 1, with the following conditional distributions forms,

$$\begin{aligned}\ln p(\beta|\alpha) &= \ln h(\beta) + \alpha^T t(\beta) - a_g(\alpha) \\ \ln p(z_i|\beta) &= \ln h(z_i) + \eta_z(\beta)^T t(z_i) - a_z(\eta_z(\beta)) \\ \ln p(x_i|z_i, \beta, \theta) &= \ln h(x_i) + \eta_x(f(z_i, \beta; \theta))^T t(x_i) - a_x(f(z_i, \beta; \theta))\end{aligned}\quad (13)$$

The main difference with respect to a standard LVMs described in Section 2.1 is the conditional distribution over the observations x_i given the local hidden variables z_i and the global parameters β . Now this conditional relationship is governed by the DNN encoded in the function f , which parameterized by a parameter vector θ ².

Models with conditional distributions as defined in Equation (13) does not belong to the exponential family because they can not be represented in the forms given by Equations (2) and (3). In consequence, the variational inference techniques detailed in Sections 2.2 and 2.3 do not apply here.

Example 3 A popular example of LVMs with DNNs are Variational Autoencoders (VAE) (Kingma & Welling, 2013). The generative part of this model can be described as follows,

$$\begin{aligned}z_i &\sim \mathcal{N}_K(0, \mathbf{I}\sigma_z) \\ x_i &\sim \mathcal{N}_D(\mu(z_i; \theta_\mu), \sigma(z_i; \theta_\sigma)),\end{aligned}$$

where $\mu(\cdot; \theta_\mu)$ and $\sigma(\cdot; \theta_\sigma)$ are DNNs functions with input z_i and output the mean vector and the covariance matrix of x_i , respectively.

This model is quite similar to the PCA model presented in Example 1. The main difference comes from the conditional distribution of x_i . In the PCA model, the mean of the Normal distribution of x_i depends linearly of z_i .

²This parameter vector is not assumed to be a random variable.

In the VAE model, the mean (and the variance) depends of z_i through a DNN parametrized by θ .

A VAE falls inside the general models class described in this section. In this case, $\alpha = \sigma_z$, $\beta = \emptyset$ and $p(z_i|\beta)$ is a Normal distribution which belongs to the exponential family. Finally, the mean and the variance (and, hence, the natural parameters) of the Normal distribution of x_i depends of z_i through a DNN with parameters $\theta = (\theta_\mu, \theta_\sigma)$.

4 Variational Inference with Deep Neural Networks

4.1 Black Box Variational Inference

When a probabilistic model contains DNNs the variational scheme presented in the previous section applies and the inference problem can be casted as an optimization problem. The main issue is that the model does not belong to the conjugate exponential family, and gradients can not be computed in closed form as shown in Equation (14).

The main challenge now is how to compute the gradient of the \mathcal{L} function (see Equation (8)) which involves computing the gradient with respect to an expectation,

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \frac{\partial \mathbb{E}_{q(\beta, z | \lambda, \phi)} [\ln p(x, z, \beta) - \ln q(z, \beta | \lambda, \phi)]}{\partial \lambda} \quad (14)$$

In the next section we review the main methods available today to compute this gradient.

4.1.1 Pathwise Gradients

Kingma et al Kingma & Welling (2013) introduced this technique to compute the \mathcal{L} 's gradients when they presented VAE. This technique tries to leverage the gradient of the model in order to better navigate trough the optimization space. But it only applies when the latent variables of the model are differentiable and can be reparametrized. A distribution $q(y|\lambda)$ is reparametrizable if it can be expressed as follows,

$$\begin{aligned} \epsilon &\sim q(\epsilon) \\ w &= w(\epsilon; \lambda) \end{aligned} \quad (15)$$

where ϵ does not depend of the λ parameter. $w(\cdot; \lambda)$ is a deterministic function which encapsulates the dependence of w with respect to λ . The problem is that only few distributions has this property. The most used one is the Normal distribution, $w \sim \mathcal{N}(\mu, \Sigma)$ can be reparametrized as $\epsilon \sim \mathcal{N}(0, I)$ and $w = \mu + L\epsilon$ where $\Sigma = LL^T$.

By exploiting this reparametrization property we can reexpress the \mathcal{L} 's gradient of Equation (14) as follows,

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \mathbb{E}_{q(\epsilon)} \left[\frac{\partial (\ln p(x, z, \beta(\epsilon; \lambda)) - \ln q(z, \beta(\epsilon; \lambda) | \lambda, \phi))}{\partial \lambda} \right] \quad (16)$$

By using Equation (16), we can derived a Monte-carlo estimator which compute unbiased estimates of the gradient by sampling from $q(\epsilon)$. The inner gradient inside the expectation can be computed because we assume hidden variables are differentiable.

Other works have extended this work to distributions which are not directly reparametrizable. For example, Naesseth et al. (2017); Ruiz et al. (2016) blablabla .

This technique does not work in case we can not find a suitable reparametrization for the involved distributions and, also, in the case the either the log-joint density function $\ln p(\mathbf{x}, \mathbf{z}, \boldsymbol{\beta})$ or the variational distribution $\ln q(\mathbf{z}, \boldsymbol{\beta}|\boldsymbol{\phi}, \boldsymbol{\lambda})$ are not differentiable. A common case comes up when the latent variables are discrete.

4.1.2 Score Function Gradients

Ranganath et al. Ranganath et al. (2014) were the first ones to approach the problem of a general variational method in the presence of non-differentiable models. They approached this problem by using a previously known technique to compute the gradient of an expectation function, score functions gradients [1]. The idea is to leverage a property of logarithms to rewrite the gradient of Equation (14) as follows,

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = \mathbb{E}_q\left[\frac{\partial \ln q(\boldsymbol{\beta}, \mathbf{z}|\boldsymbol{\lambda}, \boldsymbol{\phi})}{\partial \boldsymbol{\lambda}} (\ln p(\mathbf{x}, \mathbf{z}, \boldsymbol{\beta}) - \ln q(\mathbf{z}, \boldsymbol{\beta}|\boldsymbol{\lambda}, \boldsymbol{\phi}))\right] \quad (17)$$

As be seen, a new term appears no in the expression, $\frac{\partial \ln q(\boldsymbol{\beta}, \mathbf{z}|\boldsymbol{\lambda}, \boldsymbol{\phi})}{\partial \boldsymbol{\lambda}}$, which is known in the statistics literature as the score function. Similarly to the previous case, we can use Equation (17) to derive a Monte-carlo estimator which compute unbiased estimates of the gradient by sampling from $q(\boldsymbol{\beta}, \mathbf{z}|\boldsymbol{\lambda}, \boldsymbol{\phi})$. In opposite to the previous case, the only restriction that this method have is that the join-log-likelihood term $\ln p(\mathbf{x}, \mathbf{z}, \boldsymbol{\beta})$ can be evaluated and that $\ln q(\mathbf{z}, \boldsymbol{\beta}|\boldsymbol{\phi}, \boldsymbol{\lambda})$ is differentiable.

In spite of its generality, the main drawback of this approach is the high-variance usually associated to its gradient estimates, which make it an unfeasible approach in many relevant settings. Recent works try to address this issue. Blablablabla....

Acknowledgements

This research has been funded by the Spanish Ministry of Economy and Competitiveness, through the project TIN2015-74368-JIN and by ERDF funds.

References

- Barndorff-Nielsen, Ole. *Information and exponential families: in statistical theory*. John Wiley & Sons, 2014.
- Bishop, Christopher M. Latent variable models. In *Learning in graphical models*, pp. 371–403. Springer, 1998.
- Bishop, Christopher M. *Pattern recognition and machine learning*. springer, 2006.

- Blei, David M. Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1:203–232, 2014.
- Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.
- Borchani, Hanen, Martínez, Ana M, Masegosa, Andrés R, Langseth, Helge, Nielsen, Thomas D, Salmerón, Antonio, Fernández, Antonio, Madsen, Anders L, and Sáez, Ramón. Modeling concept drift: A probabilistic graphical model based approach. In *International Symposium on Intelligent Data Analysis*, pp. 72–83. Springer, 2015.
- Bottou, Léon. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT’2010*, pp. 177–186. Springer, 2010.
- Card, Dallas, Tan, Chenhao, and Smith, Noah A. A neural framework for generalized topic models. *arXiv preprint arXiv:1705.09296*, 2017.
- Dilokthanakul, Nat, Mediano, Pedro AM, Garnelo, Marta, Lee, Matthew CH, Salimbeni, Hugh, Arulkumaran, Kai, and Shanahan, Murray. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.
- Goodfellow, Ian, Bengio, Yoshua, Courville, Aaron, and Bengio, Yoshua. *Deep learning*, volume 1. MIT press Cambridge, 2016.
- Hoffman, Matthew D., Blei, David M., Wang, Chong, and Paisley, John. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.
- Johnson, Matthew, Duvenaud, David K, Wiltchko, Alex, Adams, Ryan P, and Datta, Sandeep R. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pp. 2946–2954, 2016.
- Kingma, Diederik P and Welling, Max. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Koller, Daphne and Friedman, Nir. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
- Linderman, Scott W, Miller, Andrew C, Adams, Ryan P, Blei, David M, Paninski, Liam, and Johnson, Matthew J. Recurrent switching linear dynamical systems. *arXiv preprint arXiv:1610.08466*, 2016.
- Masegosa, Andrés, Nielsen, Thomas D, Langseth, Helge, Ramos-Lopez, Dario, Salmerón, Antonio, and Madsen, Anders L. Bayesian models of data streams with hierarchical power priors. *arXiv preprint arXiv:1707.02293*, 2017a.
- Masegosa, Andrés R, Martinez, Ana M, Langseth, Helge, Nielsen, Thomas D, Salmerón, Antonio, Ramos-López, Darío, and Madsen, Anders L. Scaling up bayesian variational inference using distributed computing clusters. *International Journal of Approximate Reasoning*, 2017b.

- Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- Naesseth, Christian, Ruiz, Francisco, Linderman, Scott, and Blei, David. Reparameterization gradients through acceptance-rejection sampling algorithms. In *Artificial Intelligence and Statistics*, pp. 489–498, 2017.
- Ranganath, Rajesh, Gerrish, Sean, and Blei, David. Black box variational inference. In *Artificial Intelligence and Statistics*, pp. 814–822, 2014.
- Robbins, Herbert and Monro, Sutton. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.
- Ruiz, Francisco R, AUEB, Michalis Titsias RC, and Blei, David. The generalized reparameterization gradient. In *Advances in Neural Information Processing Systems*, pp. 460–468, 2016.
- Tipping, Michael E and Bishop, Christopher M. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- Wainwright, Martin J, Jordan, Michael I, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2): 1–305, 2008.
- Winn, John M. and Bishop, Christopher M. Variational message passing. *Journal of Machine Learning Research*, 6:661–694, 2005.