

---

# Black-Box $\alpha$ -Divergence Minimization

---

José Miguel Hernández-Lobato<sup>1\*</sup>

Yingzhen Li<sup>2\*</sup>

Mark Rowland<sup>2</sup>

Daniel Hernández-Lobato<sup>3</sup>

Thang Bui<sup>2</sup>

Richard E. Turner<sup>2</sup>

JMH@SEAS.HARVARD.EDU

YL494@CAM.AC.UK

MR504@CAM.AC.UK

DANIEL.HERNANDEZ@UAM.ES

TDB40@CAM.AC.UK

RET26@CAM.AC.UK

<sup>1</sup>Harvard University, <sup>2</sup>University of Cambridge, <sup>3</sup>Universidad Autónoma de Madrid

## Abstract

We present *black-box alpha* (BB- $\alpha$ ), an approximate inference method based on the minimization of  $\alpha$ -divergences between probability distributions. BB- $\alpha$  scales to large datasets since it can be implemented using stochastic gradient descent. BB- $\alpha$  can be applied to complex probabilistic models with little effort since it only requires as input the likelihood function and its gradients. These gradients can be easily obtained using automatic differentiation. By tuning divergence parameter  $\alpha$ , the method is able to interpolate between variational Bayes and an expectation propagation-like algorithm. Experiments on probit regression, neural network regression and classification problems illustrate the accuracy of the posterior approximations obtained with BB- $\alpha$ .

## 1. Introduction

Bayesian probabilistic modelling provides useful tools for making predictions from data. The formalism requires a probabilistic model  $p(\mathbf{x}|\boldsymbol{\theta})$ , parameterised by a model parameter vector  $\boldsymbol{\theta}$ , over the observations  $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$ . Bayesian inference treats  $\boldsymbol{\theta}$  as a random variable and updates the posterior belief with Bayes' rule given the prior and the observations:

$$p(\boldsymbol{\theta}|\mathcal{D}) \propto \left[ \prod_{n=1}^N p(\mathbf{x}_n|\boldsymbol{\theta}) \right] p_0(\boldsymbol{\theta}).$$

Unfortunately the computation of the posterior distribution is often intractable for many useful probabilistic models.

One can use approximate inference techniques to sidestep this difficulty. Two examples are variational Bayes (VB) (Jordan et al., 1999) and expectation propagation (EP) (Minka, 2001). These methods adjust the parameters of a tractable distribution so that it is close to the exact posterior, by finding the stationary point of an energy function. Both VB and EP are particular cases of local  $\alpha$ -divergence minimization, where  $\alpha \in (-\infty, +\infty) \setminus \{0\}$  is a parameter that specifies the divergence to be minimized (Minka, 2005). If  $\alpha \rightarrow 0$ , VB is obtained and  $\alpha = 1$  gives EP (Minka, 2005). Power EP (PEP) (Minka, 2004) corresponds to general settings of  $\alpha$  and has been shown to work very well. The optimal value for  $\alpha$  may be model, dataset and/or task specific.

Compared to VB, EP has received less attention in the machine learning community. EP constructs a posterior approximation containing local approximations to the likelihood functions. EP can provide significantly better approximations compared to VB in a variety of applications. For instance, VB provides poor approximations when non-smooth likelihood functions are used (Cunningham et al., 2011; Turner & Sahani, 2011). Moreover, EP utilises cheap local computations and message passing to obtain fixed points of the energy function, while VI often directly optimizes the typically non-convex variational free energy with gradient descent.

However, there are two issues that hinder wide deployment of EP. First, the iterative procedure requires the maintenance of all the local approximations, which has prohibitive memory cost for large datasets and big models. Second, the original EP method has no convergence guarantee to the stationary point of the energy function (Minka, 2001). Previous work has looked at the first problem, which approximates EP using factor tying/local parameter sharing through the stochastic EP/averaged EP (SEP/AEP) algorithm (Li et al., 2015; Dehaene & Barthelmé, 2015). These works provided theoretical and empirical results showing

---

\*Joint first author

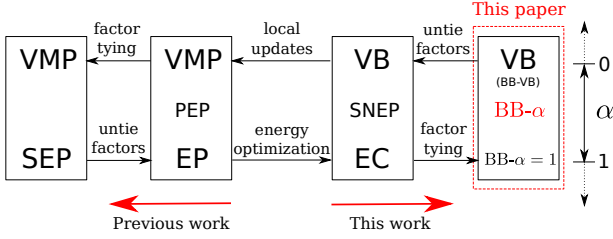


Figure 1. Connections between the proposed black-box alpha (BB- $\alpha$ ) algorithm and a number of existing approaches. EC abbreviates expectation consistent approximate inference (Opper & Winther, 2005) which is a double-loop convergent algorithm similar to Heskes & Zoeter (2002). VMP stands for variational message passing described in Minka (2005), which should be distinguished from another version in Winn & Bishop (2005). Other acronyms are explained in the main text.

that the approximations work almost as well as regular EP whilst offering substantial memory efficiencies. However, the energy function of stochastic EP is unknown, making the attempts for proving convergence in general very difficult. On the second issue, Heskes & Zoeter (2002) and Opper & Winther (2005) derived a convergent double loop algorithm based on the original min-max optimization problem. It can, however, be far slower than the message passing procedure. Teh et al. (2015) proposed the stochastic natural-gradient EP (SNEP) method, which is also double loop-like, but in practice, they only performed one-step inner loop update to speed up training.

Buoyed by the success of the seemingly severe approximation of factor tying, we have evaluated a different approach in the same spirit that applies the same idea directly to the EP energy function, rather than through the EP updates. We visualise the differences and connections of the new approach to existing methods in Figure 1. This gives further substantial advantages besides being memory efficient. In this case, convergence can be achieved by simple gradient descent instead of a double-loop procedure. On the other hand, the analytic form of the energy function enables direct applications of popular stochastic optimization methods and adaptive learning rates for large-scale learning.

Another contribution of this paper is the incorporation of black-box inference techniques to allow applications to a wide range of probabilistic models. For complicated models, the energy function of VB and EP (and the version after tying factors) might not exist in an analytic form. *Black-box* VB (Ranganath et al., 2014) sidesteps this difficulty by using Monte Carlo sampling. Similar ideas have been explored in the context of EP (Barthelmé & Chopin, 2011; Gelman et al., 2014; Xu et al., 2014). Very recently Teh et al. (2015) applied MCMC to obtain samples for moment estimates. In our paper, simple Monte Carlo is considered as a mean of approximating the energy function, although

future work will also look at more sophisticated methods including MCMC and importance sampling. Automatic differentiation techniques are also extended to the (approximate) EP context to allow fast model prototyping.

We name our approach *black-box alpha* (BB- $\alpha$ ), since it also extends to power EP which locally minimizes an  $\alpha$ -divergence. The proposed method can interpolate between VB ( $\alpha \rightarrow 0$ ) and an EP-like method ( $\alpha = 1$ ), and extrapolate beyond to  $\alpha \rightarrow \pm\infty$ . Empirical results on probit regression and neural network regression/classification problems demonstrate the scalability and accuracy of the proposed approximate (power) EP approach.

## 2. $\alpha$ -Divergence and Power EP

Let us begin by briefly reviewing the  $\alpha$ -divergence upon which our method is based. Consider two probability densities  $p$  and  $q$  of a random variable  $\theta$ ; one fundamental question is to assess how close the two distributions are. The  $\alpha$ -divergence (Amari, 1985) measures the “similarity” between two distributions, and in this paper we adopt a more convenient form<sup>1</sup> (Zhu & Rohwer, 1995):

$$D_\alpha[p||q] = \frac{1}{\alpha(\alpha-1)} \left( 1 - \int p(\theta)^\alpha q(\theta)^{1-\alpha} d\theta \right), \quad (1)$$

The following examples with different  $\alpha$  values are interesting special cases:

$$D_1[p||q] = \lim_{\alpha \rightarrow 1} D_\alpha[p||q] = \text{KL}[p||q], \quad (2)$$

$$D_0[p||q] = \lim_{\alpha \rightarrow 0} D_\alpha[p||q] = \text{KL}[q||p], \quad (3)$$

$$D_{\frac{1}{2}}[p||q] = 2 \int \left( \sqrt{p(\theta)} - \sqrt{q(\theta)} \right)^2 d\theta = 4\text{Hel}^2[p||q]. \quad (4)$$

For the first two limiting cases  $\text{KL}[p||q]$  denotes the *Kullback-Leibler (KL) divergence* given by  $\text{KL}[p||q] = \mathbb{E}_p[\log p(\theta) - \log q(\theta)]$ . In (4)  $\text{Hel}[p||q]$  denotes the *Hellinger distance* between two distributions;  $D_{\frac{1}{2}}$  is the only symmetric measure in the  $\alpha$ -divergence family.

To understand how the choice of  $\alpha$  might affect the result of approximate inference, consider the problem of approximating a complicated distribution  $p$  with a tractable distribution  $q$  by minimizing  $D_\alpha[p||q]$ . The resulting (unnormalized) approximate distribution for different  $\alpha$  values are visualized in Figure 2. This shows that when  $\alpha$  is a large positive number the approximation  $q$  tends to cover all the modes of  $p$ , while for  $\alpha \rightarrow -\infty$  (assuming the divergence is finite)  $q$  is attracted to the mode with the largest probability mass. The optimal setting of  $\alpha$  might reasonably be expected to depend on the learning task that is being considered.

<sup>1</sup>Equivalent to the original definition by setting  $\alpha' = 2\alpha - 1$  in Amari’s notation.

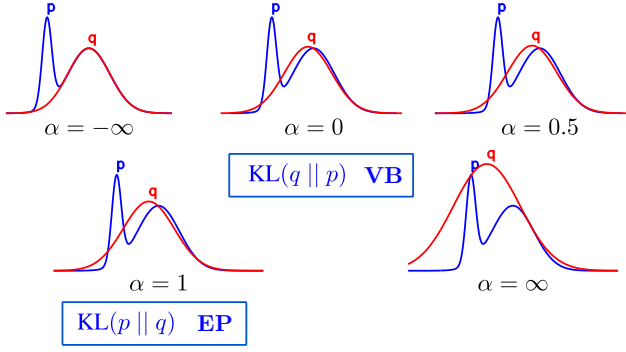


Figure 2. An illustration of approximating distributions by  $\alpha$ -divergence minimization. Here  $p$  and  $q$  shown in the graphs are unnormalized probability densities. Reproduced from [Minka \(2005\)](#). Best viewed in color.

Setting aside the analytic tractability for the computations, we note that the minimization of a global  $\alpha$ -divergence might not always be desirable. If the true posterior has many modes, a global approximation of this flavour that is refined using  $\alpha \geq 1$  will cover over the modes, and can place substantial probability in the area where the true posterior has low probability (see the last plot in Figure 2). The power EP algorithm ([Minka, 2001; 2004](#)) minimizes a set of local  $\alpha$ -divergences instead and has been shown very successful for posterior approximation.

We now give a brief review of the power EP algorithm. Recall the definition of the typically intractable posterior distribution

$$p(\theta|\mathcal{D}) \propto \left[ \prod_{n=1}^N p(\mathbf{x}_n|\theta) \right] p_0(\theta). \quad (5)$$

Here for simplicity, the prior distribution  $p_0(\theta) = \exp\{\mathbf{s}(\theta)^T \boldsymbol{\lambda}_0 - \log Z(\boldsymbol{\lambda}_0)\}$  is assumed to have an exponential family form, where  $\boldsymbol{\lambda}_0$  and  $\mathbf{s}(\theta)$  are vectors of natural parameters and sufficient statistics, respectively, and  $Z(\boldsymbol{\lambda}_0)$  is the normalization constant or partition function required to make  $p_0(\theta)$  a valid distribution. We can use power EP to approximate the true posterior  $p(\theta|\mathcal{D})$ . We assume again for convenience of the exposition that the approximate posterior is within the same exponential family as the prior so that  $q(\theta) \propto \exp\{\mathbf{s}(\theta)^T (\sum_n \boldsymbol{\lambda}_n + \boldsymbol{\lambda}_0)\}$ . Here we also define the site approximation  $f_n(\theta) = \exp\{\mathbf{s}(\theta)^T \boldsymbol{\lambda}_n\}$ , which is updated to approximate the effect of the  $n$ -th likelihood term  $p(\mathbf{x}_n|\theta)$ . In the following sections we use  $\boldsymbol{\lambda}_q$  to denote the natural parameters of  $q(\theta)$ , and in the EP context  $\boldsymbol{\lambda}_q = \sum_n \boldsymbol{\lambda}_n + \boldsymbol{\lambda}_0$ . According to [Minka \(2004\)](#) and [Seeger \(2005\)](#), the power EP energy

function with power  $\alpha$  is

$$E(\boldsymbol{\lambda}_0, \{\boldsymbol{\lambda}_n\}) = \log Z(\boldsymbol{\lambda}_0) + \left( \frac{N}{\alpha} - 1 \right) \log Z(\boldsymbol{\lambda}_q) - \frac{1}{\alpha} \sum_{n=1}^N \log \int p(\mathbf{x}_n|\theta)^\alpha \exp\{\mathbf{s}(\theta)^T (\boldsymbol{\lambda}_q - \alpha \boldsymbol{\lambda}_n)\} d\theta. \quad (6)$$

This energy is equal to minus the logarithm of the power EP approximation of the model evidence  $p(\mathcal{D})$ , that is, the normalizer of the right-hand side of (5). Therefore, minimizing (14) with respect to  $\{\boldsymbol{\lambda}_n\}$  is arguably a sensible way to tune these variational parameters. However, power EP does not directly perform gradient descent to minimizing  $E(\boldsymbol{\lambda}_0, \{\boldsymbol{\lambda}_n\})$ . Instead, it finds a stationary solution to the optimization problem by repeatedly applying the following four steps for every site approximation:

- 1 Compute the cavity distribution:

$$q^{\setminus n}(\theta) \propto q(\theta) / f_n(\theta)^\alpha,$$

$$\text{i.e. } \boldsymbol{\lambda}^{\setminus n} \leftarrow \boldsymbol{\lambda}_q - \alpha \boldsymbol{\lambda}_n;$$

- 2 Compute the “tilted” distribution by inserting the likelihood term raised to the same power  $\alpha$ :

$$\tilde{p}_n(\theta) \propto q^{\setminus n}(\theta) p(\mathbf{x}_n|\theta)^\alpha;$$

- 3 Match moments:

$$E_q[\mathbf{s}(\theta)] \leftarrow E_{\tilde{p}_n}[\mathbf{s}(\theta)];$$

- 4 Recover the site approximation  $f_n(\theta)$  by setting  $\boldsymbol{\lambda}_n \leftarrow \boldsymbol{\lambda}_q - \boldsymbol{\lambda}^{\setminus n}$ , and compute the final update for  $q(\theta)$  by  $\boldsymbol{\lambda}_q \leftarrow \sum_n \boldsymbol{\lambda}_n + \boldsymbol{\lambda}_0$ .

Notice in step 3 moment matching is equivalent to updating the  $q$  distribution by minimizing an  $\alpha$ -divergence, with the target proportional to  $p(\mathbf{x}_n|\theta) \exp\{\mathbf{s}(\theta)^T (\boldsymbol{\lambda}_q - \boldsymbol{\lambda}_n)\}$ . To see this, consider approximating some  $p$  distribution with  $q$  by minimizing the  $\alpha$ -divergence; the gradient of  $D_\alpha[p||q]$  w.r.t. the natural parameters  $\boldsymbol{\lambda}_q$  is:

$$\begin{aligned} \nabla_{\boldsymbol{\lambda}_q} D_\alpha[p||q] &= \frac{1}{\alpha} \int p(\theta)^\alpha q(\theta)^{1-\alpha} \nabla_{\boldsymbol{\lambda}_q} \log q(\theta) d\theta \\ &= \frac{1}{\alpha} (\mathbf{E}_{\tilde{p}}[\mathbf{s}(\theta)] - \mathbf{E}_q[\mathbf{s}(\theta)]), \end{aligned} \quad (7)$$

where  $\tilde{p} \propto p^\alpha q^{1-\alpha}$  and  $\alpha \neq 0$ . Substituting  $p$  with  $p(\mathbf{x}_n|\theta) \exp\{\mathbf{s}(\theta)^T (\boldsymbol{\lambda}_q - \boldsymbol{\lambda}_n)\}$  and zeroing the gradient results in step 3 that matches moments between the tilted distribution and the approximate posterior. Also [Minka \(2004\)](#) derived the stationary condition of (14) as

$$\mathbf{E}_{\tilde{p}_n}[\mathbf{s}(\theta)] = \mathbf{E}_q[\mathbf{s}(\theta)], \forall n, \quad (8)$$

so that it agrees with (7). This means at convergence  $q(\theta)$  minimizes the  $\alpha$ -divergences from all the tilted distributions to the approximate posterior.

Alternatively, Heskes & Zoeter (2002) and Oppé & Winther (2005) proposed a convergent double-loop algorithm to solve the energy minimization problem for normal EP ( $\alpha = 1$ ) (see supplementary material). This algorithm first rewrites the energy (14) as a function of the cavity parameters  $\lambda^{\setminus n}$  and adds the constraint  $(N - 1)\lambda_q + \lambda_0 = \sum_n \lambda^{\setminus n}$  that ensures agreement between the global approximation and the local component approximate factors. It then alternates between an optimization of the cavity parameters in the inner loop and an optimization of the parameters of the posterior approximation  $\lambda_q$  in the outer loop. However, this alternating optimization procedure often requires too many iterations to be useful in practice.

### 3. Approximate local minimization of $\alpha$ -divergences

In this section we introduce *black-box alpha* (BB- $\alpha$ ), which approximates power EP with a simplified objective. Now we constrain all the site parameters to be equal, i.e.  $\lambda_n = \lambda$  for all  $n$ . This is equivalent to tying all the local factors, where now  $f_n(\theta) = f(\theta)$  for all  $n$ . Then all the cavity distributions are the same with natural parameter  $\lambda^{\setminus n} = (N - \alpha)\lambda + \lambda_0$ , and the approximate posterior is parametrized by  $\lambda_q = N\lambda + \lambda_0$ . Recall that  $f_n(\theta)$  captures the contribution of the  $n$ -th likelihood to the posterior. Now with shared site parameters we are using an “average site approximation”  $f(\theta) = \exp\{s(\theta)^T \lambda\}$  that approximates the average effect of each likelihood term on the posterior. Under this assumption we rewrite the energy function (14) by replacing  $\lambda_n$  with  $\lambda$ :

$$E(\lambda_0, \lambda) = \log Z(\lambda_0) - \log Z(\lambda_q) - \frac{1}{\alpha} \sum_{n=1}^N \log \mathbf{E}_q \left[ \left( \frac{p(x_n|\theta)}{f(\theta)} \right)^\alpha \right]. \quad (9)$$

Figure 3 illustrates the comparison between the original power EP and the proposed method. Also, as there is a one-to-one correspondence between  $\lambda$  and  $\lambda_q$  given  $\lambda_0$ , i.e.  $\lambda = (\lambda_q - \lambda_0)/N$ , we can therefore rewrite (9) as  $E(\lambda_0, \lambda_q)$  using the global parameter  $\lambda_q$ .

The factor tying constraint was proposed in Li et al. (2015) and Dehaene & Barthelmé (2015) to obtain versions of EP called *stochastic EP* (SEP) and *averaged EP* (AEP), respectively, thus the new method also inherits the advantage of memory efficiency. However, applying this same idea directly to the energy function (14) results in a different class of algorithms from AEP, SEP and power SEP. The main difference is that, when an exponential family approximation is considered, SEP averages the *natural parameter* updates

obtained from each local approximation. However in this new approach the *moments*, also called the *mean parameter*, of the local approximate posteriors are averaged and then converted to the corresponding natural parameters<sup>2</sup>. To see this, one can take derivatives of  $E(\lambda_0, \lambda_q)$  w.r.t.  $\lambda_q$  and obtain the new stationary conditions:

$$\mathbf{E}_q[s(\theta)] = \frac{1}{N} \sum_{n=1}^N \mathbf{E}_{\tilde{p}_n}[s(\theta)]. \quad (10)$$

Therefore, the moments of the  $q$  distribution, i.e. the expectation of  $s(\theta)$  with respect to  $q(\theta)$ , is equal to the average of the expectation of  $s(\theta)$  across the different tilted distributions  $\tilde{p}_n(\theta) \propto p(x_n|\theta)^\alpha q^{\setminus n}(\theta)$ , for  $n = 1, \dots, N$ . It might not always be sensible to average moments, e.g. this approach is shown to be biased even in the simple case where the likelihood terms also belong to the same exponential family as the prior and approximate posterior (see supplementary material). Also such a method will cover multiple modes when applied to a Gaussian mixture model to approximate the posterior of the components’ means. But unlike SEP, the new method explicitly defines an energy function as an optimization objective, which enables analysis of convergence and applications of stochastic optimisation/adaptive learning rate methods. Moreover, it provides more controllable approximate posterior than VB (see supplementary) which is desirable.

We prove the convergence of the new approximate EP method by showing that the new energy function (9) is bounded below for  $\alpha \leq N$  when the energy function is finite, e.g. the ratio  $|p(x_n|\theta)/f(\theta)|$  is bounded for all  $\theta$  and  $x_n$ . First, using Jensen’s inequality, we can prove that the third term  $-\frac{1}{\alpha} \sum_{n=1}^N \log \mathbf{E}_q \left[ \left( \frac{p(x_n|\theta)}{f(\theta)} \right)^\alpha \right]$  in (9) is non-increasing in  $\alpha$ , because for arbitrary numbers  $0 < \alpha < \beta$  or  $\beta < \alpha < 0$  the function  $x^{\frac{\alpha}{\beta}}$  is strictly concave on  $x \geq 0$ . For convenience we shorthand this term as  $G(\alpha)$ . Then the proof is finished by subtracting from (9)  $\tilde{G}(N) := -\frac{1}{N} \sum_n \log \int p_0(\theta) p(x_n|\theta)^N d\theta$ , a function that is independent of the  $q$  distribution:

$$\begin{aligned} E(\lambda_0, \lambda_q) - \tilde{G}(N) \\ = G(\alpha) + \log Z(\lambda_0) - \log Z(\lambda_q) - \tilde{G}(N) \\ = G(\alpha) - G(N) \geq 0. \end{aligned}$$

The stationary point of (14) is expected to converge to the stationary point of (9) when more and more data are available. More precisely, as  $N$  grows, we expect  $q(\theta)$  and the cavities to become very concentrated. When this happens, the contribution of each likelihood factor  $p(x_n|\theta)$  to the

<sup>2</sup>Under a minimal exponential family assumption, there exists a one-to-one correspondence between the natural parameter and the mean parameter.



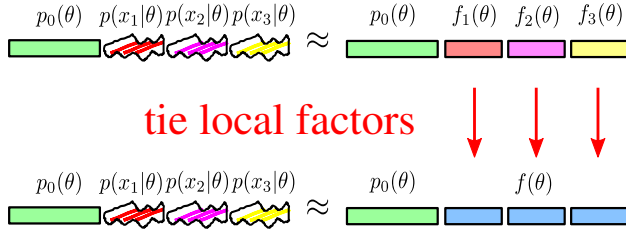


Figure 3. A cartoon for BB- $\alpha$ 's factor tying constraint. Here we assume the dataset has  $N = 3$  observations. Best seen in color.

tilted distribution  $\tilde{p}_n(\theta)$  becomes very small because the likelihood is a rather flat function when compared to the cavity distribution  $q^{\setminus n}(\theta)$ . Therefore, as the amount of data  $N$  increases, we expect all the terms  $\mathbf{E}_{\tilde{p}_n}[\mathbf{s}(\theta)]$  in (10) to be very similar to each other. When all of them are equal, we have that (10) implies (8).

As in power EP, which value of  $\alpha$  returns the best approximation depends on the particular application. In the limit that  $\alpha$  approaches zero, the BB- $\alpha$  energy (9) converges to the variational free energy:

$$\begin{aligned} & \lim_{\alpha \rightarrow 0} E(\lambda_0, \lambda_q) \\ &= \log Z(\lambda_0) - \log Z(\lambda_q) - \sum_{n=1}^N \mathbf{E}_q \left[ \log \frac{p(\mathbf{x}_n | \theta)}{f(\theta)} \right] \\ &= -\mathbf{E}_q \left[ \log \frac{\prod_n p(\mathbf{x}_n | \theta) p_0(\theta)}{\exp\{\mathbf{s}(\theta)^T \lambda_q\} / Z(\lambda_q)} \right] \\ &= -\mathbf{E}_q [\log p(\theta, \mathcal{D}) - \log q(\theta)] \\ &= E_{VI}(\lambda_0, \lambda_q). \end{aligned}$$

Note also, the  $\alpha$ -divergence as a function of  $\alpha$  is smooth in  $[0, 1]$ . Therefore, by adjusting the  $\alpha$  parameter, we can interpolate between the solutions given by variational Bayes and the solutions given by the proposed approximation to EP scheme. In the supplementary material we demonstrate this continuity in  $\alpha$  with a linear regression example.

The prior hyper-parameters  $\lambda_0$  can be optimally adjusted by minimizing (9). This can be done through a variational EM-like procedure. First we optimize the energy function w.r.t. the  $q$  distribution until convergence. Then we take the gradients w.r.t.  $\lambda_0$  and equate them to zero to obtain the update for the prior, which is  $\mathbf{E}_q[\mathbf{s}(\theta)] = \mathbf{E}_{p_0}[\mathbf{s}(\theta)]$ . However this double-loop like procedure is inefficient in practice; instead we jointly optimize the approximate posterior  $q$  and the prior distribution  $p_0$  similar to the approach of Hernández-Lobato & Hernández-Lobato (2016) for normal EP.

### 3.1. Large scale learning

When  $N$  is large, it might be beneficial to minimize (9) using stochastic optimization techniques. In particular, we

can uniformly sample a mini-batch of data  $\mathbf{S} \subseteq \{1, \dots, N\}$  and construct the noisy estimate of the energy function given by

$$\begin{aligned} E(\lambda_0, \lambda_q) &\approx \log Z(\lambda_0) - \log Z(\lambda_q) \\ &- \frac{1}{\alpha} \frac{N}{|\mathbf{S}|} \sum_{n \in \mathbf{S}} \log \mathbf{E}_q \left[ \left( \frac{p(\mathbf{x}_n | \theta)}{f(\theta)} \right)^\alpha \right]. \end{aligned} \quad (11)$$

The gradients of (11) can then be used to minimize the original objective by stochastic gradient descent. Under mild conditions, as discussed in Bottou (1998), and using a learning rate schedule  $\{\gamma_t\}$  that satisfies the Robbins-Monro conditions

$$\sum_{t=1}^{\infty} \gamma_t = \infty, \quad \sum_{t=1}^{\infty} \gamma_t^2 < \infty,$$

the stochastic optimization procedure will converge to a fixed point of the exact energy function (9).

Similar to SEP/AEP (Li et al., 2015; Dehaene & Barthelmé, 2015), BB- $\alpha$  only maintains the global parameter  $\lambda_q$  and the prior parameter  $\lambda_0$ . This has been shown to achieve a significant amount of memory saving. On the other hand, recent work on parallelising EP (Gelman et al., 2014; Xu et al., 2014; Teh et al., 2015), whether in a synchronous manner or not, extends to BB- $\alpha$  naturally. But unlike EP, which computes different cavity distributions for different datapoints, BB- $\alpha$  uses the same cavity distribution for each datapoint.

### 3.2. Black-box $\alpha$ -divergence minimization

In complicated probabilistic models, we might not be able to analytically compute the expectation over the approximate distribution  $\mathbf{E}_q \left[ \left( \frac{p(\mathbf{x}_n | \theta)}{f(\theta)} \right)^\alpha \right]$  in (11) involving the likelihood factors. However, we can obtain an estimate of these integrals by Monte Carlo. In this work we use the simplest method for doing this, but techniques like SMC and MCMC could also have the potential to be deployed. We draw  $K$  samples  $\theta_1, \dots, \theta_K$  from  $q(\theta)$  and then approximate the integrals by expectations with respect to those samples. This produces the following noisy estimate of the energy function:

$$\begin{aligned} \hat{E}(\lambda_0, \lambda_q) &= \log Z(\lambda_0) - \log Z(\lambda_q) \\ &- \frac{1}{\alpha} \frac{N}{|\mathbf{S}|} \sum_{n \in \mathbf{S}} \log \frac{1}{K} \sum_k \left( \frac{p(\mathbf{x}_n | \theta_k)}{f(\theta_k)} \right)^\alpha. \end{aligned} \quad (12)$$

Note, however, that the resulting stochastic gradients will be biased because the energy function (12) applies a non-linear transformation (the logarithm) to the Monte Carlo estimator of the integrals. Nevertheless, this bias can be reduced by increasing the number of samples  $K$ . The experiments indicate that when  $K \geq 10$  the bias is almost

negligible w.r.t. the variance from sub-sampling the data using minibatches.

There are two other tricks we have used in our implementation. The first one is the *reparameterization trick* (Kingma & Welling, 2014), which has been used to reduce the variance of the Monte Carlo approximation to the variational free energy. Consider the case of computing expectation  $\mathbb{E}_{q(\theta)}[F(\theta)]$ . This expectation can also be computed as  $\mathbb{E}_{p(\epsilon)}[F(g(\epsilon))]$ , if there exists a mapping  $g(\cdot)$  and a distribution  $p(\epsilon)$  such that  $\theta = g(\epsilon)$  and  $q(\theta)d\theta = p(\epsilon)d\epsilon$ . Now consider a Gaussian approximation  $q(\theta)$  as a running example. Since the Gaussian distribution also has an (minimal) exponential family form, there exists a one-to-one correspondence between the mean parameters  $\{\mu, \Sigma\}$  and the natural parameters  $\lambda_q = \{\Sigma^{-1}\mu, \Sigma^{-1}\}$ . Furthermore, sampling  $\theta \sim q(\theta)$  is equivalent to  $\theta = g(\epsilon) = \mu + \Sigma^{1/2}\epsilon$ ,  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ . Thus the sampling approximation  $\hat{E}(\lambda_0, \lambda_q)$  can be computed as

$$\begin{aligned} \hat{E}(\lambda_0, \lambda_q) &= \log Z(\lambda_0) - \log Z(\lambda_q) \\ &\quad - \frac{1}{\alpha} \frac{N}{|\mathbf{S}|} \sum_{n \in \mathbf{S}} \log \frac{1}{K} \sum_k \left( \frac{p(\mathbf{x}_n | g(\epsilon_k))}{f(g(\epsilon_k))} \right)^\alpha. \end{aligned} \quad (13)$$

with  $\epsilon_1, \dots, \epsilon_K$  sampled from a mean zero, unit variance Gaussian. A further trick is resampling  $\{\epsilon_k\}$  every  $M > 1$  iterations. In our experiments with neural networks, this speeds-up the training process since it reduces the transference of randomness to the GPU which makes computations slow.

Given a new probabilistic model, one can then use the proposed approach to quickly implement, in an automatic manner, an inference algorithm based on the local minimization of  $\alpha$ -divergences. For this one only needs to write code that evaluates the likelihood factors  $f_1, \dots, f_N$  in (13). After this, the most difficult task is the computation of the gradients of (13) so that stochastic gradient descent with minibatches can be used to optimize the energy function. However, the computation of these gradients can be easily automated by using automatic differentiation tools such as Autograd (<http://github.com/HIPS/autograd>) or Theano (Bastien et al., 2012). This approach allows us to quickly implement and test different modeling assumptions with little effort.

## 4. Experiments

We evaluated the proposed algorithm black-box alpha (BB- $\alpha$ ), on regression and classification problems using a probit regression model and Bayesian neural networks. We also compare with a method that optimizes a Monte Carlo approximation to the variational lower bound (Ranganath et al., 2014). This approximation is obtained in a similar way to the one described in Section 3.2, where one can

show its equivalence to BB- $\alpha$  by limiting  $\alpha \rightarrow 0$ . We call this method black-box variational Bayes (BB-VB). In the implementation of BB- $\alpha$  and BB-VB shown here, the posterior approximation  $q$  is always a factorized Gaussian (but more complex distributions can easily be handled). The mean parameters of  $q$  are initialized by independently sampling from a zero mean Gaussian with standard deviation  $10^{-1}$ . We optimize the logarithm of the variance parameters in  $q$  to avoid obtaining negative variances. We use -10 as the initial value for the log-variance parameters, which results in very low initial variance in  $q$ . This makes the stochastic optimizer initially resemble a point estimator method which finds quickly a solution for the mean parameters with good predictive properties on the training data. After this, the stochastic optimizer progressively increases the variance parameters to capture the uncertainty around the mean of  $q$ . This trick considerably improves the performance of the stochastic optimization method for BB- $\alpha$  and BB-VB. The prior  $p(\mathbf{x})$  is always taken to be a factorized Gaussian with zero mean and unit standard deviation. The implementation of each of the analyzed methods will be made available for reproducible research.

### 4.1. Probit regression

We perform experiments with a Bayesian probit regression model to validate the proposed black-box approach. We optimize the different objective functions using minibatches of size 32 and Adam (Kingma & Ba, 2014) with its default parameter values during 200 epochs. BB- $\alpha$  and BB-VB are implemented by drawing  $K = 100$  Monte Carlo samples for each minibatch. The following values  $\alpha = 1$ ,  $\alpha = 0.5$  and  $\alpha = 10^{-6}$  for BB- $\alpha$  are considered in the experiments. The performance of each method is evaluated on 50 random training and test splits of the data with 90% and 10% of the data instances, respectively.

Table 1 shows the average test log-likelihood and test error obtained by each technique in the probit regression datasets from the UCI data repository (Lichman, 2013). We also show the average rank obtained by each method across all the train/test splits of the data. Overall, all the methods obtain very similar results although BB- $\alpha$  with  $\alpha = 1.0$  seems to perform slightly better. Importantly, BB- $\alpha$  with  $\alpha = 10^{-6}$  produces the same results as BB-VB, which verifies our theory of continuous interpolations near the limit  $\alpha \rightarrow 0$ .

### 4.2. Neural network regression

We perform additional experiments with neural networks for regression with 100 units in a single hidden layer with ReLUs and Gaussian additive noise at the output. We consider several regression datasets also from the UCI data repository (Lichman, 2013). We use the same training pro-

Table 1. Probit regression experiment results

| Dataset      | Average Test Log-likelihood        |                    |                      |                                    | Average Test Error                |                   |                                   |                   |
|--------------|------------------------------------|--------------------|----------------------|------------------------------------|-----------------------------------|-------------------|-----------------------------------|-------------------|
|              | BB- $\alpha=1.0$                   | BB- $\alpha=0.5$   | BB- $\alpha=10^{-6}$ | BB-VB                              | BB- $\alpha=1.0$                  | BB- $\alpha=0.5$  | BB- $\alpha=10^{-6}$              | BB-VB             |
| Ionosphere   | -0.333 $\pm$ 0.022                 | -0.333 $\pm$ 0.022 | -0.333 $\pm$ 0.022   | <b>-0.333<math>\pm</math>0.022</b> | 0.124 $\pm$ 0.008                 | 0.124 $\pm$ 0.008 | <b>0.123<math>\pm</math>0.008</b> | 0.123 $\pm$ 0.008 |
| Madelon      | <b>-0.799<math>\pm</math>0.006</b> | -0.920 $\pm$ 0.008 | -0.953 $\pm$ 0.009   | -0.953 $\pm$ 0.009                 | <b>0.445<math>\pm</math>0.005</b> | 0.454 $\pm$ 0.004 | 0.457 $\pm$ 0.005                 | 0.457 $\pm$ 0.005 |
| Pima         | <b>-0.501<math>\pm</math>0.010</b> | -0.501 $\pm$ 0.010 | -0.501 $\pm$ 0.010   | -0.501 $\pm$ 0.010                 | <b>0.234<math>\pm</math>0.006</b> | 0.234 $\pm$ 0.006 | 0.235 $\pm$ 0.006                 | 0.235 $\pm$ 0.006 |
| Colon Cancer | <b>-2.261<math>\pm</math>0.402</b> | -2.264 $\pm$ 0.403 | -2.268 $\pm$ 0.404   | -2.268 $\pm$ 0.404                 | <b>0.303<math>\pm</math>0.028</b> | 0.307 $\pm$ 0.028 | 0.307 $\pm$ 0.028                 | 0.307 $\pm$ 0.028 |
| Avg. Rank    | <b>1.895<math>\pm</math>0.097</b>  | 2.290 $\pm$ 0.038  | 2.970 $\pm$ 0.073    | 2.845 $\pm$ 0.072                  | <b>2.322<math>\pm</math>0.048</b> | 2.513 $\pm$ 0.039 | 2.587 $\pm$ 0.031                 | 2.578 $\pm$ 0.031 |

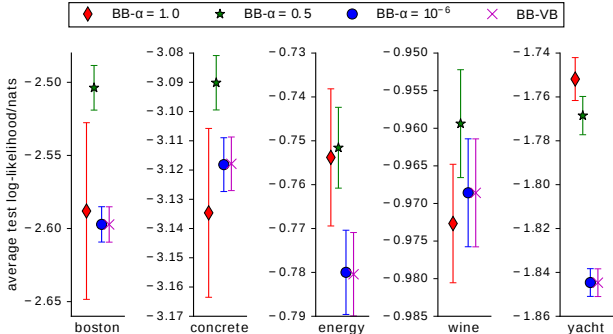


Figure 4. Average test log-likelihood and the ranking comparisons. Best viewed in color.

cedure as before, but using 500 epochs instead of 200. The noise variance is learned in each method by optimizing the corresponding objective function: evidence lower bound in BB-VB and (9) in BB- $\alpha$ .

The average test log-likelihood across 50 splits of the data into training and test sets are visualised in Figure 4. The performance of BB- $\alpha = 10^{-6}$  is again almost indistinguishable from that of BB-VB. None of the tested  $\alpha$  settings clearly dominates the other choices in all cases, indicating that the optimal  $\alpha$  may vary for different problems. However,  $\alpha = 0.5$  produces good overall results. This might be because the Hellinger distance is the only symmetric divergence measure in the  $\alpha$ -divergence family which balances the tendencies of capturing a mode ( $\alpha < 0.5$ ) and covering the whole probability mass ( $\alpha > 0.5$ ). There are no significant differences in regression error so the results are not shown.

### 4.3. Neural network classification

The next experiment considers the MNIST digit classification problem. We use neural networks with 2 hidden layers with 400 hidden units per layer, ReLUs and softmax outputs. In this problem, we initialize the mean parameters in  $q$  as recommended in Glorot & Bengio (2010) by sampling from a zero-mean Gaussian with variance given by  $2/(d_{\text{in}} + d_{\text{out}})$ , where  $d_{\text{in}}$  and  $d_{\text{out}}$  are the dimensionalities of the previous and the next layer in the neural network. In this case we also try  $\alpha = -1.0$  in BB- $\alpha$ . We use minibatches of size 250 and run the different methods for 250

epochs. BB- $\alpha$  and BB-VB use now  $K = 50$  Monte Carlo samples to approximate the expectations with respect to  $q$  in each minibatch. We implemented the neural networks in Theano and ran the different methods on GPUs using Adam with its default parameter values and a learning rate of 0.0001. In this case, to reduce the transference of data from the main memory to the GPU, we update the randomness in the Monte Carlo samples only after processing 10 minibatches instead of after processing each minibatch.

Table 2. Average Test Error and Log-likelihood in MNIST

| Setting                | Error/100   | Rank        | LL/100       | Rank        |
|------------------------|-------------|-------------|--------------|-------------|
| BB- $\alpha = 1.0$     | 1.51        | 4.97        | -5.51        | 5.00        |
| BB- $\alpha = 0.5$     | 1.44        | 4.03        | -5.09        | 4.00        |
| BB- $\alpha = 10^{-6}$ | 1.36        | 2.15        | -4.68        | 2.55        |
| BB-VB                  | 1.36        | 2.12        | -4.68        | 2.45        |
| BB- $\alpha = -1.0$    | <b>1.33</b> | <b>1.73</b> | <b>-4.47</b> | <b>1.00</b> |

Table 2 summarises the average test error and test log-likelihood obtained by each method over 20 random initializations. For non-negative alpha settings BB-VB ( $\alpha = 0$ ) returns the best result, and again BB- $\alpha = 10^{-6}$  performs almost identically to the variational inference approach BB-VB. The best performing method is BB- $\alpha$  with  $\alpha = -1$ , which is expected to move slightly towards fitting a mode. We note here that this is different from the Laplace approximation, which, depending on the curvature at the MAP solution, might return an approximate posterior that also covers spaces other than the mode. On this dataset  $\alpha = -1$  returns both higher test log-likelihood and lower test error than all the tested non-negative settings for  $\alpha$ .

### 4.4. Clean energy project data

We perform additional experiments with data from the Harvard Clean Energy Project, which is the world’s largest materials high-throughput virtual screening effort (Hachmann et al., 2014). It has scanned a large number of molecules of organic photovoltaics to find those with high power conversion efficiency (PCE) using quantum-chemical techniques. The target value within this dataset is the PCE of each molecule. The input features for all molecules in the data set are 512-bit Morgan circular fingerprints, calculated with a bond radius of 2, and derived from the canonical smiles, implemented in the RDkit. We use 50,000 molecules for

Table 3. Average Test Error and Test Log-likelihood in CEP Dataset.

| CEP Dataset         | BB- $\alpha=1.0$ | BB- $\alpha=0.5$                 | BB- $\alpha=10^{-6}$ | BB-VB            |
|---------------------|------------------|----------------------------------|----------------------|------------------|
| Avg. Error          | 1.28 $\pm$ 0.01  | <b>1.08<math>\pm</math>0.01</b>  | 1.13 $\pm$ 0.01      | 1.14 $\pm$ 0.01  |
| Avg. Rank           | 4.00 $\pm$ 0.00  | <b>1.35<math>\pm</math>0.15</b>  | 2.05 $\pm$ 0.15      | 2.60 $\pm$ 0.13  |
| Avg. Log-likelihood | -0.93 $\pm$ 0.01 | <b>-0.74<math>\pm</math>0.01</b> | -1.39 $\pm$ 0.03     | -1.38 $\pm$ 0.02 |
| Avg. Rank           | 1.95 $\pm$ 0.05  | <b>1.05<math>\pm</math>0.05</b>  | 3.40 $\pm$ 0.11      | 3.60 $\pm$ 0.11  |

training and 10,000 molecules for testing.

We consider neural networks with 2 hidden layers with 400 hidden units per layer, ReLUs and Gaussian additive noise at the output. The noise variance is fixed to 0.16, which is the optimal value according to the results reported in Pyzer-Knapp et al. (2015). The initialization and training process is the same as with the MNIST dataset, but using the default learning rate in Adam.

Table 3 summarises the average test error and test log-likelihood obtained by each method over 20 random initializations. BB- $\alpha = 10^{-6}$  obtains again very similar results to BB-VB. In this case, the best performing method is BB- $\alpha$  with  $\alpha = 0.5$ , both in terms of test error and test log-likelihood. This was also the best performing method in the experiments from Section 4.2, which indicates that  $\alpha = 0.5$  may be a generally good setting in neural network regression problems. Table 3 shows that  $\alpha = 0.5$  attains a balance between the tendency of  $\alpha = 1.0$  to perform well in terms of test log-likelihood and the tendency of  $\alpha = 10^{-6}$  to perform well in terms of test squared error.

#### 4.5. Analysis of the bias and variance in the gradients

We perform another series of experiments to analyze the bias and the variance in the gradients of (13) as a function of the number  $K$  of Monte Carlo samples used to obtain a noisy approximation of the objective function and the value of  $\alpha$  used. To estimate the bias in BB- $\alpha$  we run the Adam optimizer for 100 epochs on the Boston housing dataset as described in Section 4.2. Then, we estimate the biased gradient using  $K = 1, 5, 10$  Monte Carlo samples from  $q$ , which is repeated 1,000 times to obtain an averaged estimate. We also compute an approximation to the ground truth for the unbiased gradient by using  $K = 10,000$  Monte Carlo samples. The whole process is performed across 15 minibatches of datapoints from each split. We then define the bias in the gradient as the averaged L2 norm between the ground truth gradient and the biased gradient across these 15 minibatches, divided by the square root of the dimension of the gradient vector. This definition of bias is not 0 for methods that use unbiased estimators of the gradient, such as BB-VB, because of the variance by sampling on each minibatch. However, we expect this procedure to report larger bias values for BB- $\alpha$  than for BB-VB. Therefore, we subtract from the bias values obtained for BB- $\alpha$

Table 4. Average Bias.

| Dataset  | BB- $\alpha=1.0$ | BB- $\alpha=0.5$ | BB- $\alpha=10^{-6}$ |
|----------|------------------|------------------|----------------------|
| $K = 1$  | 0.2774           | 0.1214           | 0.0460               |
| $K = 5$  | 0.0332           | 0.0189           | 0.0162               |
| $K = 10$ | 0.0077           | 0.0013           | 0.0001               |

Table 5. Average Standard Deviation Gradient.

| Dataset  | BB- $\alpha=1.0$ | BB- $\alpha=0.5$ | BB- $\alpha=10^{-6}$ |
|----------|------------------|------------------|----------------------|
| $K = 1$  | 14.1209          | 14.0159          | 13.9109              |
| $K = 5$  | 12.7953          | 12.8418          | 12.8984              |
| $K = 10$ | 12.3203          | 12.4101          | 12.5058              |

the corresponding bias values obtained for BB-VB. This eliminates from the bias values any additional variance that is produced by having to sample to estimate the unbiased and biased gradient on each minibatch.

Table 4 shows the average bias obtained for each value of  $K$  and  $\alpha$  across the 50 splits. We observe that the bias is reduced as we increase  $K$  and as we make  $\alpha$  closer to zero. For  $K = 10$  the bias is already very low. To put these bias numbers into context, we also computed a standard deviation-like measure by the square root of the average empirical variance per dimension in the noisy gradient vector over the 15 minibatches. Table 5 shows the average values obtained across the 50 splits, where entries are almost constant as a function of  $\alpha$ , and up to 5 orders of magnitude larger than the entries of Table 4 for  $K = 10$ . This means the bias in the gradient used by BB- $\alpha$  is negligible when compared with the variability that is obtained in the gradient by subsampling the training data.

## 5. Conclusions and future work

We have proposed BB- $\alpha$  as a black-box inference algorithm to approximate power EP. This is done by considering the energy function used by power EP and constraining the form of the site approximations in this method. The proposed method locally minimizes the  $\alpha$ -divergence that is a rich family of divergence measures between distributions including the Kullback-Leibler divergence. Such a method is guaranteed to converge under certain conditions, and can be implemented by optimizing an energy function without having to use inefficient double-loop al-



gorithms. Scalability to large datasets can be achieved by using stochastic gradient descent with minibatches. Furthermore, a combination of a Monte Carlo approximation and automatic differentiation methods allows our technique to be applied in a straightforward manner to a wide range of probabilistic models with complex likelihood factors. Experiments with neural networks applied to small and large datasets demonstrate both the accuracy and the scalability of the proposed approach. The evaluations also indicate the optimal setting for  $\alpha$  may vary for different tasks. Future work should provide a theoretical guide and/or automatic tools for modelling with different factors and different  $\alpha$  values.

## Acknowledgement

JMHL acknowledges support from the Rafael del Pino Foundation. YL thanks the Schlumberger Foundation Faculty for the Future fellowship on supporting her PhD study. MR acknowledges support from UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/L016516/1 for the University of Cambridge Centre for Doctoral Training, the Cambridge Centre for Analysis. TB thanks Google for funding his European Doctoral Fellowship. DHL acknowledge support from Plan National I+D+i, Grant TIN2013-42351-P and TIN2015-70308-REDT, and from Comunidad de Madrid, Grant S2013/ICE-2845 CASI-CAM-CM. RET thanks EPSRC grant #EP/L000776/1 and #EP/M026957/1.

## References

- Amari, Shun-ichi. *Differential-Geometrical Methods in Statistic*. Springer, New York, 1985.
- Barthelmé, Simon and Chopin, Nicolas. Abc-ep: Expectation propagation for likelihoodfree bayesian computation. In *International Conference on Machine Learning*, 2011.
- Bastien, Frédéric, Lamblin, Pascal, Pascanu, Razvan, Bergstra, James, Goodfellow, Ian J., Bergeron, Arnaud, Bouchard, Nicolas, and Bengio, Yoshua. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.
- Bottou, Léon. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9):25, 1998.
- Cunningham, John P, Hennig, Philipp, and Lacoste-Julien, Simon. Gaussian probabilities and expectation propagation. *arXiv preprint arXiv:1111.6832*, 2011.
- Dehaene, Guillaume and Barthelmé, Simon. Expectation propagation in the large-data limit. *arXiv:1503.08060*, 2015.
- Gelman, Andrew, Vehtari, Aki, Jylänki, Pasi, Robert, Christian, Chopin, Nicolas, and Cunningham, John P. Expectation propagation as a way of life. *arXiv:1412.4869*, 2014.
- Glorot, Xavier and Bengio, Yoshua. Understanding the difficulty of training deep feedforward neural networks. In *International conference on artificial intelligence and statistics*, pp. 249–256, 2010.
- Hachmann, Johannes, Olivares-Amaya, Roberto, Jinich, Adrian, Appleton, Anthony L, Blood-Forsythe, Martin A, Seress, László R, Román-Salgado, Carolina, Trepte, Kai, Atahan-Evrenk, Sule, Er, Süleyman, et al. Lead candidates for high-performance organic photovoltaics from high-throughput quantum chemistry—the harvard clean energy project. *Energy & Environmental Science*, 7(2):698–704, 2014.
- Hernández-Lobato, Daniel and Hernández-Lobato, José Miguel. Scalable gaussian process classification via expectation propagation. 2016.
- Heskes, Tom and Zoeter, Onno. Expectation propagation for approximate inference in dynamic bayesian networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pp. 216–223. Morgan Kaufmann Publishers Inc., 2002.
- Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.
- Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, 2014.
- Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Li, Yingzhen, Hernandez-Lobato, Jose Miguel, and Turner, Richard E. Stochastic expectation propagation. In *Advances in Neural Information Processing Systems*, 2015.
- Lichman, M. UCI machine learning repository, 2013. URL <http://archive.ics.uci.edu/ml>.
- Minka, Thomas P. Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pp. 362–369. Morgan Kaufmann Publishers Inc., 2001.
- Minka, Thomas P. Power ep. Technical report, Technical report, Microsoft Research, Cambridge, 2004.

- Minka, Thomas P. Divergence measures and message passing. Technical report, Technical report, Microsoft Research, 2005.
- Opper, Manfred and Winther, Ole. Expectation consistent approximate inference. *The Journal of Machine Learning Research*, 6:2177–2204, 2005.
- Pyzer-Knapp, Edward O, Li, Kewei, and Aspuru-Guzik, Alan. Learning from the harvard clean energy project: The use of neural networks to accelerate materials discovery. *Advanced Functional Materials*, 25(41):6495–6502, 2015.
- Ranganath, Rajesh, Gerrish, Sean, and Blei, David. Black box variational inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pp. 814–822, 2014.
- Seeger, Matthias. Expectation propagation for exponential families. Technical report, 2005.
- Teh, Yee Whye, Hasenclever, Leonard, Lienart, Thibaut, Vollmer, Sebastian, Webb, Stefan, Lakshminarayanan, Balaji, and Blundell, Charles. Distributed bayesian learning with stochastic natural-gradient expectation propagation and the posterior server. *arXiv:1512.09327*, 2015.
- Turner, Richard E. and Sahani, Maneesh. Probabilistic amplitude and frequency demodulation. In Shawe-Taylor, J., Zemel, R.S., Bartlett, P., Pereira, F.C.N., and Weinberger, K.Q. (eds.), *Advances in Neural Information Processing Systems 24*, pp. 981–989. 2011.
- Winn, John M and Bishop, Christopher M. Variational message passing. In *Journal of Machine Learning Research*, pp. 661–694, 2005.
- Xu, Minjie, Lakshminarayanan, Balaji, Teh, Yee Whye, Zhu, Jun, and Zhang, Bo. Distributed bayesian posterior sampling via moment sharing. In *NIPS*, 2014.
- Zhu, Huaiyu and Rohwer, Richard. Information geometric measurements of generalisation. Technical report, Technical Report NCRG/4350. Aston University., 1995.

## A. The Min-Max Problem of EP

This section revisits the original EP algorithm as a min-max optimization problem. Recall in the main text that we use the  $q$  distribution of an exponential family form  $q(\boldsymbol{\theta}) \propto \exp\{\mathbf{s}(\boldsymbol{\theta})^T \boldsymbol{\lambda}_q\}$  to approximate the true posterior distribution  $p(\boldsymbol{\theta}|\mathcal{D})$ . Now we define a set of *unnormalized* cavity distributions  $q^{\setminus n}(\boldsymbol{\theta}) = \exp\{\mathbf{s}(\boldsymbol{\theta})^T \boldsymbol{\lambda}_{\setminus n}\}$  for every datapoint  $\mathbf{x}_n$ . Then according to (Minka, 2001), the EP energy function is

$$E(\boldsymbol{\lambda}_q, \{\boldsymbol{\lambda}_{\setminus n}\}) = \log Z(\boldsymbol{\lambda}_0) + (N-1) \log Z(\boldsymbol{\lambda}_q) - \sum_{n=1}^N \log \int p(\mathbf{x}_n|\boldsymbol{\theta}) q^{\setminus n}(\boldsymbol{\theta}) d\boldsymbol{\theta}. \quad (14)$$

And in practice EP finds a stationary solution to the constrained optimization problem

$$\min_{\boldsymbol{\lambda}_q} \max_{\{\boldsymbol{\lambda}_{\setminus n}\}} E(\boldsymbol{\lambda}, \{\boldsymbol{\lambda}_{\setminus n}\}), \quad \text{subject to} \quad (N-1)\boldsymbol{\lambda}_q + \boldsymbol{\lambda}_0 = \sum_{n=1}^N \boldsymbol{\lambda}_{\setminus n}, \quad (15)$$

where the constraint in (15) guarantees that the  $\{\boldsymbol{\lambda}_{\setminus n}\}$  are valid cavity parameters that are consistent with the approximate posterior. Similary for power EP the energy function has the following form:

$$\begin{aligned} E(\boldsymbol{\lambda}_q, \{\boldsymbol{\lambda}_{\setminus n}\}) &= \log Z(\boldsymbol{\lambda}_0) + \left(\frac{N}{\alpha} - 1\right) \log Z(\boldsymbol{\lambda}_q) \\ &\quad - \frac{1}{\alpha} \sum_{n=1}^N \log \int p(\mathbf{x}_n|\boldsymbol{\theta})^\alpha q^{\setminus n}(\boldsymbol{\theta}) d\boldsymbol{\theta}, \end{aligned} \quad (16)$$

and the constraint of the optimization problem changes to  $(N-\alpha)\boldsymbol{\lambda}_q + \alpha\boldsymbol{\lambda}_0 = \sum_{n=1}^N \boldsymbol{\lambda}_{\setminus n}$ .

The problem in (15) can be solved using a double-loop algorithm (Heskes & Zoeter, 2002; Opper & Winther, 2005). This algorithm alternates between an optimization of the cavity parameters  $\{\boldsymbol{\lambda}_{\setminus n}\}$  in the inner loop and an optimization of the parameters of the posterior approximation  $\boldsymbol{\lambda}_q$  in the outer loop. Each iteration of the double-loop algorithm is guaranteed to minimize the energy in (14). However, the alternating optimization of  $\boldsymbol{\lambda}_q$  and  $\{\boldsymbol{\lambda}_{\setminus n}\}$  is very inefficient to be useful in practice.

## B. Linear regression example

In this section we demonstrate several properties of BB- $\alpha$  on a toy linear regression problem; in particular, we compare the BB- $\alpha$  optimal distribution to the true posterior in the cases where the true posterior lies in the variational family considered, and in the mean-field case where the variational family is Gaussian with diagonal covariance matrix.

More specifically, we consider a linear regression model of the form

$$y_n = \boldsymbol{\theta}^\top \mathbf{x}_n + \sigma \varepsilon_n$$

where  $y_n \in \mathbf{R}$ ,  $\mathbf{x}_n \in \mathbf{R}^2$  for  $n = 1, \dots, N$ ,  $\boldsymbol{\theta} \in \mathbf{R}^2$ ,  $\sigma \in \mathbf{R}_+$ , and  $(\varepsilon_n)_{n=1}^N \stackrel{\text{iid}}{\sim} N(0, 1)$ . This model is simple enough to analytically optimise the BB- $\alpha$  energy function, and provides intuition for BB- $\alpha$  in more general contexts. We specify a prior  $p_0(\boldsymbol{\theta}) \propto \exp(-\boldsymbol{\theta}^\top \boldsymbol{\theta}/2)$  on the model weights; in this case, the posterior distribution for the model weights  $\boldsymbol{\theta}$  is given by

$$\begin{aligned} p(\boldsymbol{\theta}) &\propto p_0(\boldsymbol{\theta}) \prod_{n=1}^N p(y_n|\boldsymbol{\theta}, \mathbf{x}_n) \\ &\propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\theta} - \sum_{n=1}^N \frac{1}{2\sigma^2} (y_n - \mathbf{x}_n^\top \boldsymbol{\theta})^2\right) \\ &\propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \left(I + \frac{1}{\sigma^2} \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^\top\right) \boldsymbol{\theta} + \left(\frac{1}{\sigma^2} \sum_{n=1}^N y_n \mathbf{x}_n^\top\right) \boldsymbol{\theta}\right) \end{aligned}$$

and so the posterior is Gaussian with covariance matrix and mean given by

$$\Sigma = \left( I + \frac{1}{\sigma^2} \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^\top \right)^{-1} \quad (17a)$$

$$\mu = \left( I + \frac{1}{\sigma^2} \sum_{n=1}^N \mathbf{x}_n \mathbf{x}_n^\top \right)^{-1} \frac{1}{\sigma^2} \sum_{n=1}^N y_n \mathbf{x}_n \quad (17b)$$

### B.1. Non-recovery of posterior distribution

We first consider the case of a variational family of distributions which contains the true posterior distribution for this model. We let  $f(\boldsymbol{\theta})$  be an arbitrary 2-dimensional Gaussian distribution

$$f(\boldsymbol{\theta}) \propto \exp \left( -\frac{1}{2} \boldsymbol{\theta}^\top \Lambda \boldsymbol{\theta} + \boldsymbol{\eta}^\top \boldsymbol{\theta} \right)$$

parametrised by its natural parameters  $\Lambda \in \mathbf{R}^{2 \times 2}$  and  $\boldsymbol{\eta} \in \mathbf{R}^2$ , and consider the variational family of the form  $q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) f(\boldsymbol{\theta})^N$ . As described in the main text, the BB- $\alpha$  optimality equations for the variational distribution  $q$  are given by

$$\mathbf{E}_q [\mathbf{s}(\boldsymbol{\theta})] = \frac{1}{N} \sum_{n=1}^N \mathbf{E}_{\tilde{p}_n} [\mathbf{s}(\boldsymbol{\theta})] \quad (18)$$

where  $\mathbf{s}$  is a vector of sufficient statistics for  $q$ , and  $\tilde{p}_n(\boldsymbol{\theta}) \propto q(\boldsymbol{\theta}) (p(y_n | \boldsymbol{\theta}, \mathbf{x}_n) / f(\boldsymbol{\theta}))^\alpha$  is the tilted distribution for data point  $n$ .

Since the variational family considered is 2-dimensional Gaussian, the sufficient statistics for  $q$  are  $\mathbf{s}(\boldsymbol{\theta}) = ((\theta_i)_{i=1}^2, (\theta_i \theta_j)_{i,j=1}^{2,2})$ . We denote the solution of Equation (18) by  $q^*$ , and denote its mean and variance by  $\mu_{q^*}$  and  $\Sigma_{q^*}$  respectively; we denote the corresponding quantities for the tilted distribution  $\tilde{p}_n$  by  $\tilde{\mu}_n$  and  $\tilde{\Sigma}_n$ . Equation (18) then becomes

$$\mu_{q^*} = \frac{1}{N} \sum_{n=1}^N \tilde{\mu}_n \quad (19a)$$

$$\Sigma_{q^*} + \mu_{q^*} \mu_{q^*}^\top = \frac{1}{N} \left( \sum_{n=1}^N \tilde{\Sigma}_n + \tilde{\mu}_n \tilde{\mu}_n^\top \right) \quad (19b)$$

Or, in terms of natural parameters,

$$\Lambda_{q^*}^{-1} \boldsymbol{\eta}_{q^*} = \frac{1}{N} \sum_{n=1}^N \tilde{\Lambda}_n^{-1} \tilde{\boldsymbol{\eta}}_n \quad (20a)$$

$$\Lambda_{q^*}^{-1} + (\Lambda_{q^*}^{-1} \boldsymbol{\eta}_{q^*}) (\Lambda_{q^*}^{-1} \boldsymbol{\eta}_{q^*})^\top = \frac{1}{N} \sum_{n=1}^N \left( \tilde{\Lambda}_n^{-1} + (\tilde{\Lambda}_n^{-1} \tilde{\boldsymbol{\eta}}_n) (\tilde{\Lambda}_n^{-1} \tilde{\boldsymbol{\eta}}_n)^\top \right) \quad (20b)$$

But the natural parameters of the true posterior are the mean of the natural parameters of the tilted distributions; i.e. power EP returns the true posterior when it lies in the approximating family. Since the map from moments to natural parameters is non-linear, we deduce that the distribution  $q$  fitted according the moment matching equations (20) is not equal to the true posterior for non-zero  $\alpha$ .

### B.2. Example 1

We now provide a concrete example of this phenomenon, by performing the explicit calculations for a toy dataset consisting of two observations. We select our dataset to given by  $\mathbf{x}_1 = (1, 0)^\top$  and  $\mathbf{x}_2 = (0, 1)^\top$ , for now letting the output points  $y_{1:2}$  be arbitrary. In this case, we can read off the mean and covariance of the true posterior from (17):

$$\Sigma = \frac{\sigma^2}{1 + \sigma^2} I_2, \quad \mu = \frac{1}{1 + \sigma^2} (y_1, y_2)^\top$$



Note that in this case the true posterior has a diagonal covariance matrix. We consider fitting a variational family of distributions containing the true posterior via BB- $\alpha$ , to demonstrate that the true posterior is generally not recovered for non-zero  $\alpha$ .

We now suppose that

$$f(\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \Lambda \boldsymbol{\theta} + \boldsymbol{\eta}^\top \boldsymbol{\theta}\right)$$

is constrained to have diagonal precision matrix  $\Lambda$  (and hence covariance matrix). The optimality equation is still given by (18), but the sufficient statistics are now given by the reduced vector  $\mathbf{s}(\boldsymbol{\theta}) = ((\theta_i)_{i=1}^2, (\theta_i^2)_{i=1}^2)$ . Now, we have  $q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta})f(\boldsymbol{\theta})^N$ , and so

$$\begin{aligned} q(\boldsymbol{\theta}) &\propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\theta}\right) \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top (N\Lambda)\boldsymbol{\theta} + N\boldsymbol{\eta}^\top \boldsymbol{\theta}\right) \\ &= \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top (I + N\Lambda)\boldsymbol{\theta} + N\boldsymbol{\eta}^\top \boldsymbol{\theta}\right) \end{aligned}$$

Similarly, note that the tilted distribution  $\tilde{p}_n(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta})f(\boldsymbol{\theta})^{N-\alpha}p(y_n|\boldsymbol{\theta}, \mathbf{x}_n)^\alpha$ . We note that

$$\begin{aligned} p(y_n|\boldsymbol{\theta}, \mathbf{x}_n) &\propto \exp\left(-\frac{1}{2\sigma^2}(y_n - \mathbf{x}_n^\top \boldsymbol{\theta})^2\right) \\ &\propto \exp\left(y_n \mathbf{x}_n^\top \boldsymbol{\theta} - \frac{1}{2\sigma^2}\boldsymbol{\theta}^\top (\mathbf{x}_n \mathbf{x}_n^\top) \boldsymbol{\theta}\right) \end{aligned}$$

So we have

$$\tilde{p}_n(\boldsymbol{\theta}) \propto \exp(-\frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\theta}) \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top ((N-\alpha)\Lambda)\boldsymbol{\theta} + (N-\alpha)\boldsymbol{\eta}^\top \boldsymbol{\theta}\right) \exp\left(\frac{\alpha}{\sigma^2}y_n \mathbf{x}_n^\top \boldsymbol{\theta} - \frac{\alpha}{2\sigma^2}\boldsymbol{\theta}^\top (\mathbf{x}_n \mathbf{x}_n^\top) \boldsymbol{\theta}\right) \quad (21a)$$

$$\propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \left(I + (N-\alpha)\Lambda + \frac{\alpha}{\sigma^2}\mathbf{x}_n \mathbf{x}_n^\top\right) \boldsymbol{\theta} + \left((N-\alpha)\boldsymbol{\eta}^\top + \frac{\alpha}{\sigma^2}y_n \mathbf{x}_n^\top\right) \boldsymbol{\theta}\right) \quad (21b)$$

We can now use these expressions for the natural parameters, together with the optimality conditions (18) to fit the variational family of distributions.

### B.2.1. MATCHING FIRST MOMENTS OF THE VARIATIONAL DISTRIBUTION

Denoting  $\Lambda = \text{diag}(\lambda_1, \lambda_2)$ , setting  $N = 2$  and using the specific values of  $\mathbf{x}_{1:2}$  mentioned above, the optimality equation (20a) implies that we have

$$\frac{2\eta_i}{1 + 2\lambda_i} = \frac{1}{2} \left( \frac{(2-\alpha)\eta_i + \frac{\alpha}{\sigma^2}y_i}{1 + (2-\alpha)\lambda_i + \frac{\alpha}{\sigma^2}} \right) + \frac{1}{2} \left( \frac{(2-\alpha)\eta_i}{1 + (2-\alpha)\lambda_i} \right), \quad i = 1, 2$$

These equations are linear in the components of  $\eta$ , so we have that

$$\eta_i = \frac{(y_i \frac{\alpha}{\sigma^2}) / (1 + (2-\alpha)\lambda_i + \frac{\alpha}{\sigma^2})}{\frac{4}{1+2\lambda_i} - \frac{2-\alpha}{1+(2-\alpha)\lambda_i + \frac{\alpha}{\sigma^2}} - \frac{2-\alpha}{1+(2-\alpha)\lambda_i}}, \quad i = 1, 2$$

yielding the natural parameter  $\eta_i$  in terms of  $\lambda_i$ .

### B.2.2. MATCHING SECOND MOMENTS OF THE VARIATIONAL DISTRIBUTION

The optimality equation (20b) can now be used to recover the precision matrix parameters  $\lambda_{1:2}$ , and hence also the parameters  $\eta_{1:2}$  using the formula derived in the above section. Importantly, recall that since we are dealing only with Gaussian variational distributions with diagonal covariance matrices, we need only the diagonal elements of this matrix equation. The resulting equations for  $\lambda_{1:2}$  are quite involved, so we begin by evaluating the left-hand side of Equation (20b). Evaluating the  $i^{\text{th}}$  diagonal element (where  $i = 1$  or  $2$ ) gives

$$\frac{1}{1 + 2\lambda_i} + \frac{4\eta_i^2}{(1 + 2\lambda_i)^2}$$

We now turn our attention to the right-hand side. We again evaluate the  $i^{\text{th}}$  diagonal element, which yields

$$\frac{1}{2} \left( \frac{1}{1 + (2 - \alpha)\lambda_i + \frac{\alpha}{\sigma^2}} + \left( \frac{(2 - \alpha)\eta_i + \frac{\alpha}{\sigma^2}y_i}{1 + (2 - \alpha)\lambda_i + \frac{\alpha}{\sigma^2}} \right)^2 \right) + \frac{1}{2} \left( \frac{1}{1 + (2 - \alpha)\lambda_i} + \left( \frac{(2 - \alpha)\eta_i}{1 + (2 - \alpha)\lambda_i} \right)^2 \right)$$

Equating the above two expressions in general yields  $\lambda_i$  as a zero of a high-order polynomial, and therefore does not have an analytic solution. However, taking the data points  $y_{1:2}$  to be zero simplifies the algebra considerably, and allows for an analytic solution for the natural parameters to be reached. In this case, the optimality equation yields

$$\frac{1}{1 + 2\lambda_i} = \frac{1}{2} \left( \frac{1}{1 + (2 - \alpha)\lambda_i + \frac{\alpha}{\sigma^2}} + \frac{1}{1 + (2 - \alpha)\lambda_i} \right), \quad i = 1, 2$$

Solving this (with the constraint that  $\lambda_i > 0$ ) for the positive solution gives

$$\lambda_i = \frac{\sqrt{\alpha^2 - 2\alpha + (\sigma^2 + 1)^2} - \alpha - \sigma^2 + 1}{2\sigma^2(2 - \alpha)}, \quad i = 1, 2$$

and

$$\lambda_i = \frac{-\sqrt{\alpha^2 - 2\alpha + (\sigma^2 + 1)^2} - \alpha - \sigma^2 + 1}{2\sigma^2(2 - \alpha)}, \quad i = 1, 2 \text{ when } \alpha > 2$$

Plotting a diagonal element of the variational covariance matrix as a function of  $\alpha$  gives the curve plotted in Figure 5. This plot demonstrates that the variance of the fitted distribution increases continuously and monotonically with  $\alpha$  in the range  $(0, 2)$ .

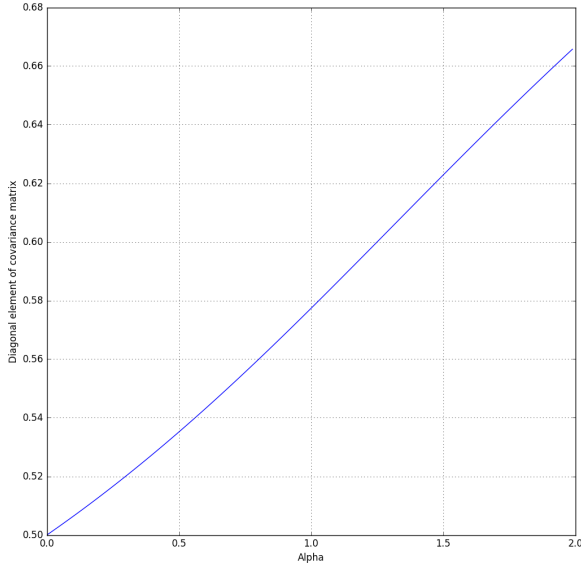


Figure 5. Diagonal element of fitted covariance matrix against  $\alpha$ , using output data points  $y_1 = y_2 = 0$  and  $\sigma^2 = 1$

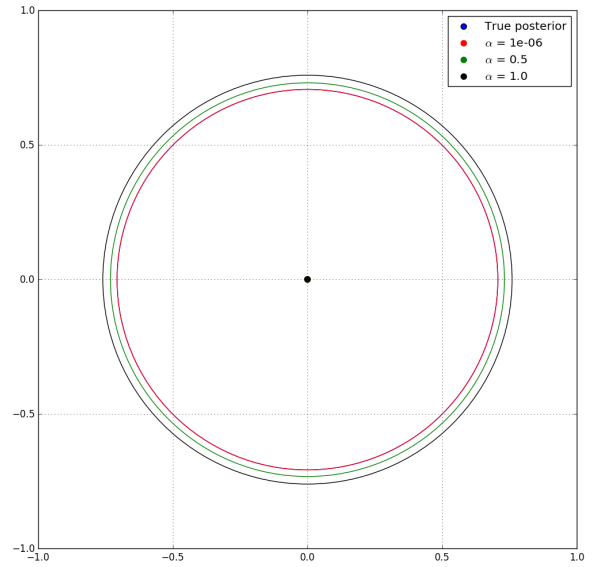


Figure 6. Plot of the mean and one standard deviation's confidence region for the true posterior and several BB- $\alpha$  approximations. Output data is set to  $y_1 = 0$ ,  $y_2 = 0$  and  $\sigma^2 = 1$

With the fitted mean and covariance matrix, we can plot the fitted approximate posterior distributions as  $\alpha$  varies, along with the true posterior - see Figure 6. Note that in the limit as  $\alpha \rightarrow 0$ , the true posterior is recovered by BB- $\alpha$ , as the corresponding  $\alpha$ -divergence converges to the KL-divergence; in fact, the true posterior and the fitted approximation for  $\alpha = 10^{-6}$  are indistinguishable at this scale. Note that for non-zero  $\alpha$ , the fitted covariance parameters differ from those of the true posterior.

### B.3. Mean-field approximations

We now demonstrate the behaviour of BB- $\alpha$  in fitting mean-field Gaussian distributions, showing that smooth interpolation between KL-like and EP-like behaviour can be obtained by varying  $\alpha$ .

## B.4. Example 2

We now consider the case where the true posterior has non-diagonal covariance matrix; we set the input values in our toy dataset to be  $\mathbf{x}_1 = (1, -1)^\top$ ,  $\mathbf{x}_2 = (-1, 1)^\top$  (again leaving the output data  $y_{1:2}$  arbitrary for now), and note from Equation (17) that this implies the mean and covariance of the true posterior are given by

$$\Sigma = \frac{1}{4 + \sigma^2} \begin{pmatrix} \sigma^2 + 2 & 2 \\ 2 & \sigma^2 + 2 \end{pmatrix}$$

$$\mu = \left( \frac{y_1 - y_2}{4 + \sigma^2}, \frac{y_2 - y_1}{4 + \sigma^2} \right)^\top$$

We fit the same variational family considered in Example 1; namely the family in which

$$f(\boldsymbol{\theta}) \propto \exp \left( -\frac{1}{2} \boldsymbol{\theta}^\top \Lambda \boldsymbol{\theta} + \eta^\top \boldsymbol{\theta} \right)$$

is constrained to have diagonal precision matrix  $\Lambda$  (and hence covariance matrix). We can now use this information, together with the form of the tilted distributions in Equation (21) and the optimality conditions in Equation (18) to fit the variational family of distributions.

### B.4.1. MATCHING FIRST MOMENTS OF THE VARIATIONAL DISTRIBUTION

Denoting  $\Lambda = \text{diag}(\lambda_1, \lambda_2)$ , substituting in  $N = 2$  and using our specific choice of data points  $\mathbf{x}_{1:2}$ , Equation (20a) yields the linear system

$$\begin{pmatrix} \frac{2}{1+2\lambda_1} & 0 \\ 0 & \frac{2}{1+2\lambda_2} \end{pmatrix} \eta = \begin{pmatrix} 1 + (2 - \alpha)\lambda_1 + \frac{\alpha}{\sigma^2} & -\frac{\alpha}{\sigma^2} \\ -\frac{\alpha}{\sigma^2} & 1 + (2 - \alpha)\lambda_2 + \frac{\alpha}{\sigma^2} \end{pmatrix}^{-1} \begin{pmatrix} (2 - \alpha)\eta + \frac{\alpha}{2\sigma^2} (y_1 - y_2) \\ (2 - \alpha)\eta + \frac{\alpha}{2\sigma^2} (y_2 - y_1) \end{pmatrix}$$

This linear system can be solved for  $\eta$ , although in general the solution is a complicated rational function of  $(\lambda_1, \lambda_2)$ . However, taking  $y_1 = y_2$  yields  $\eta = 0$ .

### B.4.2. MATCHING THE SECOND MOMENTS OF THE VARIATIONAL DISTRIBUTION

With the above choices for  $y_1$  and  $y_2$ , it follows by symmetry that we must have  $\lambda_1 = \lambda_2$ . We denote this unknown variable by  $\lambda$  in what follows. Considering the diagonal elements of the Equation (20b) then yields

$$\frac{1}{1 + 2\lambda_i} = \frac{1 + (2 - \alpha)\lambda_i + \alpha/\sigma^2}{(1 + (2 - \alpha)\lambda_i + \alpha/\sigma^2)^2 - \alpha^2/\sigma^4} + \frac{y^2 \alpha^2}{\sigma^4} \frac{(1 + (2 - \alpha)\lambda_i)^2}{(1 + (2 - \alpha)\lambda_i + \alpha/\sigma^2)^2 - \alpha^2/\sigma^4}$$

This can be re-arranged into a cubic for  $\lambda$  and thus solved analytically, at least in theory; the resulting expression for  $\lambda$  in terms of  $\alpha, \sigma$  and  $y$  is lengthy in practice and is therefore omitted here. We instead consider the case  $y = y_1 = y_2 = 0$ , where the algebra is more tractable. This results in the following equation for  $\lambda$ :

$$\frac{1}{1 + 2\lambda_i} = \frac{1 + (2 - \alpha)\lambda_i + \alpha/\sigma^2}{(1 + (2 - \alpha)\lambda_i + \alpha/\sigma^2)^2 - \alpha^2/\sigma^4}, \quad i = 1, 2$$

This equation is merely quadratic, and hence has a more easily expressible solution; solving this equation (with the constraint  $\lambda_i > 0$ ) gives the precision parameters as

$$\lambda_i = \frac{\sqrt{4\alpha^2 - 8\alpha + \sigma^4 + 4\sigma^2 + 4} - (2\alpha + \sigma^2 - 2)}{2\sigma^2(2 - \alpha)}, \quad i = 1, 2$$

and

$$\lambda_i = \frac{-\sqrt{4\alpha^2 - 8\alpha + \sigma^4 + 4\sigma^2 + 4} - (2\alpha + \sigma^2 - 2)}{2\sigma^2(2 - \alpha)}, \quad i = 1, 2 \text{ when } \alpha > 2$$

Plotting a diagonal element of the corresponding fitted covariance matrix as a function of  $\alpha$  gives the curve shown in Figure 7. Note that as before, the element exhibits a continuous, monotonic dependence on  $\alpha$  in the range  $(0, 2)$ .

With the fitted mean and covariance matrix, we can plot the fitted approximate posterior distributions as  $\alpha$  varies, along with the true posterior - see Figure 8. Note that in the limit as  $\alpha \rightarrow 0$ , the BB- $\alpha$  solution mimics the KL fit, exhibiting low variance relative to the true posterior, and as  $\alpha$  increases, so does the spread of the distribution, consistent with Figure 7.

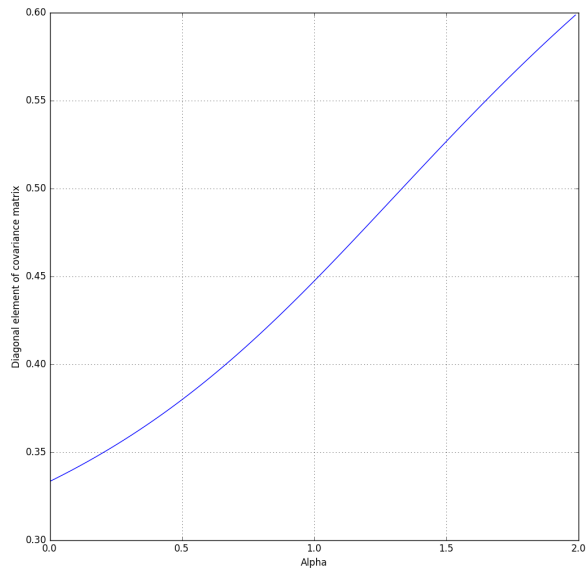


Figure 7. Diagonal element of fitted covariance matrix against  $\alpha$ , using output data points  $y_1 = 0$ ,  $y_2 = 0$  and  $\sigma^2 = 1$

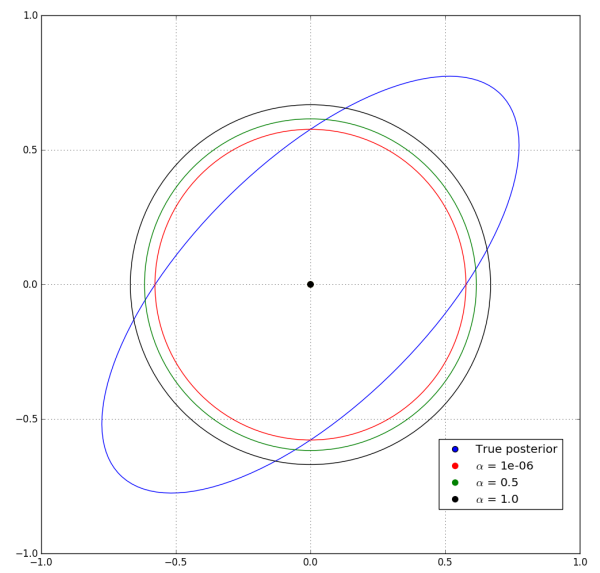


Figure 8. Plot of mean and one standard deviation's confidence region for the true posterior and several BB- $\alpha$  approximations. Output data is set to  $y_1 = 0$ ,  $y_2 = 0$  and  $\sigma^2 = 1$