Approximate inference for the loss-calibrated Bayesian

Anonymous Author(s)
Top Secret Institution(s)

Abstract

We consider the problem of approximate inference in the context of Bayesian decision theory. Traditional approaches focus on approximating general properties of the posterior, ignoring the decision task – and associated losses – for which the posterior could be used. We argue that this can be suboptimal and propose instead to *loss-calibrate* the approximate inference methods with respect to the decision task at hand. We present a general framework rooted in Bayesian decision theory to analyze approximate inference from the perspective of losses, opening up several research directions. We propose an EM-like algorithm for loss-calibrated inference and show how it can improve a standard approach to Gaussian process classification when losses are asymmetric.

1 INTRODUCTION

Bayesian methods have enjoyed a surge of popularity in machine learning over the last decade. Even though it is sometimes overlooked, the main theoretical motivations for the Bayesian paradigm are rooted in Bayesian decision theory [3], which provides a well-defined theoretical framework for rational decision making under uncertainty about a hidden parameter θ. The ingredients of Bayesian decision theory are an observation model $p(\mathcal{D}|\theta)$, a prior distribution $p(\theta)$, and a loss $L(\theta, a)$ of actions $a \in \mathcal{A}$. In this framework, the optimal action is chosen by minimizing the expected loss of the action over the posterior $p(\theta|\mathcal{D})$, that is obtained from the prior and the observation model via Bayes' rule. The independence of the posterior from the loss motivates the common practice of breaking decision making into two independent sub-problems: *inference*, whereby the posterior $p(\theta|\mathcal{D})$ is computed irrespectively of the loss; and then *decision*, whereby an action is chosen to minimize our loss given our posterior belief.

In practically interesting Bayesian models, however, the posterior is often computationally intractable and therefore one has to resort to approximate inference techniques, such as variational methods or Markov chain Monte Carlo. Most approaches to approximate inference ignore the decision theoretic loss and try to approximate the posterior based on its general features, such as matching its mode or higher order moments. While this is probably a reasonable approach for the simple losses usually considered or when the loss is unknown, they might fail to work well with asymmetric, non-trivial losses that appear in modern applications in machine learning.

The main message of the present paper is that when inference is carried out only approximately, treating (approximate) inference and decision making independently is not well motivated, and that when we know what the loss is, ignoring it in approximate inference can lead to suboptimal decisions. We note that a similar philosophy has already been widely applied in the frequentist discriminative machine learning literature, as for example with the use of *surrogate loss functions* [2, 17]. We will focus instead on the pure subjectivist Bayesian viewpoint in this paper as we are not yet aware of the existence of such an investigation in this case. The contributions of the present paper can be summarized as follows:

1. In section 2, we propose a general approximate inference framework based on Bayesian decision theory. The framework naturally gives rise to a divergence between distributions that can be seen as a loss-calibrated generalization of the Kullback-Leibler divergence for general losses. Here we will focus on the application of the framework to the predictive setting that is relevant to supervised machine learning applications.

2. In section 3, we present a loss-calibrated approximate inference algorithms for general losses by applying the variational Expectation-Maximization

Preliminary work. Under review by AISTATS 2011. Do not distribute.

algorithm on the Bayesian posterior risk.

3. In section 4, we investigate our approximation framework on the concrete setup of supervised learning. We apply the loss-calibrated EM algorithm to a Gaussian process classification model and analyze its performance in terms of the loss-calibrated framework. We show that it improves over a loss-insensitive approximate inference alternative and that the advantage of loss-calibration is more prominent when misclassification losses are asymmetric.

2 BAYESIAN DECISION THEORY

We use Bayesian statistical decision theory as the basis of our analysis (see chapter 2 of Robert [14] or chapter 1 of Berger [3] for example). We review here its main ingredients:

- a (statistical) loss $L(\theta, a)$ which gives the cost of taking action $a \in \mathcal{A}$ when the world state is $\theta \in \Theta$;
- an observation model $p(\mathcal{D}|\theta)$ which gives the probability of observing $\mathcal{D} \in \mathcal{O}$ assuming that the world state is θ;
- a prior belief $p(\theta)$ over world states.

The loss L describes the decision task that we are interested in, whereas the observation model and the prior represent our beliefs about the world. Given these, the ultimate evaluation metric for a possible action a after observing $\mathcal{D}$ is the *expected posterior loss* (also called the *posterior risk* [15]): $\mathcal{R}_{p_\mathcal{D}}(a) \doteq \int_\Theta L(\theta, a)\, p(\theta|\mathcal{D}) d\theta$. In this framework, the (Bayes) optimal action $a_{p_\mathcal{D}}$ is the one which minimizes $\mathcal{R}_{p_\mathcal{D}}$.

2.1 Supervised learning

We now relate the abstract decision theory setup with the typical supervised learning applications of machine learning. For a prediction task, the goal is to estimate a function $h : \mathcal{X} \to \mathcal{Y}$ where the output space $\mathcal{Y}$ can be discrete (classification) or continuous (regression). We suppose that we are given a fixed cost function $\ell(y, y')$ which gives the cost of predicting y' when the true output was y. We can cast this problem in the standard statistical decision theory setting by defining a suitable prediction loss, which is simply the standard generalization error from machine learning:

$$L(\theta, h) \doteq \mathbb{E}_{(x,y)\sim p(x,y|\theta)} \left[\ell\left(y, h(x)\right)\right]. \qquad (1)$$

For the observation model, we will assume that we are given a training set $\mathcal{D} = \{(x_i, y_i)_{i=1}^N\}$ of labelled observations generated i. i. d. from $p(x, y|\theta)$. The goal of the learning algorithm is then to output a function $h = \delta(\mathcal{D})$ chosen from a set of (possibly non-parametric) hypothesis $\mathcal{H}$. From the pure Bayesian point of view, the optimal action (hypothesis) is clear: it is the one which minimizes the posterior risk $h_{p_\mathcal{D}} \doteq \arg\min_{h\in\mathcal{H}} \mathcal{R}_{p_\mathcal{D}}(h)$.

2.2 General approximation framework

The quantity central to the Bayesian methodology is the posterior $p_\mathcal{D} \doteq p(\theta|\mathcal{D})$ which summarizes our uncertainty about the world. On the other hand, it is rarely computable in a tractable form, and so it is usually approximated with a tractable approximate distribution $q(\theta) \in \mathcal{Q}$. Popular approaches to this problem include sampling, variational inference – minimizes $d_{KL}(q\|p_\mathcal{D})$, expectation propagation – minimizes $d_{KL}(p_\mathcal{D}\|q)$). Most approximate inference approaches stop here, though in the context of decision theory, we still need to *act*. In practice, one usually treats the approximate q as if it was the true posterior and chooses the action which minimizes what we will call the *q-risk*:

$$\mathcal{R}_q(h) \doteq \int_\Theta q(\theta) L(\theta, h) d\theta, \qquad (2)$$

obtaining a *q-optimal* action h_q:

$$h_q \doteq \arg\min_{h\in\mathcal{H}} \mathcal{R}_q(h). \qquad (3)$$

It is clear that the posterior risk of any decision h is lower bounded by the posterior risk of the optimal action under the posterior. It therefore makes sense to say that an approximate distribution q is close to the posterior, if the posterior risk of the q-optimal decision is close to the posterior risk of the $p_\mathcal{D}$-optimal decision. Therefore we propose the following criterion for choosing the approximate q:

$$q_{\text{opt}} = \arg\min_{q\in\mathcal{Q}} \mathcal{R}_{p_\mathcal{D}}(h_q). \qquad (4)$$

In the case where $p_\mathcal{D} \in \mathcal{Q}$, $p_\mathcal{D}$ will be optimal according to this criterion. It is important to note though that the optimum is not necessarily unique. We could interpret the above criterion as minimizing the following asymmetric discrepancy measure between distributions:

$$d_L(p\|q) \doteq \mathcal{R}_p(h_q) - \mathcal{R}_p(h_p) \qquad (5)$$

Interestingly, the Kullback-Leibler divergence $d_{KL}(p\|q)$ can be interpreted as a special case of d_L for the following task: an action h is a density over Θ; the loss is the surprisal $L(\theta, h) = -\log h(\theta)$. The q-risk $R_q(h)$ then becomes the cross-entropy that measures the cost of coding $\theta \sim p$ according to q. Therefore d_{KL} is calibrated to description length, which is a sensible loss for

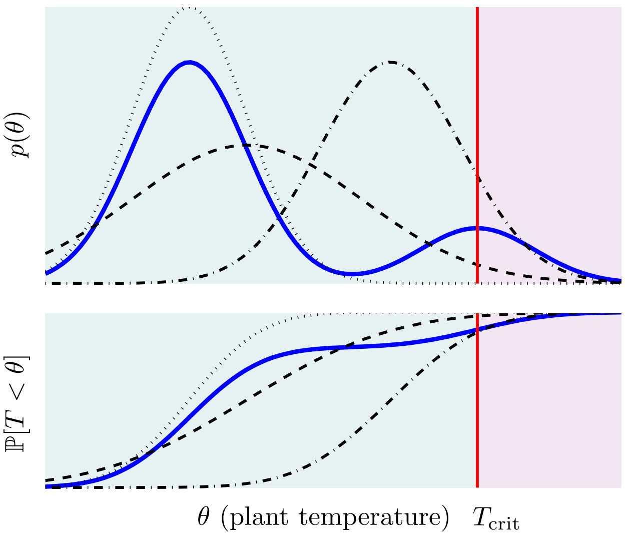

Figure 1: **Top:** Real bimodal posterior *(blue)* and three Gaussian approximations obtained by minimizing $d_{KL}(q\|p)$ *(q_1, dotted)*, $d_{KL}(p\|q)$ *(q_2, dashed)* or $d_L(p\|q)$ *(q_3, dash-dotted)* in the power plant example. **Bottom:** Cumulative distribution functions for the posterior and the three approximate distributions.

unsupervised learning, where the goal is to construct a compact meaningful summary of observed data. But this begs the natural question, whether minimizing d_L for a particular loss L provides optimal performance under other losses. We will show in section 4.2 that even in the simple Gaussian linear regression setting, minimizing the KL divergence can be suboptimal in the squared loss sense.

To illustrate the difference between traditional approaches to approximate inference and the loss-calibrated framework, consider the following simple problem. Suppose that we control a nuclear power-plant. We estimate the temperature, θ, of the plant via Bayesian inference based on measurements $\mathcal{D}$. The plant is in danger of over-heating, and as operator, we can take two actions: either shut-down the power-plant for a day or keep it running. If the temperature rises above a critical temperature T_{crit}, keeping it running will cause a nuclear meltdown, incurring a large loss $L(\theta > T_{\text{crit}}, \text{'on'})$. On the other hand, shutting down the power plant incurs a moderate loss $L(\text{'off'})$, irrespective of temperature. Given our posterior $p_{\mathcal{D}}$ and the losses, we want to compute the Bayes-optimal action that minimises the posterior risk. The posterior $p_{\mathcal{D}}(\theta)$ (figure 1, solid curve) is a complicated multi-modal distribution so we chose to approximate it with a Gaussian. Now consider how various approaches would perform in terms of their Bayesian posterior risk. Our first candidate, q_1 minimizes $d_{KL}(q\|p_{\mathcal{D}})$, which leads to a mode-seeking behaviour, and therefore concentrates around the largest mode, ignoring entirely the second, small mode around the critical temperature (figure 1, dotted curve). Minimizing $d_{KL}(p_{\mathcal{D}}\|q)$ gives a more global approximation: q_2 matches moments of the posterior, but still underestimates the probability of the temperature being above T_{crit}, thereby potentially leading to sub-optimal decision (figure 1, dashed curve). q_3 is an example of q_{opt} in this setting, resulting in the same decision as $p_{\mathcal{D}}$ (figure 1, dash-dotted curve). Note that q_3 does not model all aspects of the posterior, but it estimates the Bayes-decision well. Because there are only two possible actions in this setup, the set $\mathcal{Q}$ is split in only two halves by the function $d_L(p_{\mathcal{D}}, q)$ and so there are infinitely many q_{opt}'s that are equivalent in terms of their risk. In contrast, in the predictive setting of section 2.1 where in addition we assume $\mathcal{X}$ and $p(x)$ to be continuous, we could obtain a finer resolution $d_L(p_{\mathcal{D}}, q)$ which can potentially yield a unique optimizer.

3 LOSS-CALIBRATED EM

Minimizing d_L for a general posterior and general loss is a non-trivial problem. Expectation-propagation [9] is an approach for minimizing $d_{KL}(p_{\mathcal{D}}\|q)$ – which, as discussed, is a special case of d_L – but it relies on properties of the logarithm in the loss function, and therefore it does not generalise for general L. As an alternative, we propose a variational algorithm.

Recall that our general goal is to find an action $h_{p_{\mathcal{D}}}$ which minimizes the Bayesian posterior risk $\mathcal{R}_{p_{\mathcal{D}}}$:

$$h_{p_{\mathcal{D}}} = \operatorname*{arg\,min}_{h \in \mathcal{H}} \int_{\Theta} p(\theta|\mathcal{D}) L(\theta, h) d\theta. \tag{6}$$

The problem combines integration and optimization, and we have to find approximate solutions that produce actions with low posterior risk without having to represent the posterior exactly. One way to solve the chicken and egg problem of integration vs. optimization is to employ a strategy used by the well-known Expectation-Maximization (EM) algorithm [5] which is normally applied to maximize the marginal likelihood which is a similar intractable integral. EM can be derived from Jensen's inequality and doing coordinate ascent on a lower bound of the log-likelihood. In order to re-use this strategy here, we need to move from minimization to maximization to obtain inequalities in the correct direction. Assuming from now on that our loss function is bounded, we thus define the following *utility* function:

$$U_M(\theta, h) \doteq M - L(\theta, h) \tag{7}$$

where M is a fixed finite constant chosen so that $M > \sup_{\theta \in \Theta, h \in \mathcal{H}} L(\theta, h)$, hence $U_M(\theta, h) > 0$. In analogy

$$\begin{array}{ll}
\text{(E-step)} & q^{t+1} = \underset{q \in \mathcal{Q}}{\arg\min}\ KL\left(q \,\|\, \frac{p_{\mathcal{D}}(\cdot)U_M(\cdot, h^t)}{\mathcal{G}_{p_{\mathcal{D}}}(h^t)}\right) \\
\text{(M-step)} & h^{t+1} = \underset{h \in \mathcal{H}}{\arg\max} \int_{\Theta} q^{t+1}(\theta)\, \log U_M(\theta, h) d\theta
\end{array}$$

Table 1: Loss-EM updates

$$\begin{array}{llll}
\text{(E-step)} & q^{t+1} & = & \underset{q \in \mathcal{Q}}{\arg\min}\ KL\left(q \| p_{\mathcal{D}}\right) + \frac{\mathcal{R}_q(h^t)}{M} \\
\text{(M-step)} & h^{t+1} & = & \underset{h \in \mathcal{H}}{\arg\min}\ \mathcal{R}_{q^{t+1}}(h)
\end{array}$$

Table 2: Linearized loss-EM updates

with the q-risk $\mathcal{R}_q$, we define the *q-gain* $\mathcal{G}_q$:

$$\mathcal{G}_q(h) \doteq \int_{\Theta} q(\theta) U_M(\theta, h) d\theta. \tag{8}$$

Minimizing the q-risk is equivalent to maximizing the q-gain, as well as the log of the q-gain. So we have:

$$h_{p_{\mathcal{D}}} = \underset{h \in \mathcal{H}}{\arg\max}\ \log\left(\int_{\Theta} p_{\mathcal{D}}(\theta) U_M(\theta, h) d\theta\right) \tag{9}$$

which is the optimization problem we will approximate with (variational) EM.

3.1 Variational EM derivation

Assuming that $q(\theta) = 0 \Rightarrow p_{\mathcal{D}}(\theta) = 0$, we obtain the following lower bound from Jensen's inequality:

$$\begin{aligned}\log\left(\mathcal{G}_{p_{\mathcal{D}}}(h)\right) &= \log\left(\int_{\Theta} q(\theta)\, \frac{p_{\mathcal{D}}(\theta) U_M(\theta, h)}{q(\theta)} d\theta\right) \quad (10)\\ &\geq \int_{\Theta} q(\theta) \log\left(\frac{p_{\mathcal{D}}(\theta) U_M(\theta, h)}{q(\theta)}\right) d\theta \doteq \mathcal{L}(q, h)\end{aligned}$$

EM amounts to maximizing the lower bound functional $\mathcal{L}(q, h)$ by coordinate ascent on q and h: the E-step computes $q^{t+1} = \arg\max_{q \in \mathcal{Q}} \mathcal{L}(q, h^t)$, while the M-step computes $h^{t+1} = \arg\max_{h \in \mathcal{H}} \mathcal{L}(q^{t+1}, h)$. Moreover, the difference between the quantity we want to maximize and the lower bound is $\log\left(\mathcal{G}_{p_{\mathcal{D}}}(h)\right) - \mathcal{L}(q, h) = d_{KL}(q \| \tilde{p}_h)$ where

$$\tilde{p}_h(\theta) \doteq \frac{p_{\mathcal{D}}(\theta) U_M(\theta, h)}{\mathcal{G}_{p_{\mathcal{D}}}(h)}, \tag{11}$$

and so the E-step is equivalently minimizing $d_{KL}(q \| \tilde{p}_h)$ as h is fixed. We summarize the obtained updates in table 1 for what we will call the *loss-EM algorithm*. As in standard EM, if exact minimisation is possible in the E step, the lower bound becomes tight and the gain is guaranteed not to decrease after each full iteration. If the E-step is approximate, we can still apply the algorithm, but then we only optimize a lower bound.

3.2 Linearized loss-EM

Although loss-EM produces a decision h which has good risk, this h is not guaranteed to minimize the q-risk, for any particular q, and as such the algorithm does not provide us with a loss-calibrated approximate distribution q, as in section 2.2. Also, the objective function in the M-step can be hard to compute and minimize. To address both of these issues, we suggest another approximation. In particular, using the fact that for $M >> L$, $\log(1 - L/M) = -L/M + O(L^2/M^2)$, we can linearize the $\log U_M$ term in the loss-EM updates to obtain the linearized loss-EM updates given in table 2. Recall that M was a constant chosen by us: it does not change the optimal action $h_{p_{\mathcal{D}}}$, still it influences the behavior of the loss-EM algorithm. For large $M \to \infty$, the linearized and the original algorithms become almost equivalent. On the other hand, we can see that for a large enough M, both algorithms reduce to the standard variational inference algorithm that minimizes $d_{KL}(q \| p_{\mathcal{D}})$. Thus, we can see that the constant M acts as a parameter for our algorithm which allows us to interpolate between the standard KL approach and the loss-EM.

The linearized loss-EM algorithm is not only analytically more convenient than the original algorithm, but it also ensures that the decision in the M-step is optimal under the distribution q obtained in the previous E-step, and therefore we can use q as a loss-calibrated approximate posterior.

4 SUPERVISED LEARNING

In this section, we make our framework more concrete by investigating it in the predictive setting presented in section 2.1. We recall that in order to apply our framework, we need to specify the loss, the action space and the Bayesian observation model. In the predictive setting, the action space is a subset of functions from $\mathcal{X}$ to $\mathcal{Y}$. We will consider a fully non-parametric setting here and let $\mathcal{H}$ be the set of all such functions. We will also use Bayesian non-parametric probabilistic models based on Gaussian processes [13]. In section 4.2, we first look at Gaussian process regression. We can obtain in this case an analytic form for $p_{\mathcal{D}}$ and $\mathcal{R}_{p_{\mathcal{D}}}(h_q)$ which give us some insights about the approximation framework as well as when KL can be suboptimal. We cannot directly apply our loss-EM algorithm in this case unfortunately as the quadratic cost function is not bounded (and so $M = \infty$). We can get nevertheless useful insights which suggest future research directions for regression. In section 4.3, we consider Gaussian process classification (GPC) which will provide a test bed for the loss-EM algorithm. For both regression and classification, we will look at the discriminative regime inasmuch we are not modelling the marginal distribution of x: we assume that we are given a fixed test distribution $p(x)$ which enters in the generalization error $L(\theta, h)$ given by (1). We could think of this distribution as coming from a large unlabeled corpus of examples or from the transductive setting which specifies where we want to make predictions. In both cases, we use a Gaussian process as our prior over parameters. In order to avoid having to define the KL divergence on infinite dimensional objects though, we review in the next section how we can express our analysis using a finite dimensional parameter space.

4.1 Finite parametrization of GP

For Gaussian process regression, we use a Gaussian observation model $p(y|x, f) = \mathcal{N}(y|f(x), \sigma^2)$ and put a GP prior over the function $f : \mathcal{X} \to \mathbb{R}$; $p(f) = GP(f|0, K)$ where σ^2 is the observation noise hyperparameter and $K(\cdot, \cdot)$ is the covariance kernel for the GP. We assume that $\mathcal{D} = \{(x_i, y_i)_{i=1}^N\}$ with y_i coming iid from $p(y|x, f)$, but we *do not* assume that x_i comes from $p(x)$ – for example the inputs could even be chosen deterministically. In this predictive setting, the loss $L(f, h)$ takes the form:

$$L(f, h) = \int_{\mathcal{X}} p(x) \left(\int_{\mathcal{Y}} p(y|x, f)\ell(y, h(x))dy \right) dx. \tag{12}$$

Let $f_{\mathcal{D}}$ be the vector of values of f on the inputs of the dataset: $f_{\mathcal{D}} = (f(x_1), \ldots, f(x_N))^\top$, and let f_{rest} be the values of f on the complement of $\mathcal{D}$. Because of the above conditional independence assumptions, we have that the posterior factorizes: $p(f|\mathcal{D}) = p(f_{\text{rest}}|f_{\mathcal{D}})p(f_{\mathcal{D}}|\mathcal{D})$. By using the linearity of expectations and interchanging the order of integration, the posterior risk thus becomes:

$$\mathcal{R}_{p_{\mathcal{D}}}(h) = \tag{13}$$
$$\int_{\mathbb{R}^N} p(f_{\mathcal{D}}|\mathcal{D}) \left(\int_{\mathcal{X},\mathcal{Y}} p(x)\tilde{p}(y|x, f_{\mathcal{D}})\ell(y, h(x))dydx \right) df_{\mathcal{D}}$$

where we have defined:

$$\begin{aligned} \tilde{p}(y|x, f_{\mathcal{D}}) &\doteq \int p(y|x, f)p(f_{\text{rest}}|f_{\mathcal{D}})df_{\text{rest}} \\ &= \mathcal{N}\big(y|\mu_x(f_{\mathcal{D}}), \sigma^2 + \sigma_x^2\big) . \end{aligned} \tag{14}$$

The Gaussian expression in (14) is from standard properties of GP (basically coming from conditional independence and the conditioning formula for multivariate normals); by doing the change of variable[1] $\theta = K_{\mathcal{D}\mathcal{D}}^{-1} f_{\mathcal{D}}$, we get the expressions

$$\begin{aligned} \mu_x(\theta) &\doteq K_{x\mathcal{D}}\theta \\ \sigma_x^2 &\doteq K_{xx} - K_{x\mathcal{D}}K_{\mathcal{D}\mathcal{D}}^{-1}K_{\mathcal{D}x}, \end{aligned} \tag{15}$$

where we defined $K_{\mathcal{D}\mathcal{D}}$ as the $N \times N$ matrix with (i, j) entry $K(x_i, x_j)$, $K_{x\mathcal{D}}$ as the $1 \times N$ row vector with i^{th} entry $K(x, x_i)$, etc[2]. Because our analysis is *conditioned* on the data, in term of posterior risk optimization, we can *equivalently* redefine our probabilistic model using a finite parameter vector θ of size N as follows:

$$p(\theta) = \mathcal{N}(\theta|0, K_{\mathcal{D}\mathcal{D}}^{-1}) \tag{16}$$

$$p(y|x, \theta) = \mathcal{N}(y|\mu_x(\theta), \sigma^2 + \sigma_x^2) \tag{17}$$

with $\mu_x(\theta)$ and σ_x^2 defined (conditionally from $\mathcal{D}$) in equation (15). We can then use the loss $L(\theta, h)$ defined in term of $p(y|x, \theta)$ instead of $L(f, h)$. We stress that we are only interested in finding h which minimizes the posterior risk; we are not considering for example updating a posterior with more observations. With this finite parametrization, we are ready to analyze the q-risk for GP regression.

4.2 Gaussian process regression

Following the standard convention for regression, we consider the quadratic cost function $\ell(y, y') = (y - y')^2$. The q-risk then has the simple form:

$$\mathcal{R}_q(h) = \mathbb{E}_{x\sim p(x)} \left[Var_q[Y|x] + (\mathbb{E}_q[Y|x] - h(x))^2 \right] \tag{18}$$

where $\sigma_q^2(x) \doteq Var_q[Y|x]$ and $\mu_q(x) \doteq \mathbb{E}_q[Y|x]$ are respectively the conditional variance and conditional mean of the q-marginalized predictive likelihood that we denote by $p_q(y|x) \doteq \int_\Theta q(\theta)p(y|x, \theta)d\theta$. In the case of non-parametric h, the q-optimal action for any q can be obtained by pointwise minimization: $h_q(x) = \mu_q(x)$. We now suppose that q is a Gaussian $q(\theta) = \mathcal{N}(\mu_q, \Sigma_q)$; then $\mu_q(x) = K_{x\mathcal{D}}\mu_q$; note that the optimal action doesn't depend on Σ_q, and so the Bayesian posterior risk is agnostic to which Σ_q we use. Note that $p_{\mathcal{D}}$ is also a Gaussian, with mean $\mu_{p_{\mathcal{D}}} = (K_{\mathcal{D}\mathcal{D}} + \sigma^2 I)^{-1}\mathbf{y}$ and covariance $\Sigma_{p_{\mathcal{D}}} = K_{\mathcal{D}\mathcal{D}}^{-1} - (K_{\mathcal{D}\mathcal{D}} + \sigma^2 I)^{-1}$ (recall that we did the change of variable $\theta = K_{\mathcal{D}\mathcal{D}}^{-1} f_{\mathcal{D}}$) where $\mathbf{y}$ is the vector of outputs $(y_1, \ldots, y_N)^\top$, and so $q(\theta)$ has the capacity to exactly match $p_{\mathcal{D}}$. We are now in position to have an explicit expression for the posterior

[1] The change of variable gets rid of the inverse of $K_{\mathcal{D}\mathcal{D}}$ for prediction.

[2] We note that this formula yields $\sigma_x^2 = 0$ for $x \in \mathcal{D}$, as it should since in this case $\tilde{p}(y|x, f_{\mathcal{D}}) = p(y|x, f)$.

risk of h_q:

$$\mathcal{R}_{p_\mathcal{D}}(h_q) = \int_\mathcal{X} \underbrace{p(x)\sigma_{p_\mathcal{D}}(x)}_{\text{constant wrt } q}\, dx + (\mu_q - \mu_{p_\mathcal{D}})^\top \Lambda (\mu_q - \mu_{p_\mathcal{D}}) \tag{19}$$

where

$$\Lambda \doteq \int_\mathcal{X} p(x) K_{\mathcal{D}x} K_{x\mathcal{D}} dx \tag{20}$$

is a loss-sensitive term (i.e. is sensitive to where the test set distribution $p(x)$ lies). It is interesting to compare the posterior risk objective (19) with the KL between two Gaussians:

$$d_{KL}(q||p_\mathcal{D}) = c(\Sigma_q) + \frac{1}{2}(\mu_q - \mu_{p_\mathcal{D}})^\top \Sigma_{p_\mathcal{D}}^{-1} (\mu_q - \mu_{p_\mathcal{D}}) \tag{21}$$

where $c(\Sigma_q)$ is constant with respect to μ_q. We note that by using the block matrix inversion lemma, we can see that $\Sigma_{p_\mathcal{D}}^{-1} = K_{\mathcal{D}\mathcal{D}} + \sigma^{-2} K_{\mathcal{D}\mathcal{D}}^2$ and so is different from Λ. Even if we use the empirical distribution on $\mathcal{D}$ as the test distribution $p(x)$, then we get $\Lambda = K_{\mathcal{D}\mathcal{D}}^2/N$ which is still missing an additive $K_{\mathcal{D}\mathcal{D}}$ to become proportional to $\Sigma_{p_\mathcal{D}}^{-1}$. This means that unless $\exists q \in \mathcal{Q}$ s.t. $\mu_q = \mu_{p_\mathcal{D}}$ (i.e. we can match the mean), the minimum KL solution won't necessarily minimize the Bayesian posterior risk. Because it takes $O(N^3)$ to compute $\mu_{p_\mathcal{D}}$ due to the inversion of the kernel matrix, some proposals have been made in the GP literature to use a *sparse* μ_q instead [12]. It indeed turns out that minimizing the above KL over $\mathcal{Q}$ with a sparse restriction on μ_q can be done efficiently. In particular, if we partition the set of indices of the dataset into a fixed set S of size k for the non-zero coefficient of μ_q and T for the set of coefficients that we constraint to zero, then the minimizer of (21) has mean $\mu_{q_{\text{sp}}^{\text{KL}}} = \left(\sigma^2 K_{SS} + K_{S\mathcal{D}} K_{\mathcal{D}S}\right)^{-1} K_{S\mathcal{D}}\, \mathbf{y}$ which only requires the inversion of a $k \times k$ matrix due to fortuitous cancellation and so is computable in $O(k^3 + Nk^2)$ time. See also section 2.3.6 in Snelson [16] for the interpretation of sparse GPs as KL minimizers. On the other hand, the minimizer of (19) with sparse constraints is $\mu_{q_{\text{sp}}^{\text{opt}}} = \Lambda_{SS}^{-1} \Lambda_{S\mathcal{D}} \mu_{p_\mathcal{D}}$ which doesn't yield similar cancellations and so doesn't seem efficiently computable. It is clear in this case though that $\mu_{q_{\text{sp}}^{\text{opt}}} \neq \mu_{q_{\text{sp}}^{\text{KL}}}$ (unless $S = \mathcal{D}$) and so it leaves open how to obtain efficiently an approximate sparse solution with lower Bayesian risk. Equation (19) makes it clear though that the sparse approximations to the GP should take the test distribution $p(x)$ in consideration, especially if $p(x)$ is quite different of the training distribution in $\mathcal{D}$. We see this question as an interesting open problem.

We also mention that we can compute the linearized E-step from the linearized loss-EM algorithm of table 2 in $O(k^3 + Nk^2)$ as well in the case of sparse constraints on $\mathcal{Q}$ (while the linearized M-step is the trivial update $h^{t+1}(x) = h_{q^{t+1}}(x) = \mu_{q^{t+1}}(x)$). On the other hand, it turns out that the algorithm converges to the non loss-sensitive fixed point $\mu_{q_{\text{sp}}^{\text{opt}}}$ for any value of M, and so is not particularly useful. This is not necessarily surprising given that our derivation for the loss-EM algorithm is only valid for a bounded loss, whereas the quadratic cost function is unbounded. One could artificially bound the loss by truncating the cost function, but then we would also loose the analytic updates, therefore we have not explored this option so far.

4.3 Gaussian process classification

After having looked at an example for which we could compute the posterior analytically, we now consider one where the posterior is intractable and on which we can apply the loss-EM algorithm. We look at Gaussian process binary classification ($\mathcal{Y} = \{-1, +1\}$). We allow for an asymmetric binary cost function: the cost $\ell(y, y')$ is zero for $y = y'$ and has false positive value $\ell(-1, +1) = c_+$ and false negative value $\ell(+1, -1) = c_-$. We use the probit likelihood model $p(y|x, f) = \Phi(yf(x)) = \int_{z \leq yf(x)} N(z|0,1)dz$, i.e. Φ is the cumulative distribution function of a univariate normal and we use a GP prior on f. Using the same trick as mentioned in section 4.1, we use a finite parametrization $\theta = K_{\mathcal{D}\mathcal{D}}^{-1} f_\mathcal{D}$ and redefine the equivalent (in term of posterior risk) probabilistic model:

$$p(\theta) = N(\theta|0, K_{\mathcal{D}\mathcal{D}}^{-1}) \tag{22}$$

$$p(y|x, \theta) = \Phi\left(\frac{K_{x\mathcal{D}}\theta}{\tilde{\sigma}_x}\right) \tag{23}$$

where $\tilde{\sigma}_x^2 \doteq 1 + \sigma_x^2$ with σ_x^2 defined in equation (15). We also assume the transductive scenario where we are given a test set $\mathcal{S}$ of S points $\{x_s\}_{s=1}^S$.

We use again a Gaussian approximate posterior $q(\theta) = \mathcal{N}(\mu_q, \Sigma_q)$ which enable us to get a closed form for the marginalized predictive likelihood:

$$p_q(y|x, \theta) = \Phi\left(\frac{yK_{x\mathcal{D}}}{\tilde{\sigma}_q(x)}\right) \tag{24}$$

where $\tilde{\sigma}_q^2(x) \doteq \tilde{\sigma}_x^2 + K_{x\mathcal{D}} \Sigma_q K_{\mathcal{D}x}$ (and so unlike in the regression case, we see here that Σ_q can influence the decision boundary in the case of asymmetric cost function). The q-optimal action is then:

$$h_q(x) = \text{sign}\{K_{x\mathcal{D}}\mu_q - \tilde{\sigma}_q(x) b_c\} \tag{25}$$

where b_c is a threshold depending on the amount of cost asymmetry $b_c \doteq \Phi^{-1}\left(c_+/(c_- + c_+)\right)$. In the E-step of loss-EM, we need to minimize $-\int_\Theta q(\theta) \log \tilde{p}_{h^t}(\theta) d\theta - H(q)$ with respect to q, where $\tilde{p}_{h^t}$ is defined in equation (11) and corresponds to a loss-sensitive weighting of the posterior distribution. By analogy to a standard

```
Initialize $h^0$ to a random function.
for $t = 0$ to $T$ do
    (Laplace E-step) Maximize $\log \tilde{p}_{h^t}$ using conjugate gradient to get $\hat{\theta}$.
    Set $\mu_{q^{t+1}} = \hat{\theta}$ and $\Sigma_{q^{t+1}} = -\left(\nabla\nabla \log \tilde{p}_{h^t}(\hat{\theta})\right)^{-1}$.
    (Linearized M-step)
    Set $h^{t+1}(x_s) = h_{q^{t+1}}(x_s)$ as per (25) for $x_s \in \mathcal{S}$.
    if $h^{t+1} = h^t$ then return $h^{t+1}$.
end for
```

Table 3: Laplace Linearized Loss-EM for GPC

methodology for GP classification, we use a Laplace approximation of the intractable $\tilde{p}_{h^t}$ (which corresponds to a second order Taylor expansion of $\log \tilde{p}_{h^t}(\theta)$ around the mode $\hat{\theta}$ of $\tilde{p}_{h^t}$). This yields a Gaussian approximation $\tilde{p}_{h^t}(\theta) \simeq \mathcal{N}(\theta|\mu_{q^{t+1}}, \Sigma_{q^{t+1}})$. Hence minimizing the KL with this approximation will yield back the same Gaussian for q assuming it is unrestricted. We present the full algorithm in table 3. We use the conjugate gradient algorithm to find a local maximum of $\log \tilde{p}_{h^t}(\theta)$. We present its gradient here as it provides interesting insights on the loss-sensitivity of the algorithm:

$$\frac{\partial}{\partial\theta} \log \tilde{p}_{h^t}(\theta) = K_{\mathcal{D}\mathcal{D}}\theta + \sum_{x_i \in \mathcal{D}} a_{x_i} \frac{y_i}{p(y_i|x_i, \theta)} K_{\mathcal{D}x_i}$$
$$+ \frac{1}{S} \sum_{x_s \in \mathcal{S}} a_{x_s} \frac{h^t(x_s)\ell\left(-h^t(x_s), h^t(x_s)\right)}{U_M(\theta, h^t)} K_{\mathcal{D}x_s}, \quad (26)$$

where $a_x \doteq \tilde{\sigma}_x^{-1} N(K_{x\mathcal{D}}\theta/\tilde{\sigma}_x|0,1)$. By comparing the third term with the second, we see that the effect of the loss term on the gradient is to push the gradient in the directions of the previous decision $h^t(x_s)$ and proportional to the cost of a false prediction. Unsurprisingly, if the cost is symmetric, we expect the effect to be smaller, as we will see in our synthetic experiments.

5 EXPERIMENTS

We assessed the performance of our linearised loss-EM algorithm for GP classification (table 3) on 100 synthetic datasets. For each trial, 15 uni-variate training inputs were sampled from a uniform distribution between 0 and 1, $\mathcal{U}(0,1)$. Binary labels for these datapoints were generated according to random functions drawn form the GPC prior. To investigate the effect of the test distribution on our method, we generated three different test sets of size 1000, with inputs sampled from $\mathcal{U}(0,1)$, $\mathcal{U}(0.5, 1.5)$ and $\mathcal{U}(1,2)$ respectively (columns of table 4), but reusing the same training set. We used five different loss matrices: the loss for false negatives was constant at $c_- = 1$, the loss for false positives c_+ was varied so that the decision threshold $p_{thresh} = \frac{c_+}{c_- + c_+}$ changed linearly between 0.5 and 0.05 (rows of table 4).

For each dataset we compared three methods for approximate inference: Laplace approximation, expectation propagation (EP) and loss-EM (run separately for each loss and test set combination). Both Laplace and EP are standard approaches to GP classification [13]. To evaluate the performance of the methods we used the following criterion based on the posterior risk:

$$\tilde{R}(q) = \frac{R_{p_\mathcal{D}}(h_q) - R_{p_\mathcal{D}}(h_{p_\mathcal{D}})}{R_{p_\mathcal{D}}(\neg h_{p_\mathcal{D}}) - R_{p_\mathcal{D}}(h_{p_\mathcal{D}})} \quad (27)$$

$R_{p_\mathcal{D}}(-h_{p_\mathcal{D}})$ is the posterior risk of the classifier that always predicts the opposite than the posterior would. This provides an upper bound on the posterior risk of any classifier. Thus $\tilde{R}(q)$ is normalized to take values between 0(posterior-optimal) and 1(maximum risk), enabling us to aggregate performance measures over trials of different difficulty. We estimated $\tilde{R}(q)$ using extensive hybrid Monte Carlo sampling [10], and dropped "easy" scenarios where $\tilde{R}(q)$ were zero for all methods. The numbers reported in table 4 are the mean $\tilde{R}$ values.

We observed that loss-EM provided improvement over direct Laplace approximation to the posterior when the loss is asymmetric or when the test distribution is different from the training set. This is consistent with our expectations about loss-calibration being more critical when asymmetric or structured losses are used. Another observation is the dominance of EP. This is probably due to the fact that EP aims at minimizing $d_{KL}(p_\mathcal{D}\|q)$, whereas our particular EM algorithm is closer to optimizing $d_{KL}(q\|p_\mathcal{D})$, and is based on a crude Laplace approximation. EP is also known to be particularly effective Gaussian process classification case and superior to Laplace approximation [11]. These findings motivate future research into algorithms that minimize d_L more directly than loss-EM does.

6 DISCUSSION

6.1 Related work

As mentioned in the introduction, the discriminative machine learning community has already produced several inherently "loss-calibrated" algorithms. A common learning approach is to optimize a regularized upper bound of the empirical generalization error – or an approximation thereof, hence the name *surrogate loss* – that directly depends on the cost function, such as modern versions of large margin approaches [17]. Their objective is somewhat different inasmuch as these ap-

c_+	p_{tresh}	$\mathbf{p_{test}} = \mathcal{U}(\mathbf{0},\mathbf{1}) = \mathbf{p_{train}}$ Lapl	L-EM	EP	$\mathbf{p_{test}} = \mathcal{U}(\mathbf{0.5},\mathbf{1.5})$ Lapl	L-EM	EP	$\mathbf{p_{test}} = \mathcal{U}(\mathbf{1},\mathbf{2})$ Lapl	L-EM	EP
1.00	.5000	.0011	.0011	.0011	.0167	.0173	.0183	.0686	.0707	.0731
0.63	.3875	.0004	.0003	.0003	.0158	.0133	.0109	.0446	.0380	.0310
0.37	.2750	.0021	.0018	.0007	.0248	.0193	.0082	.0374	.0300	.0121
0.19	.1625	.0099	.0088	.0004	.0297	.0242	.0059	.0216	.0189	.0060
0.05	.0500	.0720	.0661	.0009	.0428	.0410	.0017	.0047	.0047	.0017

Table 4: Performance of Laplace approximation *(Lapl)*, Loss-EM *(L-EM)* and expectation propagation *(EP)* applied to GP classification on synthetic datasets as a function of the shift between the test and training distributions *(columns)* and the asymmetry of loss *(rows)*. Smaller numbers mean better performance (see text).

proaches are aimed at minimizing the frequentist risk – an average over possible training sets, whereas the Bayesian approach tries to make the most of the *given set of observations* by conditioning on it. We see these two approaches as complementary, rather than conflicting, and hope that our framework will attract more interest in analyzing the decision theoretic basis of Bayesian methods.

A closely related approach at midpoint between the Bayesian methodology and the frequentist one is Maximum Entropy Discrimination (MED) by Jaakkola et al. [6]. Following the more modern treatment of Jebara [7], MED aims at solving the following optimization problem (using our notation):

$$\begin{aligned} q^{MED} =& \arg\min_{q\in\mathcal{Q}} KL(q(\theta)||p(\theta)) + C\sum_i \xi_i \qquad (28)\\ & \text{s.t.} \quad \xi_i + p_q(y_i|x_i) \geq p_q(y|x_i) + \ell(y_i,y)\ \forall i \end{aligned}$$

though in practice they use $\int_\Theta q(\theta)\log p(y|x,\theta)d\theta$ rather than $p_q(y|x)$ for computational reasons. The MED optimization problem can be contrasted to our linearized E-step of table 2. MED uses the data through a hinge upper bound [8] on the empirical error (the ξ_i part) whereas we use the data $\mathcal{D}$ through the likelihood term of $p_\mathcal{D}$. The term $\mathcal{R}_q(h^t)$ can be contrasted to the ξ_i part as being a Bayesian loss on data labeled by h^t (our previous best guess) instead of the empirical error on $\mathcal{D}$ as it is for MED.

Finally, we note that Dawid [4] has provided an extensive analysis of the discrepancy d_L which we defined in (5). He analyzed its relationship to losses and 'scoring rules', and studied the question of which losses would yield a unique minimizer.

6.2 Summary and future directions

Our main goal with this paper was to emphasize that, when faced with a particular decision task and knowing what the loss is, an approximate inference method should take the loss into consideration. We took initial steps into what we believe will become a rich field of interesting research questions. We proposed a general decision theoretic framework in which we identified minimization of the loss divergence d_L as an objective of loss calibrated approximate inference. We designed a variational EM algorithm applied it in the context of non-parametric Bayesian regression and classification. Our experiments indicated that the performance of the loss-calibrated method was superior to its loss-insensitive counterpart, i. e. Laplace approximation, but was outperformed by EP, indicating that there is still room for improvement in terms of algorithmic methodology. Moreover, our experimentation with the loss-calibrated framework highlighted which key ingredients need to be considered when calibrating approximate inference to a task. Considering these ingredients, we see the following scenarios as promising applications for our framework:

non-trivial ℓ: Our experiments suggest that the loss-calibration is more pronounced in the case of asymmetric losses, which suggests that the approach has most benefits for applications where complex, structured losses are used, such as in structured prediction [1].

parametric decision boundary: restricting $\mathcal{H}$ to a parametric family – e. g. in consideration of computational efficiency – induces tradeoffs in the performance that different approximate q's can achieve. Therefore, the approximate inference algorithm needs to be calibrated to those tradeoffs.

semi-supervised learning and covariate shift: information can enter our framework through the test distribution $p(x)$ which can be arbitrarily different than the empirical distribution of training samples. This suggests that we could handle the covariate shift problem [18] with a set of unlabelled examples from the test distribution.

References

[1] G. H. Bakir, T. Hofmann, B. Schölkopf, A. J. Smola, B. Taskar, and S. V. N. Vishwanathan. *Predicting Structured Data*. The MIT Press, 2007.

[2] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473): 138–156, 2006.

[3] J. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer, New York, 1985.

[4] A. P. Dawid. Proper measures of discrepancy, uncertainty and dependence with applications to predictive experimental designs. Technical Report 139, Department of Statistical Science at University College London, 1994 (revised 1998).

[5] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[6] T. Jaakkola, M. Meila, and T. Jebara. Maximum entropy discrimination. volume 12. MIT Press, Cambridge, MA, 1999.

[7] T. Jebara. Multitask sparsity via Maximum Entropy Discrimination. *Journal of Machine Learning Research (To Appear)*, 2010.

[8] T. Joachims, T. Finley, and C.-N. Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

[9] T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, 2001. Supervisor-Picard, Rosalind.

[10] R. M. Neal. MCMC using Hamiltonian dynamics. In G. J. S. Brooks, A. Gelman and X.-L. Meng, editors, *Handbook of Markov Chain Monte Carlo*. Chapman & Hall / CRC Press, 2010.

[11] H. Nickisch and C. E. Rasmussen. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research 9*, pages 2035–2078, 2008.

[12] J. Quińonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1935–1959, 2005.

[13] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, USA, 2006.

[14] C. P. Robert. *The Bayesian Choice*. Springer, New York, 2001.

[15] M. J. Schervish. *Theory of Statistics*. Springer, New York, 1995.

[16] E. Snelson. *Flexible and efficient Gaussian process models for machine learning*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2007.

[17] I. Steinwart and A. Christmann. *Support Vector Machines*. Springer, New York, 2008.

[18] M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.