IEIE Transactions on Smart Processing and Computing

New Inference for a Multiclass Gaussian Process Classification Model using a Variational Bayesian EM Algorithm and Laplace Approximation

Wanhyun Cho[1], Sangkyoon Kim[2] and Soonyoung Park[3]

[1] Department of Statistics, Chonnam National University / Gwangju,500-757 South Koreawhcho@chonnam.ac.kr
[2] Department of Electronics Engineering, Mokpo National University / Chonnam, South Koreanarciss76@mokpo.ac.kr
[3] Department of Electronics Engineering, Mokpo National University / Chonnam, South Koreasypark@mokpo.ac.kr

***** Corresponding Author: Sangkyoon Kim

Received July 15, 2015; Revised August 5, 2015; Accepted August 24, 2015; Published August 31, 2015

** Short Paper*

** Extended from a Conference: Preliminary results of this paper were presented at the ICT-CSCC, Summer 2015. The present paper has been accepted by the editorial board through the regular reviewing process that confirms the original contribution.*

Abstract: In this study, we propose a new inference algorithm for a multiclass Gaussian process classification model using a variational EM framework and the Laplace approximation (LA) technique. This is performed in two steps, called expectation and maximization. First, in the expectation step (E-step), using Bayes' theorem and the LA technique, we derive the approximate posterior distribution of the latent function, indicating the possibility that each observation belongs to a certain class in the Gaussian process classification model. In the maximization step, we compute the maximum likelihood estimators for hyper-parameters of a covariance matrix necessary to define the prior distribution of the latent function by using the posterior distribution derived in the E-step. These steps iteratively repeat until a convergence condition is satisfied. Moreover, we conducted the experiments by using synthetic data and Iris data in order to verify the performance of the proposed algorithm. Experimental results reveal that the proposed algorithm shows good performance on these datasets.

Keywords: Multiclass gaussian process classification model, Variational bayesian EM algorithm, Laplace approximation technique, Latent function, Softmax function, Synthetic data, Iris data

1. Introduction

Gaussian process (GP) can be conveniently used to specify prior distributions of hidden functions for Bayesian inference. In the case of regression with Gaussian noise, inference can be done simply in closed form, since the posterior is also a GP. But in the case of classification, exact inference is analytically intractable because the likelihood function is given as a non-Gaussian form.

One prolific line of attack is based on approximating the non-Gaussian posterior with a tractable Gaussian distribution. Three different types of solutions have been suggested in the recent literature [1]. These are the Laplace approximation (LA) and expectation propagation (EP), Kullback-Leibler divergence minimization comprising variational bounding as a special case, and factorial approximation. First, Williams et al. proposed the use of a second-order Taylor expansion around the posterior mode to a natural way of constructing a Gaussian approximation to the log-posterior distribution [2]. The mode is taken as the mean of the approximate Gaussian. Linear terms of the log-posterior vanish because the gradient at the mode is zero. The quadratic term of the log-posterior is given by the negative Hessian matrix. Minka presented a new approximation technique (EP) for Bayesian networks [3]. This is an iterative method to find approximations based on approximate marginal moments, which can be applied to Gaussian processes. Second, Opper et al. discussed the relationship between the Laplace and variational approximations, and they show that for models with

Gaussian priors and factoring likelihoods, the number of variational parameters is actually O(N) [4]. They also considered a problem that minimizes the KL-divergence measure between the approximated posterior and the exact posterior. Gibbs et al. showed that the variational methods of Jaakkola and Jordan are applied to Gaussian processes to produce an efficient Bayesian binary classifier [5]. They obtained tractable upper and lower bounds for the unnormalized posterior density. These bounds are parameterized by variational parameters that are adjusted to obtain the tightest possible fit. Using the normalized versions of the optimized bounds, they then compute approximations to the predictive distributions. Third, Csato et al. presented three simple approximations for the calculation of the posterior mean in Gaussian process classification [6]. The first two methods are related to mean field ideas known in statistical physics. The third approach is based on a Bayesian outline approach. Finally, Kim et al. presented an approximate expectation–maximization (EM) algorithm and the EM-EP algorithm to learn both the latent function and hyper-parameters in a Gaussian process classification model [7].

We propose a new inference algorithm that can simultaneously derive both a posterior distribution of a latent function and maximum likelihood estimators of hyper-parameters in a Gaussian process classification model. The proposed algorithm is performed in two steps: called the expectation step (E-step) and the maximization step (M-step). First, in the expectation step, using the Bayesian formula and LA, we derive the approximate posterior distribution of the latent function based on learning data. Furthermore, we calculate a mean vector and covariance matrix of the latent function. Second, in the maximization step, using a derived posterior distribution of the latent function, we derive the maximum likelihood estimator for hyper-parameters necessary to define a covariance matrix. Moreover, we conducted the experiments by using synthetic data and Iris data in order to verify the performance of the proposed algorithm.

The rest of this paper is organized as follows. The next section describes a multiclass Gaussian process classification model. In the Section 3 and 4, inference scheme section, we propose a new inference method that can derive the approximate distribution for a posterior distribution of latent variables and estimate the hyper-parameters of the covariance function for prior distribution of the latent function. The section 5 includes performance evaluations and discussion of the effects of the proposed model. Finally, we conclude this paper in the last section.

2. Multiclass Gaussian Process Classification Model

We first consider a multiclass Gaussian process classification model (MGPCM). The model consists of three components: a latent function with a Gaussian process prior distribution, a multiclass response, and a link function that relates between the latent function and response mean. First, we consider the multivariate latent function. Here, we define the latent function $\mathbf{f}(\mathbf{x})$ for Gaussian process classification having C classes at a set of observations $\mathbf{x}_1,\cdots,\mathbf{x}_n$ as

$$\mathbf{f}(\mathbf{x}\,|\,\Theta)=(\mathrm{f}_1^1(\mathbf{x}),\cdots,\mathrm{f}_n^1(\mathbf{x}),\cdots,\mathrm{f}_1^c(\mathbf{x}),\cdots,\mathrm{f}_1^c(\mathbf{x}),\\ \cdots,\mathrm{f}_1^C(\mathbf{x}),\cdots,\mathrm{f}_n^C(\mathbf{x}))^T \tag{1}$$

Then, we assume a GP prior for the latent function $\mathbf{f}(\mathbf{x})$ as defined by

$$\mathbf{f}(\mathbf{x}\,|\,\Theta)\sim\mathrm{GP}(\mathbf{0},\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta)) \tag{2}$$

where $\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)$ is the covariance matrix. In this paper, we assume that the latent function $\mathbf{f}(\mathbf{x})$ represents the C classes, and the individual variables of the c -th component vector $\mathbf{f}^c(\mathbf{x})$ of latent function $\mathbf{f}(\mathbf{x})$ are uncorrelated. Therefore, the GP covariance matrix $\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)$ can be assumed from the following block diagonal form:

$$\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta)=\mathrm{diag}(\mathrm{K}^1(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta^1),\cdots,\mathrm{K}^c(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta^c),\\ \cdots,\mathrm{K}^C(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta^C)) \tag{3}$$

where

$$\mathrm{K}^{\mathrm{c}}(\mathbf{x}_i,\mathbf{x}_j\,|\,\Theta^c)=(k^c(\mathbf{x}_i,\mathbf{x}_j)\,|\,(\theta_1^c,\theta_2^c))_{(n\times n)},i,j=1,\cdots,n$$

is also the covariance matrix for the c -th component vector of the latent function.

Second, the response vector $\mathbf{Y}$ is constituted by identical independent multinomial random variables where each component variable represents a c class. That is, let us define the response vector $\mathbf{Y}$ as

$$\mathbf{Y}=(y_1^1(\mathbf{x}),\cdots,y_n^1(\mathbf{x}),\cdots,y_1^c(\mathbf{x}),\cdots,y_n^c(\mathbf{x}),\\ \cdots,y_1^C(\mathbf{x}),\cdots,y_n^C(\mathbf{x}))^T, \tag{4}$$

where a vector of response $\mathbf{Y}$ has the same length as $\mathbf{f}(\mathbf{x})$, and each component y_k^c of the c -th response vector $\mathbf{y}^c=(y_1^c,\cdots,y_n^c)^T$ for $c=1,\cdots,C$ has 1 for the class, which is the label for observation, and 0 for the other $C-1$ classes. Here, we are able to assume that the multinomial density function $p(\mathbf{Y}|\boldsymbol{\pi})$ of the response vector $\mathbf{Y}$ is given in the following form:

$$p(\mathbf{Y}|\boldsymbol{\pi})=\prod_{c=1}^{C}\prod_{k=1}^{n}(\pi_k^c)^{y_k^c} \tag{5}$$

where the indicator variable y_k^c takes one or zero with probability π_k^c and 1-π_k^c , and π_k^c denotes the probability that the k -th observation vector belongs to the particular class c.

Third, we consider the link function that specifies the relation between the latent function $\mathbf{f}(\mathbf{x})$ and the response mean vector $\mathrm{E}(\mathbf{Y}|\mathbf{f})$. Here, the link function can be defined as

$$\mathrm{E}(\mathbf{Y}|\mathbf{f}) = (\mathrm{E}(\mathbf{y}^1|\mathbf{f}),\cdots,\mathrm{E}(\mathbf{y}^c|\mathbf{f}),\cdots,\mathrm{E}(\mathbf{y}^C|\mathbf{f}))^T,$$

where

$$\mathrm{E}(\mathbf{y}^C|\mathbf{f}) = (E(\mathrm{y}_1^c|\mathbf{f}),\cdots,E(\mathrm{y}_k^c|\mathbf{f}),\cdots,E(\mathrm{y}_n^c|\mathbf{f})),$$
$$c = 1,\cdots,C,$$

And

$$E(\mathrm{y}_k^c|\mathbf{f}) = \pi_k^c = \frac{\exp(\mathrm{f}_k^c)}{\sum_{c'=1}^{C}\exp(\mathrm{f}_k^{c'})}, k = 1,\cdots,n \qquad (6)$$

3. Variational EM Framework and Laplace Approximation Method

One important issue in the Gaussian process classification model is to both derive the approximate distribution for a posterior distribution of latent variables and to estimate the hyper-parameters of the covariance function for prior distribution of the latent function. One possible approach is to consider the variational EM algorithm that is widely used in the incomplete data.

In the E-step of the variational EM algorithm, we derive the approximate Gaussian posterior $q(\mathbf{f}|\mathbf{X},\mathbf{Y},\Theta)$ for latent function value $\mathbf{f}$ using Laplace approximation. In the M-step of the variational EM algorithm, we seek an estimator of hyper-parameter Θ that can maximize a lower bound on a logarithm of the marginal likelihood $q(\mathbf{Y}|\mathbf{X},\Theta)$ using the approximate posterior $q(\mathbf{f}|\mathbf{X},\mathbf{Y},\Theta)$ obtained in the E-step. The E-step and M-step are iteratively repeated until a convergence condition is satisfied. Our algorithm is given in detail in the following sections.

3.1 Variation E-step and Laplace Approximation

First, using Bayes' rule at a variational E-step, the posterior over the latent variable $\mathbf{f}$ is given by

$$p(\mathbf{f}|\mathbf{X},\mathbf{y},\Theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X},\Theta)/p(\mathbf{y}|\mathbf{X},\Theta) \qquad (7)$$

but because the denominator $p(\mathbf{y}|\mathbf{X},\Theta)$ is independent with latent function $\mathbf{f}$, we need only consider the un-normalized posterior when maximizing with respect to $\mathbf{f}$. Taking the logarithm of the un-normalized posterior of latent function $\mathbf{f}$, it can be given as

$$\begin{aligned}\Psi(\mathbf{f}) &= \ln p(\mathbf{f}|\mathbf{Y},\mathbf{X},\Theta)\\ &= \ln p(\mathbf{f}|\mathbf{X},\Theta) + \ln p(\mathbf{Y}|\mathbf{f}) \\ &= \ln p(\mathbf{Y}|\mathbf{f}) - \frac{1}{2}\mathbf{f}^T\mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\ln|\mathbf{K}| - \frac{nC}{2}\ln 2\pi\end{aligned} \qquad (8)$$

Here, taking the first and second derivatives of Eq. (8) with respect to $\mathbf{f}$, we obtain

$$\begin{aligned}&\nabla\Psi(\mathbf{f}) = \nabla\ln p(\mathbf{Y}|\mathbf{f}) - \mathbf{K}^{-1}\mathbf{f},\\ &\nabla\nabla\Psi(\mathbf{f}) = \nabla\nabla\ln p(\mathbf{Y}|\mathbf{f}) - \mathbf{K}^{-1} = -\mathbf{W} - \mathbf{K}^{-1}\end{aligned} \qquad (9)$$

where $\mathbf{W} \equiv -\nabla\nabla\ln p(\mathbf{Y}|\mathbf{F})$ is diagonal, since the likelihood factorizes over the case.

A natural way of constructing a Gaussian approximation to the log-posterior $\Psi(\mathbf{f}) = \ln p(\mathbf{f}|\mathbf{Y},\mathbf{X},\Theta)$ is to perform a second-order Taylor expansion at the mode $\mathbf{m}_\mathbf{F}$ of the posterior, i.e.

$$\begin{aligned}\mathbf{m}_\mathbf{f} &= \arg\max_\mathbf{f} \Psi(\mathbf{f})\\ &= \arg\max_\mathbf{f} \ln p(\mathbf{Y}|\mathbf{f})p(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\end{aligned} \qquad (10)$$

It gives us the following equation:

$$\begin{aligned}\Psi(\mathbf{f}) &= \Psi(\mathbf{m}_\mathbf{f}) + \nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}}(\mathbf{f}-\mathbf{m}_\mathbf{f})\\ &\quad + \frac{1}{2}(\mathbf{f}-\mathbf{m}_\mathbf{f})^T(\nabla\nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}})(\mathbf{f}-\mathbf{m}_\mathbf{f})\\ &= \Psi(\mathbf{m}_\mathbf{f}) - \frac{1}{2}(\mathbf{f}-\mathbf{m}_\mathbf{f})^T(\mathbf{W}+\mathbf{K}^{-1})(\mathbf{f}-\mathbf{m}_\mathbf{f})\\ &\cong \ln N(\mathbf{f}|\mathbf{m}_\mathbf{f},(\mathbf{K}^{-1}+\mathbf{W})^{-1}).\end{aligned} \qquad (11)$$

Thus, we have obtained a Gaussian approximation posterior $q(\mathbf{f}|\mathbf{Y},\mathbf{X},\Theta)$ to the true posterior $p(\mathbf{f}|\mathbf{Y},\mathbf{X},\Theta)$ with mean vector $\mathbf{m}_\mathbf{f}$ and covariance matrix $\mathbf{V} = (\mathbf{K}^{-1}+\mathbf{W})^{-1}$. That is, using the Laplace approximation, the true posterior $p(\mathbf{f}|\mathbf{X},\mathbf{Y},\Theta)$ of latent function $\mathbf{f}$ is approximated as a Gaussian posterior $q(\mathbf{f}|\mathbf{X},\mathbf{Y},\Theta)$ as the following:

$$q(\mathbf{f}|\mathbf{X},\mathbf{Y},\Theta) \sim N(\mathbf{m}_\mathbf{F},\mathbf{V} = (\mathbf{K}^{-1}+\mathbf{W})^{-1}) \qquad .(12)$$

Here, the mode or maximum $\mathbf{m}_\mathbf{f}$ of the log-posterior $\Psi(\mathbf{f})$ can be found iteratively using the Newton-Rapson algorithm. That is, given an initial estimate, $\mathbf{m}_\mathbf{f}$, a new estimate is iteratively found, as follows:

$$\begin{aligned}\mathbf{m}_\mathbf{f}^{new} &= \mathbf{m}_\mathbf{f} - (\nabla\nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}})^{-1}\nabla\Psi(\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}}\\ &= \mathbf{m}_\mathbf{f} + (\mathbf{K}^{-1}+\mathbf{W})^{-1}(\nabla\ln p(\mathbf{Y}|\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{f}} - \mathbf{K}^{-1}\mathbf{m}_\mathbf{f})\\ &= (\mathbf{K}^{-1}+\mathbf{W})^{-1}(\mathbf{W}\mathbf{m}_\mathbf{f} + \nabla\ln p(\mathbf{Y}|\mathbf{f})_{\mathbf{f}=\mathbf{m}_\mathbf{F}}).\end{aligned} \qquad (13)$$

Moreover, since the log-likelihood function $\ln p(\mathbf{Y}|\mathbf{f})$

can be expressed as $\sum_{k=1}^{n} \ln p(y_k^1,\cdots,y_k^{\mathrm{C}} \mid \mathbf{f}_k)$, we obtain the following equation by differentiating the log-likelihood function $\ln p(\mathbf{Y}\mid\mathbf{f})$ with respect to $\mathbf{f}$:

$$
\begin{aligned}
\nabla_{\mathbf{f}} \ln p(\mathbf{Y}\mid\mathbf{f}) &= \nabla_{\mathbf{f}}(\sum_{k=1}^{n} \ln p(y_k^1,\cdots,y_k^{\mathrm{C}} \mid \mathbf{f}_k)) \\
&= \nabla_{\mathbf{F}}\left(\sum_{k=1}^{n}\sum_{c=1}^{C} y_i^c \mathrm{f}_i^c - \sum_{k=1}^{n} \ln(\sum_{c=1}^{C}\exp(\mathrm{f}_k^c))\right) \qquad (14)\\
&= \mathbf{Y} - \boldsymbol{\pi}
\end{aligned}
$$

where a vector $\boldsymbol{\pi}$ is defined by

$$
\begin{aligned}
&\boldsymbol{\pi}_{(nC\times 1)} = (\pi_1^1,\cdots,\pi_i^c,\cdots,\pi_n^C)^T, \\
&\pi_i^c = \frac{\exp(\mathrm{f}_i^c)}{\sum_{c^*=1}^{C}\exp(\mathrm{f}_i^{c^*})}, \quad i=1,\cdots,\mathrm{n},\ c=1,\cdots,C \qquad (15)
\end{aligned}
$$

Second, the matrix $\mathbf{W}$ can be given as

$$
\begin{aligned}
\mathbf{W} &= -\nabla\nabla \ln p(\mathbf{Y}\mid\mathbf{f}) = \left(-\frac{\partial \ln p(\mathbf{Y}\mid\mathbf{f})}{\partial\mathbf{f}\,\partial\mathbf{f}^T}\right) \qquad (16)\\
&= \mathrm{diag}(\boldsymbol{\pi}) - \boldsymbol{\Pi}^T\boldsymbol{\Pi},
\end{aligned}
$$

where $\boldsymbol{\Pi}$ is an $(n\times nC)$ matrix obtained by horizontally stacking the diagonal matrices $\left(\mathrm{diag}(\pi^c)\right), c=1,\cdots,C$. This is given in the following form:

$$
\boldsymbol{\Pi} = \begin{bmatrix}
\mathrm{diag}(\pi_1^1) & \cdots & 0 & \cdots & \mathrm{diag}(\pi_1^C) & \cdots & 0 \\
\vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\
0 & \cdots & \mathrm{diag}(\pi_n^1) & \cdots & 0 & \cdots & \mathrm{diag}(\pi_n^C)
\end{bmatrix} \qquad (17)
$$

3.2 Variational M-Step

As we assume a derived approximate Gaussian posterior $q(\mathbf{f}\mid\mathbf{X},\mathbf{Y},\Theta)$ is held fixed, we seek the new parameter values Θ^{new} that the lower bound $F(q,\Theta)$, given in the following Eq. (18) can be maximized with respect to Θ:

$$
\begin{aligned}
\ln p(\mathbf{Y}\mid\mathbf{X},\Theta) &= \ln\int p(\mathbf{f}\mid\mathbf{X},\Theta)p(\mathbf{Y}\mid\mathbf{f})d\,\mathbf{f} \\
&= \int q(\mathbf{f})\ln\left(\frac{p(\mathbf{f},\mathbf{Y}\mid\mathbf{X},\Theta)}{q(\mathbf{f})}\right)d\,\mathbf{f} \\
&\quad + \int q(\mathbf{f})\ln\left(\frac{q(\mathbf{f})}{p(\mathbf{f}\mid\mathbf{Y},\mathbf{X},\Theta)}\right)d\,\mathbf{f} \qquad (18)\\
&\geq F(q,\Theta) = \int q(\mathbf{f})\ln\left(\frac{p(\mathbf{f},\mathbf{Y}\mid\mathbf{X},\Theta)}{q(\mathbf{f})}\right)d\,\mathbf{f}.
\end{aligned}
$$

Here, the low bound $F(q,\Theta)$ can be written as

$$
\begin{aligned}
F(q,\Theta) &= \int q(\mathbf{f})\ln\left(\frac{p(\mathbf{f}\mid\mathbf{X},\Theta)p(\mathbf{Y}\mid\mathbf{f})}{q(\mathbf{f})}\right)d\,\mathbf{f} \\
&= \int q(\mathbf{f})\ln p(\mathbf{f}\mid\mathbf{X},\Theta)d\,\mathbf{f} + \int q(\mathbf{f})\ln p(\mathbf{Y}\mid\mathbf{f})d\,\mathbf{f} \\
&\qquad - \int q(\mathbf{f})\ln q(\mathbf{f})d\,\mathbf{f} \\
&= E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta)) + E_{q(\mathbf{f})}(\ln p(\mathbf{Y}\mid\mathbf{f})) + \mathrm{H}(q(\mathbf{f})). \qquad (19)
\end{aligned}
$$

Moreover, since the second term and the third term are independent with hyper-parameters Θ, we only need to maximize the first term, $E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta))$, with respect to Θ. By computing $E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta))$ using a Gaussian posterior, we obtain:

$$
\begin{aligned}
&E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta)) \\
&= -\frac{nc}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{K}(\Theta)| - \frac{1}{2}E_{q(\mathbf{f})}\left(\mathbf{f}^T\mathbf{K}(\Theta)^{-1}\mathbf{f}\right) \\
&= -\frac{nc}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{K}(\Theta)| - \frac{1}{2}\left(E_{q(\mathbf{f})}(\mathbf{f})^T\mathbf{K}(\Theta)^{-1}E_{q(\mathbf{f})}(\mathbf{f})\right) \\
&\qquad - \frac{1}{2}tr\left(\mathbf{K}(\Theta)^{-1}Cov(\mathbf{f})\right) \qquad (20)\\
&= -\frac{nc}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{K}(\Theta)| - \frac{1}{2}\left(\mathbf{m}_{\mathbf{f}}{}^T\mathbf{K}(\Theta)^{-1}\mathbf{m}_{\mathbf{f}}\right) \\
&\qquad - \frac{1}{2}tr\left(\mathbf{K}(\Theta)^{-1}Cov(\mathbf{f})\right)
\end{aligned}
$$

Here, by differentiating $E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta))$ with respect to Θ using the E-step result, we obtain

$$
\begin{aligned}
&\frac{\partial E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta))}{\partial\Theta} \\
&\quad = -\frac{1}{2}\mathrm{tr}(\mathbf{K}(\Theta)^{-1}\frac{\partial\mathbf{K}(\Theta)}{\partial\Theta}) \\
&\qquad + \frac{1}{2}\left(\mathbf{m}_{\mathbf{f}}{}^T\mathbf{K}(\Theta)^{-1}\frac{\mathbf{K}(\Theta)}{\partial\Theta}\mathbf{K}(\Theta)^{-1}\mathbf{m}_{\mathbf{f}}\right) \qquad (21)\\
&\qquad\quad + \frac{1}{2}tr\left(\mathbf{K}(\Theta)^{-1}\frac{\mathbf{K}(\Theta)}{\partial\Theta}\mathbf{K}(\Theta)^{-1}Cov(\mathbf{f})\right).
\end{aligned}
$$

Therefore, we can obtain the hyper-parameter maximizing the free energy by the following gradient update rule:

$$
\Theta^{new} = \Theta^{old} + \eta\left(\frac{\partial E_{q(\mathbf{f})}(\ln p(\mathbf{f}\mid\mathbf{X},\Theta))}{\partial\Theta}\right)_{\Theta=\Theta^{old}} \qquad (22)
$$

4. Prediction Method

Here, if we denote a vector $\mathbf{f}_*$ as the latent function value corresponding with test point $\mathbf{x}_*$, then the joint prior distribution of the training latent function $\mathbf{f}$ and the test latent function $\mathbf{f}_*$ is

$$p(\mathbf{f}_*,\mathbf{f}\mid\mathbf{x}_*,\mathbf{X},\Theta)\sim N\left(\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix}\middle|\,\mathbf{0},\begin{bmatrix}\mathbf{K} & \mathbf{K}_*\\ \mathbf{K}_*{}^T & \mathbf{k}_{**}\end{bmatrix}\right) \tag{23}$$

where

$$\mathbf{K}_* = \text{Vertical+Diag}(\mathrm{K}_*^1(\mathbf{x},\mathbf{x}_*),\cdots,\mathrm{K}_*^c(\mathbf{x},\mathbf{x}_*),\cdots,\mathrm{K}_*^C(\mathbf{x},\mathbf{x}_*)),$$
$$\mathrm{K}_*^c(\mathbf{x},\mathbf{x}_*) = (k^c(\mathbf{x}_1,\mathbf{x}_*),\cdots,k^c(\mathbf{x}_n,\mathbf{x}_*))^T, \mathrm{c}=1,\cdots,\mathrm{C},$$

and

$$\mathbf{k}_{**} = diag(k^1(\mathbf{x}_*,\mathbf{x}_*),\cdots,k^C(\mathbf{x}_*,\mathbf{x}_*))$$

Hence, given a novel test point $\mathbf{x}_*$, the posterior distribution of latent function $\mathbf{f}_*$ corresponding to a test point $\mathbf{x}_*$ can be obtained by marginalizing the latent functions of the training set:

$$\begin{aligned}p(\mathbf{f}_*\mid\mathbf{x}_*,\mathbf{Y},\mathbf{X},\Theta) &= \int p(\mathbf{f}_*,\mathbf{f}\mid\mathbf{x}_*,\mathbf{Y},\mathbf{X},\Theta)\,d\mathbf{f}\\ &= \int p(\mathbf{f}_*\mid\mathbf{f},\mathbf{x}_*,\mathbf{X},\Theta)\,p(\mathbf{f}\mid\mathbf{Y},\mathbf{X},\Theta)\,d\mathbf{f}\end{aligned} \tag{24}$$

But the posterior distribution of the latent function is unfortunately not Gaussian due to the non-Gaussian likelihood, as mentioned above. Hence, the approximate posterior distribution of the latent function is necessary. Here, if we use the Laplace approximation posterior $q(\mathbf{f}\mid\mathbf{X},\mathbf{Y},\Theta)$ to a true posterior $p(\mathbf{f}\mid\mathbf{X},\mathbf{Y},\Theta)$, we have obtained the approximate posterior distribution $q(\mathbf{f}_*\mid\mathbf{x}_*,\mathbf{X},\mathbf{Y},\Theta)$ of latent function $\mathbf{f}_*$. It is obviously given as the Gaussian with mean vector $\mathbf{K}_*^T(\mathbf{K}^{-1}+\mathbf{W})\mathbf{m}_\mathbf{F}$ and covariance matrix $\mathbf{k}_{**}-\mathbf{K}_*{}^T\mathbf{K}^{-1}\mathbf{K}_*$.

Hence, the predictive mean vector for class c of the latent function value $\mathbf{f}_*$ corresponding with test point $\mathbf{x}_*$ is given by

$$\begin{aligned}\mathrm{E}_q(\mathbf{f}_*^c\mid\mathbf{x}_*,\mathbf{X},\mathbf{Y},\Theta) &= \mathbf{K}_*^c(\mathbf{x},\mathbf{x}_*)^T(\mathbf{K}^c)^{-1}\mathbf{m}_\mathbf{F}^c\\ &= \mathbf{K}_*^c(\mathbf{x},\mathbf{x}_*)^T(\mathbf{y}^c-\boldsymbol{\pi}^c)\end{aligned} \tag{25}$$

where the last equality comes from $\mathbf{K}^{-1}\mathbf{m}_\mathbf{F}=\mathbf{Y}-\boldsymbol{\pi}$, and $(\mathbf{K}^c)^{-1}\mathbf{m}_\mathbf{F}^c=(\mathbf{y}^c-\boldsymbol{\pi}^c)$. Moreover, if these are put into vector form, then the expectation of latent function $\mathbf{f}_*$ under the Laplace approximation is given as

$$\boldsymbol{\mu}_* = \mathrm{E}_q(\mathbf{f}_*^c\mid\mathbf{x}_*,\mathbf{X},\mathbf{Y},\Theta) = \mathrm{Q}_*^T(\mathbf{y}-\boldsymbol{\pi}), \tag{26}$$

where a matrix Q_*^T is defined as the $(nC\times C)$ matrix

$$\mathbf{Q}_* = \begin{pmatrix}\mathbf{K}_*^1(\mathbf{x},\mathbf{x}_*) & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{K}_*^2(\mathbf{x},\mathbf{x}_*) & \cdots & \vdots\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{K}_*^C(\mathbf{x},\mathbf{x}_*)\end{pmatrix} \tag{27}$$

And the covariance matrix of the latent function $\mathbf{f}_*$ can be represented as

$$\begin{aligned}\boldsymbol{\Sigma}_* &= Cov_q(\mathbf{f}_*\mid\mathbf{x}_*,\mathbf{X},\mathbf{Y},\Theta)\\ &= \mathbf{k}_{**}-\mathbf{Q}_*^T(\mathbf{K}+\mathbf{W}^{-1})^{-1}\mathbf{Q}_*\end{aligned} \tag{28}$$

Therefore, we have obtained the approximate Gaussian posterior distribution $G(\boldsymbol{\mu}_*,\boldsymbol{\Sigma}_*)$ of the latent function $\mathbf{f}_*$.

Finally, in order to classify input vector $\mathbf{X}_*$ into its proper class, we first extract the n random samples $\mathbf{f}_{*1},\cdots,\mathbf{f}_{*\mathrm{n}}$ from the predictive distribution of latent function $\mathbf{f}_*$ corresponding to the input vector. Further, using the Eq. (2), we calculate the estimate of the classification probability $(\pi_{*1}{}^{c},\cdots,\pi_{*\mathrm{n}}{}^{c})$, $c=1,\cdots,C$,and compute a mean vector of these probabilities $(\bar{\pi}_*{}^{1},\cdots,\bar{\pi}_*{}^{\mathrm{C}})$. Therefore, we will classify the input vector $\mathbf{X}_*$ into the class which its classification probability is maximized. That is,

$$\bar{\pi}_*{}^{c'} = \arg\max_{1\le c\le C}(\bar{\pi}_*{}^{1},\cdots,\bar{\pi}_*{}^{\mathrm{C}}). \tag{29}$$

5. Performance Evaluation

In order to evaluate the performance improvement achieved by the proposed inference method, we consider a bivariate normal synthetic data and Iris data.

5.1 Synthetic Data

Here, we will consider four partially overlapping Gaussian sources of data in two dimensions. First, in order to train a model, we generate four classes of bivariate Gaussian random samples. One hundred sixty data points were generated by the four bivariate normal distributions with the mean vectors and covariance matrices described in Table 1. Fig. 1(a) plots these data points in a two-dimensional space.

Table 1. Mean Vector and Covariance Matrix for Each Class.

	Class 1	Class 2	Class 3	Class 4
Mean vector	(1.75,-1.0)	(-1.75,1.0)	(2,2)	(-2,-2)
Covariance matrix	$\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$	$\begin{pmatrix}1 & 0.5\\ 0.5 & 1\end{pmatrix}$	$\begin{pmatrix}1 & -0.5\\ -0.5 & 1\end{pmatrix}$	$\begin{pmatrix}1 & -0.5\\ -0.5 & 1\end{pmatrix}$

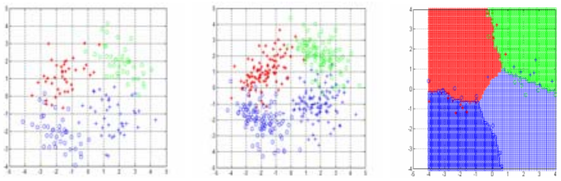

Fig. 1. (a) Training data, (b) testing data, and (c) class region and misclassification observations.

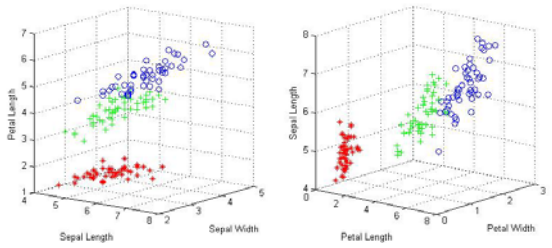

Fig. 2. Iris dataset.

Table 2. Classification of Iris Species.

	setosa	versicolor	virginica
Setosa	1	0	0
Versicolor	0	0.96	0.04
Virginica	0	0.01	0.99

Second, in order to verify the performance of the model, we generate four different classes of bivariate Gaussian random samples. Four hundred data points were generated by the bivariate normal distribution. Fig. 1(b) plots the testing data points. Fig. 1(c) shows each region and misclassification data points. We can see that it totals about 7-8% misclassification. Therefore, we know that the proposed method can completely classify the data points well.

5.2 Iris Dataset

Here, we considered real data called an Iris dataset. This dataset consists of 50 samples from each of three species of Iris flowers: setosa, versicolor and virginica. Four features were measured from each sample (length and width of sepal and petal) in centimeters. Based on the combination of the four features, we developed a GP classifier model to distinguish one species from another.

Fig. 2 shows the Iris dataset from different viewpoints. First, in order to train a model, we used a total of 90 observations from three classes. And in order to verify the performance of the model, we selected 60 samples, except for ones used in the training set.

Next, we want to measure the performance of our proposed model when classifying the Iris species. To find the best performance, we chose to find the optimal hyper-parameters at the point where the marginal likelihood has a maximum using the EM algorithm.

Table 2 shows the results of the Iris species classification. To calculate the rates, we estimate the number of correctly classified negatives and positives and divide by the total number of each species.

We had to try many experiments to get meaningful results using randomly selected samples. Experimental results reveal that the average for a successful classification rate is about 98%.

6. Conclusion

This paper proposed a new inference algorithm that can simultaneously derive both a posterior distribution of a latent function and estimators of hyper-parameters in the Gaussian process classification model. The proposed algorithm was performed in two steps: the expectation step and the maximization step. In the expectation step, using a Bayesian formula and Laplace approximation, we derived the approximate posterior distribution of the latent function on the basis of the learning data. Furthermore, we considered a method of calculating a mean vector and covariance matrix of a latent function. In the classification step, using the derived posterior distribution of the latent function, we derived the maximum likelihood estimator for hyper-parameters necessary to define a covariance matrix.

Finally, we conducted experiments by using synthetic data and Iris data in order to verify the performance of the proposed algorithm. Experimental results reveal that the proposed algorithm shows good performance on these datasets. Our future work will extend the proposed method to other video recognition problems, such as 3D human action recognition, gesture recognition, and surveillance systems.

Acknowledgement

This work was jointly supported by the National Research Foundation of the Korea Government (2014R1A1A4A0109398) and the research fund of Chonnam National University (2014-2256).

References

[1] H. Nickish et al., “Approximations for Binary Gaussian Process Classification,” *Journal of Machine Learning Research*, Vol. 9, pp. 2035-2078, 2008. Article (CrossRef Link)

[2] C. K. I. Williams et al., “Bayesian Classification with Gaussian Processes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, pp. 1103-1118, 1998. Article (CrossRef Link)

[3] T. P. Minka, “Expectation Propagation for Approximate Bayesian Inference,” *Technical Report*, Depart. of Statistics, Carnegie Mellon University, Pittsburgh, PA 15213, 2001. Article (CrossRef Link)

[4] M. Opper et al., “The Variational Gaussian Approximation Revisited,” *Neural Comput.*, Vol. 21, No. 3, pp. 786-92, 2009. Article (CrossRef Link)

[5] M. N. Gibbs et al., “Variational Gaussian Process Classifiers,” *IEEE Transactions on Neural Networks*, Vol. 11, No. 6, pp. 1458-1464, 2000. Article (CrossRef Link)

[6] L. Csato et al., “Efficient Approaches to Gaussian Process Classification,” *in Neural Information Processing Systems*, Vol. 12, pp. 251-257, MIT Press, 2000. Article (CrossRef Link)

[7] H. Kim, et al., "Bayesian Gaussian Process Classification with the EM-EP algorithm," IEEE Trans. on PAMI, Vol. 28, No. 12, pp 1948-1959, 2006. Article (CrossRef Link)

Wanhyun Cho received both a BSc and an MSc from the Department of Mathematics, Chonnam National University, Korea, in 1977 and 1981, respectively, and a PhD from the Department of Statistics, Korea University, Korea, in 1988. He is now teaching at Chonnam National University. His research interests are statistical modeling, pattern recognition, image processing, and medical image processing.

Sangkyoon Kim received a BSc, an MSc and a PhD in Electronics Engineering, Mokpo National University, Korea, in 1998, 2000 and 2015, respectively. From 2011 to 2015, he was a Visiting Professor in the Department of Information & Electronics Engineering, Mokpo National University, Korea. His research interests include image processing, pattern recognition and computer vision.

Soonyoung Park received a BSc in Electronics Engineering from Yonsei University, Korea, in 1982 and an MSc and PhD in Electrical and Computer Engineering from State University of New York at Buffalo, in 1986 and 1989, respectively. From 1989 to 1990 he was a Postdoctoral Research Fellow in the Department of Electrical and Computer Engineering at the State University of New York at Buffalo. Since 1990, he has been a Professor with the Department of Electronics Engineering, Mokpo National University, Korea. His research interests include image and video processing, image protection and authentication, and image retrieval techniques.