A Bayesian Active Learning Framework for a Two-Class Classification Problem

Pablo Ruiz[1], Javier Mateos[1], Rafael Molina[1], and Aggelos K. Katsaggelos[2]

[1] University of Granada, 18071 Granada, Spain
mataran@decsai.ugr.es
http://decsai.ugr.es/vip
[2] Northwestern University, Evanston, IL, USA

Abstract. In this paper we present an active learning procedure for the two-class supervised classification problem. The utilized methodology exploits the Bayesian modeling and inference paradigm to tackle the problem of kernel-based data classification. This Bayesian methodology is appropriate for both finite and infinite dimensional feature spaces. Parameters are estimated, using the kernel trick, following the evidence Bayesian approach from the marginal distribution of the observations. The proposed active learning procedure uses a criterion based on the entropy of the posterior distribution of the adaptive parameters to select the sample to be included in the training set. A synthetic dataset as well as a real remote sensing classification problem are used to validate the followed approach.

1 Introduction

In many real applications large collections of data are extracted whose class is unknown. Those applications include, for instance, most image classification applications, text processing, speech recognition, and biological research problems. While extracting the samples is straightforward and inexpensive, classifying each one of those samples is a tedious and often expensive task. Active learning is a supervised learning technique that attempts to overcome the labeling bottleneck by asking queries in the form of unlabeled samples to be labeled by an *oracle* (e.g., a human annotator) [10]. An active learning procedure queries only the most informative samples from the whole set of unlabeled samples. The objective is to obtain a high classification performance using as few labeled samples as possible, minimizing, this way, the cost of obtaining labeled data.

Kernel methods in general and Support Vector Machines (SVMs) in particular dominate the field of discriminative data classification [8]. This problem has also been approached from a Bayesian point of view. For example, the relevance vector machine [13] assumes a Gaussian prior over the adaptive parameters and uses the EM algorithm to estimate them. In practice, this prior enforces sparsity because the posterior distribution of many adaptive parameters is sharply peaked around zero. Lately, Gaussian Process Classification [7] has received much attention. Adopting the least-squares SVM formulation may alternatively allow to

E. Salerno, A.E. Çetin, and O. Salvetti (Eds.): MUSCLE 2011, LNCS 7252, pp. 42–53, 2012.

perform Bayesian inference on SVMs [12]. A huge benefit is obtained by applying Bayesian inference on these machines since hyperparameters may be learned directly from data using a consistent theoretical framework.

In this paper we make use of the Bayesian paradigm to tackle the problem of active learning on kernel-based two-class data classification. The Bayesian modeling and inference approach to the kernel-based classification we propose in this paper allows us to derive efficient closed-form expressions for parameter estimation and active learning.

The general two-class supervised classification problem [2] we tackle here implies a classification function of the form:

$$y(\mathbf{x}) = \boldsymbol{\phi}^\top(\mathbf{x})\mathbf{w} + b + \epsilon, \tag{1}$$

where the mapping $\boldsymbol{\phi} : \mathcal{X} \to \mathcal{H}$ embeds the observed $\mathbf{x} \in \mathcal{X}$ into a higher L-dimensional (possibly infinite) feature space $\mathcal{H}$. The output $y(\mathbf{x}) \in \{0, 1\}$ consists of a binary coding representation of its classification, $\mathbf{w}$ is a vector of size $L \times 1$ of adaptive parameters to be estimated, b represents the bias in the classification function, and ϵ is an independent realization of the Gaussian distributions $\mathcal{N}(0, \sigma^2)$.

While kernel-based classification in *static* scenarios has been extensively studied, the problem related to the emerging field of *active learning* [10] is still unsolved. Let us assume that we have access to P vectors in the feature space denoted by $\boldsymbol{\phi}(\mathbf{x}_i), i = 1, \ldots, P$ for which the corresponding output $y(\mathbf{x}_i), i = 1, \ldots, P$ can be provided by an oracle. The key is to decide which elements $\mathbf{x}_i$ to acquire from the set of P possible samples in order to build an optimal compact classifier. Active learning aims at efficiently sampling the observations space to improve the model performance by *incrementally* building training sets. Such sets are obtained by selecting from the available samples the best ones according to a selection strategy and querying the oracle only for the label of those samples. Many selection strategies have been devised in the literature, which are based on different heuristics: 1) large margin, 2) expert committee, and 3) posterior probability (see [10] for a comprehensive review). The first two approaches typically exploit SVM methods. The latter requires classifiers that can provide posterior probabilities.

In [6], a Bayesian active learning procedure for finite dimensional feature spaces is proposed. Assuming that $\boldsymbol{\phi}(\mathbf{x}_i), i = 1, \ldots, P$ has L components, the design matrix $\boldsymbol{\Phi}_{:,:}$ is of size $P \times L$, whose i^{th} row, $i = 1, \ldots, P$ is given by $\boldsymbol{\phi}(\mathbf{x}_i)^\top$. Then, a subset of size C of the L columns of $\boldsymbol{\Phi}_{:,:}$, denoted by $\boldsymbol{\Phi}_{:,I_C}$, is selected using the differential entropy instead of the response functions $y(\mathbf{x}_i)$ [6]. Notice that this approach is in contrast to other basis selection techniques which make explicit use of the response functions, for example, [3] in the context of SVM, [4] in the context of sparse representation, and [1] considering compressive sensing. To select the rows of $\boldsymbol{\Phi}_{:,I_C}$, for which the response associated to $\phi(\mathbf{x}_i)$ will be queried, a criterion based again on differential entropy is utilized (see [6] for details). See also [5] for the general theory and [9] for the use of the approach in compressive sensing.

Here, the Bayesian modeling and inference paradigm is applied to two-class classification problems which utilize kernel-based classifiers. This paradigm is used to tackle both active learning and parameter estimation for infinite dimensional feature spaces, and consequently for problems where basis selection cannot be carried out explicitly. As we will see later, the proposed approach will make extensive use of the marginal distribution of the observations to avoid dealing with infinite dimensional feature spaces and the posterior distribution of the infinite dimensional $\mathbf{w}$.

The rest of the paper is organized as follows. Section 2 introduces the models we use in our Bayesian framework. Then, in section 3, Bayesian inference is performed. We calculate the posterior distribution of $\mathbf{w}$, and propose a methodology for parameter estimation, active learning, and class prediction. Experiments illustrating the performance of the proposed approach on a synthetic and a real remote sensing classification problem are presented in section 4. Finally, section 5 concludes the paper.

2 Bayesian Modeling

Let us assume that the target variable $y(\mathbf{x}_i)$ follows the model in Eq. (1). If we already know the classification output $y(\mathbf{x}_i)$ associated with the feature samples $\boldsymbol{\phi}(\mathbf{x}_i), i = 1, \ldots, M$, with M the number of samples, we can then write

$$\mathrm{p}(\mathbf{y}|\mathbf{w}, \sigma^2) = \prod_{i=1}^{M} \mathcal{N}(y(\mathbf{x}_i)|\boldsymbol{\phi}^\top(\mathbf{x}_i)\mathbf{w} + b, \sigma^2). \tag{2}$$

Since $\mathbf{x}_i$, $i = 1, \ldots, M$, will always appear as conditioning variable, for the sake of simplicity, we have removed the dependency on $\mathbf{x}_1, \ldots, \mathbf{x}_M$ in the left-hand side of the equation. We note that, for infinite dimensional feature vectors $\phi(\mathbf{x}_i)$, $\mathbf{w}$ is infinite dimensional.

The Bayesian framework allows us to introduce information about the possible value of $\mathbf{w}$ in the form of a prior distribution. In this work we assume that each component of $\mathbf{w}$ independently follows a Gaussian distribution $\mathcal{N}(0, \gamma^2)$. When the feature vectors are infinite dimensional, we will not make explicit use of this prior distribution but still we will be able to carry out parameter estimation and active learning tasks.

3 Bayesian Inference

Bayesian inference extracts conclusions from the posterior distribution $\mathrm{p}(\mathbf{w}|\mathbf{y}, \gamma^2, \sigma^2)$. The posterior distribution of $\mathbf{w}$ is given by [2]

$$\mathrm{p}(\mathbf{w}|\mathbf{y}, \gamma^2, \sigma^2) = \mathcal{N}(\mathbf{w}|\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}\sigma^{-2}\boldsymbol{\Phi}^\top(\mathbf{y} - b\mathbf{1}), \boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}), \tag{3}$$

where

$$\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2} = (\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \gamma^{-2}\mathbf{I})^{-1}$$

and $\boldsymbol{\Phi}$ is the design matrix whose i^{th} row is $\boldsymbol{\phi}(\mathbf{x}_i)^\top$.

It is important to note that we do not need to know the form of $\boldsymbol{\Phi}$ explicitly to calculate this posterior distribution. We only need to know the Gram matrix $\mathbf{K} = \boldsymbol{\Phi}\boldsymbol{\Phi}^\top$, which is an $M \times M$ symmetric matrix with elements $\mathbf{K}_{nm} = k(\mathbf{x}_n, \mathbf{x}_m) = \phi(\mathbf{x}_n)^\top \phi(\mathbf{x}_m)$, which has to be a positive semidefinite matrix [8]. This leads to the construction of kernel functions $k(\mathbf{x}, \mathbf{x}')$ for which the Gram matrix $\mathbf{K}$ is positive semidefinite for all possible choices of the set $\{\mathbf{x}_n\}$ [11]. Note that, even if $\boldsymbol{\Phi}$ has an infinite number of columns, which correspond to the case of $\mathbf{x}_i$ being an infinite dimensional feature vector, we can still calculate $\mathbf{K}$ of size $M \times M$ by means of the kernel function. Note also that we are somewhat abusing the notation here because $\mathbf{w}$ is infinite dimensional for infinite dimensional feature vectors.

3.1 Parameter Estimation

To estimate the values of γ^2 and σ^2 we use the Evidence Bayesian approach without any prior information on these parameters. According to it, we maximize the marginal distribution obtained by integrating out the vector of adaptive parameters $\mathbf{w}$. It can easily be shown, see for instance [2], that

$$\mathrm{p}(\mathbf{y}|\gamma^2, \sigma^2) = \mathcal{N}(\mathbf{y}|b\mathbf{1}, \boldsymbol{\Sigma}_{\mathbf{y}|\gamma^2,\sigma^2}), \tag{4}$$

where

$$\boldsymbol{\Sigma}_{\mathbf{y}|\gamma^2,\sigma^2} = \gamma^2 \boldsymbol{\Phi}\boldsymbol{\Phi}^\top + \sigma^2 \mathbf{I}.$$

The value of b can be easily obtained from Eq. (4) as

$$b = \frac{1}{M} \sum_{i=1}^{M} y(\mathbf{x}_i). \tag{5}$$

Differentiating $2 \ln \mathrm{p}(\mathbf{y}|\gamma^2, \sigma^2)$ with respect to γ^2 and equating to zero, we obtain

$$\begin{aligned} &\mathbf{tr}[(\gamma^2 \boldsymbol{\Phi}\boldsymbol{\Phi}^\top + \sigma^2)^{-1} \boldsymbol{\Phi}\boldsymbol{\Phi}^\top] = \\ &\mathbf{tr}[(\mathbf{y} - b\mathbf{1})^\top (\gamma^2 \boldsymbol{\Phi}\boldsymbol{\Phi}^\top + \sigma^2 \mathbf{I})^{-1} \boldsymbol{\Phi}\boldsymbol{\Phi}^\top (\gamma^2 \boldsymbol{\Phi}\boldsymbol{\Phi}^\top + \sigma^2 \mathbf{I})^{-1} (\mathbf{y} - b\mathbf{1})]. \end{aligned} \tag{6}$$

Diagonalizing $\boldsymbol{\Phi}\boldsymbol{\Phi}^\top$, we obtain $\mathbf{U}\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{U}^\top = \mathbf{D}$, where $\mathbf{U}$ is an orthonormal matrix and $\mathbf{D}$ is a diagonal matrix with entries $\lambda_i, i = 1, \ldots, M$. We can then rewrite the above equation as

$$\sum_{k=1}^{M} \frac{\lambda_k}{\gamma^2 \lambda_k + \sigma^2} = \sum_{i=1}^{M} z_i^2 \frac{\lambda_i}{(\gamma^2 \lambda_i + \sigma^2)^2}, \tag{7}$$

where $\mathbf{U}(\mathbf{y} - b\mathbf{1}) = \mathbf{z}$ with components $z_i, i = 1, \ldots, M$.

Multiplying both sides of the above equation by γ^2 we have

$$\gamma^2 = \sum_{i=1}^{M} \frac{\frac{\lambda_i}{\gamma^2 \lambda_i + \sigma^2}}{\sum_{k=1}^{M} \frac{\lambda_k}{\gamma^2 \lambda_k + \sigma^2}} \frac{\gamma^2 z_i^2}{\gamma^2 \lambda_i + \sigma^2} = \sum_{i=1}^{M} \mu_i \frac{\gamma^2 z_i^2}{\gamma^2 \lambda_i + \sigma^2}, \tag{8}$$

where

$$\mu_i = \frac{\frac{\lambda_i}{\gamma^2\lambda_i+\sigma^2}}{\sum_{k=1}^{M}\frac{\lambda_k}{\gamma^2\lambda_k+\sigma^2}}. \tag{9}$$

Note that $\mu_i \geq 0$ and $\sum_{i=1}^{M}\mu_i = 1$.

Similarly, differentiating $2\ln \mathrm{p}(\mathbf{y}|\gamma^2,\sigma^2)$ with respect to σ^2 and equating it to zero, we obtain

$$\sum_{k=1}^{M}\frac{1}{\gamma^2\lambda_k+\sigma^2} = \sum_{i=1}^{M} z_i^2\frac{1}{(\gamma^2\lambda_i+\sigma^2)^2}. \tag{10}$$

Following the same steps we already performed to estimate γ^2, we obtain

$$\sigma^2 = \sum_{i=1}^{M}\nu_i\frac{\sigma^2 z_i^2}{\gamma^2\lambda_i+\sigma^2}, \tag{11}$$

where

$$\nu_i = \frac{\frac{1}{\gamma^2\lambda_i+\sigma^2}}{\sum_{k=1}^{M}\frac{1}{\gamma^2\lambda_k+\sigma^2}}. \tag{12}$$

Note that, again, $\nu_i \geq 0$ and $\sum_{i=1}^{M}\nu_i = 1$.

To obtain estimates of γ^2 and σ^2 we use an iterative procedure where the values of the old estimates of γ^2 and σ^2 are used on the right hand side of Equations (8) and (11) to obtain the updated values of the parameters in the left hand side of these equations. Although we have not formally established the convergence and unicity of the solution, we have not observed any convergence problems in the performed experiments. Note that to estimate γ^2 and σ^2 we have not made use of the posterior distribution of the components of $\mathbf{w}$.

3.2 Active Learning

Active learning starts with a small set of observations whose class is already known. From these observations, the posterior distribution of $\mathbf{w}$ and the parameters b, γ^2 and σ^2 can be estimated using the procedure described in the previous sections. Now we want that the system learns new observations incrementally. Let us assume that we want to add a new observation associated to $\boldsymbol{\phi}(\mathbf{x}_+)$, whose corresponding $y(\mathbf{x}_+)$ will be learned by querying the oracle. The covariance matrix of the posterior distribution of $\mathbf{w}$ when $\boldsymbol{\phi}(\mathbf{x}_+)$ is added is given by

$$\boldsymbol{\Sigma}^{\mathbf{x}_+}_{\mathbf{w}|\mathbf{y},\gamma^2\sigma^2} = (\sigma^{-2}(\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \boldsymbol{\phi}(\mathbf{x}_+)\boldsymbol{\phi}^\top(\mathbf{x}_+)) + \gamma^{-2}\mathbf{I})^{-1}.$$

Since we have a set of observations that could be added and whose class is unknown (but can be learned by querying the oracle), the objective of active learning is to select the observation that maximizes the performance of the system, minimizing in this way the number of queries answered by the oracle. To

select this new feature vector, in this paper, we propose to maximize the difference between the entropies of the posterior distribution before and after adding the new feature vector (see [6, 9]) to obtain

$$\mathbf{x}_+ = \arg\max_{\mathbf{x}} \frac{1}{2} \log |\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}||\boldsymbol{\Sigma}^{\mathbf{x}}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}|^{-1}. \tag{13}$$

Then we have

$$\begin{aligned}
&\frac{1}{2} \log |\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}||\boldsymbol{\Sigma}^{\mathbf{x}}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}|^{-1} \\
&= \frac{1}{2} \log |\mathbf{I} + \sigma^{-2}\boldsymbol{\phi}(\mathbf{x})\boldsymbol{\phi}^\top(\mathbf{x})(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \gamma^{-2}\mathbf{I})^{-1}| \\
&= \frac{1}{2} \log(1 + \sigma^{-2}\boldsymbol{\phi}^\top(\mathbf{x})(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \gamma^{-2}\mathbf{I})^{-1}\boldsymbol{\phi}(\mathbf{x})),
\end{aligned}$$

and using

$$(\sigma^{-2}\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \gamma^{-2}\mathbf{I})^{-1} = \gamma^2\mathbf{I} - \gamma^4\boldsymbol{\Phi}^\top(\sigma^2\mathbf{I} + \gamma^2\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi}, \tag{14}$$

we can finally write

$$\begin{aligned}
&\frac{1}{2} \log |\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}| \cdot |\boldsymbol{\Sigma}^{\mathbf{x}}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}|^{-1} \\
&= \frac{1}{2} \log \left(1 + \sigma^{-2}\gamma^2\boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\phi}(\mathbf{x}) - \sigma^{-2}\gamma^4\boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\Phi}^\top(\sigma^2\mathbf{I} + \gamma^2\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi}\boldsymbol{\phi}(\mathbf{x})\right) \\
&= \frac{1}{2} \log \left(1 + \sigma^{-2}\gamma^2\boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\phi}(\mathbf{x}) - \sigma^{-2}\gamma^4\boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\Phi}^\top\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2}\boldsymbol{\Phi}\boldsymbol{\phi}(\mathbf{x})\right).
\end{aligned} \tag{15}$$

Consequently, all needed quantities to select $\mathbf{x}_+$ can be calculated without knowledge of the feature vectors and the posterior distribution of the possibly infinite dimensional adaptive parameters and using only kernel functions and the marginal distribution of the observations.

Notice that, given $\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2}$, we can easily calculate the new precision matrix $\boldsymbol{\Sigma}^{-1}_{\mathbf{y},y(\mathbf{x}_+)|\gamma^2,\sigma^2}$ of the marginal distribution of $\mathbf{y}$ when the observation corresponding to $\mathbf{x}_+$ has been added. We have

$$\boldsymbol{\Sigma}^{-1}_{\mathbf{y},y(\mathbf{x}_+)|,\gamma^2,\sigma^2} = \begin{pmatrix} \mathbf{M} & -\mathbf{M}\mathbf{v}d^{-1} \\ -d^{-1}\mathbf{v}^\top\mathbf{M} & d^{-1} + d^{-2}\mathbf{v}^\top\mathbf{M}\mathbf{v} \end{pmatrix}, \tag{16}$$

with $\mathbf{v} = \gamma^2\boldsymbol{\Phi}\boldsymbol{\phi}(\mathbf{x}_+)$, $d = \sigma^2 + \gamma^2\boldsymbol{\phi}^{\mathrm{T}}(\mathbf{x}_+)\boldsymbol{\phi}(\mathbf{x}_+)$, and $\mathbf{M} = (\boldsymbol{\Sigma}_{\mathbf{y}|\gamma^2,\sigma^2} - d^{-1}\mathbf{v}\mathbf{v}^\top)^{-1}$.

To calculate $\mathbf{M}$ we use the Sherman-Morrison-Woodbury formula to obtain

$$\mathbf{M} = \boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2} - \frac{1}{-d + \mathbf{v}^\top\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2}\mathbf{v}}\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2}\mathbf{v}\mathbf{v}^\top\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2},$$

and consequently $\boldsymbol{\Sigma}^{-1}_{\mathbf{y},y(\mathbf{x}_+)|\gamma^2,\sigma^2}$ can be calculated from the previous $\boldsymbol{\Sigma}^{-1}_{\mathbf{y}|\gamma^2,\sigma^2}$ in a straightforward manner.

Hence, starting with an initial estimation of the parameters, to perform active learning we alternate between the selection of a new sample using Eq. (13) and the estimation of the unknown parameters b, γ^2, and σ^2 using the procedure described in section 3.1.

3.3 Prediction

Once the system has been trained, we want to predict the value of $y(\mathbf{x}_*)$ for a new value of $\mathbf{x}$, denoted by $\mathbf{x}_*$. To calculate this predicted value, we make use of the distribution of $\boldsymbol{\phi}^\top(\mathbf{x}_*)\mathbf{w} + b$ where the posterior distribution of $\mathbf{w}$ is given in Eq. (3). Its mean value, $\boldsymbol{\phi}^\top(\mathbf{x}_*)\mathrm{E}[\mathbf{w}] + b$, is given by

$$\boldsymbol{\phi}^\top(\mathbf{x}_*)\mathrm{E}[\mathbf{w}] + b = \boldsymbol{\phi}^\top(\mathbf{x}_*)\boldsymbol{\Sigma}_{\mathbf{w}|\mathbf{y},\gamma^2,\sigma^2}\sigma^{-2}\boldsymbol{\Phi}^\top(\mathbf{y} - b\mathbf{1}) + b, \tag{17}$$

where we have made use of Eq. (14) to obtain

$$\begin{aligned}\boldsymbol{\phi}^\top(\mathbf{x}_*)\mathrm{E}[\mathbf{w}] + b = {} & \gamma^2\sigma^{-2}\boldsymbol{\phi}^\top(\mathbf{x}_*)\boldsymbol{\Phi}^\top(\mathbf{y} - b\mathbf{1}) \\ & - \gamma^4\sigma^{-2}\boldsymbol{\phi}^\top(\mathbf{x}_*)\boldsymbol{\Phi}^\top(\sigma^2\mathbf{I} + \gamma^2\boldsymbol{\Phi}\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^\top(\mathbf{y} - b\mathbf{1}) + b,\end{aligned} \tag{18}$$

which can be calculated without knowing the feature vectors if the kernel function is known.

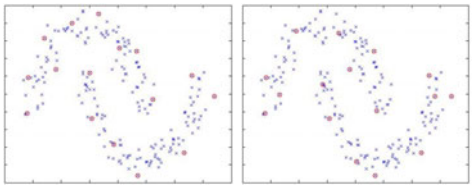
4 Experimental Results

We have tested the proposed active learning algorithm on a synthetic dataset and a real remote sensing classification problem. The synthetic data set, due to Paisley [6], consists of 200 observations, 100 from each one of the two classes, in a bi-dimensional space. The data, plotted in figure 1, is composed of two classes defined by two manifolds, which are not linearly separable in this bi-dimensional space.

We have compared the proposed active learning method with random sampling and the recently proposed Bayesian method in [6]. Random sampling was implemented using the proposed method but, instead of selecting the samples according to Eq. (13), samples are selected randomly from the available training set. In all cases, a Gaussian kernel was used, whose optimal width parameter was selected by maximizing the standard cross-validation accuracy.

We divided the full set of 200 samples into two disjoint sets of 100 randomly selected samples each, one for training and the other for testing. We started our active learning process with a seed, a single labeled sample, randomly selected from the data set, that is, $M = 1$ at the beginning and the rest of the training set was used to simulate the oracle queries. We run the three algorithms for 99 iterations adding one sample at each iteration, that is, querying the oracle one sample each time so, at the end, $M = 100$. To obtain meaningful results, the process was repeated 10 times with different randomly selected training and test sets.

The performance of the algorithms is measured utilizing the samples in the test set using the mean confusion matrix, the mean overall accuracy (OA) and OA variance, and the mean kappa index. Each cell (i, j) of the mean confusion matrix contains the mean number of samples, over the ten executions of the algorithms using the different training and test sets, belonging to the j-th class, classified in the i-th class. The overall accuracy is the proportion of correctly



(a) (b)

Fig. 1. First 15 selected samples for (a) the method in [6] and (b) the proposed method

classified samples over the total number of samples. The mean OA averages the ten OA results of the ten different algorithm executions. The variance of the OA in all the executions is reported as OA variance. The kappa index is a statistical measure, which reflects agreement between the obtained accuracy and the accuracy that would be expected by randomly classifying the samples. Unlike the Overall Accuracy, the kappa index avoids the chance effect. A value of the kappa index greater than 0.8 is considered to be "very good". Since ten runs of the algorithm are performed, the mean kappa over all the executions index is used.

In Figure 1 we show the first 15 selected samples for the method in [6] and the proposed method. It can be seen that both algorithms select samples that efficiently represent the two manifolds. Figure 2 shows the average learning curves for random sampling, the method in [6] and the proposed method. From the figure, it is clear that random sampling provides the lowest convergence rate, while the method in [6] and the proposed method have a similar learning rate to the full set overall accuracy. At convergence, when 100 samples are included in the training set, all methods have the same accuracy but the proposed method reaches this value with 18.4 samples on the average while the method in [6] needs 28.2 samples and random sampling needs 36.4 samples.

In the second experiment a real remote sensing dataset was used. Satellite or airborne mounted sensors usually capture a set of images of the same area in several wavelengths or spectral channels forming a multispectral image. This multispectral image allows for the classification of the pixels in the scene into different classes to obtain classification maps used for management, policy making and monitoring. A critical problem in remote sensing image segmentation is that few labeled pixels are typically available: in such cases, active learning may be very helpful [14].

We evaluated the methods on a real Landsat 5 TM image, whose RGB bands are depicted in Fig. 3a. The region of interest is a 1024×1024 pixels area centered in the city of Granada, in the south of Spain. The Landsat TM sensor provides a six bands multispectral image that covers RGB, near-infrared and mid-infrared ranges with a spatial resolution of 30 meters per pixel, that is, each pixels captures the energy reflected by the Earth in a square area of side equal to 30 meters. The dataset, created by the RSGIS Laboratory at the University of Granada, divides the scene into two classes, vegetation and no-vegetation. Note that the

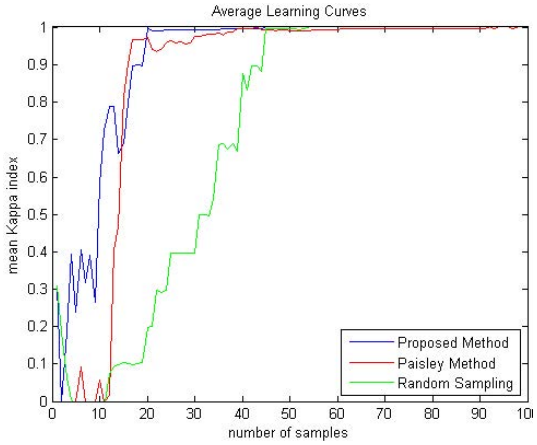
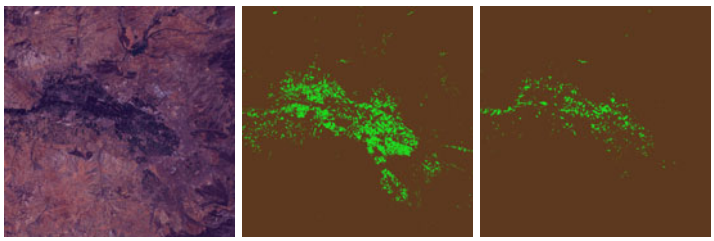


Fig. 2. Average learning curves for the active learning techniques using random sampling, the Bayesian method in [6] (Paisley method), and the proposed method for the synthetic experiment



(a) (b) (c)

Fig. 3. (a) Multispectral image, (b) classification map with the proposed method, and (c) classification map with the method in [6]. Pixels classified as vegetation are shown in green color and pixels classified as no-vegetation are shown in brown.

no-vegetation class includes bare soil that has a very similar spectral signature to vegetation making the correct classification of the pixels a challenging problem.

A total of 336 samples, whose class is precisely known by visual inspection of the images and by terrain inspection, were selected from the image, 174 samples corresponding to the vegetation class and 162 samples corresponding to the no-vegetation class. Each sample has six characteristics, each one corresponding to the mean value of a 3×3 area centered in the pixel under study for each one of the six bands that comprise the multispectral information provided by the Landsat TM satellite. Again, the same Gaussian kernel was used for all methods.

From the labeled dataset a test set of 150 samples was randomly selected, and the remaining 186 samples were used to simulate the oracle queries. We run the experiments 10 times with different training and test sets. All the algorithms

Table 1. Mean confusion matrix, mean kappa index, mean overall accuracy and its variance for ten runs of the method in [6] on different test sets

Predicted/actual	**vegetation**	**no-vegetation**
vegetation	74.4	3.5
no-vegetation	0.6	71.5

Mean Kappa = 0.9453
Mean OA = 97.27%
OA variance = 4.39×10^{-5}

Table 2. Mean confusion matrix, mean kappa index, mean overall accuracy and its variance for ten runs of the proposed method on different test sets

Predicted/actual	**vegetation**	**no-vegetation**
vegetation	74.4	2.4
no-vegetation	0.6	72.6

Mean Kappa = 0.96
Mean OA = 98.00%
OA variance = 9.87×10^{-5}

were run for 185 iterations, starting from a training set with a single labeled pixel, that is $M = 1$, and adding one pixel to the training set at each iteration (query).

Again, the proposed method is compared with random sampling and the Bayesian method in [6]. For the method in [6] we did not perform the basis selection step. We want to note that, since this basis selection procedure discards features from the samples, better results are expected when all the features are used although the computational cost will be higher.

Figure 4 shows the average learning curves. The method in [6] provides a lower convergence rate to the full set overall accuracy than the proposed method. However, the method in [6] starts learning faster than the proposed one. It may be due to the fact that the active learning is carried out in an M-dimensional feature space while the proposed method works in an infinite-dimensional space. However, at convergence, when 186 samples have been included in the training set, the proposed method performs better than the method in [6]. Note also that, at convergence, random sampling obtains the same results with the proposed method, obtaining better classification accuracy than the method in [6]. This was expected since it uses the same classification procedure as the proposed method, except for the active learning selection procedure. Note, however, that the convergence rate is much slower than the other two methods.

Figures 3b and 3c depict the classification map for the full image using the proposed method and the method in [6]. The random sampling classification is not shown since, at convergence, coincides with the proposed method. The mean of the confusion matrices as well as the mean kappa index, the mean overall accuracy, and the overall accuracy variance are shown in Tables 1 and 2, for the method in [6] and the proposed method, respectively. From these figures of merit it is clear that the proposed method discriminates better between vegetation and no-vegetation than the method in [6].

All compared methods were implemented using Matlab© and run on a Intel i7 @ 2.67GHz. The proposed method took 1.23 sec to complete the 185 iterations while the method in [6] took 48.44 sec and random sampling took 1.01 sec. It

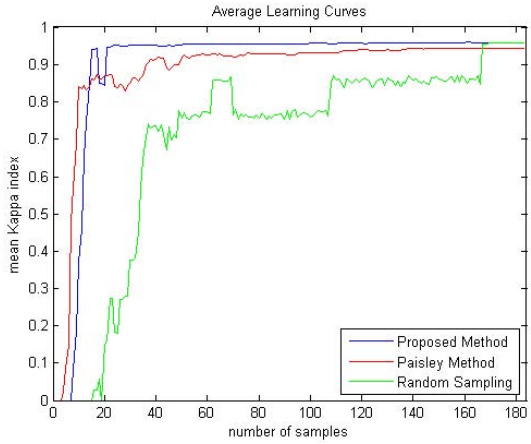


Fig. 4. Learning curve for the active learning techniques using random sampling, the Bayesian method in [6] (Paisley method), and the proposed method for the real remote sensing dataset

is worth noting that computing the precision matrix $\boldsymbol{\Sigma}^{-1}_{\mathbf{y},y(\mathbf{x}_+)|\gamma^2,\sigma^2}$ in Eq. (16) takes most of the time, which explains the similar cost between the proposed method and random sampling. It is worth noting that the proposed method provided better figures of merit than the method in [6] for both mean kappa index and mean overall accuracy, learning with less interaction with the oracle and, also, with a much lower computational cost.

5 Conclusions

We presented an active learning procedure that exploits Bayesian learning and parameter estimation to tackle the problem of two-class kernel-based data classification. Using the Bayesian modeling and inference, we developed a Bayesian method for classification both finite and infinite dimensional feature spaces. The proposed method allows us to derive efficient closed-form expressions for parameter estimation and incremental and active learning. The method was experimentally compared to other methods and its performance was assessed on remote sensing multispectral image as well as synthetic data.

Acknowledgments. This work has been supported by the Spanish research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018) and the "Consejería de Innovación, Ciencia y Empresa of the Junta de Andalucía" under contract P07-TIC-02698. We want to thank V. F. Rodríguez-Galiano and Prof. M. Chica from the RSGIS laboratory (Group RNM122 of the Junta de Andalucía), who are supported by the Spanish MICINN (CGL2010-17629), for the image of the neighborhood of the city of Granada and the classified samples that conformed the real dataset used in this paper.

References

1. Babacan, D., Molina, R., Katsaggelos, A.: Bayesian compressive sensing using Laplace priors. IEEE Transactions on Image Processing 19(1), 53–63 (2010)
2. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer (2007)
3. Burges, C.J.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
4. Elad, M.: Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing. Springer (2010)
5. MacKay, D.J.C.: Information-based objective functions for active data selection. Neural Computation 4(4), 590–604 (1992)
6. Paisley, J., Liao, X., Carin, L.: Active learning and basis selection for kernel-based linear models: A Bayesian perspective. IEEE Transactions on Signal Processing 58, 2686–2700 (2010)
7. Rasmussen, C.E., Williams, C.K.: Gaussian Processes for Machine Learning. MIT Press, NY (2006)
8. Schölkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambrigde (2002)
9. Seeger, M.W., Nickisch, H.: Compressed sensing and Bayesian experimental design. In: International Conference on Machine Learning 25 (2008)
10. Settles, B.: Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison (2009)
11. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge Univ. Press (2004)
12. Suykens, J.A.K., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J.: Least Squares Support Vector Machines. World Scientific, Singapore (2002)
13. Tipping, M.E.: The relevance vector machine. Journal of Machine Learning Research 1, 211–244 (2001)
14. Tuia, D., Volpi, M., Copa, L., Kanevski, M., Muñoz-Marí, J.: A survey of active learning algorithms for supervised remote sensing image classification. IEEE J. Sel. Topics Signal Proc. 4, 606–617 (2011)