Communications for Statistical Applications and Methods
2019, Vol. 26, No. 6, 635–644

https://doi.org/10.29220/CSAM.2019.26.6.635
Print ISSN 2287-7843 / Online ISSN 2383-4757

A model-free soft classification with a functional predictor

Eugene Lee[a], Seung Jun Shin[1,a]

[a]Department of Statistics, Korea University, Korea

Abstract

Class probability is a fundamental target in classification that contains complete classification information. In this article, we propose a class probability estimation method when the predictor is functional. Motivated by Wang *et al.* (*Biometrika*, **95**, 149–167, 2007), our estimator is obtained by training a sequence of functional weighted support vector machines (FWSVM) with different weights, which can be justified by the Fisher consistency of the hinge loss. The proposed method can be extended to multiclass classification via pairwise coupling proposed by Wu *et al.* (*Journal of Machine Learning Research*, **5**, 975–1005, 2004). The use of FWSVM makes our method model-free as well as computationally efficient due to the piecewise linearity of the FWSVM solutions as functions of the weight. Numerical investigation to both synthetic and real data show the advantageous performance of the proposed method.

Keywords: functional data, Fisher consistency, support vector machines, probability estimation

1. Introduction

Binary classification is frequently encountered in machine learning applications. Whaba (2002) categorizes the binary classification into hard and soft classification. Hard classification predicts a class label directly and the support vector machine (SVM) (Vapnik, 1995) that trains the decision boundary falls in this category. However, soft classification seeks the class probability denoted by $p(\mathbf{x}) = P(Y = 1|\mathbf{X} = \mathbf{x})$ where $Y \in \{-1, 1\}$ and $\mathbf{X} \in \mathbb{R}^p$ are a binary response and a p-dimensional predictor, respectively. Another popular binary classifier, logistic regression is a canonical example of soft classification. The soft classification is more difficult than the hard classification since $p(\mathbf{x}) \in [0, 1]$ is a more informative quantity with higher resolution than a dichotomous $Y \in \{-1, 1\}$.

Recent applications often regarded a predictor as a function, $x(t)$ rather than a vector. It is not possible to observe the functional predictor completely in the sample level; therefor, we are given a set of their realizations denoted by $\mathbf{x}_i(\mathbf{t}_i) = (x_i(t_{i1}), \ldots, x_i(t_{id_i}))^T$, $i = 1, \ldots, n$ with $\mathbf{t}_i = (t_{i1}, \ldots, t_{id_i})^T$ being a grid at which the function for the i^{th} example, $x_i(t)$ is evaluated. With functional predictors, several binary classification methods have been developed by extending conventional binary classifiers to the functional context. James (2002) proposed a functional generalized linear model which includes the functional logistic regression (FLR). FLR is a soft classification method but requires a distributional assumption on the response which may not be valid in practice. Rossi and Villa (2006) and Park *et al.* (2008) proposed functional support vector machines (FSVM). The FSVM showed a promising performance for the prediction, yet could not estimate the class probability.

We propose a model-free soft classification method with a functional predictor. In particular, we extend the idea of Wang *et al.* (2007) where class probability is estimated by training a sequence

[1] Corresponding author: Department of Statistics, Korea University, 145 Anam-Ro, Sungbuk-Gu, Seoul 02841, Korea. E-mail: sjshin@korea.ac.kr

Published 30 November 2019 / journal homepage: http://csam.or.kr

of weighted SVM (WSVM) for different weights. We first introduce a weighted version of FSVM proposed by Park *et al.* (2008) which we call functional WSVM (FWSVM). The class probability is then estimated by training a sequence of FWSVM for different weights as suggested by Wang *et al.* (2007). For the computation, we exploit the piecewise linearity of the FWSVM solution as function of the weight parameter, which improves the computational efficiency of the probability estimator. Finally, we extend our method to multi-class classification via the pairwise coupling algorithm proposed by Wu *et al.* (2004).

The rest of the article is organized as follows. In Section 2, we provide a review of the probability estimation scheme based on WSVM that serves as a building block of our proposal. In Section 3, we propose a class probability estimator in binary classification with a functional predictor, and then extend the idea to the multi-class classification via a pairwise coupling algorithm. In Section 4, we conduct simulation studies to evaluate the finite sample performance of the proposed method, and illustration to real data in Section 5. Finally, concluding summaries are given in Section 6.

2. Probability estimation via weighted support vector machine

We start with a brief review of a model-free class probability estimation scheme based on the WSVM proposed by Wang *et al.* (2007), which serves as a building block for our proposal developed in the following Section 3.

Suppose we are given a set of training samples $\{y_i, \mathbf{x}_i\} \in \{-1, +1\} \times \mathbb{R}^p$, for $i = 1, \ldots, n$. For the sample estimation, we assuming that the classification function f_π resides on the reproducing kernel Hilbert space (RKHS) (Wahba, 1990), $\mathcal{H}_K$ generated by a positive definite kernel $K(\mathbf{x}, \mathbf{x}')$. Namely, the WSVM solves

$$\hat{f}_\pi(\mathbf{x}) = \underset{f \in \mathcal{H}_K}{\operatorname{argmin}} \sum_{i=1}^{n} w_\pi(y_i) H_1\{y_i f(\mathbf{x}_i)\} + \frac{\lambda}{2}\|f\|^2_{\mathcal{H}_K}, \tag{2.1}$$

where $H_1(u) = [1 - u]_+$ denotes the hinge loss and $w_\pi(y)$ is a weight function that takes $1 - \pi$ when $y = 1$ and π otherwise, with $\pi \in [0, 1]$ being a weight parameter controlling the relative importance between the positive and negative classes. According to *Represent Theorem* (Kimeldorf and Wahba, 1971) the minimizer of (2.1) must have the following finite form:

$$f_\pi(\mathbf{x}) = \frac{1}{\lambda}\left\{\alpha_0 + \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i)\right\}. \tag{2.2}$$

Plugging (2.2) into (2.1), the WSVM (2.1) is equivalently rewritten as the finite dimensional optimization problem:

$$(\hat{\alpha}_{0,\pi}, \hat{\boldsymbol{\alpha}}_\pi) = \underset{\alpha_0, \boldsymbol{\alpha}}{\operatorname{argmin}} \sum_{i=1}^{n} w_\pi(y_i) H_1\{y_i f_\pi(\mathbf{x}_i)\} + \frac{1}{2\lambda}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{2.3}$$

A connection between decision function $f_\pi(\mathbf{x})$ from the WSVM and the corresponding class probability $p(\mathbf{x})$ is guided by Fisher consistency of the hinge loss. Fisher consistency of the WSVM states that for a given $\mathbf{X} = \mathbf{x}$,

$$\operatorname{sign}\{f^*(\mathbf{x})\} = \operatorname{sign}\{p(\mathbf{x}) - \pi\}, \tag{2.4}$$

where $f^*(\mathbf{x}) = \text{argmin}_f E[w_\pi(Y)H_1\{Yf(\mathbf{X})\} \mid \mathbf{X} = \mathbf{x}]$. Fisher consistency implies that the population minimizer of the weighted hinge risk yields the corresponding Bayes classifier.

Fisher consistency (2.4) provides a natural way to recover $p(\mathbf{x})$ by training a series of WSVMs with different values of weight π, as described in the following. Given $\mathbf{X} = \mathbf{x}$, $\hat{f}_\pi(\mathbf{x})$ can be viewed as a continuous function of $\pi \in [0, 1]$. Since $\hat{f}_0(\mathbf{x}) > 0$ and $\hat{f}_1(\mathbf{x}) < 0$ for all $\mathbf{x}$, we can always find π^* such that $\hat{f}_{\pi^*}(\mathbf{x}) = 0$. By the Fisher consistency of the hinge loss (2.4), π^* can be used as an estimator of $p(\mathbf{x})$. In order to obtain π^*, Wang *et al.* (2007) proposed the following procedure. For a given grid of π, $0 < \pi_1 < \cdots < \pi_M < 1$, a series of the corresponding WSVM solutions denoted by $\hat{f}_m, m = 1, \ldots, M$ is trained, where the subscript m is used to denote the WSVM solution with $\pi = \pi_m$. Finally, the class probability estimator $\hat{p}(\mathbf{x})$ is given by

$$\hat{p}(\mathbf{x}) = \frac{1}{2}(\bar{\pi}(\mathbf{x}) + \underline{\pi}(\mathbf{x})),$$

where $\bar{\pi}(\mathbf{x}) = \text{argmax}_{\pi_m}[\text{sign}\{\hat{f}_m(\mathbf{x})\} = 1]$ and $\underline{\pi}(\mathbf{x}) = \text{argmax}_{\pi_m}[\text{sign}\{\hat{f}_m(\mathbf{x})\} = -1]$.

3. Proposed method

3.1. Binary classification

We are given a set of binary responses $y_i \in \{-1, 1\}$ and functional predictors $\mathbf{x}_i(\mathbf{t}_i) = (x_i(t_{i1}), \ldots, x_i(t_{id_i}))^T$ where $\mathbf{t}_i = (t_{i1}, \ldots, t_{id_i})$, $i = 1, \ldots, n$. Assuming the functional predictor $x_i(t)$ is square integrable on a finite interval, i.e., $x_i \in L^2[0, T]$ for $T < \infty$, it can be expressed as $x_i(t) = \sum_{m=1}^{\infty} c_{i,m}\phi_m(t)$ for a given basis system $\{\phi_l\}_{m=1}^{\infty}$. A popular choice of the basis system includes Fourier, B-spline, and cubic-spline basis. Given a basis system, one can readily estimate $x_i(t)$ from the observed $\mathbf{x}_i$ by $\hat{x}_i(t) = \sum_{m=1}^{M} \hat{c}_{i,m}\phi(t_j), i = 1, \ldots, n$ for sufficiently large M, where

$$\hat{\mathbf{c}}_i = (\hat{c}_{i,1}, \ldots, \hat{c}_{i,M})^T = \underset{\mathbf{c}_i}{\text{argmin}} \sum_{j}^{d_i} \left\{ x_i(t_j) - \sum_{m=1}^{M} c_{i,m}\phi(t_j) \right\}^2.$$

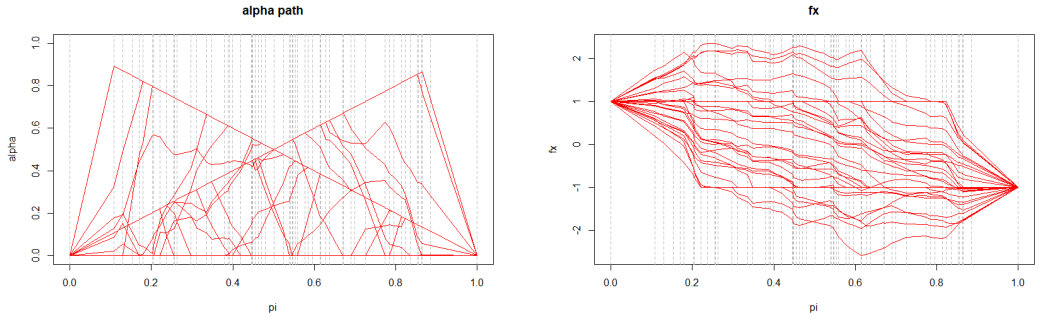
Park *et al.* (2008) suggest to use the inner product to measure the similarity between functional predictors. The proposed kernel is

$$K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \langle \hat{x}_i, \hat{x}_j \rangle = \hat{\mathbf{c}}_i' \Phi \hat{\mathbf{c}}_j, \tag{3.1}$$

where $\Phi = (\int_0^T \phi_i(t)\phi_j(t)d_t)_{i,j=1,\ldots,M}$ inner product matrix of M basis systems. Notice that (3.1) can be regarded as a linear kernel for functional predictors.

Now, it is straightforward to define the FWSVM as the WSVM (2.3) with the kernel given in (3.1). In this regard, FWSVM is a version of WSVM (2.3). One distinguishing feature of the WSVM and the FWSVM is the piecewise linearity of $(\hat{\alpha}_{0,\pi}, \hat{\boldsymbol{\alpha}}_\pi)$ as a function of π for any given $\lambda > 0$ (Wang *et al.*, 2007). This enables us to efficiently recover the entire trajectories of $(\hat{\alpha}_{0,\pi}, \hat{\boldsymbol{\alpha}}_\pi)$, which we call π-path. Note also that $\hat{f}_\pi(\mathbf{x})$ is a linear function of $(\hat{\alpha}_{0,\pi}, \hat{\boldsymbol{\alpha}}_\pi)$ and thus piecewise linear in π. Shin *et al.* (2014) showed that $(\hat{\alpha}_{0,\pi}, \hat{\boldsymbol{\alpha}}_\pi)$ are jointly piecewise linear in λ and π.

Figure 1 shows the illustration of the piecewise linear solution paths of (a) $\hat{\alpha}_{i,\pi}$, and (b) $\hat{f}_\pi(\mathbf{x}_i), i = 1, \ldots, n$ for the FWSVM obtained from a simulated data. Given the π-path, the entire trajectory of $\hat{f}_\pi(\mathbf{x})$ on $\pi \in [0, 1]$ for an arbitrary given $\mathbf{x}$ readily follows and the corresponding probability estimator $\hat{\pi}$ that solves $\hat{f}_\pi(\mathbf{x}) = 0$ is obtained.



(a) π-path of $\hat{\alpha}_{i,\pi}$, $i = 1, \ldots, n$ (b) π-path of $\hat{f}_{\pi}(x_i)$, $i = 1, \ldots, n$

Figure 1: *Piecewise linear solution paths for the FWSVM solution as function of π. FWSVM = functional weighted support vector machines.*

3.2. Multiclass classification

We extend the proposed method to the multi-class problem with a K-class categorical response $Y \in \{1, \ldots, K\}$ by applying the pairwise coupling algorithm described in the following. Let p_k denote the k^{th} class probability, i.e., $P(Y = k \mid \mathbf{X} = \mathbf{x}), k = 1, \ldots, K$, the pairwise coupling algorithm estimates $\mathbf{p} = (p_1, \ldots, p_K)^T$ from the pairwise class probabilities, $r_{kl} = P(Y = k \mid Y \in \{k, l\}, \mathbf{X} = \mathbf{x}), \forall k < l$. By construction, we have $r_{lk}p_k = r_{kl}p_l, \forall k \neq l$. This leads to solve the following problem to estimate $\mathbf{p}$:

$$\hat{\mathbf{p}} = \underset{\mathbf{p}}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{l \neq k} (\hat{r}_{lk}p_k - \hat{r}_{kl}p_l)^2, \quad \text{s.t.} \sum_{k=1}^{K} p_k = 1, \tag{3.2}$$

where $\hat{r}_{ij}$ denotes an estimator of r_{ij}. Wu *et al.* (2004) further showed that (3.2) is equivalently rewritten as

$$\hat{\mathbf{p}} = \underset{\mathbf{p}}{\operatorname{argmin}} \frac{1}{2} \boldsymbol{p}^T \mathbf{Q} \boldsymbol{p}, \quad \text{s.t.} \sum_{k=1}^{K} p_k = 1, \tag{3.3}$$

where $\mathbf{Q}$ is the K-dimensional square matrix whose $(k, l)^{th}$ element is given by $\sum_{s \neq k} \hat{r}_{sk}^2$ if $k = l$ and $-\hat{r}_{lk}\hat{r}_{kl}$ otherwise. The alternative formulation (3.3) can be readily solved in an iterative way. We refer Section 4 of Wu *et al.* (2004) for complete details on the pairwise coupling algorithm.

Finally, we estimate r_{kl} by applying our method for binary response as described in Section 3.1 to the subset of data whose response is either k or l for all $k < l$, $k, l = 1, \ldots, K$ then the class probability $\mathbf{p}$ can be estimated from (3.3).

4. Simulation

We conduct a simulation study to evaluate the finite-sample performance of the proposed probability estimator. We first generate a binary response Y taking 1 with probability 0.5 and -1 otherwise, then the functional predictors x from the following Gaussian process model:

$$X(t) \mid Y = y \sim \mathrm{GP}(\mu_y(t), \delta \times \Sigma(t, s)),$$

where $\mu_y(t)$ denotes the mean function indexed by class label $y \in \{+1, -1\}$, $\Sigma(t, s)$ is the covariance kernel, and $\delta > 0$ is a constant that controls the noise level. We used ten equally-spaced grid for t of the functional predictor on $[0, \pi]$. Notice that π in this section represents the constant $3.141592\cdots$, not the weight in the FWSVM. For the choice of covariance kernel Σ we consider independent $\Sigma(s,t) = \mathbb{1}\{s = t\}$; compound symmetric $\Sigma(s,t) = 1$ if $s = t$ and ρ_{cs} otherwise; and autoregressive structures $\Sigma(s,t) = \rho_{\text{ar}}^{|t-s|}$. We use $\rho_{\text{cs}} = 0.3$ and $\rho_{\text{ar}} = 0.7$. The noise level δ is set to be either 0.3 or 0.5.

We compute the thee quantities as performance measure for the independent test set $(y'_j, \mathbf{x}'_j)$, $j = 1, \ldots, n'$: cross entropy (CE), absolute probability difference (PD), and weighted absolute probability difference (WD), respectively defined as:

- CE: $-\sum_{j=1}^{n'} \left[\mathbb{1}\{y'_j = 1\} \log(\hat{p}(\mathbf{x}_j)) + \mathbb{1}\{y'_j = -1\} \log\{1 - \hat{p}(\mathbf{x}'_j)\}\right]$;
- PD: $\sum_{j=1}^{n'} \left|p(\mathbf{x}'_j) - \hat{p}(\mathbf{x}'_j)\right|$;
- WD: $\sum_{j=1}^{n'} w_j \left|p(\mathbf{x}'_j) - \hat{p}(\mathbf{x}'_j)\right|$ where $w_j = \sqrt{p(\mathbf{x}'_j)(1 - p(\mathbf{x}'_j))}$.

All the three performance measures yield the smaller values for the better performance in estimating the class probability. Finally, we generated $n \in \{100, 300\}$ training and $n' = 300$ test examples independently under all combinations of the three covariance kernels Σ and the mean functions $\mu_y(t)$ to be described in the following subsections, and compared the finite performance of the proposed method (FWSVM) to the FLR (James, 2002) in terms of CE, PD, and WD.

For training the FWSVM, the B-spline basis system with 10 equally spaced knots is employed and λ is chosen via the grid search to minimize the cross-validated CE. The π-path is then explicitly computed for the given λ, and our probability estimator defined by $\pi \in [0, 1]$ such that $\hat{f}_\pi(\mathbf{x}) = 0$ directly follows for an arbitrary given $\mathbf{x}$.

4.1. Binary classification

In binary classification, we consider four different mean functions as follows.

- (B1) $\mu_-(t) = t$, $\mu_+(t) = t + 2$
- (B2) $\mu_-(t) = t$, $\mu_+(t) = -t + 1$
- (B3) $\mu_-(t) = \sin(t)$, $\mu_+(t) = \sin(t) + 2$
- (B4) $\mu_-(t) = \sin(\pi t/2)$, $\mu_+(t) = \cos(\pi t/2)$

Model (B1) and (B2) are linear while (B3) and (B4) are nonlinear. The mean functions of two classes are parallel in (B1) and (B3), and crossed in (B2) and (B4). Table 1 reports the comparison results to FLR under the independent covariance kernel function. One can observe that the proposed method outperforms FLR for all scenarios under consideration. The results are similar for other two different covariance kernels which are relegated to Supplementary Materials to avoid redundancy.

4.2. Multi-class classification

We consider three-class classification with four different mean functions as:

- (M1) $\mu_1(t) = t$, $\mu_2(t) = t + 1$, $\mu_3(t) = t + 2$

Table 1: Simulation results for binary classification with the independent covariance kernel

Model	n	δ	CE		PD		WD	
			FWSVM	FLR	FWSVM	FLR	FWSVM	FLR
(B1)	100	0.3	0.001(0.000)	0.031(0.004)	0.001(0.000)	0.029(0.004)	0.000(0.000)	0.000(0.000)
		0.5	0.000(0.000)	0.041(0.005)	0.000(0.000)	0.039(0.005)	0.000(0.000)	0.000(0.000)
	300	0.3	0.000(0.000)	0.029(0.002)	0.000(0.000)	0.028(0.002)	0.000(0.000)	0.000(0.000)
		0.5	0.000(0.000)	0.039(0.003)	0.000(0.000)	0.037(0.003)	0.000(0.000)	0.000(0.000)
(B2)	100	0.3	0.042(0.011)	0.130(0.011)	0.032(0.004)	0.068(0.007)	0.010(0.001)	0.017(0.002)
		0.5	0.110(0.018)	0.182(0.014)	0.046(0.006)	0.087(0.008)	0.016(0.002)	0.023(0.002)
	300	0.3	0.025(0.007)	0.125(0.007)	0.039(0.003)	0.063(0.004)	0.012(0.001)	0.016(0.001)
		0.5	0.086(0.015)	0.174(0.009)	0.054(0.019)	0.099(0.030)	0.018(0.005)	0.024(0.004)
(B3)	100	0.3	0.001(0.000)	0.031(0.004)	0.001(0.000)	0.029(0.004)	0.000(0.000)	0.000(0.000)
		0.5	0.000(0.000)	0.041(0.005)	0.000(0.000)	0.039(0.005)	0.000(0.000)	0.000(0.000)
	300	0.3	0.000(0.000)	0.029(0.002)	0.000(0.000)	0.028(0.002)	0.000(0.000)	0.000(0.000)
		0.5	0.000(0.000)	0.039(0.003)	0.000(0.000)	0.037(0.003)	0.000(0.000)	0.000(0.000)
(B4)	100	0.3	0.034(0.007)	0.124(0.010)	0.031(0.007)	0.080(0.018)	0.009(0.002)	0.017(0.002)
		0.5	0.095(0.016)	0.174(0.013)	0.041(0.005)	0.089(0.008)	0.014(0.002)	0.022(0.002)
	300	0.3	0.020(0.006)	0.120(0.007)	0.032(0.002)	0.066(0.004)	0.009(0.001)	0.016(0.001)
		0.5	0.074(0.014)	0.166(0.008)	0.038(0.004)	0.083(0.005)	0.013(0.001)	0.021(0.001)

Averaged values of CE, PD, and WD over 100 independent repetitions are reported along with the corresponding standard errors in parentheses. CE = cross entropy; PD = absolute probability difference; WD = weighted absolute probability difference; FWSVM = functional weighted support vector machines; FLR = functional logistic regression.

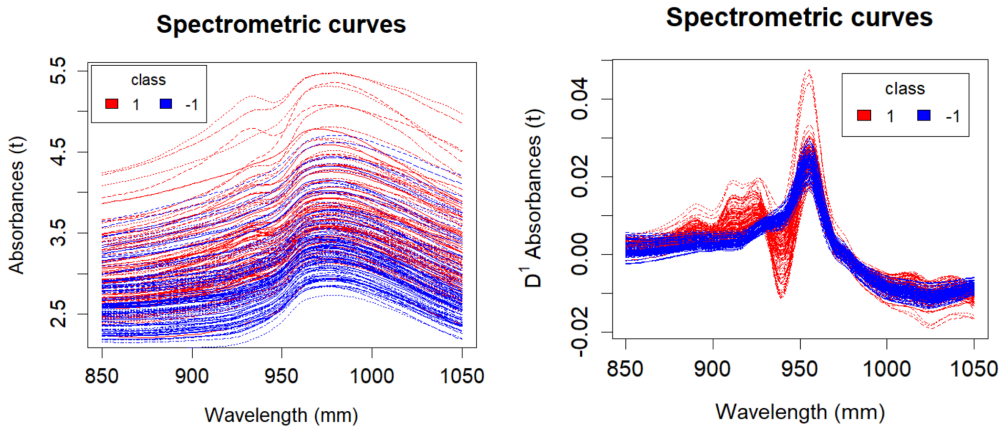
Table 2: Simulation results for multi-class classification with the independent covariance kernel

Model	n	δ	CE		PD	
			FWSVM	FLR	FWSVM	FLR
(M1)	200	0.3	0.002(0.001)	0.100(0.005)	0.002(0.001)	0.090(0.005)
		0.5	0.007(0.003)	0.138(0.007)	0.005(0.001)	0.117(0.006)
	500	0.3	0.000(0.000)	0.097(0.004)	0.001(0.000)	0.087(0.004)
		0.5	0.005(0.003)	0.134(0.006)	0.003(0.001)	0.114(0.005)
(M2)	200	0.3	0.005(0.002)	0.094(0.006)	0.005(0.001)	0.079(0.005)
		0.5	0.020(0.006)	0.129(0.007)	0.010(0.002)	0.100(0.006)
	500	0.3	0.002(0.002)	0.092(0.005)	0.005(0.001)	0.078(0.004)
		0.5	0.015(0.005)	0.127(0.006)	0.009(0.002)	0.099(0.005)
(M3)	200	0.3	0.002(0.001)	0.100(0.005)	0.002(0.001)	0.090(0.005)
		0.5	0.007(0.003)	0.138(0.007)	0.005(0.001)	0.117(0.006)
	500	0.3	0.000(0.000)	0.097(0.004)	0.001(0.000)	0.087(0.004)
		0.5	0.005(0.003)	0.134(0.006)	0.003(0.001)	0.114(0.005)
(M4)	200	0.3	0.002(0.001)	0.108(0.007)	0.002(0.001)	0.096(0.006)
		0.5	0.007(0.003)	0.149(0.009)	0.005(0.001)	0.125(0.007)
	500	0.3	0.000(0.000)	0.106(0.005)	0.001(0.000)	0.094(0.004)
		0.5	0.004(0.003)	0.146(0.007)	0.003(0.001)	0.123(0.005)

Averaged values of CE and PD over 100 independent repetitions are reported along with the corresponding standard errors in parentheses. CE = cross entropy; PD = absolute probability difference; FWSVM = functional weighted support vector machines; FLR = functional logistic regression.

- (M2) $\mu_1(t) = 2t, \quad \mu_2(t) = -2t, \quad \mu_3(t) = 0.5t$
- (M3) $\mu_1(t) = \sin(t), \quad \mu_2(t) = \sin(t) + 1, \quad \mu_3(t) = \sin(t) + 2$
- (M4) $\mu_1(t) = \sin(3t), \ \mu_2(t) = -\sin(3t) + 1, \ \mu_3(t) = \cos(t/10)$

Analogous to the binary classification model (B1)–(B4), Model (M1) and (M2) have linear and (B3) and (B4) have nonlinear mean functions for each classes. The mean functions are parallel in (M1) and (M3), and crossed in (M2) and (M4). Table 2 reports the comparison results under the independent



(a) Original predictor (b) Deriatives

Figure 2: *Tecator data: functional predictors obtained by employing the B-spline basis system. Derivative looks more informative for the classification.*

covariance kernel. For the multi-class cases, WD measure is not considered due to the ambiguity in determining suitable weights. Similar to the binary cases, our method outperforms the FLR in all scenarios under consideration. Again, the results for the other covariance structures are relegated to Supplementary Materials.

Our method shows advantageous performance in estimating class probability estimator in both binary and multi-class classifications.

5. Real data illustration

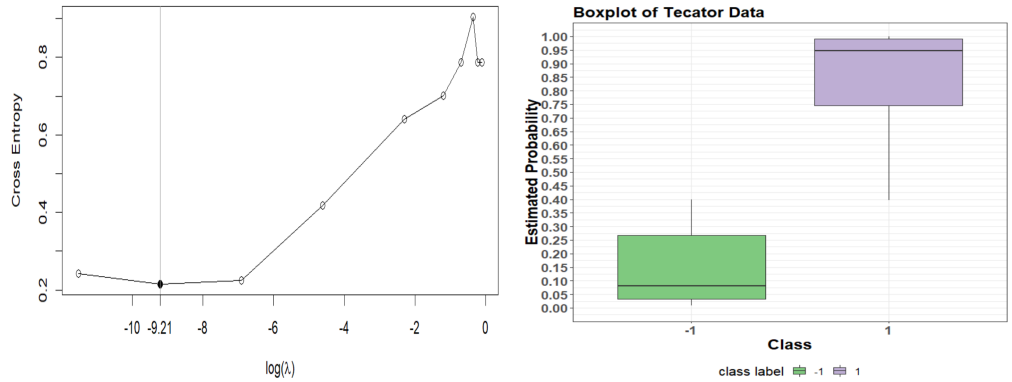
In this section, we illustrate our method to two real data sets: tecator data for binary, and phoneme data for multi-class classification.

5.1. Tecator data

Tecator data consists of 215 meat samples. Response variable is the percentage of fat content in meat, and we dichotomize it: take 1 if it is greater than 15 and -1 otherwise. Functional predictor is the absorbance values measured on each 100 wavelengths. Figure 2 depicts the (a) functional predictors and their (b) derivatives, both of which are obtained by employing a B-spline basis with 10 equally spaced knots. We decide to use derivatives as predictors because the two classes are more clearly separated in terms of derivatives. We employ the grid search to find an optimal λ in FWSVM, and Figure 3(a) depicts the cross-validated CE for different values of λ. The optimal λ is selected as $\log \lambda = -9.21$. Finally, Figure 2(b) depicts boxplots of cross-validated probability estimates for the two classes and we can observe that the proposed method performs well in the sense that the distributions of the estimated class probabilities for test examples are clearly separated according to their true class labels.

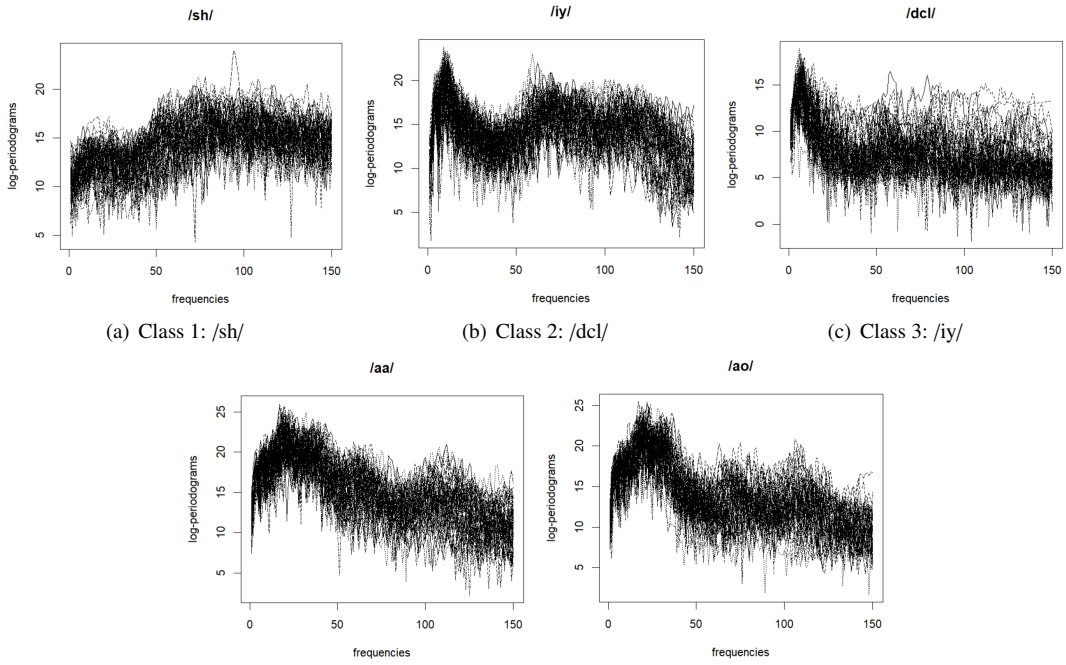
5.2. Phoneme data

Phoneme data consists of 500 samples with five different classes, containing 100 samples in each class. The response variable represents five phonemes: /sh/, /dcl/, /iy/, /aa/, and /ao/. The predictor



(a) Cross-validated CE (b) Boxplot of $\hat{p}(\mathbf{x})$

Figure 3: *Tecator data: (a) depicts the cross-validated CE for different values of* λ *which is minimized at* $\log \lambda = -9.21$. *(b) compares the boxplots of cross-validated probability estimates for the two classes. CE = cross entropy.*



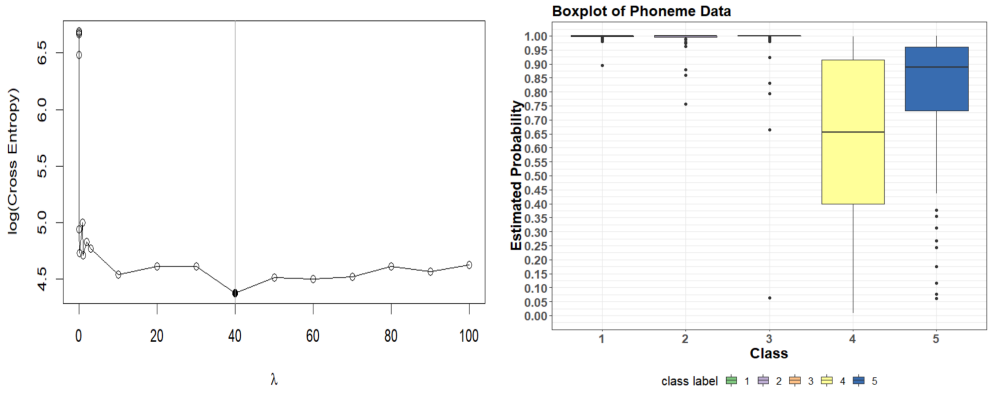
(a) Class 1: /sh/ (b) Class 2: /dcl/ (c) Class 3: /iy/

(d) Class 4: /aa/ (e) Class 5: /ao/

Figure 4: *Phoneme data: illustrations of functional predictors obtained by employing B-spline. Class 4 and 5 look difficult to distinguish.*

is functional and composed of values of log-periodogram observed at each 150 discretized frequency points. Figure 4 depicts the functional predictors in each classes of the Phoneme data estimated from the B-spline system with 10 equally spaced knots.

We obtain an optimal $\lambda = 40$ via grid search as done in Section 4.1. Figure 5(a) depicts the cross-



(a) Cross-validated CE (b) Boxplots of $\hat{p}_k(\mathbf{x}), k = 1, \ldots, 5$

Figure 5: *Phoneme data: sub-panel (a) depicts the cross-validated CE. (b) compares the cross-validated class probability estimates of* $P(Y_{i_k} = k \mid \mathbf{X} = \mathbf{x}_{i_k}), i_k \in \{i : Y_i = k\}$ *for the observations in the* k^{th} *class,* $k = 1, \ldots, 5$*. CE = cross entropy.*

validated CE. Figure 5(b) compares the cross-validated class probability estimates of $P(Y_{i_k} = k \mid \mathbf{X} = \mathbf{x}_{i_k}), i_k \in \{i : Y_i = k\}$ for the observations in the k^{th} class, $k = 1, \ldots, 5$. Although it seems relatively difficult to classify class 4 and 5 as expected from Figure 4(d) and (e), the proposed method performs well for Phoneme data with five-class classification problem.

6. Conclusion

In this article, we propose a model-free approach to estimate class probability when the predictor is functional, by extending the idea of Wang *et al.* (2007) to the functional context. The proposed model is model-free but also computationally efficient by employing the piecewise linearity of the FWSVM solutions. Numerical illustrations shows that the proposed method is advantageous to FLR which relies on a model assumption.

Acknowledgements

This work was supported by a National Research Foundation of Korea (NRF) grant funded by the Korea government (MIST), Grant number 2018R1D1A1B07043034 and No.2019R1A4A1028134.

References

James GM (2002). Generalized linear models with functional predictors, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **64**, 411–432.

Kimeldorf G and Wahba G (1971). Some results on Tchebycheffian spline functions, *Journal of Mathematical Analysis and Applications*, **33**, 82–95.

Park C, Koo JY, Kim S, Sohn I, and Lee JW (2008). Classification of gene functions using support vector machine for time-course gene expression data, *Computational Statistics & Data Analysis*, **52**, 2578–2587.

Rossi F and Villa N (2006). Support vector machine for functional data classification, *Neurocomputing*, **69**, 730–742.

Shin SJ, Wu Y, and Zhang HH (2014). Two-dimensional solution surface for weighted support vector machines, *Journal of Computational and Graphical Statistics*, **23**, 383–402.

Vapnik VN (1995). *The Nature of Statistical Learning Theory*, Springer-Verlag, Berlin, Heidelberg.

Wahba G (1990). *Spline Models for Observational Data*, Society for Industrial and Applied Mathematics, Philadelphia.

Wahba G (2002). Soft and hard classification by reproducing kernel Hilbert space methods, *Proceedings of the National Academy of Sciences*, **99**, 16524–16530.

Wang J, Shen X, and Liu Y (2007). Probability estimation for large-margin classifiers, *Biometrika*, **95**, 149–167.

Wu TF, Lin CJ, and Weng RC (2004). Probability estimates for multi-class classification by pairwise coupling, *Journal of Machine Learning Research*, **5**, 975–1005.

Received August 26, 2019; Revised October 6, 2019; Accepted October 7, 2019