L_2 Kernel Classification

JooSeuk Kim, *Student Member*, *IEEE*, and Clayton D. Scott, *Member*, *IEEE*

Abstract—Nonparametric kernel methods are widely used and proven to be successful in many statistical learning problems. Well-known examples include the kernel density estimate (KDE) for density estimation and the support vector machine (SVM) for classification. We propose a kernel classifier that optimizes the L_2 or integrated squared error (ISE) of a "difference of densities." We focus on the Gaussian kernel, although the method applies to other kernels suitable for density estimation. Like a support vector machine (SVM), the classifier is sparse and results from solving a quadratic program. We provide statistical performance guarantees for the proposed L_2 kernel classifier in the form of a finite sample oracle inequality and strong consistency in the sense of both ISE and probability of error. A special case of our analysis applies to a previously introduced ISE-based method for kernel density estimation. For dimensionality greater than 15, the basic L_2 kernel classifier performs poorly in practice. Thus, we extend the method through the introduction of a natural regularization parameter, which allows it to remain competitive with the SVM in high dimensions. Simulation results for both synthetic and real-world data are presented.

Index Terms—Kernel methods, sparse classifiers, integrated squared error, difference of densities, SMO algorithm.

1 Introduction

In the binary classification problem, we are given realizations $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ of a jointly distributed pair $(\mathbf{X}, Y)$, where $\mathbf{X} \in \mathbb{R}^d$ is a pattern and $Y \in \{-1, +1\}$ is a class label. The goal of classification is to build a classifier, i.e., a function taking $\mathbf{X}$ as input and outputting a label such that some measure of performance is optimized. Kernel classifiers [1] are an important family of classifiers that have drawn much recent attention for their ability to represent nonlinear decision boundaries and to scale well with increasing dimension d. A kernel classifier (without offset) has the form

$$g(\mathbf{x}) = \operatorname{sign}\left\{\sum_{i=1}^{n} \alpha_i Y_i k(\mathbf{x}, \mathbf{X}_i)\right\},$$

where α_i are parameters and k is a kernel function. For example, support vector machines (SVMs) without offset have this form [2], as does the standard kernel density estimate (KDE) plug-in rule.

In this paper, we employ an L_2 or integrated squared error (ISE) criterion to design the coefficients α_i of a kernel classifier. Like the SVM, L_2 kernel classifiers are the solutions of convex quadratic programs (QPs) that can be solved efficiently using standard decomposition algorithms. In addition, the classifiers are sparse, meaning most of the coefficients $\alpha_i = 0$, which has advantages for representation and evaluation efficiency. The L_2 objective function also has appealing geometric interpretations in that it estimates a hyperplane in kernel feature space. Unlike the SVM, the most basic version of our method has no free parameters to be set by the user, except perhaps the kernel bandwidth parameter. However, this basic L_2 kernel classifier is not competitive with the SVM for problems of dimensionality exceeding 15-20. Thus, we also extend the method to incorporate a regularization parameter, which allows it to remain competitive with the SVM in high dimensions.

We provide statistical performance guarantees for the proposed L_2 kernel classifier. The linchpin of our analysis is a new concentration inequality bounding the deviation of a cross-validation-based ISE estimate from the true ISE. This bound is then applied to prove an oracle inequality and consistency in both ISE and probability of error. In addition, as a special case of our analysis, we are able to deduce performance guarantees for the method of L_2 kernel density estimation described in [3], [4], which has not previously been analyzed.

1.1 Related Work

The ISE criterion has a long history in the literature on bandwidth selection for kernel density estimation [5] and more recently in parametric estimation [6]. The use of ISE for optimizing the weights of a KDE via quadratic programming was first described in [3] and later rediscovered in [4]. In [7], an ℓ_1 penalized ISE criterion was used to aggregate a finite number of predetermined densities. Linear and convex aggregation of densities, based on an L_2 criterion, are studied in [8], where the densities are based on a finite dictionary or an independent sample. In contrast, our proposed method allows data-adaptive kernels, and does not require an independent (holdout) sample.

In classification, some connections relating SVMs and ISE are made in [9], although no new algorithms are proposed. The application of the ISE-based kernel method to classification problem is first studied in [10], where each class-conditional density is estimated separately and plugged into the final classifier. However, our ISE criterion is a more natural choice for classification in that we directly estimate the difference of densities (DOD). It also leads to interesting geometric interpretations and relationships between our method and SVMs.

- *The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122. E-mail: {stannum, clayscot}@umich.edu.*

Manuscript received 20 Nov. 2008; revised 11 May 2009; accepted 20 Aug. 2009; published online 18 Nov. 2009.
Recommended for acceptance by D. Schuurmans.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-11-0801.
Digital Object Identifier no. 10.1109/TPAMI.2009.188.

The "difference of densities" perspective has been applied to classification in other settings by several authors. In [11] and [12], a difference of densities is used to find smoothing parameters or kernel bandwidths. In [13], conditional densities are chosen among a parameterized set of densities to maximize the average (bounded) density differences. The relationship between the consistency of ISE and the consistency of the probability of error is studied in [14]. Finally, Pelckmans et al. [15] consider a kernel classifier that maximizes the average (as opposed to worst case) empirical margin. The resulting classifier amounts to an estimate of the difference of densities having uniform α_is.

1.2 Organization

Section 2 introduces our L_2 criterion for classification and formulates the criterion as a quadratic program. Statistical performance guarantees are presented in Section 3. Geometric interpretations for the proposed method are provided in Section 4. Extension and variations of the basic method are presented in Section 5, including one extension that makes the method competitive in higher dimensions at the expense of an extra regularization parameter. We demonstrate experimental results in Section 6. Conclusions are offered in the final section. The appendices contain proofs of theorems, and an efficient Sequential Minimal Optimization (SMO) algorithm for implementing the classifier and can be found in the IEEE Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2009.188. Matlab software implementing our algorithm, including fast C++/Mex code for the core computations, is available at http://www-personal.umich.edu/~stannum/l2kernel.zip. Preliminary versions of this work appeared in [16], [17].

2 L_2 Kernel Classification

Let $f_+(\mathbf{x})$ and $f_-(\mathbf{x})$ denote the class-conditional densities of the pattern given the label. From decision theory, the optimal classifier has the form

$$g^*(\mathbf{x}) = \mathrm{sign}\{f_+(\mathbf{x}) - \gamma f_-(\mathbf{x})\}. \tag{1}$$

Denote the "DOD" by $d_\gamma(\mathbf{x}) := f_+(\mathbf{x}) - \gamma f_-(\mathbf{x})$.

Here, we view γ as a fixed parameter to be set by the user to reflect prior class probabilities and class-conditional error costs. For example, if we are interested in minimizing the probability of error, γ should be set to $\gamma^* = \frac{1-p}{p}$, where $0 < p < 1$ is the prior probabilities of the positive class. If p is unknown, we may set γ to be the natural empirical estimate for γ^*. We analyze this exact strategy in Section 3, and also employ it in our experiments in Section 6.

Recall that we are given realizations $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$, where $\mathbf{X}_i \in \mathbb{R}^d$ is a pattern and $Y_i \in \{-1, +1\}$ is a class label. For convenience, we relabel Y so that it belongs to $\{1, -\gamma\}$ and denote $I_+ = \{i \mid Y_i = +1\}$ and $I_- = \{i \mid Y_i = -\gamma\}$. The class-conditional densities are modeled as KDEs with *variable weights* $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$:

$$\widehat{f}_+(\mathbf{x};\boldsymbol{\alpha}) = \sum_{i\in I_+} \alpha_i k_\sigma(\mathbf{x}, \mathbf{X}_i),$$

$$\widehat{f}_-(\mathbf{x};\boldsymbol{\alpha}) = \sum_{i\in I_-} \alpha_i k_\sigma(\mathbf{x}, \mathbf{X}_i),$$

with constraints $\boldsymbol{\alpha} \in A$ where

$$A = \left\{ \boldsymbol{\alpha} \Big| \sum_{i\in I_+} \alpha_i = \sum_{i\in I_-} \alpha_i = 1, \quad \alpha_i \geq 0 \quad \forall i \right\}$$

and

$$k_\sigma(\mathbf{x}, \mathbf{X}_i) = \left(2\pi\sigma^2\right)^{-d/2} \exp\left\{ -\frac{\|\mathbf{x} - \mathbf{X}_i\|^2}{2\sigma^2} \right\}$$

is the Gaussian kernel with bandwidth $\sigma > 0$. In general, σ is a tuning parameter that will need to set using standard model selection strategies, such as cross validation. As explained in Section 5, other kernels besides the Gaussian also fit naturally into our framework.

We take as our goal estimating $d_\gamma(\mathbf{x})$ directly with $\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) := \widehat{f}_+(\mathbf{x};\boldsymbol{\alpha}) - \gamma \widehat{f}_-(\mathbf{x};\boldsymbol{\alpha})$, rather than estimating $f_+(\mathbf{x})$ and $f_-(\mathbf{x})$ separately and "plug-in" to (1) as in [10]. In particular, we propose estimating $\boldsymbol{\alpha}$ by minimizing the L_2 distance or ISE between the model $\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})$ and the truth $d_\gamma(\mathbf{x})$. The ISE associated with $\boldsymbol{\alpha}$ is

$$\begin{aligned} ISE(\boldsymbol{\alpha}) &= \|\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - d_\gamma(\mathbf{x})\|^2_{L_2} \\ &= \int \left(\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - d_\gamma(\mathbf{x})\right)^2 d\mathbf{x} \\ &= \int \widehat{d}^2_\gamma(\mathbf{x};\boldsymbol{\alpha}) d\mathbf{x} - 2\int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) d_\gamma(\mathbf{x})\, d\mathbf{x} \\ &\quad + \int d^2_\gamma(\mathbf{x})\, d\mathbf{x}. \end{aligned}$$

Since we do not know the true $d_\gamma(\mathbf{x})$, we need to estimate the second term in the above equation

$$H(\boldsymbol{\alpha}) \triangleq \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) d_\gamma(\mathbf{x})\, d\mathbf{x}, \tag{2}$$

by $H_n(\boldsymbol{\alpha})$ which will be explained in detail in Section 2.1. Then, the empirical ISE becomes

$$\widehat{ISE}(\boldsymbol{\alpha}) = \int \widehat{d}^2_\gamma(\mathbf{x};\boldsymbol{\alpha})\, d\mathbf{x} - 2H_n(\boldsymbol{\alpha}) + \int d^2_\gamma(\mathbf{x})\, d\mathbf{x}. \tag{3}$$

Now, $\widehat{\boldsymbol{\alpha}}$ is defined as

$$\widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}\in A} \widehat{ISE}(\boldsymbol{\alpha}), \tag{4}$$

and the final classifier will be

$$g(\mathbf{x}) = \begin{cases} +1, & \widehat{d}_\gamma(\mathbf{x};\widehat{\boldsymbol{\alpha}}) \geq 0, \\ -\gamma, & \widehat{d}_\gamma(\mathbf{x};\widehat{\boldsymbol{\alpha}}) < 0. \end{cases} \tag{5}$$

2.1 Estimation of $H(\boldsymbol{\alpha})$

In this section, we propose a method of estimating $H(\boldsymbol{\alpha})$ in (2). The basic idea is to view $H(\boldsymbol{\alpha})$ as an expectation and estimate it using a sample average. In [16], the resubstitution estimator for $H(\boldsymbol{\alpha})$ was used. However, since this estimator is biased, we use a leave-one-out cross-validation (LOOCV) estimator which is unbiased and facilitates our theoretical analysis. Note that the DOD can be expressed as

$$\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) = \widehat{f}_+(\mathbf{x}) - \gamma \widehat{f}_-(\mathbf{x}) = \sum_{i=1}^n \alpha_i Y_i k_\sigma(\mathbf{x}, \mathbf{X}_i).$$

Then,

$$\begin{aligned}H(\boldsymbol{\alpha}) &= \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) d_\gamma(\mathbf{x}) d\mathbf{x}\\ &= \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) f_+(\mathbf{x}) d\mathbf{x} - \gamma \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) f_-(\mathbf{x})\ d\mathbf{x}\\ &= \int \sum_{i=1}^n \alpha_i Y_i k_\sigma(\mathbf{x},\mathbf{X}_i)\ f_+(\mathbf{x}) d\mathbf{x}\\ &\quad - \gamma \int \sum_{i=1}^n \alpha_i Y_i k_\sigma(\mathbf{x},\mathbf{X}_i) f_-(\mathbf{x}) d\mathbf{x}\\ &= \sum_{i=1}^n \alpha_i Y_i h(\mathbf{X}_i),\end{aligned}$$

where

$$\begin{aligned}h(\mathbf{X}_i) \triangleq &\int k_\sigma(\mathbf{x},\mathbf{X}_i)\ f_+(\mathbf{x})\ d\mathbf{x}\\ &- \gamma \int k_\sigma(\mathbf{x},\mathbf{X}_i)\ f_-(\mathbf{x})\ d\mathbf{x}.\end{aligned} \tag{6}$$

We estimate each $h(\mathbf{X}_i)$ in (6) for $i = 1, \dots, n$ using leave-one-out cross validation

$$\widehat{h}_i \triangleq \begin{cases} \dfrac{1}{N_+ - 1} \displaystyle\sum_{j \in I_+, j \neq i} k_\sigma(\mathbf{X}_j, \mathbf{X}_i) \\ \quad - \dfrac{\gamma}{N_-} \displaystyle\sum_{j \in I_-} k_\sigma(\mathbf{X}_j, \mathbf{X}_i), \quad i \in I_+, \\ \dfrac{1}{N_+} \displaystyle\sum_{j \in I_+} k_\sigma(\mathbf{X}_j, \mathbf{X}_i) \\ \quad - \dfrac{\gamma}{N_- - 1} \displaystyle\sum_{j \in I_-, j \neq i} k_\sigma(\mathbf{X}_j, \mathbf{X}_i), \quad i \in I_-, \end{cases}$$

where $N_+ = |I_+|$, $N_- = |I_-|$. Then, the estimate of $H(\boldsymbol{\alpha})$ is $H_n(\boldsymbol{\alpha}) = \sum_{i=1}^n \alpha_i Y_i \widehat{h}_i$. We emphasize that here cross validation is employed as a method of estimation and is distinct from any procedure that may be used for tuning the bandwidth σ.

2.2 Optimization

The optimization problem (4) can be formulated as a quadratic program. The first term in (3) is

$$\begin{aligned}\int \widehat{d}_\gamma^2(\mathbf{x};\boldsymbol{\alpha})\ d\mathbf{x} &= \int \left(\sum_{i=1}^n \alpha_i Y_i k_\sigma(\mathbf{x},\mathbf{X}_i) \right)^2 d\mathbf{x}\\ &= \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j Y_i Y_j \int k_\sigma(\mathbf{x},\mathbf{X}_i) k_\sigma(\mathbf{x},\mathbf{X}_j)\ d\mathbf{x}\\ &= \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j Y_i Y_j k_{\sqrt{2}\sigma}(\mathbf{X}_i,\mathbf{X}_j),\end{aligned}$$

by the convolution theorem for Gaussian kernels [18]. As we have seen in Section 2.1, the second term $H_n(\boldsymbol{\alpha})$ in (3) is linear in $\boldsymbol{\alpha}$ and can be expressed as $\sum_{i=1}^n \alpha_i c_i$, where $c_i = Y_i \widehat{h}_i$. Finally, since the third term does not depend on $\boldsymbol{\alpha}$, the optimization problem (4) becomes the following QP:

$$\widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha} \in A} \quad \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j Y_i Y_j k_{\sqrt{2}\sigma}(\mathbf{X}_i,\mathbf{X}_j) - \sum_{i=1}^n c_i \alpha_i. \tag{7}$$

We refer to the resulting classifier as L2QP (L2 classification via Quadratic Programming). Since the Gaussian kernel is positive definite [1], the objective function in (7) is strictly convex if the $\mathbf{X}_i$s are distinct, and thus it has a unique solution. As discussed in [4], quadratic programs derived from ISE-based criteria induce sparse solutions, and the nonzero α_is tend to be concentrated in regions of space with greater probability mass. Another explanation of this is presented in Section 4. The QP (7) is similar in some respects to the dual QP of the 2-norm SVM with hinge loss [2]. However, unlike the SVM, (7) does not include a regularization parameter, and therefore the computational cost required for training the L2QP classifier will typically be less than that of the SVM. The QP can be solved by a variant of the Sequential Minimal Optimization (SMO) algorithm [19] explained in Appendix A.

3 Statistical Performance Analysis

We give theoretical performance guarantees for our proposed method. We assume that $\{\mathbf{X}_i\}_{i \in I_+}$ and $\{\mathbf{X}_i\}_{i \in I_-}$ are i.i.d. samples from $f_+(\mathbf{x})$ and $f_-(\mathbf{x})$, respectively, and treat N_+ and N_- as deterministic variables n_+ and n_- such that $n_+ \to \infty$ and $n_- \to \infty$ as $n \to \infty$. Proofs are found in Appendices B-D, which can be found in the IEEE Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2009.188.

3.1 Concentration Inequality for $H_n(\boldsymbol{\alpha})$

Lemma 1. *Conditioned on $\mathbf{X}_i$, $\widehat{h}_i$ is an unbiased estimator of $h(\mathbf{X}_i)$, i.e.,*

$$\mathbf{E}\left[\widehat{h}_i \middle| \mathbf{X}_i\right] = h\ (\mathbf{X}_i).$$

Furthermore, for any $\epsilon > 0$

$$\begin{aligned}&\mathbf{P}\left\{ \sup_{\boldsymbol{\alpha} \in A} \left| H_n(\boldsymbol{\alpha}) - H(\boldsymbol{\alpha}) \right| > \epsilon \right\}\\ &\quad \leq\ 2n \left(e^{-c(n_+ - 1)\epsilon^2} + e^{-c(n_- - 1)\epsilon^2} \right),\end{aligned}$$

where $c = 2\left(\sqrt{2\pi}\sigma\right)^{2d} / (1 + \gamma)^4$.

Lemma 1 implies that $H_n(\boldsymbol{\alpha}) \to H(\boldsymbol{\alpha})$ almost surely for all $\boldsymbol{\alpha} \in A$ simultaneously, provided that σ, n_+ and n_- evolve as functions of n such that $n_+ \sigma^{2d} / \ln n \to \infty$ and $n_- \sigma^{2d} / \ln n \to \infty$.

3.2 Oracle Inequality

Next, we establish on oracle inequality, which relates the performance of our estimator to that of the best possible kernel classifier.

Theorem 1. *Let $\epsilon > 0$ and set $\delta = \delta(\epsilon) = 2n(e^{-c(n_+ - 1)\epsilon^2} + e^{-c(n_- - 1)\epsilon^2})$ where $c = 2(\sqrt{2\pi}\sigma)^{2d}/(1+\gamma)^4$. Then, with probability at least $1 - \delta$,*

$$ISE(\widehat{\boldsymbol{\alpha}}) \leq \inf_{\boldsymbol{\alpha} \in A} ISE(\boldsymbol{\alpha}) + 4\epsilon.$$

Proof. From Lemma 1, with probability at least $1 - \delta$,

$$|ISE(\boldsymbol{\alpha}) - \widehat{ISE}(\boldsymbol{\alpha})| \leq 2\epsilon, \quad \forall \boldsymbol{\alpha} \in A,$$

by using the fact $ISE(\boldsymbol{\alpha}) - \widehat{ISE}(\boldsymbol{\alpha}) = 2(H_n(\boldsymbol{\alpha}) - H(\boldsymbol{\alpha}))$. Then, with probability at least $1 - \delta$, for all $\boldsymbol{\alpha} \in A$, we have

$$ISE(\widehat{\boldsymbol{\alpha}}) \le \widehat{ISE}(\widehat{\boldsymbol{\alpha}}) + 2\epsilon \le \widehat{ISE}(\boldsymbol{\alpha}) + 2\epsilon \le ISE(\boldsymbol{\alpha}) + 4\epsilon,$$

where the second inequality holds from the definition of $\widehat{\boldsymbol{\alpha}}$. This proves the theorem. □

3.3 ISE Consistency

Next, we have a theorem stating that $ISE(\widehat{\boldsymbol{\alpha}})$ converges to zero in probability.

Theorem 2. *Suppose that for $f = f_+$ and f_-, the Hessian $\mathcal{H}_f(\mathbf{x})$ exists and each entry of $\mathcal{H}_f(\mathbf{x})$ is piecewise continuous and square integrable. If σ, n_+, and n_- evolve as functions of n such that $\sigma \to 0$, $n_+\sigma^{2d}/\ln n \to \infty$, and $n_+\sigma^{2d}/\ln n \to \infty$, then $ISE(\widehat{\boldsymbol{\alpha}}) \to 0$ in probability as $n \to \infty$.*

This result intuitively follows from the oracle inequality since the standard Parzen window density estimate is consistent and uniform weights are among the simplex A. The rigorous proof is presented in Appendix C, which can be found in the IEEE Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2009.188.

3.4 Bayes Error Consistency

In classification, we are ultimately interested in minimizing the probability of error. The consistency with respect to the probability of error could be easily shown if we set γ to $\gamma^* = \frac{1-p}{p}$ and apply Theorem 3 in [14], where $0 < p < 1$ is the prior probability of the positive class. However, since p is unknown, we must estimate γ^*. Let us now assume $\{\mathbf{X}_i\}_{i=1}^n$ is an i.i.d. sample from $f(\mathbf{x}) = pf_+(\mathbf{x}) + (1-p)f_-(\mathbf{x})$. Then, N_+ and N_- are binomial random variables, and we may estimate γ^* as $\gamma = \frac{N_-}{N_+}$. The next theorem says the L_2 kernel classifier is consistent with respect to the probability of error.

Theorem 3. *Suppose that the assumptions in Theorem 2 are satisfied. In addition, suppose that $f_- \in L_2(\mathbb{R})$, i.e., $\|f_-\|_2 < \infty$. Let $\gamma = N_-/N_+$ be an estimate of $\gamma^* = \frac{1-p}{p}$. If σ evolves as a function of n such that $\sigma \to 0$ and $n\sigma^{2d}/\ln n \to \infty$ as $n \to \infty$, then the L_2 kernel classifier is consistent. In other words, given training data $\mathbf{D}_n = ((\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n))$, the classification error*

$$L_n = \mathbf{P}\{\mathrm{sign}\{\widehat{d}_\gamma(\mathbf{X}; \widehat{\boldsymbol{\alpha}})\} \ne Y \mid \mathbf{D}_n\}$$

converges to the Bayes error L^* in probability as $n \to \infty$.

The proof is given in Appendix D, which can be found in the IEEE Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2009.188.

3.5 Application to Density Estimation

By setting $\gamma = 0$, our goal becomes estimating f_+ and we recover the L_2 kernel density estimate of [3], [4] using leave-one-out cross validation. Given an i.i.d. sample $\mathbf{X}_1, \dots, \mathbf{X}_n$ from $f(\mathbf{x})$, the L_2 kernel density estimate of $f(\mathbf{x})$ is defined as

$$\widehat{f}(\mathbf{x}; \widehat{\boldsymbol{\alpha}}) = \sum_{i=1}^n \widehat{\alpha}_i k_\sigma(\mathbf{x}, \mathbf{X}_i),$$

with $\widehat{\alpha}_i$s optimized such that

$$\widehat{\boldsymbol{\alpha}} = \mathop{\arg\min}_{\substack{\sum \alpha_i = 1\\ \alpha_i \ge 0}} \ \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i\alpha_j k_{\sqrt{2}\sigma}(\mathbf{X}_i, \mathbf{X}_j) - \sum_{i=1}^n \alpha_i \left(\frac{1}{n-1} \sum_{j \ne i} k_\sigma(\mathbf{X}_i, \mathbf{X}_j) \right).$$

Our concentration inequality, oracle inequality, and L_2 consistency result immediately extend to provide the same performance guarantees for this method. In particular, we state the following corollaries:

Corollary 1. *Let $\epsilon > 0$ and set $\delta = \delta(\epsilon) = 2ne^{-c(n-1)\epsilon^2}$, where $c = 2(\sqrt{2\pi}\sigma)^{2d}$. Then, with probability at least $1 - \delta$,*

$$\int \left(\widehat{f}(\mathbf{x}; \widehat{\boldsymbol{\alpha}}) - f(\mathbf{x})\right)^2 d\mathbf{x} \le \inf_{\substack{\sum \alpha_i = 1\\ \alpha_i \ge 0}} \int \left(\widehat{f}(\mathbf{x}; \boldsymbol{\alpha}) - f(\mathbf{x})\right)^2 d\mathbf{x} + 4\epsilon.$$

Corollary 2. *Suppose that the Hessian $\mathcal{H}_f(\mathbf{x})$ of a density function $f(\mathbf{x})$ exists and each entry of $\mathcal{H}_f(\mathbf{x})$ is piecewise continuous and square integrable. If $\sigma \to 0$ and $n\sigma^{2d}/\ln n \to \infty$ as $n \to \infty$, then*

$$\int \left(\widehat{f}(\mathbf{x}; \widehat{\boldsymbol{\alpha}}) - f(\mathbf{x})\right)^2 d\mathbf{x} \to 0$$

in probability.

4 Geometric Interpretations

In this section, we present two geometric interpretations of the L2QP classifier.

4.1 Separating Hyperplane in Kernel Feature Space

The first interpretation views the QP (7) as the dual of a primal problem defined in a kernel feature space. The corresponding primal problem is

$$\begin{aligned} &\min_{\mathbf{w}, \xi_+, \xi_-} \ \frac{1}{2}\|\mathbf{w}\|^2 + \xi_+ + \xi_- \\ &\quad \text{s.t.} \quad Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle \ge c_i - \xi_+, \quad \text{for } i \in I_+, \\ &\qquad\quad\ Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle \ge c_i - \xi_-, \quad \text{for } i \in I_-, \end{aligned} \tag{8}$$

where $\Phi_\sigma(\mathbf{x})$ is the implicit kernel mapping into the feature space associated with the Gaussian kernel Hilbert space [1].

This primal formulation differs from that of the standard 2-norm SVM with hinge loss (and without offset) in the following aspects: First, in the right-hand sides of the constraints, c_is appear instead of 1. This means if c_i is larger (i.e., $\mathbf{X}_i$ is accurately classified by the Parzen window plug-in formula), this modified SVM places more emphasis on correctly classifying $\mathbf{X}_i$. Second, there exist only two slack variables, ξ_+ and ξ_-, one per class, and these are not required to be nonnegative. Finally, after finding the optimal solution $\widehat{\mathbf{w}} = \sum_{i=1}^n \widehat{\alpha}_i Y_i \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)$, the final classifier takes the sign of the inner product between $\widetilde{\mathbf{w}} = \sum_{i=1}^n \widehat{\alpha}_i Y_i \Phi_\sigma(\mathbf{X}_i)$ and $\Phi_\sigma(\mathbf{x})$, not between $\widehat{\mathbf{w}}$ and $\Phi_{\sqrt{2}\sigma}(\mathbf{x})$, i.e.,

$$g(\mathbf{x}) = \mathrm{sign}\{\langle \widetilde{\mathbf{w}}, \Phi_\sigma(\mathbf{x})\rangle\} = \mathrm{sign}\left\{\sum_{i=1}^n \widehat{\alpha}_i Y_i k_\sigma(\mathbf{x}, \mathbf{X}_i)\right\}.$$

The primal offers another explanation of why the points with nonzero α_is are concentrated in regions of space with greater probability mass. First note that since we are minimizing ξ_+ and ξ_-, they satisfy

$$\xi_+ = \max_{i\in I_+}\{c_i - Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle\},$$
$$\xi_- = \max_{i\in I_-}\{c_i - Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle\}.$$

As α_i is the Lagrangian multiplier associated with each constraint, the optimal α_i should satisfy the Karush-Kuhn-Tucker (KKT) conditions, in particular the complimentary slackness condition. Thus, for nonzero $\alpha_i > 0$, the associated constraint should be met with equality, i.e.,

$$\begin{aligned} c_i - Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle &= \xi_+ \\ &= \max_{j\in I_+}\{c_j - Y_j \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_j)\rangle\} \quad \text{for } \alpha_i > 0,\ i \in I_+, \\ c_i - Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle &= \xi_- \\ &= \max_{j\in I_-}\{c_j - Y_j \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_j)\rangle\} \quad \text{for } \alpha_i > 0,\ i \in I_-. \end{aligned}$$

Therefore, we can see that if c_i is larger, $c_i - Y_i \cdot \langle \mathbf{w}, \Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle$ is more likely to be a maximum value and thus the corresponding α_i is nonzero. Since $c_i = Y_i\widehat{h}(\mathbf{X}_i)$, where $\widehat{h}$ is the Parzen window plug-in estimate of d_γ, it tends to be largest in regions of space with high probability mass.

In Section 5, we introduce an extension of the L_2 kernel classification that amounts to augmenting the primal with an additional parameter multiplying the slack variables.

4.2 Weighted Centroids in Kernel Feature Space

Another interpretation can be obtained by expressing the L_2 criterion itself in the kernel feature space, not considering it as a dual problem. Define

$$\mathbf{m}_\sigma^+ = \frac{1}{N_+}\sum_{i\in I_+}\Phi_\sigma(\mathbf{X}_i), \quad \mathbf{m}_\sigma^- = \frac{1}{N_-}\sum_{i\in I_-}\Phi_\sigma(\mathbf{X}_i),$$
$$\mathbf{m}_\sigma^+(\boldsymbol{\alpha}) = \sum_{i\in I_+}\alpha_i\Phi_\sigma(\mathbf{X}_i), \quad \mathbf{m}_\sigma^-(\boldsymbol{\alpha}) = \sum_{i\in I_-}\alpha_i\Phi_\sigma(\mathbf{X}_i).$$

With this notation, by adding the constant term

$$\left(\frac{1}{N_+ - 1} + \frac{\gamma^2}{N_- - 1}\right) k_\sigma(\mathbf{0},\mathbf{0}),$$

the L_2 objective function may be expressed as

$$\begin{aligned} &\frac{1}{2}\left\|\mathbf{m}_{\sqrt{2}\sigma}^+(\boldsymbol{\alpha}) - \gamma\mathbf{m}_{\sqrt{2}\sigma}^-(\boldsymbol{\alpha})\right\|^2 \\ &\quad - \left\langle \mathbf{m}_\sigma^+(\boldsymbol{\alpha}), \frac{N_+}{N_+ - 1}\mathbf{m}_\sigma^+ - \gamma\mathbf{m}_\sigma^- \right\rangle \\ &\quad - \gamma\left\langle \mathbf{m}_\sigma^-(\boldsymbol{\alpha}), \mathbf{m}_\sigma^+ - \gamma\frac{N_-}{N_- - 1}\mathbf{m}_\sigma^- \right\rangle. \end{aligned}$$

Since $\frac{N_+ - 1}{N_+}$ and $\frac{N_- - 1}{N_-}$ approach 1 as n_+ and n_- go to ∞, for large n, (7) is equivalent to

$$\begin{aligned} \widehat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}\in A} \quad &\frac{1}{2}\left\|\mathbf{m}_{\sqrt{2}\sigma}^+(\boldsymbol{\alpha}) - \gamma\mathbf{m}_{\sqrt{2}\sigma}^-(\boldsymbol{\alpha})\right\|^2 \\ &- \langle \mathbf{m}_\sigma^+(\boldsymbol{\alpha}) - \gamma\mathbf{m}_\sigma^-(\boldsymbol{\alpha}), \mathbf{m}_\sigma^+ - \gamma\mathbf{m}_\sigma^- \rangle. \end{aligned}$$

This has an appealing geometric interpretation. The first term, by itself, gives rise to the max-margin hyperplane in feature space in the case of separable data [20], [21]. In particular, because of the constraints $\sum_{i\in I_+}\alpha_i = \sum_{i\in I_-}\alpha_i = 1$ and $\alpha_i \geq 0,\ \forall i$, the first term is minimized when $\mathbf{m}_{\sqrt{2}\sigma}^+(\boldsymbol{\alpha})$ and $\mathbf{m}_{\sqrt{2}\sigma}^-(\boldsymbol{\alpha})$ are on the boundaries of their respective convex hulls, giving rise to the maximum margin separating hyperplane. The second term tries to align $\mathbf{m}_\sigma^+(\boldsymbol{\alpha}) - \gamma\mathbf{m}_\sigma^-(\boldsymbol{\alpha})$ with $\mathbf{m}_\sigma^+ - \gamma\mathbf{m}_\sigma^-$, which is the normal vector defining the nearest centroid classifier. Interestingly, with $\gamma = 1$, the nearest centroid classifier in feature space is identical to the Parzen window plug-in classifier [1] up to an offset term. Thus, we may say that the second term regularizes the SVM (an alternative to the SVM's soft-margin-based regularization), or the first term sparsifies the Parzen window. Note, however, that the first and second terms involve different kernel bandwidths so that the two terms correspond to different Hilbert spaces.

5 Variations and Extensions

5.1 Weighted L_2 Distance in Fourier Domain

One variation of the L2QP classifier is obtained by minimizing the weighted L_2 distance in the Fourier domain. For density estimation, weighted ISE applied to characteristic functions was previously considered in a parametric setting in [22], [23]. We denote the Fourier transforms of $\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})$ and $d_\gamma(\mathbf{x})$ by $\widehat{D}_\gamma(\omega;\boldsymbol{\alpha})$ and $D_\gamma(\omega)$, respectively, each of which is a difference of characteristic functions. Define the weighted L_2 distance associated with $\boldsymbol{\alpha}$

$$ISE_\lambda(\boldsymbol{\alpha}) := \int\left|\widehat{D}_\gamma(\boldsymbol{\omega};\boldsymbol{\alpha}) - D_\gamma(\boldsymbol{\omega})\right|^2 e^{-\lambda^2\omega^2}d\boldsymbol{\omega},$$

where $\lambda \geq 0$ is a fixed parameter. The effect of the weighting term $e^{-\lambda^2\omega^2}$ and the choice of λ will be discussed below. We may write

$$\begin{aligned} &ISE_\lambda(\boldsymbol{\alpha}) \\ &= \int\left|\widehat{D}_\gamma(\boldsymbol{\omega};\boldsymbol{\alpha})e^{-\lambda^2\omega^2/2} - D_\gamma(\boldsymbol{\omega})e^{-\lambda^2\omega^2/2}\right|^2 d\boldsymbol{\omega} \\ &= 2\pi\int\left(\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) * k_\lambda(\mathbf{x},0) - d_\gamma(\mathbf{x}) * k_\lambda(\mathbf{x},0)\right)^2 d\mathbf{x} \\ &= 2\pi\int\Bigg(\sum_{i=1}^n \alpha_i Y_i k_{\sqrt{\sigma^2+\lambda^2}}(\mathbf{x},\mathbf{X}_i) \\ &\quad - \int d_\gamma(\mathbf{x}')k_\lambda(\mathbf{x},\mathbf{x}')d\mathbf{x}'\Bigg)^2 d\mathbf{x}, \end{aligned} \tag{9}$$

where the second equality holds by Parseval's theorem and $*$ denotes convolution.

After expanding the square in (9), the first term becomes

$$\begin{aligned} &\int\left(\sum_{i=1}^n \alpha_i Y_i k_{\sqrt{\sigma^2+\lambda^2}}(\mathbf{x},\mathbf{X}_i)\right)^2 d\mathbf{x} \\ &= \sum_{i=1}^n\sum_{j=1}^n \alpha_i\alpha_j Y_i Y_j k_\rho(\mathbf{X}_i,\mathbf{X}_j), \end{aligned}$$

where $\rho = \sqrt{2\sigma^2 + 2\lambda^2}$ by the convolution theorem for Gaussian kernels [18]. The second term can be written

$$\begin{aligned}
&\int \left(\sum_{i=1}^{n} \alpha_i Y_i k_{\sqrt{\sigma^2+\lambda^2}}(\mathbf{x},\mathbf{X}_i)\right) \\
&\qquad \cdot \left(\int d_\gamma(\mathbf{x}') k_\lambda(\mathbf{x},\mathbf{x}')\, d\mathbf{x}'\right) d\mathbf{x} \\
&= \sum_{i=1}^{n} \alpha_i Y_i \\
&\qquad \cdot \int d_\gamma(\mathbf{x}') \left(\int k_{\sqrt{\sigma^2+\lambda^2}}(\mathbf{x},\mathbf{X}_i)\; k_\lambda\;(\mathbf{x},\mathbf{x}')\, d\mathbf{x}\right) d\mathbf{x}' \\
&= \sum_{i=1}^{n} \alpha_i Y_i \int d_\gamma(\mathbf{x}')\; k_{\sqrt{\sigma^2+2\lambda^2}}(\mathbf{x}',\mathbf{X}_i)\; d\mathbf{x}' \approx \sum_{i=1}^{n} \alpha_i \tilde{c}_i,
\end{aligned}$$

where we used leave-one-out cross-validation estimate in the last step and

$$\tilde{c}_i \triangleq \begin{cases}
Y_i\Bigg(\dfrac{1}{N_+ - 1}\displaystyle\sum_{j\in I_+, j\neq i} k_{\sqrt{\sigma^2+2\lambda^2}}(\mathbf{X}_j,\mathbf{X}_i) \\
\qquad - \dfrac{\gamma}{N_-}\displaystyle\sum_{j\in I_-} k_{\sqrt{\sigma^2+2\lambda^2}}(\mathbf{X}_j,\mathbf{X}_i)\Bigg), \; i \in I_+, \\
Y_i\Bigg(\dfrac{1}{N_+}\displaystyle\sum_{j\in I_+} k_{\sqrt{\sigma^2+2\lambda^2}}(\mathbf{X}_j,\mathbf{X}_i) \\
\qquad - \dfrac{\gamma}{N_- - 1}\displaystyle\sum_{j\in I_-, j\neq i} k_{\sqrt{\sigma^2+2\lambda^2}}(\mathbf{X}_j,\mathbf{X}_i)\Bigg), \; i \in I_-.
\end{cases}$$

Therefore, an empirical minimizer of the weighted L^2 distance $ISE_\lambda(\boldsymbol{\alpha})$ is obtained by solving

$$\widehat{\boldsymbol{\alpha}} = \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in A} \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j Y_i Y_j k_\rho(\mathbf{X}_i,\mathbf{X}_j) - \sum_{i=1}^{n} \tilde{c}_i \alpha_i.$$

From the Fourier domain definition of $ISE_\lambda(\boldsymbol{\alpha})$, we may interpret the Gaussian weight function $e^{-\lambda^2\omega^2}$ as a low-pass filter that deemphasizes high-frequency content in the unknown densities. Thus, larger values of λ place more emphasis on the slowly varying features of $d_\gamma(\mathbf{x})$. A similar interpretation results if we consider the effect of λ in the $\mathbf{x}$ domain. In (9), we see that $\boldsymbol{\alpha}$ is chosen to optimize the L^2 distance between an $\boldsymbol{\alpha}$-weighted DOD with kernel bandwidth $\sqrt{\sigma^2+\lambda^2}$ and a uniformly weighted DOD with kernel bandwidth λ. That is, the "target" DOD is increasingly smooth as λ increases.

We refer to this method as L2QP-k, where k determines λ through $\lambda = k\cdot\sigma$. Since L2QP-0 corresponds to the previous L2QP, L2QP-k is a generalization of L2QP method. Our experiments have primarily focused on $\lambda = 0$ and $\lambda = \sigma$, the latter being motivated by the belief that the "target" DOD and final classifier should be accurately represented by the same kernel bandwidth. Our evidence thus far suggests that both of these choices of λ, as well as others much larger, lead to comparable classifiers. We have observed, however, that smaller values of λ tend to yield sparser classifiers.

5.2 L_2 Criterion with Inequality Constraints

Our theoretical analysis carries through if we replace the constraint set $A = \{\boldsymbol{\alpha} : \alpha_i \geq 0,\ \sum_{i\in I_+}\alpha_i = \sum_{i\in I_-}\alpha_i = 1\}$ with the set

$$A' = \left\{\boldsymbol{\alpha} : \alpha_i \geq 0,\ \left(1 - \sum_{i\in I_+}\alpha_i\right) = \gamma\left(1 - \sum_{i\in I_-}\alpha_i\right) \geq 0\right\}.$$

By requiring $\left(1 - \sum_{i\in I_+}\alpha_i\right) = \gamma\left(1 - \sum_{i\in I_-}\alpha_i\right)$, we still enforce that d_γ integrate to the true value of $1-\gamma$. However, by allowing the coefficients in each class to sum to less than one, we allow for the possibility that some positive and negative coefficients might "cancel out" in regions of space where f_+ and f_- overlap. This could potentially lead to even sparser solutions.

5.3 L_2 Criterion without Constraints

Since our goal is classification and not density estimation, it is not necessary that $\widehat{f}_+(\mathbf{x};\boldsymbol{\alpha})$ and $\widehat{f}_-(\mathbf{x};\boldsymbol{\alpha})$ be proper density estimates, and hence the constraints $\boldsymbol{\alpha}\in A$ may be dropped. In this case, the unconstrained quadratic objective function, in the matrix/vector form,

$$\frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} - \widetilde{\mathbf{c}}^T\boldsymbol{\alpha},$$

is minimized by the solution of

$$Q\boldsymbol{\alpha} = \widetilde{\mathbf{c}}, \tag{10}$$

where $\widetilde{\mathbf{c}} = [\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_n]^T$, $Q := (Y_iY_jK_{ij})_{i,j=1}^n$, and $K := (k_\rho(\mathbf{X}_i,\mathbf{X}_j))_{i,j=1}^n$. If K is positive definite and $\gamma \neq 0$, then Q is also positive definite, and thus the objective is strictly convex.

The optimization problem now becomes the problem of solving a linear system of (10). It is similar in that respect to the 2-norm SVM with squared error loss, or least-squares SVM (LS-SVM) [24], but again does not include a regularization parameter. The resulting L2LE-k (L2 classification via Linear system of Equations) classifier is not sparse, again like the LS-SVM. Since Q is positive definite, (10) can be solved efficiently by the conjugate gradient descent (CGD) algorithm [25].

5.4 Other Kernels

Our methodology allows for any kernel $k(\mathbf{x},\mathbf{x}')$ such that $k(\mathbf{x},\mathbf{x}') \geq 0$ and, for any fixed $\mathbf{x}'$, $\int k(\mathbf{x},\mathbf{x}')d\mathbf{x} = 1$, e.g., the multivariate Cauchy kernel,

$$k_\sigma(\mathbf{x},\mathbf{X}_i) = \frac{\Gamma(\frac{1+d}{2})}{\pi^{(d+1)/2}\cdot\sigma^d}\left(1 + \frac{\|\mathbf{x}-\mathbf{X}_i\|^2}{\sigma^2}\right)^{-\frac{1+d}{2}},$$

or the multivariate Laplacian kernel,

$$k_\sigma(\mathbf{x},\mathbf{X}_i) = \frac{1}{(2\sigma)^d}\exp\left\{-\frac{\|\mathbf{x}-\mathbf{X}_i\|_1}{\sigma}\right\}.$$

The L_2 kernel classifier is still the solution of

$$\widehat{\boldsymbol{\alpha}} = \operatorname*{arg\,min}_{\boldsymbol{\alpha}\in A} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_jY_iY_jK_{ij} - \sum_{i=1}^{n} c_i\alpha_i,$$

where

$$K_{ij} = \int k(\mathbf{x},\mathbf{X}_i)k(\mathbf{x},\mathbf{X}_j)d\mathbf{x},$$

and c_i is as before.

We make two important observations regarding this QP. First, from the identity

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j Y_i Y_j K_{ij} = \int\left(\sum_{i=1}^{n}\alpha_i Y_i k(\mathbf{x},\mathbf{X}_i)\right)^2 d\mathbf{x},$$

we see that the matrix $(Y_iY_jK_{ij})_{i,j=1}^n$ is always positive definite, and therefore the QP is strictly convex, provided the $\mathbf{X}_i$ are distinct. Second, it is desirable that K_{ij} be easily computable. For some kernels, like the Gaussian, the integral has a closed form expression. For example, the multivariate Cauchy kernel satisfies [26]

$$k_{2\sigma}(\mathbf{X}_i,\mathbf{X}_j) = \int k_\sigma(\mathbf{x},\mathbf{X}_i)\cdot k_\sigma(\mathbf{x},\mathbf{X}_j)\, d\mathbf{x},$$

and the multivariate Laplacian (product) kernel satisfies

$$\begin{aligned}&\int k_\sigma(\mathbf{x},\mathbf{X}_i)\cdot k_\sigma(\mathbf{x},\mathbf{X}_j)d\mathbf{x}\\ &= \frac{1}{(4\sigma)^d}\prod_{l=1}^{d}\left(1+\frac{|\mathbf{X}_{i,l}-\mathbf{X}_{j,l}|}{\sigma}\right)\exp\left\{-\frac{\|\mathbf{X}_i-\mathbf{X}_j\|_1}{\sigma}\right\}.\end{aligned}$$

For kernels without such a formula, values of the integral may still be precomputed and stored. For radially symmetric kernels, such as an alternative multivariate Laplacian kernel [27], $k_\sigma(\mathbf{x},\mathbf{X}_i) = C\cdot\exp(-\|\mathbf{x}-\mathbf{X}_i\|/\sigma)$, where C is a normalizing constant, this entails a simple one-dimensional table, as K_{ij} will depend only on $\|\mathbf{X}_i-\mathbf{X}_j\|$. We experimented briefly with multivariate Cauchy kernels, but did not see significant differences compared to the Gaussian.

5.5 Regularization for High-Dimensional Data

Our experimental results show that our L2QP thus far discussed perform poorly on most high-dimensional data. Similarly, in [10], where the class-conditional densities are estimated separately based on the L_2 criterion, the authors only consider low-dimensional data (the 20-dimensional German data set was reduced to 7-dimensional). In this section, we offer an explanation for this phenomenon. We also present a variation that significantly improves the performance in high dimensions at the expense of introducing a new regularization parameter that must be tuned.

To understand the impact of dimension on L2QP, it is important to realize that the method involves Gaussian kernels of bandwidth $\sqrt{2}\sigma$ (quadratic term) and σ (linear term). The normalizing constants for these kernels are $(4\pi\sigma^2)^{-d/2}$ and $(2\pi\sigma^2)^{-d/2}$, respectively. The ratio of the second normalizing constant to the first one is $\sqrt{2}^d$. In other words, the ratio exponentially increases as a function of dimension and thus in high-dimensional data the linear term in (7) dominates the quadratic term. In this case, minimizing (7) causes a few data points associated with larger c_is to monopolize the weights and yields a too sparse solution.

To address this problem, we introduce a new parameter $\eta > 0$ that balances the linear term and the quadratic term

$$\min_{\boldsymbol{\alpha}\in A}\quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j Y_iY_j k_{\sqrt{2}\sigma}(\mathbf{X}_i,\mathbf{X}_j) - \frac{1}{\eta}\sum_{i=1}^{n}c_i\alpha_i.$$

The corresponding primal is

$$\begin{aligned}\min_{\mathbf{w},\xi_+,\xi_-}\quad & \frac{1}{2}\|\mathbf{w}\|^2 + \eta(\xi_+ + \xi_-)\\ \text{s.t.}\quad & Y_i\cdot\langle\mathbf{w},\Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle \ge c_i - \xi_+, \quad \text{for } i\in I_+\\ & Y_i\cdot\langle\mathbf{w},\Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle \ge c_i - \xi_-, \quad \text{for } i\in I_-.\end{aligned} \tag{11}$$

Therefore, (8) can be thought as a special case of (11) where the regularization parameter η is set to 1. In the primal point of view, η controls the trade-off between the complexity of the classifier, $\|\mathbf{w}\|^2$ and how much the classifier fits to Parzen window plug-in classifier $c_i - Y_i\cdot\langle\mathbf{w},\Phi_{\sqrt{2}\sigma}(\mathbf{X}_i)\rangle$. This new algorithm may also be viewed as minimizing an estimated of a modified ISE, given by

$$\begin{aligned}ISE^\eta(\boldsymbol{\alpha}) &= \left\|\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - \frac{1}{\eta}d_\gamma(\mathbf{x})\right\|_{L_2}^2\\ &= \int\left(\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - \frac{1}{\eta}d_\gamma(\mathbf{x})\right)^2 d\mathbf{x}.\end{aligned}$$

This new method may also be combined with the Fourier domain extension discussed previously, and we refer to resulting classifier as L2QP$_\eta$-k.

6 Experiments

We implement our methods[1] (L2QP-0, L2QP-1, L2QP$_\eta - 0$, L2QP$_\eta - 1$) based on LIBSVM [28] by modifying an SMO subroutine (see Appendix A, which can be found in the IEEE Computer Society Digital Library at http://doi.ieeecom-putersociety.org/10.1109/TPAMI.2009.188). For comparison, we also experiment with the 2-norm SVM with hinge loss (S-SVM, S for "soft margin"), the 2-norm SVM with hinge loss and $C \to \infty$ (H-SVM, H for "hard margin"), and a plug-in classifier based on Parzen window density estimates (Parzen).

To illustrate some of the basic properties of L_2 kernel classifiers, we first experiment with one-dimensional data. Both classes are equally likely and

$$\begin{aligned}f_+(\mathbf{x}) &= 0.2\phi(\mathbf{x};4,\sqrt{2}) + 0.8\phi(\mathbf{x};8,1),\\ f_-(\mathbf{x}) &= 0.7\phi(\mathbf{x};0,1) + 0.3\phi(\mathbf{x};10,\sqrt{2}),\end{aligned}$$

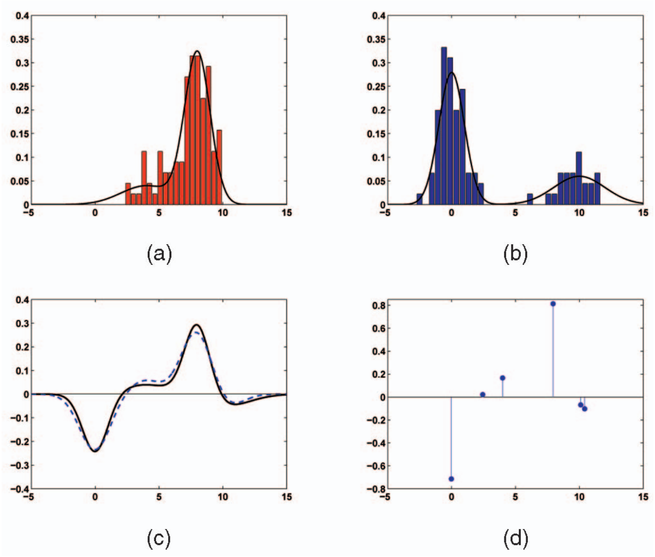
where $\phi(\mathbf{x};\mu,\sigma)$ is a univariate Gaussian pdf with mean μ and variance σ^2. We build a L2QP-0 classifier from 200 training samples. To find a classifier with the smallest probability error, we set $\gamma = N_-/N_+$ and use fivefold cross validation to estimate the bandwidth σ from a logarithmically spaced grid of 50 points from 10^{-2} to 10^1.

The results are shown in Fig. 1. The estimate $\widehat{d}_\gamma(\mathbf{x};\widehat{\boldsymbol{\alpha}})$ is fairly close to the true $d_\gamma(\mathbf{x})$. For $\widehat{\alpha}_i > 0$, $\widehat{\alpha}_iY_i$ are shown at the corresponding $\mathbf{X}_i$ in Fig. 1d and the number of nonzero weights is 9.

Next, we demonstrate our algorithms on 18 artificial and real-world benchmark data sets, available online[2] [28], [29].

[1] The code is available at http://www-personal.umich.edu/~stannum/l2kernel.zip.

[2] http://ida.first.fhg.de/projects/bench/ for the first 13 data sets and http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/ for the last five data sets.



(a) (b)

(c) (d)

Fig. 1. (a) $f_+(\mathbf{x})$ and histogram of its samples. (b) $f_-(\mathbf{x})$ and histogram of its samples. (c) $d_\gamma(\mathbf{x})$ (solid line) and $\widehat{d}_\gamma(\mathbf{x};\widehat{\alpha})$ (dashed line). (d) Sparsity of the proposed method.

TABLE 1
General Information About Benchmark Data Sets

Dataset	# of training data	# of test data	input dimension
Banana	400	4900	2
B. Cancer	200	77	9
Diabetes	468	300	8
F. Solar	666	400	9
German	700	300	20
Heart	170	100	13
Image	1300	1010	18
Ringnorm	400	7000	20
Splice	1000	2175	60
Thyroid	140	75	5
Titanic	150	2051	3
Twonorm	400	7000	20
Waveform	400	4600	21
Adult	3000	3000	123
Ionosphere	251	100	34
mushrooms	4124	4000	112
Sonar	108	100	60
Web	3000	3000	300

There are 100 randomly permuted partitions of each data set into training and test sets (20 for Image, Splice, Adult, Mushrooms, and Web). The dimension and sample sizes[3] of each data set are summarized in Table 1. We set $\gamma = N_-/N_+$ to minimize the probability of error. The parameters to be tuned are σ (all methods), C (S-SVM), and η (L2QP$_\eta$-k, $k = 0, 1$). The following grids were used: For L2QP-0, L2QP-1, and Parzen, we search a logarithmically spaced grid of 50 points from 10^{-2} to 10^1 for σ. For the SVMs, we search the grid $2^{-2}, 2^{-1}, \ldots, 2^7$ for σ and for S-SVM we searched $2^{-5}, 2^{-3}, \ldots, 2^{15}$ for C. For L2QP$_\eta$-0 and L2QP$_\eta$-1, we searched a logarithmically spaced grid of 11 points from 10^{-2} to 10^1 for σ, and a logarithmically spaced grid of 10 points from 1 to $\sqrt{2}^d$ for η. The grids were chosen to ensure that the two-parameter methods searched grids of the same size. The parameters were taken to be the same for all partitions. Each parameter was determined by taking the median estimate based on the first five training sets. On each of these training sets, we use five-fold cross validation to determine the best parameters.

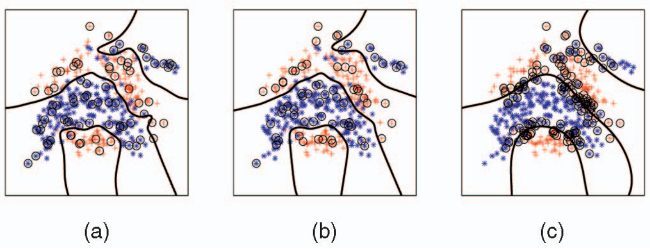
For the "banana" data set, we plot the decision boundary of the L2QP-0, L2QP-1, and S-SVM in Fig. 2 along with training samples. The number of training samples is 400 and the first partition of the data set is used. The numbers of nonzero weights of each method are 77, 66, and 142, respectively. The decision boundaries of L2QP-0 and L2QP-1 slightly differ in that L2QP-1 shows smoother boundary than L2QP-0.

The results for all the data sets are presented in Tables 2, 3, and 4. They show the average probability of error, the average percentage of nonzero coefficients (reflecting the sparsity), and training time over all permutations, respectively. Time indicates the total time required to build a classifier, including the cross-validation search for free parameters.

From these results, we can see that the L2QP-0 and L2QP-1 methods show comparable performance to SVMs except on some high-dimensional data sets, e.g., German, Image, Splice, Waveform, Adult, and Ionosphere. For low-dimensional data sets, the default value $\eta = 1$ works well, but for dimensionality exceeding 15, this default method tends to be too sparse, as explained in Section 5.5. Significantly improved performance on high-dimensional data results from optimizing η. The L2QP$_\eta$-0 and L2QP$_\eta$-1 are comparable to SVMs for almost all data sets; their prediction accuracy is 2-3 percent worse on average. The primary exception is the Splice data. A likely explanation for this is that the data set consists of only categorical features, and thus density-based methods may not be suitable.

Training time shows L2QP-0 and L2QP-1 are significantly faster than the SVM, a reflection of not having to search for an additional regularization parameter. Regarding sparsity, the L_2 methods are often much sparser than the SVMs. One noticeable exception is the Ringnorm data. We have discovered that allowing the two classes to have separate bandwidths (which easily fits within our framework) leads to greatly improved performance here, as well as sparse L_2 classifiers. To maintain a uniform presentation, however, we do not present detailed results for this extension.

Finally, we remark that the hard margin SVM was considered as alternative method having only one tuning parameter, like L2QP-0 and L2QP-1. In reality, however, we were only able to implement H-SVM by taking C very large in the S-SVM. Since the problem is not feasible for C too large, depending on σ, it was actually necessary to search for C after all (not reflected in reported runtimes). In



(a) (b) (c)

Fig. 2. Decision boundary along with positive samples (+) and negative samples (*) for banana data set. Points whose corresponding α_i are nonzero are enclosed by ○. (a) L2QP-0, (b) L2QP-1, (c) S-SVM.

[3] The Adult and Web data sets were subsampled owing to their large size.

TABLE 2
Probability of Error

	H-SVM	S-SVM	L2QP-0	L2QP-1	L2QP$_\eta$-0	L2QP$_\eta$-1	Parzen
Banana	10.9±0.5	10.8±0.5	11.1±0.6	*10.7±0.4*	10.7±0.5	**10.6±0.4**	11.3±0.6
B.Cancer	27.4±4.8	26.9±4.6	26.5±4.6	26.4±4.4	27.6±4.7	*25.2±4.2*	**24.7±4.2**
Diabetes	*23.8±1.9*	**23.2±1.8**	26.5±2.4	26.8±2.4	26.2±2.3	26.6±1.9	26.0±2.1
F.Solar	37.8±4.6	**32.3±1.8**	35.7±3.4	35.5±1.8	37.3±1.8	*34.0±2.0*	36.2±1.9
German	*24.2±2.2*	**24.2±2.1**	29.4±2.1	28.4±2.7	26.1±2.8	25.5±2.6	25.3±2.5
Heart	19.4±4.0	**15.6±3.4**	17.5±4.2	16.8±3.6	17.4±4.0	*16.7±3.8*	18.0±3.5
Image	*3.3±0.7*	**3.0±0.7**	28.7±7.8	9.3±4.5	3.5±0.6	3.7±0.5	3.4±0.5
Ringnorm	**1.6±0.1**	*2.0±0.2*	2.4±0.2	2.4±0.2	2.4±0.2	2.4±0.2	2.4±0.2
Splice	**10.8±0.7**	*11.3±0.6*	38.9±4.3	36.0±5.2	19.2±1.5	29.1±10.1	26.0±1.9
Thyroid	5.3±2.3	5.0±2.2	5.2±2.2	4.9±2.3	4.6±2.4	*4.3±2.5*	**4.2±2.1**
Titanic	*22.4±1.0*	22.8±0.7	23.0±0.4	22.9±0.6	23.2±1.5	23.2±1.7	**22.2±1.1**
Twonorm	3.6±0.6	*2.9±0.2*	6.9±3.6	3.9±0.4	3.3±0.7	5.4±4.0	**2.5±0.2**
Waveform	13.2±1.2	**10.0±0.4**	14.2±0.8	13.5±1.2	11.6±0.7	11.2±1.1	*10.7±0.8*
Adult	*15.9±0.8*	**15.7±0.8**	19.5±1.5	21.6±1.2	18.3±0.8	20.0±0.8	18.4±0.6
Ionosphere	*5.7±2.4*	**5.5±2.1**	29.4±4.1	29.6±4.2	12.9±3.8	9.5±2.8	13.2±3.3
Mushrooms	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0
Sonar	**15.5±3.7**	*15.6±3.6*	16.9±3.7	16.8±3.6	16.3±3.8	16.2±3.6	*15.6±3.6*
Web	2.3±0.3	**1.9±0.3**	2.9±0.3	2.9±0.3	3.6±1.3	2.9±0.3	*2.1±0.3*

Best method in bold face, second best emphasized.

TABLE 3
Percentage of Nonzero Weights

	H-SVM	S-SVM	L2QP-0	L2QP-1	L2QP$_\eta$-0	L2QP$_\eta$-1	Parzen
Banana	26.2±2.5	38.5±2.2	19.9±1.5	16.5±1.1	13.5±1.0	12.7±1.0	100±0.0
B.Cancer	46.9±2.5	55.2±2.9	3.1±0.9	6.7±1.0	17.2±1.9	23.3±2.3	100±0.0
Diabetes	43.1±1.6	52.3±1.7	3.2±0.6	16.3±1.2	6.5±1.1	9.1±1.0	100±0.0
F.Solar	60.6±3.9	76.5±1.8	30.0±2.5	46.7±2.9	46.7±2.4	53.9±3.2	100±0.0
German	52.7±1.6	57.7±1.5	0.5±0.1	6.9±0.5	63.7±2.7	40.7±1.8	100±0.0
Heart	27.8±3.1	50.7±3.0	1.8±0.5	10.4±1.7	20.0±3.5	22.9±3.5	100±0.0
Image	30.2±0.9	7.9±0.8	7.8±2.0	75.8±23.2	79.8±1.0	73.0±1.0	100±0.0
Ringnorm	53.9±3.3	21.5±1.4	100±0.0	100±0.0	100±0.0	100±0.0	100±0.0
Splice	70.8±1.4	17.8±1.9	0.4±0.1	0.3±0.1	40.0±4.7	17.4±1.1	100±0.0
Thyroid	26.7±2.2	32.7±2.4	31.0±2.7	30.8±2.5	60.5±3.1	57.9±3.0	100±0.0
Titanic	45.3±6.5	48.0±6.8	36.5±11.5	44.0±8.2	70.2±4.9	70.4±4.1	100±0.0
Twonorm	19.7±2.1	13.9±1.1	0.7±0.2	5.2±1.2	3.9±0.9	4.5±0.7	100±0.0
Waveform	23.9±2.8	33.9±2.4	96.0±7.0	39.5±4.4	88.6±3.8	42.9±2.6	100±0.0
Adult	32.0±1.4	37.2±1.4	0.13±0.03	0.11±0.02	0.81±0.14	4.1±0.3	100±0.0
Ionosphere	56.7±2.1	34.3±1.7	1.2±0.4	3.1±3.7	32.6±7.2	53.9±2.0	100±0.0
Mushrooms	4.5±0.2	97.4±0.2	100±0.0	100±0.0	51.0±0.7	18.4±0.7	100±0.0
Sonar	89.2±2.5	89.6±2.3	100±0.0	100±0.0	100±0.0	99.2±0.7	100±0.0
Web	7.4±0.5	7.4±0.6	8.2±0.5	8.2±0.5	90.9±1.1	8.2±0.5	100±0.0

TABLE 4
Time (s): Runtime, Including Cross-Validation Search for a Regularization Parameter Where Appropriate and Training Time for All Permutations

	H-SVM	S-SVM	L2QP-0	L2QP-1	L2QP$_\eta$-0	L2QP$_\eta$-1	Parzen
Banana	4140.14	82.01	37.29	51.22	89.17	124.84	-
B.Cancer	2181.51	24.48	10.52	10.80	29.22	47.07	-
Diabetes	30222.68	165.03	52.10	50.42	139.38	135.02	-
F.Solar	15058.31	335.95	79.93	88.83	363.91	849.55	-
German	7310.32	237.69	137.66	142.33	431.41	423.13	-
Heart	173.17	6.60	8.54	8.27	22.72	23.25	-
Image	2047.28	1056.40	397.61	418.93	1353.30	1367.40	-
Ringnorm	105.68	127.04	46.36	48.44	144.95	145.96	-
Splice	2583.97	1323.10	387.71	371.54	1329.09	1386.79	-
Thyroid	8.65	12.12	5.22	4.93	12.78	12.54	-
Titanic	173.12	239.78	4.53	4.55	14.63	15.58	-
Twonorm	12.46	95.32	46.38	44.97	141.88	139.34	-
Waveform	425.29	127.46	47.15	45.21	143.47	140.28	-
Adult	11705.50	10924.42	1016.06	989.54	9929.85	16359.32	-
Ionosphere	15.87	64.89	9.41	9.48	67.01	68.53	-
Mushrooms	861.73	13337.38	3027.31	2944.65	28368.16	29076.56	-
Sonar	4.42	24.16	2.95	3.16	19.25	18.70	-
Web	499.02	5086.96	871.05	858.64	8703.72	11241.11	-

addition, the running time for large C was far greater than that of any other approach.

7 Conclusion

In this paper, the L_2 kernel classification method is proposed which minimizes the L_2 distance between the true unknown difference of densities $d_\gamma(\mathbf{x})$ and an estimator $\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})$. Like the SVM, it is the solution of a convex quadratic program and has a sparse representation.

Through the development of a novel concentration inequality, we have established statistical performance guarantees on the L_2 kernel classifier. The results also specialize to give performance guarantees for an existing method of L_2 kernel density estimation. The oracle inequality here has been applied to deduce consistency of the procedure (in both ISE and probability of error), but we suspect it may also yield adaptive rates of convergence.

Although formulated in terms of the L_2 distance on the difference of densities, the L_2 kernel classifier has geometric interpretations that more clearly reveal similarities and differences to the SVM. One of these interpretations motivates the incorporation of a regularization parameter into the approach, which allows the method to remain competitive with the SVM for dimensionality $d > 15$.

Acknowledgments

The authors would like to thank the anonymous reviewers for several constructive comments. This work was supported in part by US National Science Foundation (NSF) Grant CCF-0830490.

References

[1] B. Schölkopf and A.J. Smola, *Learning with Kernels.* MIT Press, 2002.

[2] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning,* vol. 20, no. 3, pp. 273-297, 1995.

[3] D. Kim, "Least Squares Mixture Decomposition Estimation," unpublished doctoral dissertation, Dept. of Statistics, Virginia Polytechnic Inst. and State Univ., 1995.

[4] M. Girolami and C. He, "Probability Density Estimation from Optimally Condensed Data Samples," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 10, pp. 1253-1264, Oct. 2003.

[5] B.A. Turlach, "Bandwidth Selection in Kernel Density Estimation: A Review," Technical Report 9317, C.O.R.E. and Inst. de Statistique, Université Catholique de Louvain, 1993.

[6] D.W. Scott, "Parametric Statistical Modeling by Minimum Integrated Square Error," *Technometrics,* vol. 43, pp. 274-285, 2001.

[7] F. Bunea, A.B. Tsybakov, and M.H. Wegkamp, "Sparse Density Estimation with l_1 Penalties," *Proc. 20th Ann. Conf. Learning Theory,* pp. 530-543, 2007.

[8] P.H. Rigollet and A.B. Tsybakov, "Linear and Convex Aggregation of Density Estimators," https://hal.ccsd.cnrs.fr/ccsd-00068216, 2004.

[9] R. Jenssen, D. Erdogmus, J.C. Principe, and T. Eltoft, "Towards a Unification of Information Theoretic Learning and Kernel Method," *Proc. IEEE Workshop Machine Learning for Signal Processing,* 2004.

[10] C. He and M. Girolami, "Novelty Detection Employing an L_2 Optimal Nonparametric Density Estimator," *Pattern Recognition Letters,* vol. 25, pp. 1389-1397, 2004.

[11] P. Hall and M.P. Wand, "On Nonparametric Discrimination Using Density Differences," *Biometrika,* vol. 75, no. 3, pp. 541-547, Sept. 1988.

[12] M. Di Marzio and C.C. Taylor, "Kernel Density Classification and Boosting: An L_2 Analysis," *Statistics and Computing,* vol. 15, pp. 113-123, Apr. 2005.

[13] P. Meinicke, T. Twellmann, and H. Ritter, "Discriminative Densities from Maximum Contrast Estimation," *Proc. Advances in Neural Information Processing Systems,* vol. 15, pp. 985-992, 2002.

[14] C.T. Wolverton and T.J. Wagner, "Asymptotically Optimal Discriminant Functions for Pattern Classification," *IEEE Trans. Information Theory,* vol. 15, no. 2, pp. 258-265, Mar. 1969.

[15] K. Pelckmans, J.A.K. Suykens, and B. De Moor, "A Risk Minimization Principle for a Class of Parzen Estimators," *Proc. Advances in Neural Information Processing Systems,* vol. 20, Dec. 2007.

[16] J. Kim and C. Scott, "Kernel Classification via Integrated Squared Error," *Proc. IEEE Workshop Statistical Signal Processing,* Aug. 2007.

[17] J. Kim and C. Scott, "Performance Analysis for L_2 Kernel Classification," *Proc. Advances in Neural Information Processing Systems,* vol. 21, Dec. 2008.

[18] M.P. Wand and M.C. Jones, *Kernel Smoothing.* Chapman & Hall, 1995.

[19] J.C. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," Technical Report MSR-TR-98-14, Apr. 2001.

[20] D. Crisp and C. Burges, "A Geometric Interpretation of ν-SVM Classifiers," *Proc. Neural Information Processing Systems,* vol. 12, 1999.

[21] K. Bennett, N. Cristianini, J. Shawe-Taylor, and D. Wu, "Enlarging the Margins in Perceptron Decision Trees," *Machine Learning,* vol. 41, pp. 295-313, 2000.

[22] A.S. Paulson, E.W. Holcomb, and R.A. Leitch, "The Estimation of the Parameters of the Stable Laws," *Biometrika,* vol. 62, pp. 163-170, 1975.

[23] C.R. Heathcote, "The Integrated Squared Error Estimation of Parameters," *Biometrika,* vol. 64, pp. 255-264, 1977.

[24] J.A.K. Suykens and J. Vandewalle, "Least Squares Support Vector Machine Classifiers," *Neural Processing Letters,* vol. 44, no. 8, pp. 293-300, June 1999.

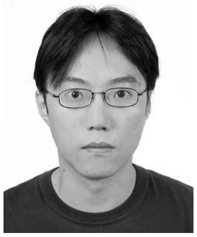
[25] J.R. Schechuk, "An Introduction to the Conjugate Gradient Method without the Agonizing Pain," Technical Report MSR-TR-98-14, Aug. 1994.

[26] D. Berry, K. Chaloner, and J. Geweke, *Bayesian Analysis in Statistics and Econometrics: Essays in Honor of Arnold Zellner.* Wiley, 1996.

[27] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf, "Kernel Methods for Measuring Independence," *J. Machine Learning Research,* vol. 6, pp. 2075-2129, 2005.

[28] C.-C. Chang and C.-J. Lin, *LIBSVM: A Library for Support Vector Machines,* http://www.csie.ntu.edu.tw/cjlin/libsvm, 2001.

[29] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An Introduction to Kernel-Based Learning Algorithms," *IEEE Trans. Neural Networks,* vol. 12, no. 2, pp. 181-201, Mar. 2001.



JooSeuk Kim received the BS degree in electrical engineering from Seoul National University in 2002, and the MSE degree in electrical engineering and computer science from the University of Michigan, Ann Arbor, in 2007. He was a software developer in AhnLab, Inc., from 2002 to 2004. He is currently working toward the PhD degree at the University of Michigan, Ann Arbor. His research interests include machine learning statistical learning theory and kernel methods. He is a student member of the IEEE.

Clayton D. Scott received the AB degree in mathematics from Harvard University in 1998, and the MS and PhD degrees in electrical engineering from Rice University in 2000 and 2004, respectively. He was a postdoctoral fellow in the Department of Statistics at Rice, and is currently an assistant professor in the Department of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor. His research interests include machine learning theory and applications. He is a member of the IEEE.