KERNEL CLASSIFICATION VIA INTEGRATED SQUARED ERROR

JooSeuk Kim and Clayton D. Scott

Dept. of EECS, University of Michigan, Ann Arbor, MI, USA
E-mail: {stannum, clayscot}@umich.edu

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 an integrated squared error (ISE) criterion based on a "difference of densities" formulation. Our classifier is sparse, like SVMs, and performs comparably to state-of-the-art kernel methods. Furthermore, and unlike SVMs, the ISE criterion does not require the user to set any unknown regularization parameters. As a consequence, classifier training is faster than for support vector methods.

Index Terms— kernel methods, integrated squared error, sparse classifiers, quadratic programming, difference of densities

1. INTRODUCTION

In the 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 or feature vector 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 has the form

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

where α_i are parameters and k is a kernel function, often taken to be a Gaussian kernel. For example, support vector machines (without offset) have this form, as does the standard kernel density estimate (KDE) plug-in classifier.

In this paper we employ an L^2 or integrated squared error (ISE) criterion to design the coefficients α_i of a kernel classifier with Gaussian kernel. Our ISE-based classifiers perform comparably to existing kernel methods while possessing a number of desirable properties. First, like the SVM, the classifiers are sparse, meaning most of the coefficients $\alpha_i = 0$, which has advantages for representation and evaluation efficiency. Second, and unlike the SVM, there are no free parameters to be set by the user except perhaps the kernel bandwidth parameter. In contrast, the SVM has a second regularization parameter, usually denoted by C, that can drastically affect the classifier's performance if not set properly, and requires a time consuming cross-validation search to set in practice.

Our approach can be summarized as follows. Let $f_-(\mathbf{x})$ and $f_+(\mathbf{x})$ denote the class-conditional densities of the pattern given the label. From decision theory we know the optimal classifier has the form

$$g^*(x) = \text{sign}\left\{f_+(\mathbf{x}) - \gamma f_-(\mathbf{x})\right\}, \tag{1}$$

where γ incorporates prior class probabilities and class-conditional error costs (in the Bayesian setting) or a desired tradeoff between false positives and false negatives. Denote the "difference of densities" $d_\gamma(\mathbf{x}) := f_+(\mathbf{x}) - \gamma f_-(\mathbf{x})$. We model the class-conditional densities as KDEs with *variable weights*. Rather than estimating these densities separately and plugging in to (1), we minimize the L^2 distance between $d_\gamma(\mathbf{x})$ and $\widehat{d}_\gamma(\mathbf{x}) := \widehat{f}_+(\mathbf{x}) - \gamma\widehat{f}_-(\mathbf{x})$ directly. This is motivated by the understanding that classification is easier than density estimation, and also in the hope of obtaining sparser classifiers. The estimation problem is thus reduced to the minimization of a quadratic objective function in $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_n)$. Depending on whether we constrain the implicit density estimates to be proper densities, we must solve either a constrained or an unconstrained quadratic program, and efficient algorithms are available in either case. The respective classifiers are qualitatively similar to the two-norm SVM with hinge loss and least-squares SVM, respectively, but the actual objective functions and classifiers are quite different.

The ISE criterion has a long history in the literature on bandwidth selection for kernel density estimation [2] and more recently in parametric estimation [3]. The use of ISE for optimizing the weights of a KDE via quadratic programming was first described in [4] and later rediscovered in [5]. Some connections relating SVMs and ISE are made in [6], although no new algorithms are proposed. Final, the "difference of densities" perspective has been applied to classification in other settings by [7], [8], and [9].

2. L^2 KERNEL CLASSIFICATION

We model the class-conditional densities with Gaussian kernel as

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

where $I_+ = \{i \mid y_i = +1\}$ and $I_- = \{i \mid y_i = -1\}$ and

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

The ISE associated with $\boldsymbol{\alpha}$ is

$$\|\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - d_\gamma(\mathbf{x})\|_{L^2}^2 = \int \left(\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha}) - d_\gamma(\mathbf{x})\right)^2 d\mathbf{x}$$
$$= \int \widehat{d}_\gamma^2(\mathbf{x};\boldsymbol{\alpha}) - 2\widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})\, d_\gamma(\mathbf{x}) + d_\gamma^2(\mathbf{x})\, d\mathbf{x} \qquad (2)$$

The first term in (2) becomes

$$\int \widehat{d}_\gamma^2(\mathbf{x};\boldsymbol{\alpha})\, d\mathbf{x} = \int \left(\sum_{i=1}^n \alpha_i \tilde{y}_i k_\sigma(\mathbf{x},\mathbf{x}_i)\right)^2 d\mathbf{x}$$
$$= \sum_{i=1}^n \sum_{j=1}^n \alpha_i\alpha_j\tilde{y}_i\tilde{y}_j \int k_\sigma(\mathbf{x},\mathbf{x}_i)\, k_\sigma(\mathbf{x},\mathbf{x}_j)\, d\mathbf{x}$$
$$= \sum_{i=1}^n \sum_{j=1}^n \alpha_i\alpha_j\tilde{y}_i\tilde{y}_j k_{\sqrt{2}\sigma}(\mathbf{x}_i,\mathbf{x}_j)$$

where $\tilde{y}_i = 1$ for $i \in I_+$, $-\gamma$ for $i \in I_-$ by the convolution theorem for Gaussian kernels.

For the second term in (2), we approximate the expected value to the sample average so that

$$\int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})\, d_\gamma(\mathbf{x})\, d\mathbf{x}$$
$$= \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})\, f_+(\mathbf{x})\, d\mathbf{x} - \gamma \int \widehat{d}_\gamma(\mathbf{x};\boldsymbol{\alpha})\, f_-(\mathbf{x})\, d\mathbf{x}$$
$$\approx \frac{1}{n_+}\sum_{i\in I_+} \widehat{d}_\gamma(\mathbf{x}_i;\boldsymbol{\alpha}) - \frac{\gamma}{n_-}\sum_{i\in I_-} \widehat{d}_\gamma(\mathbf{x}_i;\boldsymbol{\alpha})$$
$$= \sum_{i=1}^n \alpha_i\tilde{y}_i \left(\frac{1}{n_+}\sum_{j\in I_+} k_\sigma(\mathbf{x}_i,\mathbf{x}_j) - \frac{\gamma}{n_-}\sum_{j\in I_-} k_\sigma(\mathbf{x}_i,\mathbf{x}_j)\right)$$

where $n_+ = |I_+|, n_- = |I_-|$ and $|\cdot|$ means cardinality. Since the third term in (2) does not depend on $\boldsymbol{\alpha}$, the minimization of (2) now becomes the minimization of

$$\frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i\alpha_j\tilde{y}_i\tilde{y}_j k_{\sqrt{2}\sigma}(\mathbf{x}_i,\mathbf{x}_j) - \sum_{i=1}^n c_i\alpha_i \qquad (3)$$

or

$$\frac{1}{2}\boldsymbol{\alpha}^T Q\boldsymbol{\alpha} - \mathbf{c}^T\boldsymbol{\alpha} \qquad (4)$$

where $\mathbf{c} = [c_1, c_2, \cdots, c_n]^T$ and

$$Q := (\tilde{y}_i\tilde{y}_j K_{ij})_{i,j=1}^n, \quad K := \left(k_{\sqrt{2}\sigma}(\mathbf{x}_i,\mathbf{x}_j)\right)_{i,j=1}^n$$
$$c_i = \tilde{y}_i\left(\frac{1}{n_+}\sum_{j\in I_+} k_\sigma(\mathbf{x}_i,\mathbf{x}_j) - \frac{\gamma}{n_-}\sum_{j\in I_-} k_\sigma(\mathbf{x}_i,\mathbf{x}_j)\right).$$

It can be shown that if K is positive definite and $\gamma \neq 0$, then the matrix Q is also positive definite.

2.1. Proper densities

If we require the estimates $\widehat{f}_+(\mathbf{x};\boldsymbol{\alpha})$ and $\widehat{f}_-(\mathbf{x};\boldsymbol{\alpha})$ to be proper densities, α_i's should satisfy $\sum_{i\in I_+}\alpha_i = \sum_{i\in I_-}\alpha_i = 1, \quad \alpha_i \geq 0 \quad \forall i$. With these constraints, the quadratic objective function (3) is minimized by solving the following quadratic program (QP)

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \alpha_i\alpha_j\tilde{y}_i\tilde{y}_j k_{\sqrt{2}\sigma}(\mathbf{x}_i,\mathbf{x}_j) - \sum_{i=1}^n c_i\alpha_i \qquad (5)$$
$$\text{s.t} \quad \sum_{i\in I_+}\alpha_i = \sum_{i\in I_-}\alpha_i = 1, \quad \alpha_i \geq 0, \quad \forall\, i.$$

The constraints enforce most α_i's to be zero and therefore the resulting L2QP (L2 classification via Quadratic Programming) classifier is sparse. The quadratic programming in L2QP classifier and two-norm SVM with hinge loss are similar and both classifiers are sparse. However, L2QP does not include a regularization parameter. The support vectors of the SVM are on or near the decision boundary whereas nonzero α_i's in L2QP correspond to regions of space with greater probability mass. The QP can be solved by a variant of the Sequential Minimal Optimization (SMO) algorithm (see Appendix).

2.2. Improper densities

If we allow improper density estimates, we do not impose any constraint on α_i's. In this case, the unconstrained quadratic objective function (4) is minimized when the derivative of (4) is equal to zero:

$$Q\boldsymbol{\alpha} = \mathbf{c}. \qquad (6)$$

The optimization problem now becomes the problem of solving the linear system of equations (6). It is similar to that of least-squares SVM (LS-SVM) [10]. The resulting L2LE (L2 classification via Linear system of Equations) is not sparse like LS-SVM. If Q is positive definite, (6) can be solved by the Conjugate Gradients method [14].

3. EXPERIMENTS

In the first example, we experiment with 1 dimensional input data. Both classes are equally likely and

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

where $\phi(\mathbf{x}; \mu, \sigma)$ is a univariate Gaussian pdf with mean μ and variance σ^2. We build classifiers from 200 training samples via L2QP and L2LE. To find a classifier with the smallest probability error, we set $\gamma = n_-/n_+$ and use 5-fold-cross validation for bandwidth σ.

The results are shown in Fig.1. The estimates $\widehat{d}_\gamma(\mathbf{x}; \boldsymbol{\alpha})$ of L2QP and L2LE are fairly close to the true $d_\gamma(\mathbf{x})$. (e) and (f) shows the advantage of L2QP classifier over KDE plug-in classifier. In the plug-in classifier, the weights of $\widehat{f}_+(\mathbf{x}; \boldsymbol{\alpha})$ and $\widehat{f}_-(\mathbf{x}; \boldsymbol{\alpha})$ are separately learned by ISE minimization [5]. The number of non-zero weights for the plug-in classifier is 16 while L2QP classifier requires 9 training samples.

Next, we demonstrate our algorithms on six artificial and real world benchmark datasets, available online[1] [11] and compare the results with the 2-norm SVM with hinge loss, implemented using LIBSVM [12]. There are 100 partitions of each dataset into training and test sets. A brief summary of each dataset is shown in Table 1.

Table 1. General information about benchmark datasets

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

Parameters are set as follows. We set $\gamma = n_-/n_+$ to minimize the probability of error. The kernel bandwidth σ is taken to be the same for all 100 partitions. For L2QP and L2LE, it is determined by taking the median estimated bandwidth based on the first five training sets. On each of these training sets, we search for the bandwidth over a logarithmically spaced grid of 50 points from 10^{-2} to 10^1 and use 5-fold-cross validation to determine the best bandwidth. The parameters of the SVM are obtained in a similar way but the grid points used to search for σ and C are $2^{-2}, 2^{-1}, \ldots, 2^7$ and $2^{-5}, 2^{-3}, \ldots, 2^{15}$, respectively.[2]

Only 'banana' has 2 input dimensions among those datasets and we plot the decision boundary of L2QP and L2LE classifier in Fig.2, along with training samples. The number of training samples is 400 and the number of non-zero weights using proper density estimates and improper density estimates are 75 and 400, respectively. L2QP method has more reasonable extrapolations than L2LE method, but these differences may be minor if the training data are truly representative.

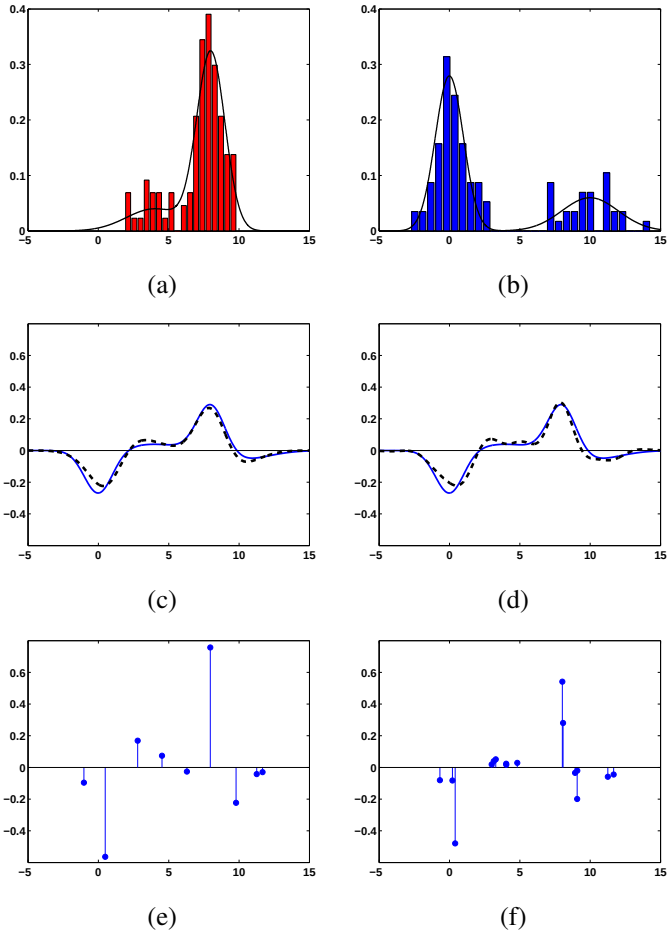


Fig. 1. (a) $f_+(\mathbf{x})$ and histogram of its samples (b) $f_-(\mathbf{x})$ and histogram of its samples (c) L2QP: $d_\gamma(\mathbf{x})$ (solid line) and $\widehat{d}_\gamma(\mathbf{x})$ (dashed line) (d) L2LE: $d_\gamma(\mathbf{x})$ (solid line) and $\widehat{d}_\gamma(\mathbf{x})$ (dashed line) (e) Sparsity of L2QP (f) Sparsity of plug-in classifier

The results for all the datasets is presented in Table 2. It shows the selected parameters and the average sparsity, time, and the probability of error over 100 permutations. Sparsity is the percentage of nonzero weights. Time indicates the average time required to build a classifier, including the cross-validation search for free parameters.

4. CONCLUSION

The experiment shows that the L2QP methods are often much sparser than the SVM. Indeed, the SVM sparsity is typically

[1] http://ida.first.fhg.de/projects/bench/

[2] The kernel is defined as $k(\mathbf{x}, \mathbf{x}') = e^{-\|\mathbf{x}-\mathbf{x}'\|/2\sigma^2}$ and C is a regularization parameter.

Table 2. Experimental results for the datasets

		σ	C	sparsity (%)	time (s)	prob. of error
Banana	SVM	1	2	39.54	82.01	10.8±0.5
	L2QP	0.222	-	20.08	125.61	11.1±0.6
	L2LE	0.450	-	100.00	13.67	11.0±0.5
B.Cancer	SVM	16	2048	55.15	24.48	26.9±4.6
	L2QP	1.600	-	3.05	141.48	26.5±4.6
	L2LE	1.600	-	100.00	4.42	25.3±4.1
Diabetes	SVM	16	128	52.32	165.03	23.2±1.8
	L2QP	1.048	-	4.79	75.98	27.1±2.1
	L2LE	1.389	-	100.00	16.16	27.4±2.4
F.Solar	SVM	8	8	76.46	335.95	32.3±1.8
	L2QP	0.023	-	45.39	133.77	35.8±1.9
	L2LE	0.910	-	100.00	36.75	35.1±1.9
German	SVM	16	8	57.66	237.69	24.2±2.1
	L2QP	1.842	-	0.39	190.13	29.3±2.0
	L2LE	1.600	-	100.00	47.45	26.9±2.5
Heart	SVM	16	8	50.73	6.60	15.6±3.4
	L2QP	2.121	-	1.42	15.01	17.5±4.2
	L2LE	1.842	-	100.00	3.62	18.4±3.8

greater than 50%, while the sparsity of L2QP is often less that 10%. As for probability of error, the SVM is overall the best, but the proposed L^2 methods are occasionally better and almost always within the standard error of the SVM. L2QP requires greater training time than L2LE due to the different optimization algorithms for $\boldsymbol{\alpha}$. The SMO algorithm used for L2QP requires $O(n^2)$ steps [13] while the CGD algorithm for L2LE requires only $O(n)$ steps [14]. Since SVM is implemented in C [12], while L^2 methods are implemented in MATLAB, it is difficult to give a precise comparison with the SVM. However, on 3 of the 6 datasets, the SVM is already the slowest. This would seem to indicate that C implementations of our methods will be significantly faster, a reflection of not having to search for an additional regularization parameter.

In conclusion, L^2 kernel classifiers based on density differences are comparable to SVMs in terms of performance but offer significantly greater sparsity and training efficiency.

5. APPENDIX: SMO ALGORITHM

Sequential Minimal Optimization (SMO) is a simple algorithm that can quickly solve the SVM QP problem without any extra matrix storage and without using time-consuming numerical QP optimization steps [13]. SMO decomposes the overall QP problem into the smallest possible optimization problem. This sub-problem can be solved analytically. An appropriate variant of SMO to solve (5) is detailed below following [5].

Given $\boldsymbol{\alpha}$, the algorithm optimizes two variables of $\boldsymbol{\alpha}$ with other variables fixed. Two variables to be optimized should be chosen from $\{\alpha_i \mid i \in I_-\}$ or $\{\alpha_i \mid i \in I_+\}$. Otherwise, the variables which we are trying to optimize cannot change since the other variables are fixed and due to the constraints $\sum_{i\in I_-} \alpha_i = 1$ and $\sum_{i\in I_+} \alpha_i = 1$. Suppose that we choose two variables from $\{\alpha_i \mid i \in I_+\}$. For notational convenience, assume the two variables are α_1 and α_2 and $1, 2 \in I_+$. Then, (5) reduces to

$$\begin{aligned} \min_{\alpha_1,\alpha_2} \quad & \frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2}\alpha_i\alpha_j Q_{ij} + \sum_{i=1}^{2} d_i\alpha_i + D \\ \text{s.t} \quad & \alpha_1, \alpha_2 \geq 0, \quad \sum_{i=1}^{2}\alpha_i = \Delta \end{aligned}$$

where $D = \frac{1}{2}\sum_{i=3}^{n}\sum_{j=3}^{n}\alpha_i\alpha_j Q_{ij} - \sum_{i=3}^{n} c_i\alpha_i$ and

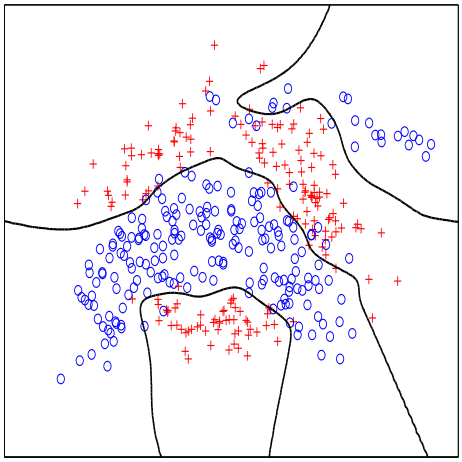
$$d_i = \sum_{j=3}^{n}\alpha_j Q_{ij} - c_i, \quad \Delta = 1 - \sum_{i\in I_+}\alpha_i.$$

We discard D, which is independent of α_1 and α_2, and eliminate α_1 to obtain

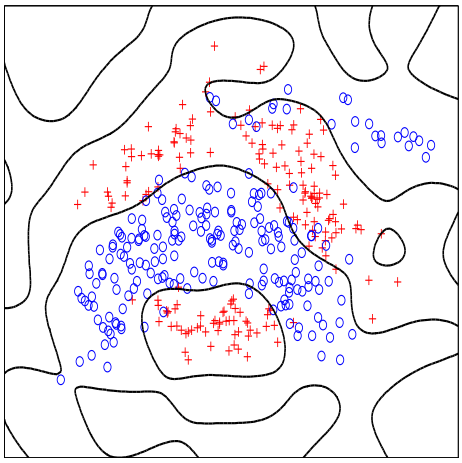
$$\begin{aligned} \min_{\alpha_2} \quad & \frac{1}{2}(\Delta - \alpha_2)^2 Q_{11} + \alpha_2(\Delta - \alpha_2)Q_{12} \\ & + \frac{1}{2}\alpha_2^2 Q_{22} + (\Delta - \alpha_2)d_1 + \alpha_2 d_2 \\ \text{s.t} \quad & 0 \leq \alpha_2 \leq \Delta. \end{aligned} \tag{7}$$

Since the objective function is quadratic and convex in one variable α_2, we can take the derivative of (7) and set it equal to zero. Then,

$$\alpha_2 = \frac{\Delta(Q_{11} - Q_{12}) + d_1 - d_2}{Q_{11} - 2Q_{12} + Q_{22}}. \tag{8}$$



(a) L2QP



(b) L2LE

Fig. 2. Decision boundary along with positive samples (+) and negative samples (o)

Let $\boldsymbol{\alpha}^*$ denote the value before the optimization step. If we define $O_i := Q_{i1}\alpha_1^* + Q_{i2}\alpha_2^* + d_i = \sum_{j=1}^n \alpha_i^* Q_{ij} - c_i$, then (8) can be expressed as the update equation

$$\alpha_2 = \alpha_2^* + \frac{O_1 - O_2}{Q_{11} - 2Q_{12} + Q_{22}}. \tag{9}$$

If α_2 is outside $[0, \Delta]$, we truncate it so that it is within $[0, \Delta]$. After finding α_2, α_1 can be recovered from $\alpha_1 = \Delta - \alpha_2$.

The optimality condition and the choice of α_i's can be found in the following way. There are three cases when choosing α_1 and α_2 : (a) Both are zero, (b) One is positive and the other is zero, (c) Both are positive.

Case (a): α_1 and α_2 are not updated because of non-negativity constraints.

Case (b): Assume that α_2 is zero. From (9), α_2 is updated only when $O_1 - O_2 > 0$ and so is α_1

Case (c): α_1 and α_2 are updated only when $O_1 \neq O_2$.

The objective value will strictly decrease if and only if α_1 and α_2 are updated after optimization step. Therefore, the optimal solution should satisfy

$$\begin{aligned} O_i \geq O_j \quad &\text{for} \quad \alpha_i = 0,\ \alpha_j > 0 \\ O_i = O_j \quad &\text{for} \quad \alpha_i, \alpha_j > 0. \end{aligned} \tag{10}$$

The convergence to the global minimum is thus guaranteed by choosing two α_i's which do not satisfy (10) for each optimization step. The optimization procedure for two variables from $\{\alpha_i \in I_-\}$ is similar.

6. REFERENCES

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

[2] B.A. Turlach, "Bandwidth selection in kernel density estimation: A review," *Technical Report 9317, C.O.R.E. and Institut de Statistique, Université Catholique de Louvain*, 1993.

[3] David W.Scott, "Parametric statistical modeling by minimum integrated square error," *Technometrics 43*, pp. 274–285, 2001.

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

[5] Mark Girolami and Chao He, "Probability density estimation from optimally condensed data samples," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1253–1264, OCT 2003.

[6] Robert Jenssen, Deniz Erdogmus, Jose C.Principe, and Torbjørn Eltoft, "Towards a unification of information theoretic learning and kernel method," in *Proc. IEEE Workshop on Machine Learning for Signal Processing (MLSP2004),* Sao Luis, Brazil.

[7] Peter Hall and Matthew P.Wand, "On nonparametric discrimination using density differeces," *Biometrika*, vol. 75, no. 3, pp. 541–547, Sept 1988.

[8] P. Meinicke, T. Twellmann, and H. Ritter, "Discriminative densities from maximum contrast estimation," in *Advances in Neural Information Proceeding Systems 15,* Vancouver, Canada, 2002, pp. 985–992.

[9] M. Di Marzio and C.C. Taylor, "Kernel density classification and boosting: an l2 analysis," *Statistics and Computing*, vol. 15, pp. 113–123(11), April 2005.

[10] Suykens J.A.K. and Vandewalle J., "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 44, no. 8, pp. 293–300, Jun 1999.

[11] G. Rätsch K. Tsuda K.-R. Müller, S. Mika and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, pp. 181–201, Mar 2001.

[12] Chih-Chung Chang and Chih-Jen Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[13] John C.Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," *Technical Report MSR-TR-98-14*, April 2001.

[14] Jonathan R.Schechuk, "An introduction to the conjugate gradient method without the agonizing pain," *Technical Report MSR-TR-98-14*, August 1994.