Reduced Support Vector Machines: A Statistical Theory

Yuh-Jye Lee
Computer Science & Information Engineering
National Taiwan University of Science and Technology
Taipei 106, Taiwan
yuh-jye@mail.ntust.edu.tw

Su-Yun Huang
Institute of Statistical Science
Academia Sinica
Taipei 115, Taiwan
syhuang@stat.sinica.edu.tw

November 15, 2005

Abstract

In dealing with large datasets the reduced support vector machine (RSVM) was proposed for the practical objective to overcome the computational difficulties as well as to reduce the model complexity.

In this article, we study the RSVM from the viewpoint of robust design for model building and consider the nonlinear separating surface as a mixture of kernels. The RSVM uses a reduced model representation instead of a full one. Our main results center on two major themes. One is on the robustness of the random subset mixture model. The robustness is judged by a few criteria: (1) model variation measure, (2) model bias (deviation) between the reduced model and the full model and (3) test power in distinguishing the reduced model from the full one. The other is on the spectral analysis of the reduced kernel. We compare the eigen-structures of the full kernel matrix and the approximation kernel matrix. The approximation kernels are generated by uniform random subsets. The small discrepancies between them indicate that the approximation kernels can retain most of the relevant information for learning tasks in the full kernel. We focus on some statistical theory of the reduced set method mainly in the context of the RSVM. The use of a uniform random subset is not limited to the RSVM. This approach can act as a supplemental-algorithm on top of a basic optimization algorithm, wherein the actual optimization takes place on the subset-approximated data. The statistical properties discussed in this paper are still valid.

Key words and phrases: canonical angles, kernel methods, maximinity, minimaxity, model complexity, reduced set, Monte-Carlo sampling, Nyström approximation, spectral analysis, support vector machines, uniform design, uniform random subset.

1 Introduction

In recent years support vector machines (SVMs) with linear or nonlinear kernels [4, 8, 40] have become one of the most promising learning algorithms for classification as well as for regression [11, 26, 27, 37, 18], which are two fundamental tasks in data mining [45]. Via the use of kernel mapping, variants of SVM have successfully incorporated effective and flexible nonlinear models. There are some major difficulties that confront large data problems due to dealing with a fully dense nonlinear kernel matrix. To overcome computational difficulties some authors have proposed low-rank approximation to the full kernel matrix [35, 44]. As an alternative, Lee and Mangasarian have proposed the method of reduced support vector machine (RSVM) [20]. The key ideas of the RSVM are as follows. Prior to training, it randomly selects a portion of dataset as to generate a thin rectangular kernel matrix. Then it uses this much smaller rectangular kernel matrix to replace the full kernel matrix in the nonlinear SVM formulation. Computational time, as well as memory usage, is much less demanding for RSVM than that for a conventional SVM using the full kernel matrix. As a result, the RSVM also simplifies the characterization of the nonlinear separating surface. The numerical comparisons in [20, 18] and later in Section 3 show that the RSVM though has higher training errors than the conventional SVM, it has comparable test errors, sometimes even slightly smaller. In other words, the RSVM has comparable, or sometimes slightly better, generalization ability. This phenomenon can be interpreted by the Minimum Description Length [31] as well as the Occam's razor [33].

The technique of using a reduced kernel matrix has been successfully applied to other kernel-based learning algorithms, such as proximal support vector machine [14], ϵ-smooth support vector regression (ϵ-SSVR) [18], Lagrangian support vector machine [28], least-square support vector machine [38, 39, 23]. Also, there were experimental studies on RSVM [23, 18] that showed the computing-time efficiency of RSVM. The RSVM results reported in [23] can be further improved if a stratified random subset was drawn first and then the one-against-one binary problems were solved using corresponding support vectors in this stratified subset. Since the RSVM has reduced the model complexity by using a much smaller rectangular kernel matrix, we suggest using a larger weight parameter to enforce better data fidelity. The numerical test in [20] on the Adult dataset [3] shows that the sample standard deviation of test set correctness for 50 replicate runs is less than 0.001. The replicate runs are based on 50 randomly chosen (with replacement) different reduced sets with the size of 1% of original dataset. In fact, the smallness of the sample standard deviation can be used as guidance for determining the size of the reduced set.

In this article we study the RSVM from the viewpoint of robust design for model building and consider the nonlinear separating surface as a mixture of kernels. The RSVM uses a reduced model representation instead of a full model. Our main results center on two major themes. One is on the robustness of the random subset mixture model and the other is on the spectral analysis of the reduced kernel. The robustness is judged by a few criteria. (1) Consider a class of mixture models built via certain Monte-Carlo sampling schemes. Among this class the RSVM mixture model via uniform random sampling minimizes a model variation measure. This uniform random subset selection in RSVM also has a link to the popular uniform design, which is a space filling design [12]. The space filling design is known to be robust against the worst possible scenario and can reduce the model bias (deviation). (2) The RSVM mixture model via uniform random sampling minimizes the maximal model bias (deviation) between the reduced model and the full model. (3) It also maximizes the minimal test power in distinguishing the reduced model from the full model. As for the spectral analysis we will compare the eigen-structures of the full kernel matrix and the approximation kernel where the approximation kernels are generated by uniform random subsets. The small discrepancies between them can provide an evidence that the approximation kernels can retain most of the relevant information for learning tasks in the full kernel.

We briefly outline the contents of this article. Section 2 provides the main ideas and formulation for RSVM. Section 3 discusses the reduced set mixture models with kernel bases drawn from a Monte-Carlo sampling scheme. The integrated variance of the Monte-Carlo sampling scheme is used to judge the model variation. The uniform random sampling is the optimality scheme among a certain class. Section 4 gives a comparison study on the spectral analysis of reduced kernels by uniform random subset and full kernel matrix. Section 5 further provides the uniform randomness certain minimaxity and maximinity properties. We discuss the applicability of the uniform random subset method to other kernel-based algorithms in Section 6. Section 7 concludes the articles. All proofs are placed in the Appendix.

A word about our notation and background material is given below. All vectors are column vectors unless otherwise specified or transposed to a row vector by a prime superscript $'$. For a vector $x = (x_1, \ldots, x_n) \in R^n$, the plus function x_+ is defined componentwise as $(x_+)_j = \max\,\{0, x_j\}$. The scalar (inner) product of two vectors $x, z \in R^n$ will be denoted by $x'z$ and the p-norm of x will be denoted by $\|x\|_p$. For a matrix $A \in R^{m\times n}$, A_i is the ith *row* of A. A column vector of ones of arbitrary dimension will be denoted by $\mathbf{1}$. For $A \in R^{m\times n}$ and $B \in R^{n\times l}$, the kernel $K(A, B)$ maps $R^{m\times n} \times R^{n\times l}$ into $R^{m\times l}$. In particular, if x and y are column vectors in R^n then, $K(x', y)$ is a real number, $K(A, x)$ is a *column* vector in R^m and $K(A, A')$ is an $m \times m$ matrix.

2 RSVM formulation

Consider the problem of classifying points into two classes, A_- and A_+. We are given a training dataset $\{(x^i, y_i)\}_{i=1}^m$, where $x^i \in \mathcal{X} \subset R^n$ is an input vector and $y_i \in \{-1, 1\}$ is a class label, indicating one of the two classes, A_- and A_+, to which the input point belongs. We represent these data points by an $m \times n$ matrix A, where the ith row A_i corresponds to the ith input data point. We use alternately A_i (a row vector) and x^i (a column vector) for the same ith data point depending on the convenience. We use an

$m \times m$ diagonal matrix D, $D_{ii} = y_i$, to specify the corresponding class membership of each input point. The main goal of the classification problem is to find a classifier that can predict correctly the unseen class labels for new data inputs. It can be achieved by constructing a linear or nonlinear separating surface, $f(x) = 0$, which is implicitly defined by a kernel function. We classify a test point x to A_+ if $f(x) \geq 0$, otherwise, to A_-. We will focus on nonlinear case in this paper. In conventional SVM as well as many kernel-based learning algorithms [4, 8, 40] generating a nonlinear separating surface has to deal with a fully dense kernel matrix with the size of the number of training examples. When training a nonlinear SVM on a massive dataset, the huge and dense full kernel matrix will lead to some computational difficulties below:

(P1) the size of the mathematical programming problem;

(P2) the dependence of the nonlinear separating surface on most of the dataset, which creates unwieldy storage problems that hinder the use of nonlinear kernels for massive datasets.

To avoid these difficulties and to cut down model complexity, the RSVM uses a very small random subset of size $\tilde{m}$, where $\tilde{m} \ll m$, for building up the separating surface which plays a similar role of support vectors. We denote this random subset by $\tilde{A}$, which is used to generate a much smaller rectangular matrix $K(A, \tilde{A}') \in R^{m \times \tilde{m}}$. The reduced kernel matrix is served to replace the full kernel matrix $K(A, A')$ to cut problem size, computing time and memory usage as well as to simplify the characterization of the nonlinear separating surface.

We now briefly describe the RSVM formulation, which is derived from the generalized support vector machine (GSVM) [25] and the smooth support vector machine (SSVM) [21]. The RSVM starts from a standard 2-norm soft margin SVM, and next it appends the term $\gamma^2/2$ to the objective function to be minimized and results in the following minimization problem:

$$\begin{aligned} \min_{(u,\gamma,\varepsilon)} \quad & \frac{C}{2}\|\varepsilon\|_2^2 + \frac{1}{2}(\|u\|_2^2 + \gamma^2) \\ \text{subject to} \quad & D\{K(A,A')Du - \mathbf{1}\gamma\} \geq \mathbf{1} - \varepsilon \\ & \varepsilon \geq 0, \end{aligned}$$

where C is a positive number for balancing *training error* and the *regularization term* in the objective function. We call it as *weight parameter*. We note that the nonnegative constraint $\varepsilon \geq 0$ can be removed because of the term $\|\varepsilon\|_2^2$ in the objective function of the minimization problem. If we let $v = Du$, then $\|v\|_2^2 = \|u\|_2^2$. Thus, the problem above is equivalent to

$$\min_{(v,\gamma,\varepsilon)} \frac{C}{2}\|\varepsilon\|_2^2 + \frac{1}{2}(\|v\|_2^2 + \gamma^2) \tag{1}$$

$$\text{subject to} \;\; D\{K(A,A')v - \mathbf{1}\gamma\} \geq \mathbf{1} - \varepsilon. \tag{2}$$

At a solution, ε takes the form $\varepsilon = (\mathbf{1} - D\{K(A,A')v - \mathbf{1}\gamma\})_+$. Next we convert the problem given by (1) and (2) into an equivalent SVM, which is an unconstrained optimization problem as follows:

$$\min_{(v,\gamma)\in R^{m+1}} \frac{C}{2}\|(\mathbf{1} - D\{K(A,A')v - \mathbf{1}\gamma\})_+\|_2^2 + \frac{1}{2}(\|v\|_2^2 + \gamma^2). \tag{3}$$

Instead of using the full kernel matrix $K(A,A')$, we replace it with a reduced kernel matrix $K(A,\tilde{A}')$, where $\tilde{A}$ consists of $\tilde{m}$ random columns from A, and the problem becomes

$$\min_{(\tilde{v},\gamma)\in R^{\tilde{m}+1}} \frac{C}{2}\|(\mathbf{1} - D\{K(A,\tilde{A}')\tilde{v} - \mathbf{1}\gamma\})_+\|_2^2 + \frac{1}{2}(\|\tilde{v}\|_2^2 + \gamma^2). \tag{4}$$

In solving the RSVM (4), a smooth approximation $p(z,\alpha)$ to the plus function is used [21]. The p function defined below can provide a very accurate approximation. The RSVM then solves the following approximate unconstrained minimization problem for a general kernel $K(A, \tilde{A}')$:

$$\min_{(\tilde{v},\gamma)\in R^{\tilde{m}+1}} \frac{C}{2}\|p(\mathbf{1} - D\{K(A, \tilde{A}')\tilde{v} - \mathbf{1}\gamma\}, \alpha)\|_2^2 + \frac{1}{2}(\|\tilde{v}\|_2^2 + \gamma^2), \tag{5}$$

where $p(z, \alpha)$ is defined componentwise by

$$\{\text{the } j\text{th component of } p(z, \alpha)\} = z_j + \frac{1}{\alpha}\log(1 + \exp\{-\alpha z_j\}),\ \alpha > 0,\ j = 1, \ldots, m. \tag{6}$$

The function $p(z, \alpha)$ converges to $(z)_+$ as α goes to infinity. Since the RSVM has already reduced the model complexity via using a much smaller rectangular kernel matrix (corresponding to using less support vectors in constructing decision boundaries), we will suggest *to use a larger weight parameter C in RSVM than in a conventional SVM.* The solution to the minimization problem (4) or (5) leads to a nonlinear separating surface of the form

$$\sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) - \gamma = 0. \tag{7}$$

In fact, the reduced set $\tilde{A}$ is not necessary to be a subset of training set [17]. The minimization problem (5) retains the strong convexity and differentiability properties in the space, $R^{\tilde{m}+1}$, of $(\tilde{v}, \gamma)$ for any arbitrary *rectangular* kernel. Hence we can apply the Newton-Armijo method [21] directly to solve (5) and the existence and uniqueness of the optimal solution are also guaranteed. Moreover, the computational complexity of solving problem (5) by the Newton-Armijo method is $O(\tilde{m}^3)$ while solving the nonlinear SSVM with the full square kernel is $O(m^3)$ [21]. Typically, $\tilde{m} \ll m$. The numerical test in [20] on the Adult dataset [3] shows that sample standard deviation of test set correctness for 50 replicate runs using $\tilde{A} \in R^{326\times123}$ out of $A \in R^{32562\times123}$ is less than 0.001. In fact, the smallness of the standard error can be used as a guidance to determining $\tilde{m}$.

In summary, the RSVM can be split into two parts. First, it selects a small random subset $\{K(\tilde{A}_1, \cdot), K(\tilde{A}_2, \cdot), \cdots, K(\tilde{A}_{\tilde{m}}, \cdot)\}$ from the full-data bases $\{K(A_i, \cdot)\}_{i=1}^m$

for building the separating surface prior to training. While the conventional SVMs use a set of support vectors which are determined after training for building the surface. When projected onto the separating surface, the full-data bases are likely highly correlated with possibly heavy overlaps, which makes room for model reduction. Secondly, the RSVM determines the best coefficients of the selected kernel functions by solving the unconstrained minimization problem (4) or (5) using the entire dataset so that the surface will adapt to the whole data. Hence, even the RSVM uses only a small portion of kernel bases, it can still keep most of the relevant pattern information given by the entire training set. We will discuss this issue again from a low-rank approximation point of view in Section 4.

3 Reduced set mixture models and Monte-Carlo sampling for kernel bases

The nonlinear SVM uses a full-kernel representation for the discriminant function:

$$f(x) = \sum_{i=1}^{m} v_i K(A_i, x) - \gamma. \tag{8}$$

It is a linear combination of basis functions, $\{1\} \cup \{K(A_i, \cdot)\}_{i=1}^{m}$. The coefficient v_i is determined by solving a quadratic programming problem and the data points with corresponding *nonzero* coefficient v_i are called support vectors. It is often desirable to have less number of support vectors. The reduced set approach cuts down the model complexity and fits a reduced model:

$$f(x) = \sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) - \gamma. \tag{9}$$

We call $\{K(A_i, \cdot)\}_{i=1}^{m}$ the set of full-data bases and $\{K(\tilde{A}_i, \cdot)\}_{i=1}^{\tilde{m}}$ a reduced set.

Concerning the choice of a reduced set prior to the training process, a simple way is the uniform random subset used in the RSVM [20]. It randomly selects a small portion of basis functions from the full set to generate the reduced model, while it fits the entire dataset. Each candidate basis in the full set has equal chance of being selected. This uniform random selection is simple and straightforward without resorting to any search algorithm for optimal bases. There have been experimental results showing its ability for modeling classification surfaces [14, 20, 23] as well as regression surfaces [18]. In this article, we will mainly focus on its statistical properties and theory.

Before proceed any further, we define some more notations and state some conditions necessary for later statistical analysis and theory.

- **Definition 1 (Training design for inputs)** *Let ξ_T be a probability measure on input space $\mathcal{X} \subset R^n$. Assume that training inputs follow this probability distribution ξ_T. In statistical term, ξ_T is called a training design for input measurements.*

  The training inputs x^i, for $i = 1, \ldots, m$, are assumed i.i.d. realizations from ξ_T. After observing the training inputs, a discrete empirical version of the training design is given by $\xi_{T_m}(x) = m^{-1} \sum_{i=1}^{m} \delta(x - x^i)$, where $\delta(\cdot)$ puts probability mass one at zero. In this article we do not distinguish between the generic training design and its empirical version, and use a unified notation ξ_T for both unless otherwise specified.

- Let $\mathcal{H}$ denote the reproducing kernel Hilbert space generated by the kernel $K(x', z)$. Let $\mathcal{D} := \{K(\cdot, z)\}_{z \in \mathcal{X}}$ denote a dictionary for $\mathcal{H}$.

- **Definition 2 (Sampling design for bases selection)** *Let ξ be a probability measure on $\mathcal{X}$. As will be seen later in (12), ξ is used as a Monte-Carlo sampling scheme for kernel bases from the dictionary $\mathcal{D}$ to construct the discriminant surface. We call this probability measure ξ a sampling design for bases selection.*

- Assume ξ_T and ξ are two equivalent measures, indicated by $\xi_T \equiv \xi$.[1] Denote the Radon-Nikodym derivative of ξ with respect to ξ_T by $p(x) := d\xi(x)/d\xi_T(x)$. Let $\mathcal{P}$ be the collection of all such probability measures:

$$\mathcal{P} := \{\xi : \text{probability measure on } \mathcal{X} \text{ satisfying } \xi \equiv \xi_T\}. \tag{10}$$

  Again, we will not distinguish between the generic sampling design and its empirical version and use a unified notation ξ.

- For convenience, in Theorem 1 and Corollary 1, a Gaussian kernel

$$K_\sigma(x', z) = (2\pi\sigma^2)^{-n/2} \exp\{-\|x - z\|^2/(2\sigma^2)\}$$

  is assumed. We often suppress the subscript σ in K_σ and let $K(x - z) := K_\sigma(x', z)$. The value of $K(x - z)$ represents the inner product of resulting vectors of x and z in the feature space after a nonlinear mapping implicated and defined by the Gaussian kernel. We can also interpret it as a measure of similarity between x and z. In other words, $K(A_i, A)$ records the similarity between A_i and all training inputs. In particular, if we use the Gaussian kernel in the RSVM, we can interpret the RSVM as an instance-based learning algorithms [31]. The reduced kernel matrix arranges only the similarity between reduced set and the entire training dataset. In contrast to the k-nearest neighbor algorithm using the simple voting strategy, the RSVM uses the weighted voting strategy, where weights and threshold are determined by a training procedure. That is, if the weighted sum, $\tilde{v}'K(\tilde{A}, x)$, is greater than a threshold, γ, the point x is classified as a positive example.

[1] That is, ξ is absolutely continuous with respect to ξ_T and conversely, ξ_T is also absolutely continuous with respect to ξ. The equivalence of two measures insures the existence of the Radon-Nikodym derivatives $d\xi(x)/d\xi_T(x)$ and $d\xi_T(x)/d\xi(x)$.

Theorem 1 and Corollary 1 can easily extend to translation-type kernels, i.e., kernels of the form $K(x', z) = K(x - z)$.

As seen in (8) and (9), the underlying discriminant function $f(x)$ is modeled as a mixture of kernels plus an offset term. We assume the following modeling for the discriminant function via the mixing distribution ξ_T:

$$f(x) = \int K(x', z)v(z)d\xi_T(z) - \gamma, \tag{11}$$

where $v(z)$ is a coefficient function assumed to satisfy $\int v^2(z)d\xi_T(z) < \infty$. By re-expressing the above mixture via an arbitrary mixing distribution $\xi \in \mathcal{P}$, we have

$$f(x) = \int K(x', z)v(z)\frac{d\xi_T(z)}{d\xi(z)}\ d\xi(z) - \gamma := \int f(x, z)\ d\xi(z) - \gamma, \tag{12}$$

where $f(x, Z) := K(x, Z)v(Z)/p(Z)$ with $Z \in \mathcal{X}$ following a probability distribution ξ. Bases, sampled via the random mechanism $K(x, Z)$ with $Z \sim \xi$, are used to build a reduced model. The sum of certain realizations of $f(x, Z)$ is a Monte-Carlo sampling approximation to the presumably true underlying discriminant function $f(x)$ in (11). A natural and simple way to evaluate such a Monte-Carlo approximation is through its integrated variance [24]:

$$\int Var_\xi\{f(x, Z)\}dx = \int\int K^2(x', z)v^2(z)/p^2(z)\ d\xi(z)dx - \int (f(x) + \gamma)^2 dx. \tag{13}$$

Thus the above quantity is used to assess the model robustness. The smaller it is, the better stability (less variability) the model built from ξ will retain. Therefore, we aim to find a good sampling design ξ with as small integrated model variance as possible. As the quantity $\int (f(x) + \gamma)^2 dx$ does not involve ξ, we may drop it from the expression (13) in the minimization step and use the following measure of model variation.

Definition 3 (Model variation measure) *We use the following measure to assess the model variation due to the sampling design ξ:*

$$\begin{aligned} V(\xi) &:= \int\int K^2(x', z) v^2(z)/p^2(z) \; d\xi(z) dx \\ &= \int\int K^2(x', z) v^2(z)/p(z) \; d\xi_T(z) dx. \end{aligned} \tag{14}$$

(If $K^2(x', z)v^2(z)/p(z)$ is not integrable, set $V(\xi) = \infty$.)

By Lemma 1 in the Appendix, it is easy to get the optimal design which minimizes $V(\xi)$ over the set $\mathcal{P}$. The resulting optimal sampling design has its pdf with respect to ξ_T given by

$$d\xi_{opt}/d\xi_T = p(z) \;\propto\; |v(z)| \left(\int K^2(x', z) dx\right)^{1/2} \;\propto\; |v(z)|,$$

if $K(x', z)$ is a translation-type kernel, e.g., the Gaussian kernel case.[2]

Theorem 1 (Optimal sampling design) *Assume that the kernel employed is of translation-type. The ideal optimal sampling design for bases selection is given by ξ_{opt} with pdf $d\xi_{opt}(z)/d\xi_T(z) = p(z) \propto |v(z)|$.*

The coefficient function $v(z)$ is not known and has yet to be estimated in the training process. Here we use a constant as a reference function to replace $|v(z)|$ and the resulting $p(z)$ is simply a uniform pdf with respect to ξ_T. There are two main reasons of using a constant reference function for $|v(z)|$. One is to indicate no prior preference or information on $v(z)$ prior to training. The other is to reflect the intrinsic mechanism in SVM that tends to minimize the 2-norm, $\int v^2(z) d\xi_T(z)$, of $|v(z)|$. For an arbitrary fixed scale $\int |v(z)| d\xi_T(z) = c$, say $c = 1$, the 2-norm of $|v(z)|$ is minimized when $|v(z)|$ is a constant function.

[2] For Gaussian kernel on a compact set $\mathcal{X}$, a constant $\int K^2(x', z) dx$ is not true. However, for z away from the boundary, the function $\int K^2(x', z) dx$ is near a constant. For z near the boundary, some boundary correction should be made prior to forming the mixture model and also prior to training. See Appendix for more detailed discussion.

Corollary 1 *Prior to training process, a constant function is used as a reference for the coefficient function $|v(z)|$. Then the optimal sampling design for bases selection is given by $p(z) = constant$, that is, $\xi_{opt} = \xi_T$. This justifies the uniform sampling scheme (on training data) used in the original RSVM of Lee and Mangasarian [20].*

Remark 1 *The result, $p(z) \propto |v(z)|$, in Theorem 1 says that we should sample more kernel bases at points with large coefficients. These large coefficient points are potential support vectors. However, prior to training, we do not know where these potential support vectors are. Though sequential adaptive algorithms for optimal bases selection are available [35, 19, 5], they are more time-consuming and require some search algorithms, which are not the scope of this article. Here we simply point out that there is a simple and economic alternative, namely, the uniform random subset approach. Though, a posteriori this uniform random sampling scheme is not the optimal one, the true a posteriori optimal one is $p(z) \propto |v(z)|$. However, as prior to training, we do not have information on which data points have better potential to be support vectors than others.*

Another interpretation for the constant $|v(z)|$ is as follows. Suppose that the training inputs come from a mixture of two distributions. With probability π_+ the training inputs are from the positive group having pdf $f_+(x)$ and with probability $\pi_- = 1 - \pi_+$ the training inputs are from the negative group having pdf $f_-(x)$. The Rosenblatt-Parzen [34, 32] kernel estimators for $f_+(x)$ and $f_-(x)$ are

$$\hat{f}_+(x) = \frac{1}{n_+} \sum_{i \in A_+} K(A_i, x), \quad \hat{f}_-(x) = \frac{1}{n_-} \sum_{i \in A_-} K(A_i, x).$$

The prior probabilities of group assignment can be estimated by frequency counts:

$$\hat{\pi}_+ = \frac{n_+}{n}, \quad \hat{\pi}_- = \frac{n_-}{n}.$$

An ad hoc decision boundary can be given by (comparing two pdfs, adding an offset *term to balance what might be biased)*

$$\hat{\pi}_+ \hat{f}_+(x) - \hat{\pi}_- \hat{f}_-(x) - \gamma = 0,$$

which corresponds to a constant $|v(z)|$.

Remark 2 *A friendly reminder to readers who are interested in implementing the reduced set approach on top of their own kernel-based learning algorithms is that a stratified random sampling should be taken, especially when the numbers of classes are quite a few. That is, the uniform random subset should be done within each class and then be combined together. The stratified sampling is a "must" for multiple classification in order to reduce the variance due to Monte-Carlo sampling, especially for problems with large numbers of classes [24].*

This random subset approach can drastically cut down the model complexity, while the sampling design helps to guide the bases selection in terms of minimal model variation (14). However, we remind the readers that the quantity $V(\xi)$ does not account for variance incurred in parameter estimation, but only for the model variation caused by bases sampling. The resulting optimal sampling design in Corollary 1 is a *uniform design* with respect to ξ_T, which seeks basis points uniformly distributed *over training points*. The use of uniform design has been popular since 1980. See articles [12] and [13] for a nice survey of theory and application on uniform design. The uniform design is a space filling design and it seeks to obtain maximal model robustness.

In the original RSVM article [20], it has experimental comparison of the RSVM vs. the full-kernel SSVM. As a supplement, here we add some further comparison with the LIBSVM [6]. We run numerical tests on nine datasets (five for classification and four for regression) which are available on [3, 9, 10]. Results in Table 1 below show that models generated by the RSVM consistently have slightly smaller testing errors. All training and testing errors are the ten-fold cross-validation average. The parameters used in all support vector machines are determined by a 2-dimensional mesh tuning procedure.

Dataset	RSVM			SSVM			LIBSVM		
	Train Error	Test Error	CPU Sec.	Train Error	Test Error	CPU Sec.	Train Error	Test Error	CPU Sec.
Ionosphere	0.0369	0.0411	0.313	0.0136	0.0471	2.578	0.0041	0.0471	1.016
(351x34)	Reduced set size: 36						Number of SVs: 183		
Pima	0.2276	0.2211	0.433	0.2218	0.2237	199.166	0.2231	0.2237	1.688
(768x8)	Reduced set size: 39						Number of SVs: 394		
Image	0.0127	0.0260	3.031	0.0015	0.0294	1526.700	0.0140	0.0294	4.750
(2310x18)	Reduced set size: 116						Number of SVs: 344		
Mushroom	0.1049	0.1063	287.584	N/A	N/A	N/A	0.1047	0.1073	736.891
(8124x22)	Reduced set size: 400						Number of SVs: 1757		
Tree	0.0802	0.0919	672.390	N/A	N/A	N/A	0.0902	0.0939	507.391
(12392x18)	Reduced set size: 620						Number of SVs: 2676		
Comp-Activ	0.0281	0.0317	1.458	0.0293	0.0313	28.143	0.0287	0.0320	8.593
(1000x21)	Reduced set size: 150						Number of SVs: 862		
Kin-fh	0.1313	0.1354	1.586	0.1300	0.1357	26.161	0.1227	0.1359	7.055
(1000x32)	Reduced set size: 150						Number of SVs: 654		
Comp-Activ	0.0270	0.0285	87.045	N/A	N/A	N/A	0.0266	0.0285	436.027
(8192x21)	Reduced set size: 410						Number of SVs: 6901		
Kin-fh	0.1285	0.1322	62.954	N/A	N/A	N/A	0.1275	0.1320	699.251
(8192x32)	Reduced set size: 410						Number of SVs: 5277		

Table 1: **Numerical comparisons of RSVM, SSVM and LIBSVM. The first five datasets are binary classification problems and the rest are regression problems. We used Gaussian kernel in all numerical tests."N/A" indicates the result is not available because of computational difficulties.**

In all numerical tests above, the size of reduced set is smaller than the number of support vectors in LIBSVM. This indicates that the RSVM uses fewer number of kernel bases to generate the discriminant function. The RSVM tends to have a simpler model and need a smaller number of function evaluations when predicts a new unlabeled data point. This is an advantage in the testing phase of learning tasks. Although in many cases the RSVM has a slightly bigger training error, it does not scarify any predicting accuracy on all test datasets. The basic RSVM phenomenon is that by using a much simpler model *we sacrifice a little bit of bias to reduce the variance of the predicted values, and hence may improve the overall prediction accuracy* (quote from [16]).

4 Spectral analysis of reduced kernels versus the full kernel

In this section, we attempt to explain why the RSVM can perform well successful from a spectral analysis point of view. In order to avoid dealing with the huge and dense full kernel matrix in SVM, a low-rank approximation to the full kernel matrix which is known as the Nyström approximation has been proposed in many sophisticated ways [35, 44]. That is,

$$K(A,A') \approx K(A,\tilde{A}')K(\tilde{A},\tilde{A}')^{-1}K(\tilde{A},A') = \tilde{K}. \tag{15}$$

We denote the Nyström approximation of $K(A,A')$ by $\tilde{K}$ in the rest of this article. Applying this approximation, for a vector $v \in R^m$,

$$K(A,A')v \approx K(A,\tilde{A}')K(\tilde{A},\tilde{A}')^{-1}K(\tilde{A},A')v = K(A,\tilde{A}')\tilde{v}, \tag{16}$$

where $\tilde{v} = K(\tilde{A},\tilde{A})^{-1}K(\tilde{A},A')v$. In the RSVM, $\tilde{v}$ is directly determined by fitting the entire dataset and the Nyström approximation is generated by the uniform random

Dataset & Size (Train, Reduced)	Largest Eigen-v. of $K(A, A')$	Largest Eigen-v. of $\tilde{K}$	Max-diff of Eigenvalue	Rel-diff of Trace
Ionosphere (351,36)	162.649	162.096	1.0315	0.210
Pima (768, 39)	606.547	606.536	0.387	0.0026
Image (2310,116)	1303.2	1303.1	1.496	0.0210
Comp-Act(1000) (1000, 150)	961.364	961.359	0.040	0.00034
Kin-fh(1000) (1000, 150)	953.160	953.154	0.008	0.00087

Table 2: **Spectral comparison of full kernel $K(A, A')$ and the Nyström approximation $\tilde{K}$. The relative difference of the trace is defined as $\frac{\textbf{trace}(K-\tilde{K})}{\textbf{trace}(K)}$.**

subset.

We measure the discrepancy between the full kernel and the uniform-random-subset Nyström approximation via a few quantities like the maximum difference of eigenvalues and the relative difference of the trace of full kernel $K(A, A')$ and $\tilde{K}$ (denoted as Max-diff of Eigenvalue and Rel-diff of Trace, respectively in Table 2) on five real datasets, which have manageable sizes of full kernel eigenvalues and eigenvectors. We summarize the results in Table 2. In order to have a better understanding of the differences of their spectral behaviors, we plot four figures for each dataset. Each figure has four subfigures. In subfigure (a), it shows the difference between each pair of eigenvalues. In subfigure (b) and (c), we plot the eigenvalues of the full and the approximate kernels. We split them into two parts and skip the largest eigenvalue because the different scales of eigenvalues. We try to provide more details about them. In subfigure (d), we plot the squared root of eigenvalue of the full kernel times the two norm of the difference of each pair of eigenvectors. That is, we plot $(k, \sqrt{\lambda_k} \cdot \|e_k - \tilde{e_k}\|_2)$, where λ_k is the kth eigenvalue of $K(A, A')$ and e_k is the corresponding eigenvector. $\tilde{e_k}$ denotes the kth eigenvector of the Nyström approximation by uniform random subset. Note that $\sqrt{\lambda_k} \cdot \|e_k - \tilde{e_k}\|_2 = \sqrt{2\lambda_k(1 - \cos\theta_k)}$, where θ_k is the angle between e_k and $\tilde{e}_k$. These angles $\{\theta_1, \theta_2, \theta_3, \ldots\}$ are known as the leading canonical angles between K and $\tilde{K}$ [7].

We find from these figures that the quality of the approximation will depend on the rate of decay of the eigenvalues of the full kernel matrix. Also, the numerical simulations indicate that the reduced kernel generated by the uniform random selection scheme retains most of the relevant information. These observations might give an explanation why the RSVM can provide a good discriminant and regression function estimation in supervised learning tasks.

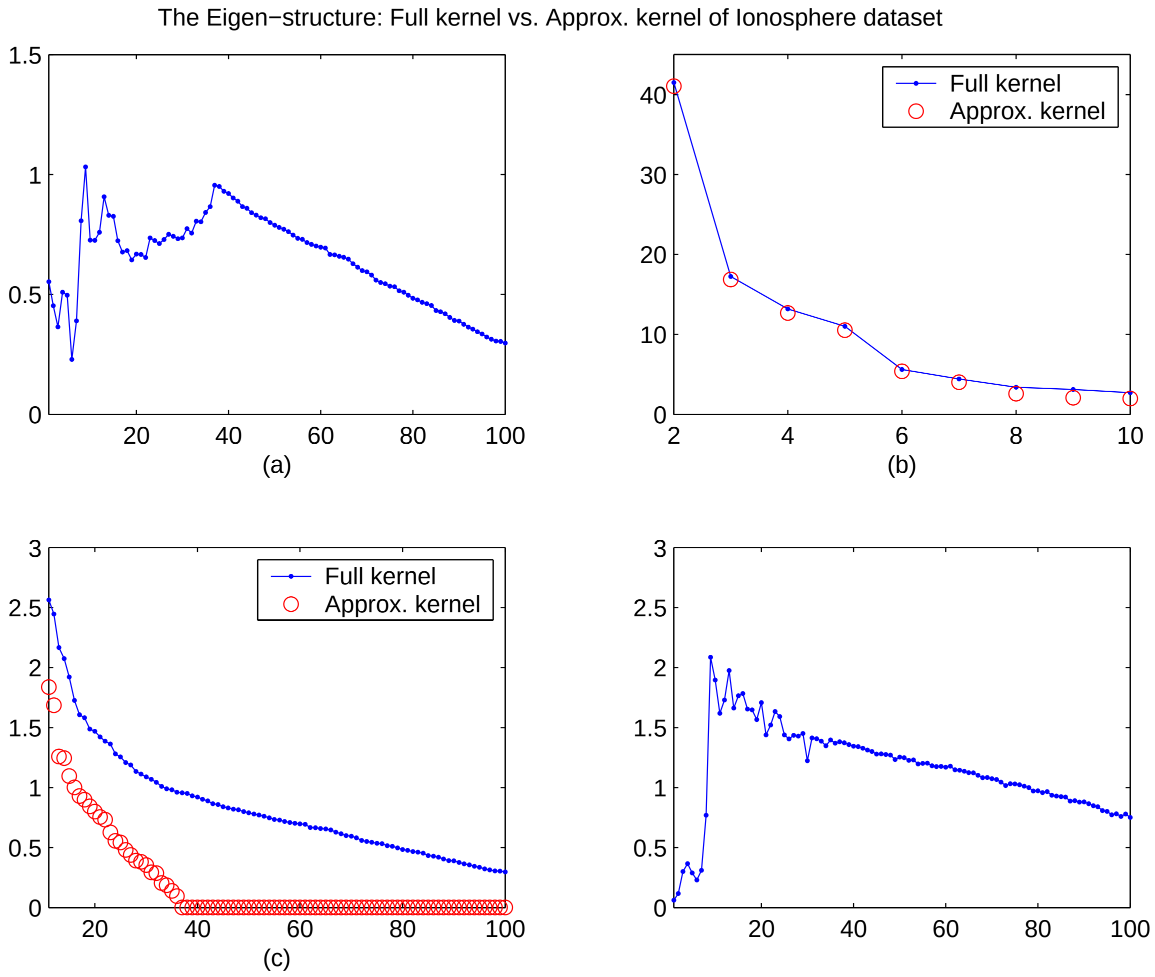

Figure 1: The spectral analysis of Ionosphere dataset

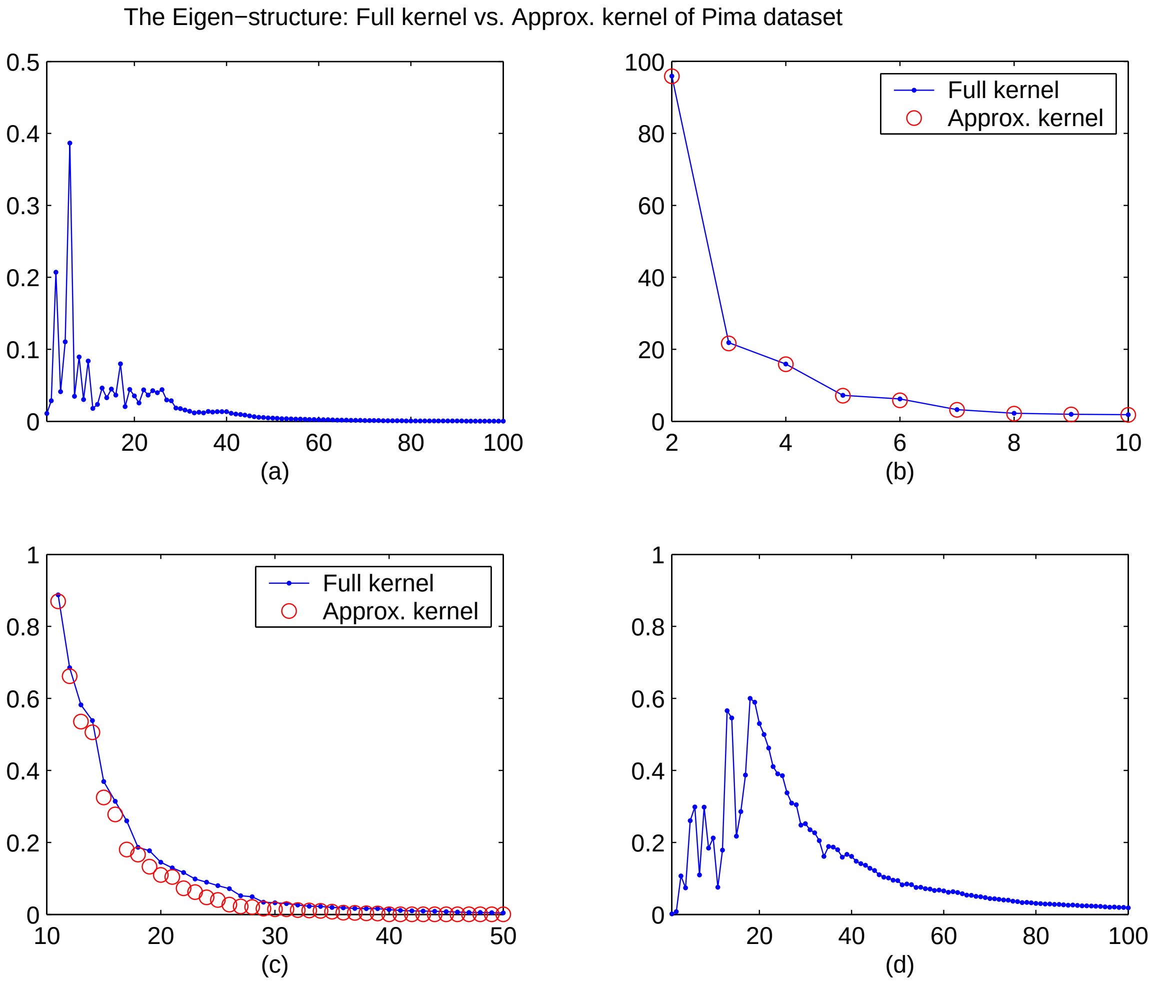

Figure 2: The spectral analysis of Pima dataset

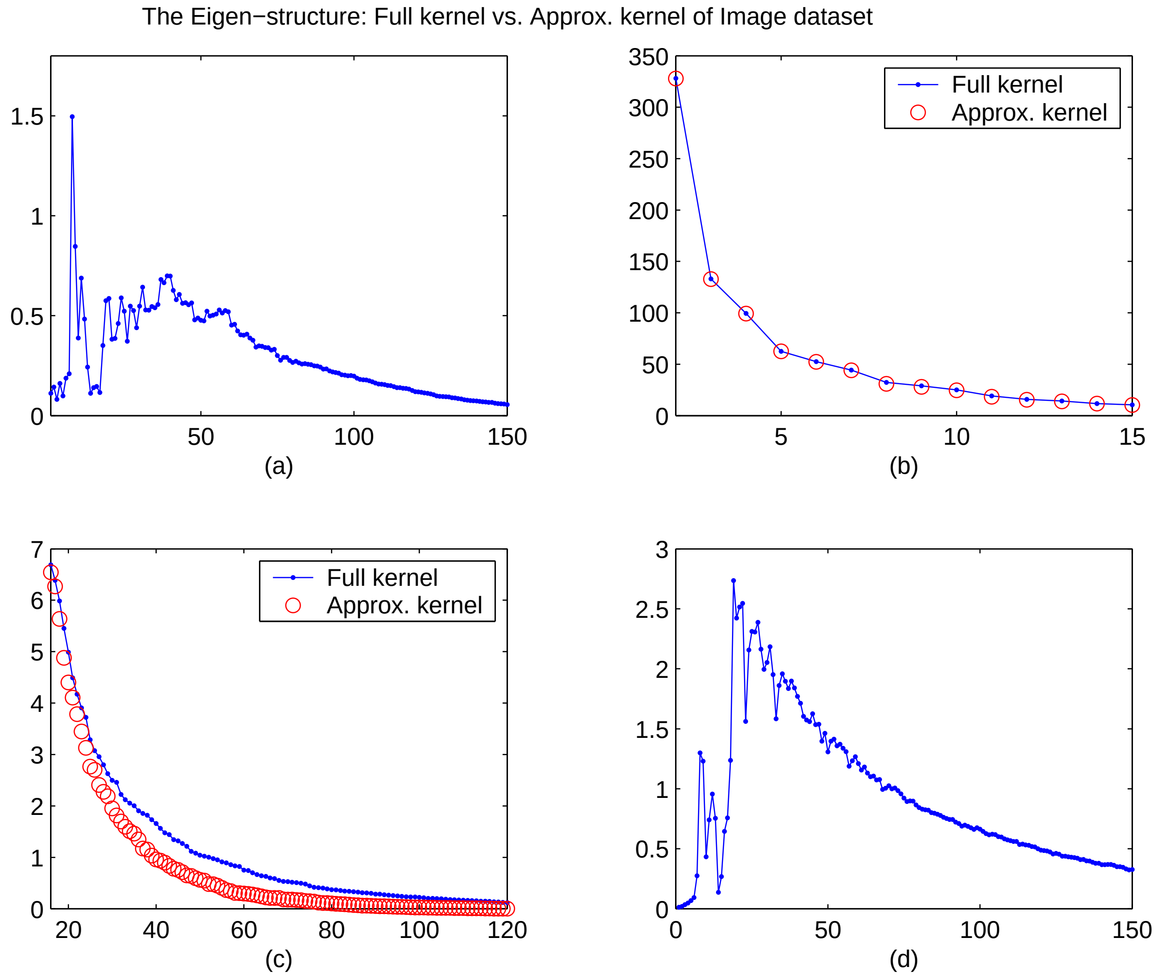

Figure 3: The spectral analysis of Image dataset

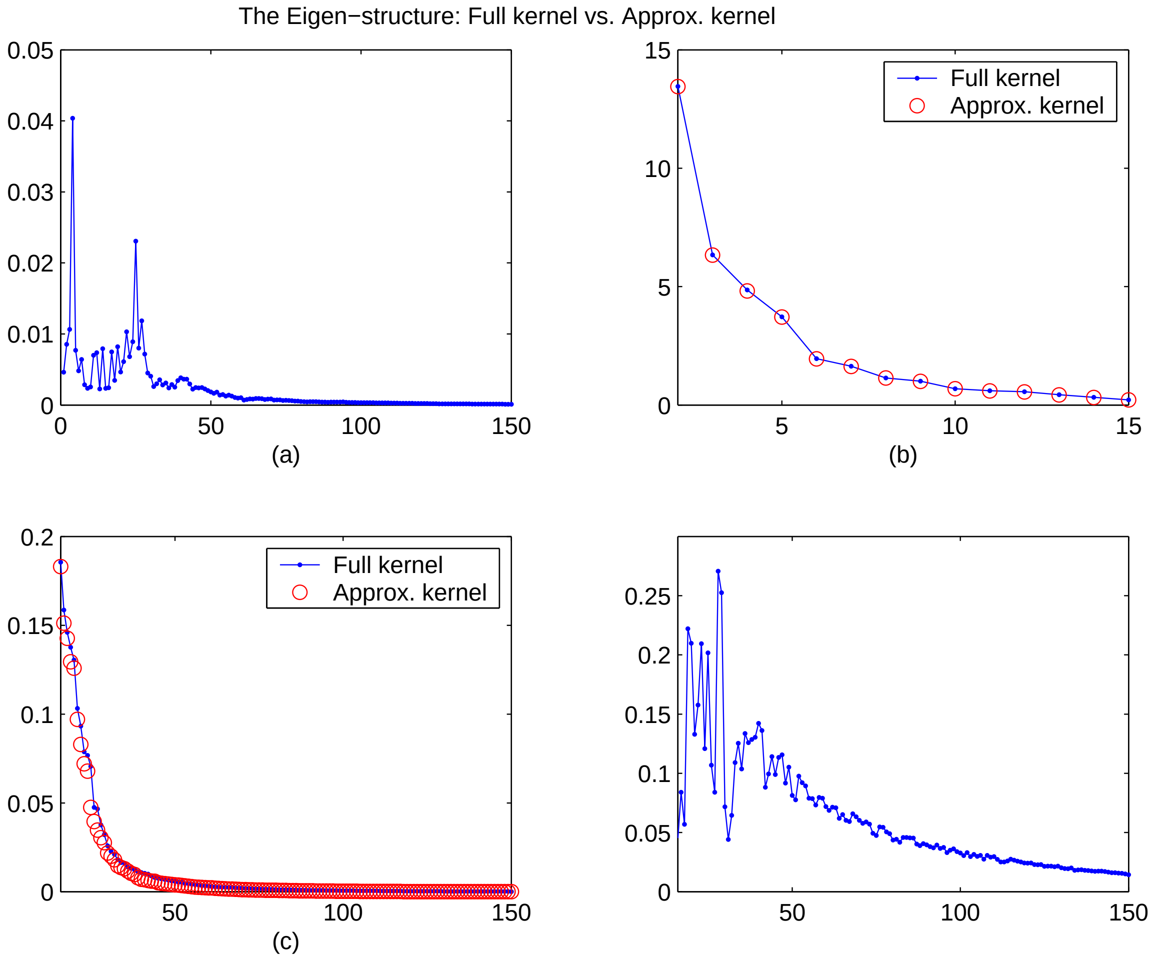

Figure 4: The spectral analysis of CompActiv (1000) dataset

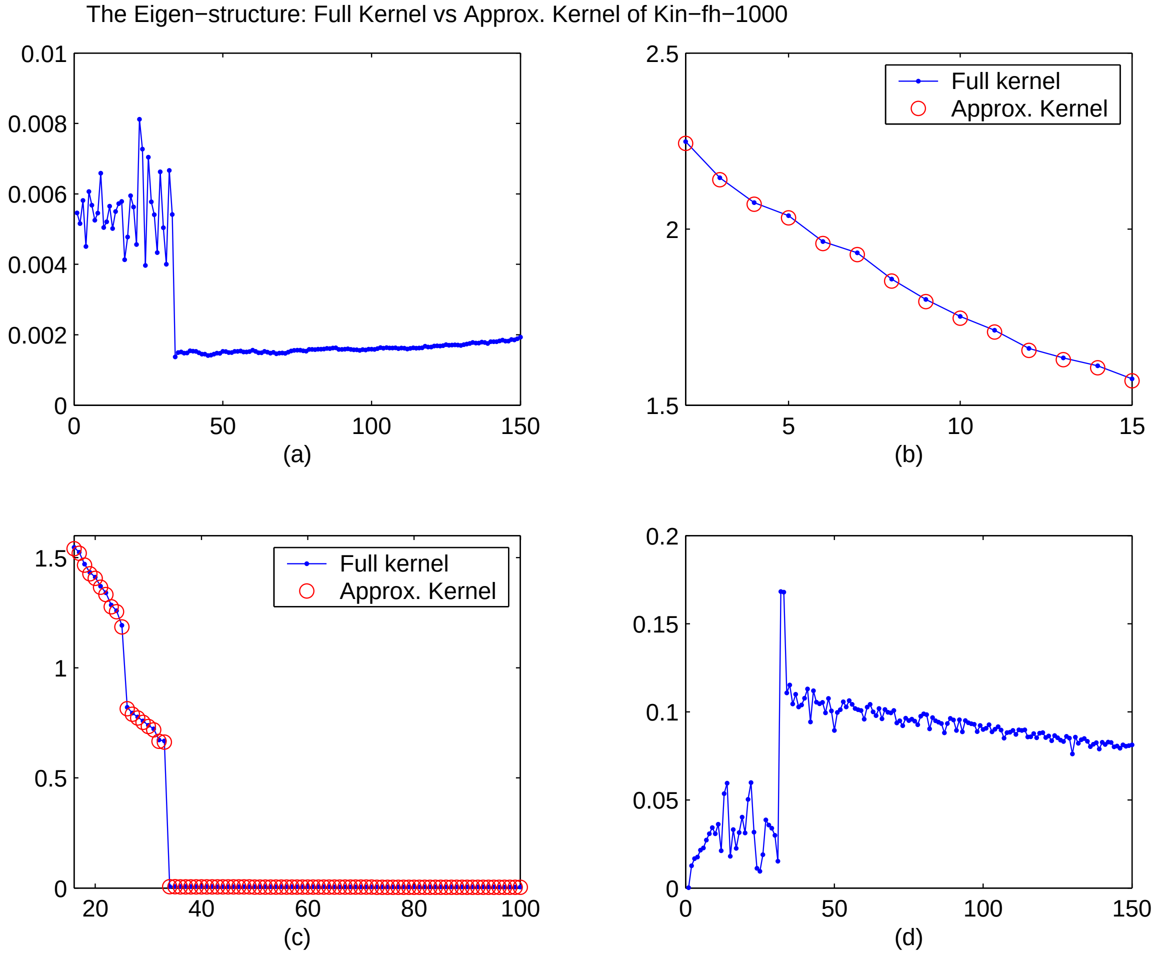

Figure 5: The spectral analysis of Kin-fh (1000) dataset

5 Minimaxity and maximinity

In this section we will show that the uniform sampling design in Corollary 1 possesses some other robustness properties, namely, the minimaxity and the maximinity. The ideas are taken from robust designs [41, 42, 43, 22, 1, 2] and customized into the context of the RSVM (4). Recall the full and reduced models:

$$\text{full model :} \qquad f(x) = \sum_{i=1}^{m} v_i K(A_i, x) - \gamma, \tag{17}$$

$$\text{reduced model :} \qquad f(x) = \sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) - \gamma. \tag{18}$$

Using kernel K corresponds to mapping the data into a feature Hilbert space with a map: $\Phi : \mathcal{X} \to \mathcal{U}$.[3] In the feature space $\mathcal{U}$, the normal vector of the SVM separating hyperplane, $w'u - \gamma = 0$, can be expanded in terms of support vectors. The full model has the following full expansion for the normal vector

$$\text{full pre-image expansion :} \;\; w = \sum_{i=1}^{m} v_i \Phi(A_i), \tag{19}$$

while the reduced model has the reduced expansion

$$\text{reduced pre-image expansion :} \;\; w = \sum_{i=1}^{\tilde{m}} \tilde{v}_i \Phi(\tilde{A}_i). \tag{20}$$

Expansions given by (19) and (20) are called pre-image expansions [36]. We now try to measure the error induced by the reduced-set approximation. For a given vector $v \in R^m$, we consider the minimization problem

$$\min_{\tilde{v} \in R^{\tilde{m}}} \| \sum_{i=1}^{m} v_i K(A_i, x) - \sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) \|^2_{\mathcal{H}(d\xi_T)}. \tag{21}$$

[3] Here we assume the kernel spectral is done with respect to the measure ξ_T, i.e., $K(x, z) = \sum_i \lambda_i \phi_i(x) \phi_i(z)$ with $\int \phi_i^2(x) d\xi_T = 1$ and $\int \phi_i(x)\phi_j(x) d\xi_T = 0$ for $i \neq j$. The corresponding Hilbert space is denoted by $\mathcal{H}(d\xi_T)$.

By reproducing property,

$$\| \sum_{i=1}^{m} v_i K(A_i, x) - \sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) \|^2_{\mathcal{H}(d\xi_T)} = v' K(A, A') v - 2v' K(A, \tilde{A}') \tilde{v} + \tilde{v}' K(\tilde{A}, \tilde{A}') \tilde{v}.$$

Then the vector $\tilde{v}^* = K(\tilde{A}, \tilde{A}')^{-1} K(\tilde{A}, A') v \in R^{\tilde{m}}$ solves problem (21). Hence, the errors induced by the reduced-set approximation (18) and (20) are given respectively by

$$\min_{\tilde{v} \in R^{\tilde{m}}} \| \sum_{i=1}^{m} v_i K(A_i, x) - \sum_{i=1}^{\tilde{m}} \tilde{v}_i K(\tilde{A}_i, x) \|^2_{\mathcal{H}(d\xi_T)} = v'(K - \tilde{K}) v,$$

where $K = K(A, A')$ and $\tilde{K}$ is defined as in (15), and similarly

$$\min_{\tilde{v} \in R^{\tilde{m}}} \| \sum_{i=1}^{m} v_i \Phi(A_i) - \sum_{i=1}^{\tilde{m}} \tilde{v}_i \Phi(\tilde{A}_i) \|^2_{\mathcal{U}} = v'(K - \tilde{K}) v.$$

They lead to the same quantity.

Below we define some more notations. All the ξ's are assumed in the class $\mathcal{P}$ given by (10). The full data A is assumed fixed and the reduced set size $\tilde{m}$ is also assumed fixed.

- $\mathcal{F}(A) :=$ linear span$\{K(A_i, \cdot)\}_{i=1}^{m}$, which is the full model given training inputs A. $\mathcal{R}(\tilde{A}) :=$ linear span$\{K(\tilde{A}_i, \cdot)\}_{i=1}^{\tilde{m}}$, which is a reduced model by the subset $\tilde{A} \subset A$. We will suppress the set notation and simply use $\mathcal{F}$ and $\mathcal{R}$, if no confusion.

- $\tilde{K}_\xi := \int_{\mathcal{X}^{\tilde{m}}} K(A, \tilde{A}') K(\tilde{A}, \tilde{A}')^{-1} K(\tilde{A}, A') d\xi(\tilde{A})$, where $d\xi(\tilde{A}) = d\xi(\tilde{A}_1) \cdots d\xi(\tilde{A}_{\tilde{m}})$. $\tilde{K}_\xi$ is an $m \times m$ matrix representing the average effect of low-rank approximation by ξ-drawn subset of size $\tilde{m}$. Note that when we use $\tilde{K}_{\xi_T}$, it also means a low-rank approximation by ξ_T-drawn subset of size $\tilde{m}$, but rather not means the full kernel.

- $\mathcal{F}_\eta^- := \left\{ f(x) = v' K(A, x) \in \mathcal{F} : v'(K - \tilde{K}_{\xi_T}) v \le \eta^2 \right\}$. It specifies a region in $\mathcal{F}$. After removing the average effect explained by ξ-drawn reduced model, we put a bound on what is left unexplained. Later, when we compare the full model and a ξ-drawn reduced model via a certain bias measure, this bound is used to prevent the bias measure from escaping to infinity.

- $\mathcal{F}_\eta^+ := \left\{ f(x) = v'K(A,x) : v'(K - \tilde{K}_{\xi_T})v \geq \eta^2 \right\}$. It specifies a region in $\mathcal{F}$. Again, after removing the average effect explained by reduced model, what is left unexplained has at least $\mathcal{H}(d\xi_T)$-norm of size η. Later, in a lack of fit test (i.e., checking if the ξ-drawn reduced model provides an adequate approximation for functions in $\mathcal{F}_\eta^+$) also via a bias measure, the η-distance bounds the bias measure away from zero. In other words, the η-distance guarantees a minimum (> 0) distinguishability between the ξ-drawn reduced model and $\mathcal{F}_\eta^+$.

- For $f(x) = v'K(A,x) \in \mathcal{F}$, define $\mathcal{B}(f, \tilde{A}) := \|f(x) - P_{\mathcal{R}} f\|^2_{\mathcal{H}(d\xi_T)}$, where $P_{\mathcal{R}} f(x)$ is the projection of f onto $\mathcal{R}$ and is given by

  $$P_{\mathcal{R}} f(x) = v'K(A, \tilde{A}')K(\tilde{A}, \tilde{A}')^{-1}K(\tilde{A}, x).$$

  It is derived by solving a minimization problem similar to (21). This quantity $\mathcal{B}(f, \tilde{A})$ reflects the bias measure induced by approximating $f(x) = v'K(A,x)$ using the reduced model by subset $\tilde{A}$.

- $\mathcal{B}(f, \xi) := \int_{\mathcal{X}^{\tilde{m}}} \mathcal{B}(f, \tilde{A}) d\xi(\tilde{A})$. It is straightforward to show that $\mathcal{B}(f, \xi) = v'(K - \tilde{K}_\xi)v$. The quantity $\mathcal{B}(f, \xi)$ reflects an average bias measure induced by approximating $f(x) = v'K(A,x)$ using the ξ-drawn subset.

There are other means of defining alternative classes $\mathcal{F}_\eta^-$ and $\mathcal{F}_\eta^+$ and bias measure $\mathcal{B}(f, \xi)$, see, for instance, [41, 42, 43, 22, 1, 2]. The minimaxity and maximinity discussed below depend on the particular way of specifying alternative classes and also on the definition of bias measure.

Theorem 2 (Minimaxity) *The minimax sampling design which minimizes the maximum bias $\mathcal{B}(f,\xi)$ for $f \in \mathcal{F}_\eta^-$ is achieved by $\xi = \xi_T$, i.e.,*

$$\sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f,\xi_T) = \inf_{\xi\in\mathcal{P}} \sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f,\xi). \tag{22}$$

Equivalently, the uniform design with respect to ξ_T is the minimax sampling design.

Theorem 3 (Maximinity) *For $f \in \mathcal{F}_\eta^+$, the maximin sampling design for $\mathcal{B}(f,\xi)$ is achieved by $\xi = \xi_T$, i.e.,*

$$\inf_{f\in\mathcal{F}_\eta^+} \mathcal{B}(f,\xi_T) = \sup_{\xi\in\mathcal{P}} \inf_{f\in\mathcal{F}_\eta^+} \mathcal{B}(f,\xi). \tag{23}$$

Both the minimaxity and maximinity are some qualities that are robust against the worst case scenarios. Theorem 2 says that the worst expected L_2 error, introduced by the approximation $Kv \approx \tilde{K}_\xi v$, is minimized by the Nyström approximation using uniform random subset. In judging if a reduced set approximation $Kv \approx \tilde{K}_\xi v$ is adequate or not, Theorem 3 says that, for a least distinguishable $f \in \mathcal{F}_\eta^+$, the Nyström approximation by uniform random subset has the best testing power for a lack of fit (i.e., the best ability to detect an inadequacy for a least distinguishable f). Theorems 2 and 3 tell us that the uniform design (a space filling design) is robust against the worst case. All the statistical properties discussed in this article, including the optimal bases sampling design, the minimaxity and maximinity, are all prior to training phenomena. Though the worst case scenarios or the prior to training phenomena might not sound practical in real experiments, the spectral analysis for real datasets in Section 4 further provides some practical view for the method. Experimental results show that the uniform random subset often provides a reasonably well approximation to the full kernel in ordinary cases.

6 Applicability to other kernel-based algorithms

For many other kernel-based algorithms, one has to face the same problems (P1) and (P2) stated earlier. Several authors have suggested the use of low-rank approximations to the full kernel matrix for very large problems [20, 35, 44]. These low-rank approximations all have used a thin rectangular matrix $K_{m\tilde{m}}$ consisting of a subset of $\tilde{m}(\ll m)$ columns drawn from the full kernel matrix K. Lee and Mangasarian [20], Williams and Seeger [44] suggest to pick the subset randomly and uniformly over all columns. Smola and Schölkopf [35] consider finding an optimal subset. Since finding an optimal subset is a combinatorial problem involving $C(m, \tilde{m})$ possibilities and an exhaustive search is too expensive to carry out, they use a probabilistic speedup algorithm. The trick they use is: First draw a random subset of a fixed size and then pick the best basis (column) from this set. The search goes on till the stopping criterion is reached. For either Lee and Mangasarian's, or Williams and Seeger's random subset, or Smola and Schölkopf's *a priori* random subset at each iteration in search for an optimal basis within, our theorems and optimality properties are valid referring to the random mechanism of the subset selection.

For problems solved in the primal space, e.g., the proximal SVM [14], the least-square SVM [38, 39], the kernel Fisher discriminant [30, 29], the random subset method works by replacing the full kernel with the reduced kernel $K(A, \tilde{A}')$ and also by cutting down the corresponding number of parameters. For problems solved in the dual space, we form the approximate kernel matrix $\tilde{K} = K(A, \tilde{A}')K(\tilde{A}, \tilde{A}')^{-1}K(\tilde{A}, A')$ to replace the full matrix K. For instance, a reduced set approach for the Lagrangian SVM [28] has been implemented in [23]. To obtain the primal solution $\tilde{v}$, we should solve $K(A, \tilde{A}')\tilde{v} = \tilde{K}v$, where v is the dual solution.

Though we have discussed the statistical properties of the uniform random subset approach mainly on the context of the reduced SVM, the use of a uniform random subset is not limited to the RSVM. The uniform random subset approach can act as a supplemental-algorithm on top of a basic optimization algorithm, wherein the actual optimization takes place on the subset-approximated data. The statistical properties discussed above are still valid.

7 Conclusion

We study the RSVM from a robust design point of view and measure the discrepancy between the full kernel and the uniform-random-subset generated Nyström approximation in order to provide a better understanding of the reduced kernel technique. Our main results center on two major themes. One is on the robustness of the random subset mixture model and the other is on the spectral analysis of the reduced kernel. We show that uniformly and randomly selecting the reduced set of RSVM is the optimal sampling strategy for recruiting kernel bases in the sense of minimizing a model variation measure. We further provide two optimal properties for the RSVM, namely, the minimaxity and maximinity. As for a practical view of the RSVM in action, we have provided some spectral analysis. We compare the eigen-structures of the full kernels and the approximation kernels on five real datasets. The approximation kernels are generated by uniform random subsets. The discrepancies such as the differences between each pair of eigenvalues and eigenvectors, the relative difference of the trace are very small. These results indicate that the uniform random subset kernels can retain most of the relevant information for the learning tasks in the full kernel. We have tested the RSVM on nine datasets and compared the results to the conventional nonlinear SVM solved by the LIBSVM. Although in many cases the RSVM has a slightly bigger training error, it does not scarify any predicting accuracy on all test datasets. Furthermore, the reduced set size is smaller than the number of support vectors in conventional SVM results. Thus, the RSVM uses a simpler model and is more economic in predicting new unlabeled data. This is an advantage in the testing phase of learning tasks. The usage of uniform random subset is not limited to the RSVM. The statistical properties discussed remain valid for other kernel-based algorithms combined with a uniform random subset approximation.

8 Appendix

Lemma 1 *For a given nonnegative function $t(z)$, let $\mathcal{P}_t$ denote the collection of probability density functions $p(z) \in \mathcal{P}$, where $\mathcal{P}$ is as defined in (10), such that $\int_{\mathcal{X}}(t(z)/p(z))d\xi_T(z) < \infty$. (Here $0/0$ is defined to be zero.) Then the solution to the following optimization problem*

$$\arg\min_{p\in\mathcal{P}_t} \int_{\mathcal{X}} (t(z)/p(z))d\xi_T(z)$$

is given by $p(z) = c^{-1}\sqrt{t(z)}$, where $c = \int_{\mathcal{X}} \sqrt{t(z)}\, d\xi_T(z)$.

Proof: Since $\int_{\mathcal{X}}(t(z)/p(z))d\xi_T(z) < \infty$ and $\int_{\mathcal{X}} p(z)d\xi_T(z) = 1$, by Hölder inequality, we have

$$\int_{\mathcal{X}} (t(z)/p(z))\, d\xi_T(z) = \int_{\mathcal{X}} (t(z)/p(z))\, d\xi_T(z) \cdot \int_{\mathcal{X}} p(z)d\xi_T(z) \geq \left(\int_{\mathcal{X}} \sqrt{t(z)}d\xi_T(z)\right)^2 .$$

Equality holds if and only if, there exists some nonzero constant β such that

$$t(z)/p(z) = \beta p(z) \;\; a.e. \text{ with respect to } \xi_T.$$

Since $p(z)$ is a pdf, then equality holds if and only if $p(z) = c^{-1}\sqrt{t(z)}$. □

In the rest of proofs, we let $c_0^2 = \mathbf{1}'K(A, A')\mathbf{1}$ and $\rho_0 = \mathbf{1}'\tilde{K}_{\xi_T}\mathbf{1}/c_0^2 \leq 1$. Assume that $\tilde{m} \ll m$, so that $K - \tilde{K}_{\xi_T}$ is nonnegative definite and $0 \leq \rho_0 < 1$.

Lemma 2 $\sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f, \xi_T) = \eta^2$.

Proof: For $f \in \mathcal{F}_\eta^-$, we have $\mathcal{B}(f, \xi_T) \le \eta^2$. The proof can be completed by finding an $f_0 \in \mathcal{F}_\eta^-$ such that the above equality holds. Let $f_0(x) = v_0' K(A, x)$ with $v_0 = \eta \mathbf{1}/(c_0\sqrt{1-\rho_0}\ )$. It is easily verified that $v_0'(K(A, A') - \tilde{K}_{\xi_T})v_0 = \eta^2$. Here $\tilde{K}_{\xi_T} = \int K(A, \tilde{A}')K(\tilde{A}, \tilde{A}')^{-1}K(\tilde{A}, A')d\xi_T(\tilde{A})$. □

Lemma 3 $\inf_{f\in\mathcal{F}_\eta^+} \mathcal{B}(f, \xi_T) = \eta^2$.

Proof: Similarly, for any $f \in \mathcal{F}_\eta^+$, we have $\mathcal{B}(f, \xi_T) \ge \eta^2$. Again by taking $f_0(x) = v_0' K(A, x)$ with $v_0 = \eta\mathbf{1}/(c_0\sqrt{1-\rho_0}\ )$, we have $v_0'(K(A, A) - \tilde{K}_{\xi_T})v_0 = \eta^2$. □

Proof for Theorems 2 and 3: We will show that

(1) $\inf_{\xi\in\mathcal{P}} \sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f, \xi) = \sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f, \xi_T)$ and

(2) $\sup_{\xi\in\mathcal{P}} \inf_{f\in\mathcal{F}_\eta^+} \mathcal{B}(f, \xi) = \inf_{f\in\mathcal{F}_\eta^+} \mathcal{B}(f, \xi_T)$.

(1) From Lemma 2 we have $\sup_{f\in\mathcal{F}_\eta^-} \mathcal{B}(f, \xi_T) = \eta^2$. Part (1) can be shown by finding an $f_0 \in \mathcal{F}_\eta^-$ such that $\mathcal{B}(f_0, \xi) \ge \eta^2$ for all ξ, i.e. finding an $f_0(x) = v_0' K(A, x)$ such that

$$\begin{aligned} v_0'(K - \tilde{K}_{\xi_T})v_0 &\le \eta^2, \\ v_0'(K - \tilde{K}_{\xi})v_0 &\ge \eta^2, \ \forall\xi. \end{aligned}$$

For $\xi \ne \xi_T$, let $\nu = \xi_T - \xi$, a signed measure. By Hahn decomposition theorem there exists a measurable set E such that ν is positive on E and negative on E^c. Let $v_0^* = (v_{01}^*, \ldots, v_{0m}^*)'$ be given by

$$v_{0i}^* = \begin{cases} 1, & \text{if } A_i \in E, \\ 0, & \text{if } A_i \in E^c. \end{cases}$$

Let $v_0 = \eta v_0^* / \sqrt{v_0^{*\prime}(K - \tilde{K}_{\xi_T})v_0^*}$. Then $\mathcal{B}(f_0, \xi_T) = \eta^2$. Since $\xi = \xi_T - \nu$ and ν is positive on E, we have $v_0'(\tilde{K}_{\xi_T} - \tilde{K}_\xi)v_0 = v_0'\tilde{K}_\nu v_0 \ge 0$. Thus,

$$\mathcal{B}(f_0, \xi) = v_0'(K - \tilde{K}_\xi)v_0 = v_0'(K - \tilde{K}_{\xi_T} + \tilde{K}_\nu)v_0 \ge \mathcal{B}(f_0, \xi_T) = \eta^2.$$

(2) From Lemma 3 we have $\inf_{f\in\mathcal{F}_\eta^+}\mathcal{B}(v,\xi_T)=\eta^2$. Part (2) can be shown by finding an $f_0\in\mathcal{F}_\eta^+$ such that $\mathcal{B}(f_0,\xi)\le\eta^2$ for all ξ, i.e. finding an $f_0(x)=v_0'K(A,x)$ such that

$$\begin{aligned} v_0'(K-\tilde{K}_{\xi_T})v_0 &\ \ge\ \eta^2, \\ v_0'(K-\tilde{K}_{\xi})v_0 &\ \le\ \eta^2,\ \forall\xi. \end{aligned}$$

Let ν, E and E^c be as defined above. Let $v_0^*=(v_{01}^*,\ldots,v_{0m}^*)'$ be given by

$$v_{0i}^*=\begin{cases} 0, & \text{if } A_i\in E, \\ 1, & \text{if } A_i\in E^c. \end{cases}$$

Let $v_0=\eta v_0^*/\sqrt{v_0^{*\prime}(K-\tilde{K}_{\xi_T})v_0^*}$. Then $\mathcal{B}(f_0,\xi_T)=\eta^2$. Since $\xi=\xi_T-\nu$ and ν is negative on E^c, we have $v_0'(\tilde{K}_{\xi_T}-\tilde{K}_\xi)v_0=v_0'\tilde{K}_\nu v_0\le 0$. Thus,

$$\mathcal{B}(f_0,\xi)=v_0'(K-\tilde{K}_\xi)v_0=v_0'(K-\tilde{K}_{\xi_T}+\tilde{K}_\nu)v_0\le\mathcal{B}(f_0,\xi_T)=\eta^2.$$

The proof is completed. □

Kernel boundary correction. The issue of making boundary kernels and doing boundary correction in nonparametric estimation and prediction is a pretty subtle problem and there is vast statistical literature. See, e.g., Gasser, Müller and Mammitzsch [15] and references therein. Here we only provide some most basic ideas of boundary kernels and boundary correction. Let B be the set of the so called boundary points, which consist of points on and near the boundary. For simplicity consider translation-type kernels that are probability density functions. Boundary kernels are a collection of continuous varying kernels (with respect to z) $\{K(\cdot,z)\}_{z\in B}$ satisfying $\int K(x',z)dx=1$. For instance, one simple way to make Gaussian boundary kernels is to fold the part of kernel outside $\mathcal{X}$ by reflection and add it to the interior part. Another way is simply to rescale kernels near the boundary and make them integrate to one. To correct the boundary effects, one conceptually simple way is to replace the ordinary kernels with boundary kernels for points z near the boundary.

Acknowledgment

The authors wish to acknowledge the helpful comments and suggestions by three anonymous referees, which have enhanced the paper. The authors also thank Olvi Mangasarian for his suggestions and Hwang, Chii-Ruey for discussion and comments on the sampling design and spectral analysis and for reading the manuscript.

References

[1] S. Biedermann and H. Dette. Minimax optimal designs for nonparametric regression - a further optimality property of the uniform distribution. *Advances in Model-oriented Design and Analysis*, pages 13–20, 2001.

[2] S. Biedermann and H. Dette. Optimal designs for testing the functional form of a regression via nonparametric estimation techniques. *Statist. Probab. Letters*, 52:215–224, 2001.

[3] C. L. Blake and C. J. Merz. UCI repository of machine learning databases, 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[4] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[5] C.-C. Chang and Y.-J. Lee. Generating the reduced set by systematic sampling. In Zheng Rong Yang, Richard Everson, and Hujun Yin, editors, *Proceedings of the Fifth International Conference on Intelligent Data Engineering and Automated Learning(LNCS 3177)*, pages 714–719, Exeter, UK, 2004. Springer–Verlag.

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

[7] F. Chatelin. *Eigenvalues of Matrices.* John Wiley & Sons, Chichester, 1993.

[8] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines.* Cambridge University Press, Cambridge, 2000.

[9] Delve. Comp-activ dataset, data for evaluating learning in valid experiments. http://www.cs.toronto.edu/∼delve/data/comp-activ/desc.html.

[10] Delve. Kin-family of datasets, data for evaluating learning in valid experiments. http://www.cs.toronto.edu/∼delve/data/kin/desc.html.

[11] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems -9-*, pages 155–161, Cambridge, MA, 1997. MIT Press.

[12] K. T. Fang, D. K. J. Lin, P. Winker, and Y. Zhang. Uniform design: theory and application. *Technometrics*, 42:237–248, 2000.

[13] K. T. Fang, Y. Wang, and P. M. Bentler. Some applications of number-theoretic methods in statistics. *Statistical Science*, 9:416–428, 1994.

[14] G. Fung and O. L. Mangasarian. Proximal support vector machine classifiers. In F. Provost and R. Srikant, editors, *Proceedings KDD-2001: Knowledge Discovery and Data Mining, August 26-29, 2001, San Francisco, CA*, pages 77–86, New York, 2001. Asscociation for Computing Machinery. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-02.ps.

[15] T. Gasser, H-G. Müller, and V. Mammitzsch. Kernels for nonparametric curve estimation. *J. Roy. Statist. Soc. B*, 47:238–252, 1985.

[16] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer-Verlag, New York, 2001.

[17] L.-R. Jen and Y.-J. Lee. Clustering model selection for reduced support vector machines. In Zheng Rong Yang, Richard Everson, and Hujun Yin, editors, *Proceedings of the Fifth International Conference on Intelligent Data Engineering and Automated Learning(LNCS 3177)*, pages 720–725, Exeter, UK, 2004. Springer–Verlag.

[18] Y.-J. Lee, W.-F. Hsieh, and C.-M. Huang. ϵ-SSVR: A smooth vector machine for ϵ-insensitive regression. *IEEE Transactions on Knowledge and Data Engineering*, 17:678–685, 2005.

[19] Y.-J. Lee, H.-Y. Lo, and S.-Y. Huang. Incremental reduced support vector machine. In *International Conference on Informatics Cybernetics and System (ICICS 2003)*, Kaohsiung, Taiwan, 2003.

[20] Y.-J. Lee and O. L. Mangasarian. RSVM: Reduced support vector machines. Technical Report 00-07, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, July 2000. Proceedings of the First SIAM International Conference on Data Mining, Chicago, April 5-7, 2001, CD-ROM Proceedings. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-07.ps.

[21] Y.-J. Lee and O. L. Mangasarian. SSVM: A smooth support vector machine. *Computational Optimization and Applications*, 20:5–22, 2001. Data Mining Institute, University of Wisconsin, Technical Report 99-03. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/99-03.ps.

[22] K. C. Li. Robust regression designs when the design space consists of finitely many points. *Ann. Statist.*, 12:269–282, 1984.

[23] K.-M. Lin and C.-J. Lin. A study on reduced support vector machines. *IEEE Transactions on Neural Networks*, 14:1449–1459, 2003.

[24] J. S. Liu. *Monte Carlo Strategies in Scientific Computing.* Springer–Verlag, New York, 2001.

[25] O. L. Mangasarian. Generalized support vector machines. In A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 135–146, Cambridge, MA, 2000. MIT Press. ftp://ftp.cs.wisc.edu/math-prog/tech-reports/98-14.ps.

[26] O. L. Mangasarian and D. R. Musicant. Large scale kernel regression via linear programming. Technical Report 99-02, Data Mining Institute, Computer Sciences Department, University of Wisconsin, Madison, Wisconsin, August 1999. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/99-02.ps.

[27] O. L. Mangasarian and D. R. Musicant. Robust linear and support vector regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):950–955, 2000. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/99-09.ps.

[28] O. L. Mangasarian and D. R. Musicant. Lagrangian support vector machines. *Journal of Machine Learning Research*, 1:161–177, 2001. ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-06.ps.

[29] S. Mika. *Kernel Fisher Discriminants.* PhD thesis, Electrical Engineering and Computer Science, Technische Universität Berlin, 2002.

[30] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller. Fisher discriminant analysis with kernels. *Neural Networks for Signal Processing*, IX:41–48, 1999.

[31] T. M. Mitchell. *Machine Learning.* McGraw-Hill, Boston, 1997.

[32] E. Parzen. On estimation of a probability density function and mode. *Ann. Math. Statist.*, 35:1065–1076, 1962.

[33] C. E. Rasmussen and Z. Ghahramani. Occam's razor. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 294–300, Cambridge, MA, 2001. MIT Press.

[34] M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *Ann. Math. Statist.*, 27:832–835, 1956.

[35] A. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. In *Proc. 17th International Conf. on Machine Learning*, pages 911–918. Morgan Kaufmann, San Francisco, CA, 2000.

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

[37] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14:199–222, 2004.

[38] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[39] J. A. K. Suykens and J. Vandewalle. Multiclass least squares support vector machines. In *Proceedings of IJCNN'99*, pages CD–ROM, Washington, DC, 1999.

[40] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.

[41] D. P. Wiens. Designs for approximately linear regression: Two optimality properties of uniform designs. *Statist. & Probab. Letters*, 12:217–221, 1991.

[42] D. P. Wiens. Minimax designs for approximately linear regression. *J. Statist. Plann. Inference*, 31:353–371, 1992.

[43] D. P. Wiens. Minimax robust designs and weights for approximately specified regression models with heteroscedastic errors. *J. Amer. Statist. Assoc.*, 93:1440–1450, 1998.

[44] C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 682–688, Cambridge, MA, 2001. MIT Press.

[45] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufmann, San Francisco, CA, 1999.