JMLR: Workshop and Conference Proceedings 20 (2011) 165–180 Asian Conference on Machine Learning

Approximate Model Selection for Large Scale LSSVM

Lizhong Ding LIZHONGDING@GMAIL.COM
Shizhong Liao SZLIAO@TJU.EDU.CN
School of Computer Science and Technology
Tianjin University, Tianjin 300072, P. R. China

Editor: Chun-Nan Hsu and Wee Sun Lee

Abstract

Model selection is critical to least squares support vector machine (LSSVM). A major problem of existing model selection approaches of LSSVM is that the inverse of the kernel matrix need to be calculated with $O(n^3)$ complexity for each iteration, where n is the number of training examples. It is prohibitive for the large scale application. In this paper, we propose an approximate approach to model selection of LSSVM. We use multilevel circulant matrices to approximate the kernel matrix so that the fast Fourier transform (FFT) can be applied to reduce the computational cost of matrix inverse. With such approximation, we first design an efficient LSSVM algorithm with $O(n\log(n))$ complexity and theoretically analyze the effect of kernel matrix approximation on the decision function of LSSVM. We further show that the approximate optimal model produced with the multilevel circulant matrix is consistent with the accurate one produced with the original kernel matrix. Under the guarantee of consistency, we present an approximate model selection scheme, whose complexity is significantly lower than the previous approaches. Experimental results on benchmark datasets demonstrate the effectiveness of approximate model selection.

Keywords: Model Selection, Matrix Approximation, Multilevel Circulant Matrices, Least Squares Support Vector Machine

1. Introduction

Support vector machine (SVM) (Vapnik, 1998) is a learning system for training linear learning machines in the kernel-induced feature spaces, while controlling the capacity to prevent overfitting by generalization theory. It can be formulated as a quadratic programming problem with linear inequality constraints. The least squares support vector machine (LSSVM) (Suykens and Vandewalle, 1999) is a least squares version of SVM, which considers equality constraints instead of inequalities for classical SVM. As a result, the solution of LSSVM follows directly from solving a system of linear equations, instead of quadratic programming.

Model selection is an important issue in LSSVM research. It involves the selection of kernel function and associated kernel parameters and the selection of regularization parameter. In spite of regularization parameter, Micchelli and Pontil (2006) proposed a kernel selection method to obtain an optimal kernel by minimizing a cost functional over a set of kernels; Crammer et al. (2003) used the boosting paradigm to construct kernel function from data. Typically, the form of kernel function will be determined as several types, such as polynomial kernel and radial basis function (RBF) kernel. In this situation, the selection of kernel function amounts to tuning the kernel parameters. Model selection can be

reduced to the selection of kernel parameters and regularization parameter which minimize the expectation of test error (Chapelle and Vapnik, 2000). We usually refer to these parameters collectively as *hyperparameters*. Common model selection approaches mainly adopt a nested two-layer inference (Guyon et al., 2010), where the inner layer trains the classifier for fixed hyperparameters and the outer layer tunes the hyperparameters to minimize the generalization error. The generalization error can be estimated either via testing on some unused data (hold-out testing or cross validation) or via a theoretical bound (Vapnik and Chapelle, 2000; Chapelle et al., 2002).

The k-fold cross validation gives an excellent estimate of the generalization error (Duan et al., 2003) and the extreme form of cross validation, leave-one-out (LOO), provides an almost unbiased estimate of the generalization error (Luntz and Brailovsky, 1969). However, the naive model selection strategy based on cross validation, which adopts a grid search in the hyperparameters space, unavoidably brings high computational complexity, since it would train LSSVM for every possible value of the hyperparameters vector. Minimizing the estimate bounds of the generalization error is an alternative to model selection, which is usually realized by the gradient descent techniques. The commonly used estimate bounds include span bound (Vapnik and Chapelle, 2000) and radius margin bound (Chapelle et al., 2002). Generally, these methods using the estimate bounds reduce the whole hyperparameters space to a search trajectory in the direction of gradient descent, to accelerate the outer layer of model selection, but multiple times of LSSVM training have to be implemented in the inner layer to iteratively attain the minimal value of the estimates. Training LSSVM is equivalent to computing the inverse of a full $n \times n$ matrix, so its complexity is $O(n^3)$, where n is the number of training examples. Therefore, it is prohibitive for the large scale problems to directly train LSSVM for every hyperparameters vector on the search trajectory. Consequently, efficient model selection approaches via the acceleration of the inner computation are imperative.

As pointed out in (Chapelle et al., 2002; Cawley and Talbot, 2010), a model selection criterion is not required to be an unbiased estimate of the generalization error, instead the primary requirement is merely for the minimum of the model selection criterion to provide a reliable indication of the minimum of the generalization error in hyperparameters space. Therefore, we argue that it is sufficient to calculate an approximate criterion that can discriminate the optimal model from the candidates. Such considerations drive the proposal of approximate model selection approach for LSSVM.

Since the high computational cost for calculating the inverse of a kernel matrix is a major problem, we consider to approximate a kernel matrix by a "nice" matrix with a significantly lower computational cost when calculating its inverse. The multilevel circulant matrix is a good choice since its built-in periodicity allows the multi-dimensional fast Fourier transform (FFT) to be utilized in calculating its inverse with complexity of $O(n \log(n))$ (Song and Xu, 2010b; Song, 2010). Taking advantage of computational virtue of such approximation, we propose an efficient algorithm for solving LSSVM and derive an upper error bound to measure the effect of such approximation on the decision function of LSSVM. We further take a model selection criterion as an example to demonstrate the consistency of approximate model selection. With the guarantee of consistency, we present an efficient approximate model selection scheme. It conforms to the two-layer iterative procedure, but the inner computation can be realized in $O(n \log(n))$ complexity instead of $O(n^3)$.

By experiments on 10 benchmark datasets, we show that approximate model selection can significantly improve the efficiency of model selection, and meanwhile guarantee low generalization error.

The rest of the paper is organized as follows. In Section 2, we give a brief introduction of LSSVM and a reformulation of it. In Section 3, we present an efficient algorithm for solving LSSVM. In Section 4, we analyze the effect of kernel matrix approximation on the decision function of LSSVM. In Section 5, we demonstrate the consistency of approximate model selection. In Section 6, we propose an approximate model selection scheme for LSSVM. In Section 7, we report experimental results. The last section gives the conclusion.

2. Least Squares Support Vector Machine

We use $\mathcal{X}$ to denote the input space and $\mathcal{Y}$ the output domain. Usually we will have $\mathcal{X} \subseteq \mathbb{R}^d$, $\mathcal{Y} = \{-1, 1\}$ for binary classification. The training set is denoted by

$$\mathcal{S} = ((\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_l, y_l)) \in (\mathcal{X} \times \mathcal{Y})^l .$$

We seek to construct a linear classifier, $f(\boldsymbol{x}) = \boldsymbol{w} \cdot \phi(\boldsymbol{x}) + b$, in a feature space $\mathcal{F}$, defined by a feature mapping of the input space, $\phi : \mathcal{X} \to \mathcal{F}$. The parameters of the linear classifier, $(\boldsymbol{w}, b)$, are given by the minimizer of a regularized least-squares training function

$$L = \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{1}{2\mu}\sum_{i=1}^{l}[y_i - \boldsymbol{w} \cdot \phi(\boldsymbol{x}_i) - b]^2, \tag{1}$$

where $\mu > 0$ is called regularization parameter. The basic training algorithm for LSSVM (Suykens and Vandewalle, 1999; Van Gestel et al., 2004) views the regularized loss function (1) as a constrained minimization problem

$$\begin{aligned} \min \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{1}{2\mu}\sum_{i=1}^{l}\varepsilon_i^2, \\ \text{s.t.} \quad & \varepsilon_i = y_i - \boldsymbol{w} \cdot \phi(\boldsymbol{x}_i) - b. \end{aligned} \tag{2}$$

Further, we can obtain the dual form of Equation (2) as follows

$$\sum_{j=1}^{l}\alpha_j \phi(\boldsymbol{x}_j) \cdot \phi(\boldsymbol{x}_i) + b + \mu\alpha_i = y_i, \quad i = 1, 2, \ldots, l, \tag{3}$$

where $\sum_{i=1}^{l}\alpha_i = 0$. Noting that $\phi(\boldsymbol{x}_j) \cdot \phi(\boldsymbol{x}_i)$ corresponds to the kernel function $K(x_i, x_j)$, we can write Equation (3) in a matrix form

$$\begin{bmatrix} \boldsymbol{K}_l + \mu\boldsymbol{I}_l & \mathbf{1} \\ \mathbf{1}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}, \tag{4}$$

where $\boldsymbol{K}_l = [K(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1}^{l}$, $\boldsymbol{I}_l$ is the $l \times l$ identity matrix, $\mathbf{1}$ is a column vector of l ones, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_l)^{\mathrm{T}} \in \mathbb{R}^l$ is a vector of Lagrange multipliers, and $\boldsymbol{y} \in \mathcal{Y}^l$ is the label vector.

If we let $\boldsymbol{K}_{\mu,l} = \boldsymbol{K}_l + \mu \boldsymbol{I}_l$, Equation (4) is equivalent to

$$\begin{bmatrix} \boldsymbol{K}_{\mu,l} & \mathbf{1} \\ \mathbf{1}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}. \tag{5}$$

The matrix on the left-hand side is not positive definite, so Equation (5) cannot be directly solved. However, we can write the first row of Equation (5) as

$$\boldsymbol{K}_{\mu,l}(\boldsymbol{\alpha} + \boldsymbol{K}_{\mu,l}^{-1}\mathbf{1}b) = \boldsymbol{y}. \tag{6}$$

Therefore, $\boldsymbol{\alpha} = \boldsymbol{K}_{\mu,l}^{-1}(\boldsymbol{y} - \mathbf{1}b)$ and replacing $\boldsymbol{\alpha}$ in the second row of Equation (5) we can obtain

$$\mathbf{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,l}^{-1}\mathbf{1}b = \mathbf{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,l}^{-1}\boldsymbol{y}. \tag{7}$$

The system of linear equations (5) can then be re-written as

$$\begin{bmatrix} \boldsymbol{K}_{\mu,l} & \mathbf{0} \\ \mathbf{0}^{\mathrm{T}} & \mathbf{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,l}^{-1}\mathbf{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} + \boldsymbol{K}_{\mu,l}^{-1}\mathbf{1}b \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \mathbf{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,l}^{-1}\boldsymbol{y} \end{bmatrix}. \tag{8}$$

Equation (8) can be solved as follows: First solve

$$\boldsymbol{K}_{\mu,l}\boldsymbol{\rho} = \mathbf{1} \qquad \text{and} \qquad \boldsymbol{K}_{\mu,l}\boldsymbol{\nu} = \boldsymbol{y}. \tag{9}$$

Since $\boldsymbol{K}_{\mu,l} = \boldsymbol{K}_l + \mu \boldsymbol{I}_l$ is positive definite, the inverse of the matrix $\boldsymbol{K}_{\mu,l}$ exists.

The solution $(\boldsymbol{\alpha}, b)$ of Equation (5) are then given by

$$b = \frac{\mathbf{1}^{\mathrm{T}}\boldsymbol{\nu}}{\mathbf{1}^{\mathrm{T}}\boldsymbol{\rho}} \qquad \text{and} \qquad \boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho}b. \tag{10}$$

The decision function of LSSVM can be written as $f(\boldsymbol{x}) = \sum_{i=1}^{\ell} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b$.

If Equation (9) is solved, we can easily obtain the solution of LSSVM. However, the complexity of calculating the inverse of the matrix $\boldsymbol{K}_{\mu,l}$ is $O(l^3)$. In the following, we will demonstrate that multilevel circulant matrices can be used to speed up this process.

3. Approximating LSSVM using Multilevel Circulant Matrices

We first review the notion of multilevel matrices (Tyrtyshnikov, 1996). For $m, n \in \mathbb{N}$, let $\mathbb{N}_n := \{0, 1, \ldots, n-1\}$ and $\mathbb{R}^{m\times n}$ be the set of $m \times n$ real-valued matrices. For a fixed positive integer p and $\boldsymbol{n} := [n_0, n_1, \ldots, n_{p-1}] \in \mathbb{N}^p$, we set

$$\Pi_{\boldsymbol{n}} := n_0 n_1 \ldots n_{p-1}, \qquad \mathbb{N}_{\boldsymbol{n}} := \mathbb{N}_{n_0} \times \mathbb{N}_{n_1} \times \cdots \times \mathbb{N}_{n_{p-1}}.$$

A multilevel matrix is defined recursively. According to (Tyrtyshnikov, 1996), a matrix $A_{\boldsymbol{n}}$ is called a p-level matrix of level order $\boldsymbol{n}$ if it consists of $n_0 \times n_0$ blocks and each block is a $(p-1)$-level matrix of level order $[n_1, n_2, \ldots, n_{p-1}]$. To point to the entries of the multilevel matrix $A_{\boldsymbol{n}}$, we use multi-indices. For any $\boldsymbol{j} := [j_s : s \in \mathbb{N}_p], \boldsymbol{l} := [l_s : s \in \mathbb{N}_p] \in \mathbb{N}_{\boldsymbol{n}}$, we write

$$A_{\boldsymbol{n}} := [a_{\boldsymbol{j},\boldsymbol{l}} : \boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}],$$

where $(j_s, l_s), s \in \mathbb{N}_p$ is the location at level s.

In the following, we restrict the kernel to be radial basis function (RBF) kernel. We assume that there exists a real-valued function $k \in L^1(\mathbb{R})$ on $\mathcal{X}$ such that $K(t,s) = k(\|t - s\|_2)$ for all $t, s \in \mathcal{X}$, where $\|\cdot\|_2$ denotes the Euclidean norm. We point out that k is always an even function, since we have $K(t,s) = K(s,t)$, $s, t \in \mathcal{X}$, from the definition of kernels.

We present the kernel matrix in multilevel notation. For a set $\mathcal{D} \subseteq \mathcal{X}$, we relabel it as $\mathcal{D} := \{\boldsymbol{x_j} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\}$ for some $\boldsymbol{n} \in \mathbb{N}^p$. The number of elements of $\mathcal{D}$ is $\Pi_{\boldsymbol{n}}$. The kernel matrix is rewritten as

$$\boldsymbol{K_n} = [k(\|\boldsymbol{x_j} - \boldsymbol{x_l}\|_2) : \boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}].$$

We now describe the definition of multilevel circulant matrices (Davis, 1979). We begin with the definition of circulant matrices. A circulant matrix is an $n \times n$ matrix $\boldsymbol{C}_n := [c_{j,l} : j, l \in \mathbb{N}_n]$, where $c_{j,l} = c_{l-j}$ for any $j, l \in \mathbb{N}_n$ and $c_j = c_{j-n}$ for $0 \le j \le n-1$. More specifically, it takes the form

$$\begin{bmatrix} c_0 & c_1 & \dots & c_{n-1} \\ c_{n-1} & c_0 & \dots & c_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & \dots & c_0 \end{bmatrix}.$$

Clear, a circulant matrix is completely determined by its first row, so we write

$$\boldsymbol{C}_n := \mathrm{circ}[c_j : j \in \mathbb{N}_n]. \tag{11}$$

A block circulant matrix of type (m, n) is an $mn \times mn$ matrix of the form

$$\begin{bmatrix} \boldsymbol{A}_0 & \boldsymbol{A}_1 & \dots & \boldsymbol{A}_{m-1} \\ \boldsymbol{A}_{m-1} & \boldsymbol{A}_0 & \dots & \boldsymbol{A}_{m-2} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{A}_1 & \boldsymbol{A}_2 & \dots & \boldsymbol{A}_0 \end{bmatrix},$$

where each block $\boldsymbol{A}_j$, $j \in \mathbb{N}_m$ is an $n \times n$ matrix. A multilevel circulant matrix is defined recursively (Davis, 1979). A multilevel circulant matrix of level 1 is an ordinary circulant matrix. For any $s \in \mathbb{N}$, an $(s+1)$-level circulant matrix is a block circulant matrix whose blocks are s-level circulant matrices. More specifically, for $\boldsymbol{n} \in \mathbb{N}^p$, $\boldsymbol{A_n} := [a_{\boldsymbol{j},\boldsymbol{l}} \in \mathbb{N}_{\boldsymbol{n}}]$ is called a p-level circulant matrix if, for any $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$,

$$a_{\boldsymbol{j},\boldsymbol{l}} = a_{l_0 - j_0 (\mathrm{mod}\ n_0), \dots, l_{p-1} - j_{p-1} (\mathrm{mod}\ n_{p-1})}.$$

Note that a p-level circulant matrix is completely determined by its first row $a_{\boldsymbol{0},\boldsymbol{l}}$, where $\boldsymbol{0} := (0, \dots, 0)^{\mathrm{T}} \in \mathbb{R}^p$. We will write

$$\boldsymbol{A_n} := \mathrm{circ}_{\boldsymbol{n}}[a_{\boldsymbol{l}} : \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}],$$

where $a_{\boldsymbol{l}} := a_{\boldsymbol{0},\boldsymbol{l}}$, for $\boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$.

We now construct a multilevel circulant matrix $\boldsymbol{U_n}$ which approximates the given kernel matrix $\boldsymbol{K_n}$ (Song and Xu, 2010b; Song, 2010). For an $\boldsymbol{n} \in \mathbb{N}^p$, we choose a sequence of positive numbers $\boldsymbol{h_n} := [h_{\boldsymbol{n},j} : j \in \mathbb{N}_p] \in \mathbb{R}^p$, and define

$$t_{\boldsymbol{j}} := k(\|[j_s h_{\boldsymbol{n},s} : s \in \mathbb{N}_p]\|_2), \quad \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}. \tag{12}$$

For any $\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}$ and $s \in \mathbb{N}_p$, we introduce the sets $D_{\boldsymbol{j},s} := \{0\}$ if $j_s = 0$, and $D_{\boldsymbol{j},s} := \{j_s, n_s - j_s\}$ if $1 \le j_s \le n_s - 1$, and let $D_{\boldsymbol{j}} := D_{\boldsymbol{j},0} \times D_{\boldsymbol{j},1} \times \cdots \times D_{\boldsymbol{j},p-1}$. We then define

$$u_{\boldsymbol{j}} := \sum_{\boldsymbol{l} \in D_{\boldsymbol{j}}} t_{\boldsymbol{l}}, \quad \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}, \tag{13}$$

and

$$\boldsymbol{U_n} := \mathrm{circ}_{\boldsymbol{n}}[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]. \tag{14}$$

For LSSVM, we need to solve the inverse of $\boldsymbol{K_n} + \mu \boldsymbol{I_n}$. To reduce the computational cost, we intend to use the inverse of $\boldsymbol{U_n} + \mu \boldsymbol{I_n}$ as an approximation of the inverse of $\boldsymbol{K_n} + \mu \boldsymbol{I_n}$. We know that $\boldsymbol{K_n} + \mu \boldsymbol{I_n}$ is invertible, but the invertibility of $\boldsymbol{U_n} + \mu \boldsymbol{I_n}$ is not so obvious. Actually, Song and Xu (2010b) have proved that when $\boldsymbol{n}$ is large enough, which means all of its components are large enough, $\boldsymbol{U_n} + \mu \boldsymbol{I_n}$ is positive definite and invertible.

We further introduce two lemmas about the eigenvalues and eigenvectors of a multilevel circulant matrix (Tyrtyshnikov, 1996; Davis, 1979).

Lemma 1 *The eigenvalues of a p-level circulant matrix* $\boldsymbol{A_n} := \mathrm{circ}_{\boldsymbol{n}}[a_{\boldsymbol{l}} : \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}]$ *are given by*

$$\lambda_{\boldsymbol{j}} = \sum_{\boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}} a_{\boldsymbol{l}} \mathrm{e}^{2\pi i \sum_{s \in \mathbb{N}_p} \frac{j_s l_s}{n_s}}, \quad \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}},$$

where $i := \sqrt{-1}$.

Lemma 2 *Suppose that* $\boldsymbol{A_n}$ *is a multilevel matrix of order* $\boldsymbol{n}$ *and* $\boldsymbol{a} := [a_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ *is the first column of* $\boldsymbol{A_n}$. *Then* $\boldsymbol{A_n}$ *is a p-level circulant matrix of level order* $\boldsymbol{n}$ *if and only if*

$$\boldsymbol{A_n} = \frac{1}{\Pi_{\boldsymbol{n}}} \Phi^* \mathrm{diag}(\Phi \boldsymbol{a}) \Phi,$$

where $\Phi := F_{n_0} \otimes F_{n_1} \otimes \cdots \otimes F_{n_{p-1}}$, $\otimes$ *denotes the Kronecker product of matrices and* $F_m := \left[\mathrm{e}^{\mathrm{i}\frac{2\pi st}{m}} : s, t \in \mathbb{N}_m\right]$, *for* $m \in \mathbb{N}$.

From above two lemmas, we can find that the eigenvalues and eigenvectors of multilevel circulant matrices can be expressed in a multi-dimensional discrete Fourier transform (DFT) form. Therefore, their calculation can be realized efficiently by using the multi-dimensional fast Fourier transform (FFT). In the following, we will show how this computational advantage can be applied to obtain an efficient algorithm for solving LSSVM.

From Lemma 2, the multilevel circulant matrix $\boldsymbol{U_n}$ can be represented as

$$\boldsymbol{U_n} = \frac{1}{\Pi_{\boldsymbol{n}}} \Phi^* \mathrm{diag}(\boldsymbol{v}) \Phi, \tag{15}$$

where

$$\boldsymbol{v} = \Phi[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]. \tag{16}$$

It follows that

$$(\boldsymbol{U}_{\boldsymbol{n}} + \mu \boldsymbol{I}_{\boldsymbol{n}})^{-1} = \frac{1}{\Pi_{\boldsymbol{n}}} \Phi^* \mathrm{diag}\left(\frac{1}{v_{\boldsymbol{j}} + \mu} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\right) \Phi. \tag{17}$$

We next present an algorithm of solving LSSVM.

Algorithm 1: Approximating LSSVM using Multilevel Circulant Matrices

Input: $\boldsymbol{y} := \{y_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\}$, $[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$, k, μ;
Output: $(\boldsymbol{\alpha}, b)$;
1: Calculate $\boldsymbol{v}$ according to (16) by using multi-dimensional FFT;
2: Calculate $\boldsymbol{\eta} := \Phi[\mathbf{1},\ \boldsymbol{y}]$ using multi-dimensional FFT, where $\mathbf{1}$ is a vector of all ones;
3: Calculate $\boldsymbol{\tau} := \mathrm{diag}\left(\frac{1}{v_{\boldsymbol{j}} + \mu} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\right) \boldsymbol{\eta}$;
4: Calculate $[\boldsymbol{\rho},\ \boldsymbol{\nu}] = \frac{1}{\Pi_{\boldsymbol{n}}} \Phi^* \boldsymbol{\tau}$ according to (17) using multi-dimensional FFT;
5: Calculate $b = \frac{\mathbf{1}^{\mathrm{T}} \boldsymbol{\nu}}{\mathbf{1}^{\mathrm{T}} \boldsymbol{\rho}}$, $\boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho} b$;
return $(\boldsymbol{\alpha}, b)$;

We estimate the computational complexity of Algorithm 1 in next theorem.

Theorem 3 *The computational complexity of Algorithm 1 is* $O(\Pi_{\boldsymbol{n}} \log(\Pi_{\boldsymbol{n}}))$.

Proof The computational complexity of the multidimensional FFT is $O(l \log(l))$, where l is the total number of data points. It follows that the computational complexity of step 1, 2 and 4 in Algorithm 1 is $O(\Pi_{\boldsymbol{n}} \log(\Pi_{\boldsymbol{n}}))$, since each of them is the multidimensional FFT of $\Pi_{\boldsymbol{n}}$ data points. The complexity of step 3 is $O(\Pi_{\boldsymbol{n}})$, since it is the product of a vector with a diagonal matrix. In step 5, the multiplication and subtraction between two vectors need to be done, so its complexity is $O(\Pi_{\boldsymbol{n}})$. Therefore, the total complexity is $O(\Pi_{\boldsymbol{n}} \log(\Pi_{\boldsymbol{n}}))$. ■

4. Error Analysis

In this section, we analyze the effect of kernel matrix approximation on the decision function generated by LSSVM.

We assume that kernel approximation is only used in training. At testing time the true kernel function is used (Cortes et al., 2010). The decision function f derived with the exact kernel matrix is defined by

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} \alpha_{\boldsymbol{j}} K(\boldsymbol{x}, \boldsymbol{x}_{\boldsymbol{j}}) + b = \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{k}_x \\ 1 \end{bmatrix},$$

where $\boldsymbol{k}_x = [K(\boldsymbol{x}, \boldsymbol{x}_{\boldsymbol{j}}) : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$. For simplicity, we assume the offset b to be a constant φ. We define $\kappa > 0$ such that $K(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa$.

To analyze the effect of approximation, we need to introduce a class of matrices whose entries have an exponential decay property (Song and Xu, 2010b).

We first define "distances" of an entry in a multilevel matrix to its diagonal, to its upper right corner and to its lower left corner at each level. For $t \in \mathbb{N}_2$, $m \in \mathbb{N}$, $j, l \in \mathbb{N}_m$, we set

$$d_m(t, j, l) := t|j - l| + (1 - t)(m - |j - l| - 1),$$

and for $\boldsymbol{t} \in \mathbb{N}_2^p$, $\boldsymbol{n} \in \mathbb{N}^p$, $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$, let

$$\boldsymbol{d}_{\boldsymbol{n}}(\boldsymbol{t}, \boldsymbol{j}, \boldsymbol{l}) := [d_{n_s}(t_s, j_s, l_s) : s \in \mathbb{N}_p].$$

We remark that, for any $s \in \mathbb{N}_p$, $d_{n_s}(1, j_s, l_s)$ is the distance of the entry at the position (j_s, l_s) to the diagonal at level s and $d_{n_s}(0, j_s, l_s)$ is the distance to the upper right and lower left corners at level s.

We now give the definition of the class of matrices whose entries have an exponential decay as their distances defined above increase. For any $\boldsymbol{n} \in \mathbb{N}^p$, $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$ and $r > 0$, let

$$E_{r,p,\boldsymbol{n}}(\boldsymbol{j}, \boldsymbol{l}) := \sum_{\boldsymbol{t} \in \mathbb{N}_2^p} \mathrm{e}^{-r\|\boldsymbol{d}_{\boldsymbol{n}}(\boldsymbol{t}, \boldsymbol{j}, \boldsymbol{l})\|_2}. \tag{18}$$

In what follows, we write $\mathcal{A} := \{\boldsymbol{A}_{\boldsymbol{n}} : \boldsymbol{n} \in \mathbb{N}^p\}$.

Definition 4 *A sequence of positive definite matrices $\mathcal{A}$ belongs to $\mathcal{E}_r$ for a positive constant r if it satisfies the following conditions:*

(i) there exists a positive constant κ such that $\|\boldsymbol{A}_{\boldsymbol{n}}^{-1}\|_2 \le \kappa$ for any $\boldsymbol{n} \in \mathbb{N}^p$;

(ii) there exists a positive constant c such that for any $\boldsymbol{n} \in \mathbb{N}^p$ and $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$,

$$|a_{\boldsymbol{j},\boldsymbol{l}}| \le c E_{r,p,\boldsymbol{n}}(\boldsymbol{j}, \boldsymbol{l}).$$

Since solving LSSVM is equivalent to solving the linear systems, we further introduce a weight function of the location of an entry in a multilevel vector, which is used to analyze the approximation behavior in solving the linear systems.

For an $\boldsymbol{n} \in \mathbb{N}^p$, $\boldsymbol{j} \in \mathbb{N}_n$, let

$$W_{\mathcal{E}_r,\boldsymbol{n}}(\boldsymbol{j}) := \mathrm{e}^{r\nu_{\boldsymbol{n}}(\boldsymbol{j})}, \quad r > 0, \tag{19}$$

where $\nu_{\boldsymbol{n}}(\boldsymbol{j}) := \left\|\frac{\boldsymbol{n}}{2} - \boldsymbol{j}\right\|_2$.

Furthermore, we define the associated weighted norm of the multilevel matrix. For a multilevel vector $\boldsymbol{y}_{\boldsymbol{n}}, \boldsymbol{n} \in \mathbb{N}^p$, we let

$$\|\boldsymbol{y}_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}} := \sup_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} W_{\mathcal{E}_r,\boldsymbol{n}}(\boldsymbol{j}) \left|(\boldsymbol{y}_{\boldsymbol{n}})_{\boldsymbol{j}}\right|, \quad r > 0. \tag{20}$$

The "induced norm" of multilevel matrix $\boldsymbol{A}_{\boldsymbol{n}}, \boldsymbol{n} \in \mathbb{N}^p$ is defined by

$$\|\boldsymbol{A}_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}} := \sup\{\|\boldsymbol{A}_{\boldsymbol{n}}\boldsymbol{y}_{\boldsymbol{n}}\|_\infty : \|\boldsymbol{y}_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}} = 1\}, \quad r > 0.$$

Our analysis of the effect of kernel matrix approximation on the decision function of LSSVM is based on the convergence analysis of the approximation of kernel matrices by multilevel circulant matrices. The following theorem demonstrates the convergency of such approximation (Song and Xu, 2010b). We let $\boldsymbol{K}_{\mu,\boldsymbol{n}} := \boldsymbol{K}_{\boldsymbol{n}} + \mu\boldsymbol{I}_{\boldsymbol{n}}$, $\boldsymbol{U}_{\mu,\boldsymbol{n}} := \boldsymbol{U}_{\boldsymbol{n}} + \mu\boldsymbol{I}_{\boldsymbol{n}}$.

Theorem 5 *If the following assumptions*

(H1) there exist positive constants c_1 *and* c_2 *such that* $|k(s)| \le c_1 \mathrm{e}^{-c_2|s|}, s \in \mathbb{R}$;

(H2) there exist constants $h_0 > 0$ *and* $c_0 \in \mathbb{R}$ *such that*

$$\|x_{\boldsymbol{j}} - x_{\boldsymbol{l}}\|_2 \ge h_0 \|\boldsymbol{j} - \boldsymbol{l}\|_2 + c_0, \quad \boldsymbol{n} \in \mathbb{N}^p,\ \boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}};$$

(H3) there exists a positive constant h *such that* $h_{\boldsymbol{n},j} \ge h$, *for all* $\boldsymbol{n} \in \mathbb{N}^p$ *and* $j \in \mathbb{N}_p$;

(H4) there exist positive constants c_3 *and* β *such that* $|k(s) - k(t)| \le c_3 |s-t|^\beta, s, t \in \mathbb{R}$

hold, then for each $r > 0$ *there exists a positive constant* r_0 *such that, for any* $0 < r' < \frac{r_0}{4}$ *and all* $\boldsymbol{n} \in \mathbb{N}^p$, *we have*

$$\|\boldsymbol{K}_{\mu,\boldsymbol{n}}^{-1} - \boldsymbol{U}_{\mu,\boldsymbol{n}}^{-1}\|_{W_{\mathcal{E}_{r_0}}} \le c \left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}}^\beta + 1 \right) \mathrm{e}^{-r' \boldsymbol{n}_{\min}},$$

for some positive constant c, *where* $\boldsymbol{n}_{\min} := \min\{n_s : s \in \mathbb{N}_p\}$, $X_{\boldsymbol{n}} := [\|\boldsymbol{x}_{\boldsymbol{j}} - \boldsymbol{x}_{\boldsymbol{l}}\|_2 : \boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}]$, $M_{\boldsymbol{n}} := [\|[(j_s - l_s) h_{\boldsymbol{n},s} : s \in \mathbb{N}_p]\|_2 : \boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}]$, *for any* $\boldsymbol{n} \in \mathbb{N}^p$.

RBF kernels which have the form $k(x) = \mathrm{e}^{-\gamma x^2}$, $x \in \mathbb{R}$, for $\gamma > 0$ satisfy assumptions (H1) and (H4).

We know that the key of solving LSSVM is to solve

$$\boldsymbol{K}_{\mu,\boldsymbol{n}} \boldsymbol{\rho} = \mathbf{1} \qquad \text{and} \qquad \boldsymbol{K}_{\mu,\boldsymbol{n}} \boldsymbol{\nu} = \boldsymbol{y}.$$

If we use the multilevel circulant matrix $\boldsymbol{U}_{\boldsymbol{n}}$ to replace $\boldsymbol{K}_{\boldsymbol{n}}$, the solution could be different. Let $\boldsymbol{\rho}'$ denote the solution obtained using the approximate matrix $\boldsymbol{U}_{\boldsymbol{n}}$. We can write

$$\boldsymbol{\rho} - \boldsymbol{\rho}' = \boldsymbol{K}_{\mu,\boldsymbol{n}}^{-1} \mathbf{1} - \boldsymbol{U}_{\mu,\boldsymbol{n}}^{-1} \mathbf{1} = (\boldsymbol{K}_{\mu,\boldsymbol{n}}^{-1} - \boldsymbol{U}_{\mu,\boldsymbol{n}}^{-1}) \mathbf{1}. \tag{21}$$

Thus, using the Theorem 5, $\|\boldsymbol{\rho}' - \boldsymbol{\rho}\|$ can be bounded as follows:

$$\begin{aligned} \|\boldsymbol{\rho}' - \boldsymbol{\rho}\|_{W_{\mathcal{E}_{r_0}}} &\le \|\boldsymbol{K}_{\mu,\boldsymbol{n}}^{-1} - \boldsymbol{U}_{\mu,\boldsymbol{n}}^{-1}\|_{W_{\mathcal{E}_{r_0}}} \|\mathbf{1}\|_{W_{\mathcal{E}_{r_0}}} \\ &\le c \left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}}^\beta + 1 \right) \mathrm{e}^{-r' \boldsymbol{n}_{\min}} \|\mathbf{1}\|_{W_{\mathcal{E}_{r_0}}}. \end{aligned} \tag{22}$$

From Equation (20) and (19),

$$\|\mathbf{1}\|_{W_{\mathcal{E}_{r_0}}} = \sup_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} W_{\mathcal{E}_{r_0},\boldsymbol{n}}(\boldsymbol{j}) = \sup_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} \mathrm{e}^{r_0 \nu_{\boldsymbol{n}}(\boldsymbol{j})} = \mathrm{e}^{\frac{1}{2} r_0 \Omega},$$

where $\Omega = \left(n_0^2 + n_1^2 + \cdots + n_{p-1}^2\right)^{\frac{1}{2}}$. Therefore,

$$\begin{aligned} \|\boldsymbol{\rho}' - \boldsymbol{\rho}\|_{W_{\mathcal{E}_{r_0}}} &\le c \left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}}^\beta + 1 \right) \mathrm{e}^{-r' \boldsymbol{n}_{\min}} \mathrm{e}^{\frac{1}{2} r_0 \Omega} \\ &= c \left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}}^\beta + 1 \right) \mathrm{e}^{\frac{1}{2} r_0 \Omega - r' \boldsymbol{n}_{\min}}. \end{aligned} \tag{23}$$

Replacing $\mathbf{1}$ with $\boldsymbol{y}$, we can obtain the similar bound for $\|\boldsymbol{\nu}' - \boldsymbol{\nu}\|$,

$$\|\boldsymbol{\nu}' - \boldsymbol{\nu}\|_{W_{\mathcal{E}_{r_0}}} \le c \left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|_{W_{\mathcal{E}_r}}^\beta + 1 \right) \mathrm{e}^{\frac{1}{2} r_0 \Omega - r' \boldsymbol{n}_{\min}}. \tag{24}$$

As the assumptions, no approximation affects $\boldsymbol{k}_x$ and the offset b is a constants φ, so the approximate decision function f' is given by $f'(\boldsymbol{x}) = [\boldsymbol{\alpha}'; \varphi]^{\mathrm{T}}[\boldsymbol{k}_x; 1]$. Therefore,

$$\begin{aligned} f'(\boldsymbol{x}) - f(\boldsymbol{x}) &= \left(\begin{bmatrix} \boldsymbol{\alpha}' \\ \varphi \end{bmatrix}^{\mathrm{T}} - \begin{bmatrix} \boldsymbol{\alpha} \\ \varphi \end{bmatrix}^{\mathrm{T}} \right) \begin{bmatrix} \boldsymbol{k}_x \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\alpha}' - \boldsymbol{\alpha} \\ 0 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{k}_x \\ 1 \end{bmatrix} \\ &= (\boldsymbol{\alpha}' - \boldsymbol{\alpha})^{\mathrm{T}} \boldsymbol{k}_x. \end{aligned}$$

It is easy to see that the norm $\|\cdot\|_{W\mathcal{E}_r}$ satisfies the triangle inequality. We can obtain

$$\|f'(\boldsymbol{x}) - f(\boldsymbol{x})\|_{W\mathcal{E}_{r_0}} \leq \|\boldsymbol{\alpha}' - \boldsymbol{\alpha}\|_{W\mathcal{E}_{r_0}} \|\boldsymbol{k}_x\|_{W\mathcal{E}_{r_0}}. \tag{25}$$

For RBF kernel, $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa = 1$, so $\|\boldsymbol{k}_x\|_{W\mathcal{E}_{r_0}} \leq \mathrm{e}^{\frac{1}{2}r_0\Omega}$.

From Equation (10), we know that $\boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho}b = \boldsymbol{\nu} - \boldsymbol{\rho}\varphi$, so

$$\begin{aligned} \|\boldsymbol{\alpha}' - \boldsymbol{\alpha}\|_{W\mathcal{E}_{r_0}} &\leq \|\boldsymbol{\nu}' - \boldsymbol{\nu}\|_{W\mathcal{E}_{r_0}} + \varphi\|\boldsymbol{\rho} - \boldsymbol{\rho}'\|_{W\mathcal{E}_{r_0}} \\ &\leq c(1+\varphi)\left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|^{\beta}_{W\mathcal{E}_r} + 1\right) \mathrm{e}^{\frac{1}{2}r_0\Omega - r'\boldsymbol{n}_{\min}}. \end{aligned}$$

Therefore, we have

$$\begin{aligned} \|f'(\boldsymbol{x}) - f(\boldsymbol{x})\|_{W\mathcal{E}_{r_0}} &\leq c(1+\varphi)\left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|^{\beta}_{W\mathcal{E}_r} + 1\right) \mathrm{e}^{\frac{1}{2}r_0\Omega - r'\boldsymbol{n}_{\min}} \mathrm{e}^{\frac{1}{2}r_0\Omega} \\ &= c(1+\varphi)\left(\|X_{\boldsymbol{n}} - M_{\boldsymbol{n}}\|^{\beta}_{W\mathcal{E}_r} + 1\right) \mathrm{e}^{r_0\Omega - r'\boldsymbol{n}_{\min}}. \end{aligned} \tag{26}$$

Equation (26) measures the effect of kernel matrix approximation on the decision function generated by LSSVM. It enables us to bound the relative performance of LSSVM when the multilevel circulant matrix is used to approximate the kernel matrix.

5. Consistency of Approximate Model Selection

In this section, we take a model selection approach proposed by Song and Xu (2010a) as an example to demonstrate the consistency of approximate model selection when multilevel circulant matrices are used.

LSSVM is a regularized kernel-based learning algorithm, so we could define a cost functional as

$$Q(f, K) := \frac{1}{\Pi_{\boldsymbol{n}}} \left[\sum_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} (f(\boldsymbol{x}_{\boldsymbol{j}}) - y_{\boldsymbol{j}})^2 + \mu(f, f)_{\mathcal{H}_K} \right],$$

where $\mathcal{H}_K$ is the reproducing kernel Hilbert space (RKHS) with inner product $(\cdot,\cdot)_{\mathcal{H}_K}$. For a given kernel K, the target function f_K is chosen to be the minimizer of the above functional in $\mathcal{H}_K$. That is,

$$f_K := \arg\min_{f \in \mathcal{H}_K} Q(f, K).$$

The observed output $\tilde{\boldsymbol{y}} := [\tilde{y}_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ is corrupted by some random noises, that is,

$$\tilde{y}_{\boldsymbol{j}} = y_{\boldsymbol{j}} + \xi_{\boldsymbol{j}}, \quad \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}},$$

where $\boldsymbol{y} := [y_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ is the unknown true output and $\xi_{\boldsymbol{j}}$ is a random variable with mean 0 and variance σ^2. Let

$$R(K, \tilde{\boldsymbol{y}}) := \min_{f \in \mathcal{H}_K} \frac{1}{\Pi_{\boldsymbol{n}}} \left[\sum_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} (f(\boldsymbol{x}_{\boldsymbol{j}}) - \tilde{y}_{\boldsymbol{j}})^2 + \mu (f, f)_{\mathcal{H}_K} \right]. \tag{27}$$

Since $\xi_{\boldsymbol{j}}$ is a random variable, so are $\tilde{y}_{\boldsymbol{j}}$ and $R(K, \tilde{\boldsymbol{y}})$. A cost functional $R(K)$ is defined as the expectation of $R(K, \tilde{\boldsymbol{y}})$, namely,

$$R(K) := E[R(K, \tilde{\boldsymbol{y}})]. \tag{28}$$

$R(K)$ can be divided into two parts as follows (Song and Xu, 2010a):

$$\begin{aligned} R(K) &= M(\boldsymbol{K}_{\boldsymbol{n}}) + V(\boldsymbol{K}_{\boldsymbol{n}}) \\ &= \frac{\mu}{\Pi_{\boldsymbol{n}}} \boldsymbol{y} (\boldsymbol{K}_{\boldsymbol{n}} + \mu \boldsymbol{I}_{\boldsymbol{n}})^{-1} \boldsymbol{y}^{\mathrm{T}} + \frac{\mu \delta^2}{\Pi_{\boldsymbol{n}}} \sum_{\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}} \frac{1}{\lambda_{\boldsymbol{j}} + \mu}, \end{aligned} \tag{29}$$

where $\lambda_{\boldsymbol{j}}, \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}$ are the eigenvalues of $\boldsymbol{K}_{\boldsymbol{n}}$.

Therefore, for a prescribed set of kernel functions $\mathcal{K}$, we can take $R(K)$ as a model selection criterion to select the optimal kernel function K^* by minimizing it, i.e.

$$K^* = \arg\min_{K \in \mathcal{K}} R(K).$$

For equation (29), computing $M(\boldsymbol{K}_{\boldsymbol{n}})$ requires finding the inverse of the full matrix $\boldsymbol{K}_{\boldsymbol{n}} + \mu \boldsymbol{I}_{\boldsymbol{n}}$ and computing $V(\boldsymbol{K}_{\boldsymbol{n}})$ requires calculating all eigenvalues of $\boldsymbol{K}_{\boldsymbol{n}}$. Therefore, computing $R(K)$ is computationally expensive. We could also use a circulant matrix $\boldsymbol{U}_{\boldsymbol{n}}$ to replace the kernel matrix $\boldsymbol{K}_{\boldsymbol{n}}$ to reduce the computational cost. However, we need a theoretical guarantee to show the rationality of such approximation. To this end, we introduce the following theorem (Song, 2010).

Theorem 6 *If the assumptions (H1), (H2), (H3) in Theorem 5 and*

(H5) there exist positive constants c_1 and c_2 such that for any $\boldsymbol{n} \in \mathbb{N}^p$, $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$,

$$|\|\boldsymbol{x}_{\boldsymbol{j}} - \boldsymbol{x}_{\boldsymbol{l}}\|_2 - \|[(j_s - l_s) h_{\boldsymbol{n},s} : s \in \mathbb{N}_p]\|_2| \le c_1 \sum_{s \in \mathbb{N}_p} \left(\mathrm{e}^{-c_2 \delta_{n_s}(j_s)} + \mathrm{e}^{-c_2 \delta_{n_s}(l_s)} \right),$$

where $\delta_n(j) = \frac{n}{2} - |\frac{n}{2} - j|$,

hold, then there exists a positive constant c such that for any $\boldsymbol{n} \in \mathbb{N}^p$,

$$|V(\boldsymbol{U}_{\boldsymbol{n}}) - V(\boldsymbol{K}_{\boldsymbol{n}})| \le c (\boldsymbol{n}_{\min})^{-1}.$$

If in addition, there exist positive constants c_3 and r_1 such that for any $\boldsymbol{n} \in \mathbb{N}^p$ and $\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}$,

$$|y_{\boldsymbol{j}}| \le c_3 \mathrm{e}^{-r_1 \nu_{\boldsymbol{n}}(\boldsymbol{j})},$$

then there exist positive constants c and r such that for any $\boldsymbol{n} \in \mathbb{N}^p$

$$|M(\boldsymbol{U}_{\boldsymbol{n}}) - M(\boldsymbol{K}_{\boldsymbol{n}})| \le c \Pi_{\boldsymbol{n}}^{-1/2} \mathrm{e}^{-r \boldsymbol{n}_{\min}}.$$

We denote $\widetilde{R}(K) = M(\boldsymbol{U_n}) + V(\boldsymbol{U_n})$. By Theorem 6 and the triangle inequality, we could directly derive the following theorem.

Theorem 7 *If the assumptions (H1), (H2), (H3), (H5) hold, and there exist positive constants c and r such that for any $\boldsymbol{n} \in \mathbb{N}^p$ and $\boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}$, $|y_{\boldsymbol{j}}| \leq c\mathrm{e}^{-r\nu_{\boldsymbol{n}}(\boldsymbol{j})}$, then*

$$\lim_{\boldsymbol{n}\to\infty} |\widetilde{R}(K) - R(K)| = 0,$$

where $\boldsymbol{n} \to \infty$ means all of its components goes to infinity.

Theorem 7 shows that, for the regularized kernel-based learning algorithm (such as LSSVM), if we use the approximate model selection criterion $\widetilde{R}(K)$ produced with multilevel circulant matrices, we could also obtain the optimal kernel as the accurate model selection criterion $R(K)$ does, which shows the consistency of approximate model selection.

6. Approximate Model Selection for LSSVM

In previous section, we introduce a model selection approach, which focuses on the kernel selection. The selection of regularization parameter μ has not been discussed. Actually, there are many model selection approaches, which could simultaneously select the kernel and regularization parameter, such as the cross validation, radius margin bound (Chapelle et al., 2002), PRESS criterion (Cawley and Talbot, 2004) and so on. However, when optimizing model selection criteria, all these approaches need to train LSSVM in the inner layer for every iteration.

In this section, we discuss the problem of approximate model selection. We argue that for model selection purpose, it is sufficient to calculate an approximate criterion that can discriminate the optimal models from candidates. This argument has been supported by Theorem 7. In the following, we present an approximate model selection scheme, as shown in Algorithm 2. We use the RBF kernel $K(\boldsymbol{x_j}, \boldsymbol{x_l}) = \exp\left(-\gamma\|\boldsymbol{x_j} - \boldsymbol{x_l}\|^2\right)$, $\boldsymbol{j}, \boldsymbol{l} \in \mathbb{N}_{\boldsymbol{n}}$ to describe the scheme.

Algorithm 2: Approximate Model Selection Scheme for LSSVM

Input: $\mathcal{D} := \{\boldsymbol{x_j} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\}$, $\boldsymbol{y} := \{y_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}\}$;
Output: $(\gamma, \mu)_{\text{opt}}$;
Initialize: $(\gamma, \mu) = (\gamma^0, \mu^0)$;
repeat
- **1:** Calculate $[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ according to (12) and (13);
- **2:** Calculate $\boldsymbol{\alpha}$ and b for LSSVM with fixed (γ, μ) using Algorithm 1;
- **3:** Calculate model selection criterion T using $\boldsymbol{\alpha}$ and b;
- **4:** Update (γ, μ) to minimize T;

until *the criterion T is minimized* ;
return $(\gamma, \mu)_{\text{opt}}$;

Let S denote the iteration steps of optimizing model selection criteria. The complexity of solving LSSVM by calculating the inverse of the kernel matrix is $O(\Pi_{\boldsymbol{n}}^3)$. For radius margin bound or span bound (Chapelle et al., 2002), a standard LSSVM needs to be solved in the inner layer for each iteration, so the total complexity of these two methods is $O(S\Pi_{\boldsymbol{n}}^3)$.

For PRESS criterion (Cawley and Talbot, 2004), the inverse of kernel matrix still need to be calculated for each iteration, so its complexity is $O(S\Pi_{\boldsymbol{n}}^3)$. From Theorem 3, we know that using Algorithm 1, we could solve LSSVM in $O(\Pi_{\boldsymbol{n}} \log(\Pi_{\boldsymbol{n}}))$ complexity. The complexity of calculating $[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ is $O(p2^p\Pi_{\boldsymbol{n}})$, where p is the number of levels of $\boldsymbol{U}_{\boldsymbol{n}}$. p is a small constant and usually set to be 2 or 3. Therefore, if we use the above model selection criteria in the outer layer, the complexity of approximate model selection is $O(S\Pi_{\boldsymbol{n}}(p2^p + \log(\Pi_{\boldsymbol{n}})))$. For t-fold cross validation, let S_γ and S_μ denote the grid steps of γ and μ. If LSSVM is directly solved, the complexity of t-fold cross validation is $O(tS_\gamma S_\mu \Pi_{\boldsymbol{n}}^3)$. However, the complexity of approximate model selection using t-fold cross validation as outer layer criterion will be $O(tS_\gamma\Pi_{\boldsymbol{n}}(p2^p + S_\mu \log(\Pi_{\boldsymbol{n}})))$, since the calculation of $[u_{\boldsymbol{j}} : \boldsymbol{j} \in \mathbb{N}_{\boldsymbol{n}}]$ only need the kernel parameter γ.

7. Experiments

In this section, we present experiments on several benchmark datasets to demonstrate the effectiveness of approximate model selection.

7.1. Experimental Scheme

The benchmark datasets used in our experiments are introduced in Rätsch et al. (2001) and widely used for the model selection purpose (Chapelle et al., 2002; Chen et al., 2009), as shown in Table 1. For each dataset, there are 100 random training and test pre-defined partitions[1] (except 20 for the Image dataset). The use of multiple benchmarks means that the evaluation is more robust as the selection of data sets that provide a good match to the inductive bias of a particular classifier becomes less likely. Likewise, the use of multiple partitions provides robustness against sensitivity to the sampling of data to form training and test sets.

Table 1: Datasets used in experiments.

Dataset	Features	Training	Test	Replications
Thyroid	5	140	75	100
Titanic	3	150	2051	100
Heart	13	170	100	100
Breast	9	200	77	100
Banana	2	400	4900	100
Twonorm	20	400	7000	100
Diabetes	8	468	300	100
Flare solar	9	666	400	100
German	20	700	300	100
Image	18	1300	1010	20

In Rätsch et al. (2001), model selection is performed on the first five training sets of each dataset. The median values of the hyperparameters over these five sets are then determined and subsequently used to evaluate the error rates throughout all 100 partitions. However, for this experimental scheme, some of the test data is no longer statistically "pure" since it has

1. http://www.fml.tuebingen.mpg.de/Members/raetsch/benchmark

been used during model selection. Furthermore, the use of median of the hyperparameters would introduce an optimistic bias (Cawley and Talbot, 2010). In our experiments, we perform model selection on the training set of each partition, then train the classifier with the obtained optimal hyperparameters on the same training set, and finally evaluate the classifier on the corresponding test set. Therefore, we can obtain 100 test error rates for each dataset (except 20 for the Image dataset). The statistical analysis of these test error rates is conducted to evaluate the performance of the model selection approach. This experimental scheme is rigorous and can avoid the major flaws of the previous one (Cawley and Talbot, 2010). All experiments are performed on a Core2 Quad PC, with 2.33GHz CPU and 4GB memory.

7.2. Effectiveness

Following the experimental setup in Section 7.1, we perform model selection respectively using 5-fold cross validation (5-fold CV) and approximate 5-fold CV, that is, approximate model selection by minimizing 5-fold CV error (as shown in Algorithm 2). The CV is performed on a 13×11 grid of (γ, μ) respectively varying in $[2^{-15}, 2^{9}]$ and $[2^{-15}, 2^{5}]$ both with step 2^2. The number of levels of multilevel circulant matrices approximation is 2.

We compare effectiveness of two model selection approaches. Effectiveness includes efficiency and generalization. Efficiency is measured by average computation time for model selection. Generalization is measured by the mean test error rate (TER) of the classifiers trained with the optimal hyperparameters produced by different model selection approaches.

Results are shown in Table 2. We use the z statistic of TER (Cawley and Talbot, 2007) to estimate the statistical significance of differences in performance. Let $\bar{x}$ and $\bar{y}$ represent the means of TER of two approaches, and e_x and e_y the corresponding standard errors, then the z statistic is computed as $z = (\bar{x} - \bar{y})/\sqrt{\mathrm{e}_x^2 + \mathrm{e}_y^2}$ and $z = 1.64$ corresponds to a 95% significance level. From Table 2, approximate 5-fold CV is significantly outperformed by 5-fold CV for none of 10 datasets. Besides, according to the Wilcoxon signed rank test (Demšar, 2006), neither of 5-fold CV and approximate 5-fold CV is statistically superior at the 95% level of significance.

However, Table 2 also shows that approximate 5-fold CV is more efficient than 5-fold CV on all datasets. It is worth noting that the larger the training set size is, the efficiency gain is more obvious, which is in accord with the results of complexity analysis.

8. Conclusion

In this paper, multilevel circulant matrices were first introduced into the model selection problem. A brand new approximate model selection approach of LSSVM was proposed, which fully exploits the theoretical and computational virtue of multilevel circulant matrices. We designed an efficient algorithm for solving LSSVM and bounded the effect of kernel matrix approximation on the decision function of LSSVM. We demonstrated the consistency of approximate model selection. With consistency as a theoretical support, we presented an approximate model selection scheme and analyzed its complexity as compared with other classic model selection approaches. This complexity shows the promise of the application of

Table 2: Comparison of computation time and test error rate (TER) of 5-fold cross validation (5-fold CV) and approximate 5-fold CV

Dataset	5-fold CV		Approximate 5-fold CV	
	Time(s)	TER(%)	Time(s)	TER(%)
Thyroid	0.938	5.000±2.580	0.774	4.773±2.291
Titanic	0.854	22.534±0.688	0.850	22.897±1.427
Heart	0.986	16.200±3.259	0.889	18.920±4.576
Breast	1.475	26.831±4.578	1.010	27.831±5.569
Banana	6.287	10.781±0.721	1.858	11.283±0.992
Twonorm	6.362	2.560±0.310	1.937	2.791±0.566
Diabetes	10.042	23.493±1.663	2.104	26.386±4.501
Flare solar	18.172	34.172±1.863	4.200	36.440±2.752
German	23.058	23.793±2.283	4.239	25.080±2.375
Image	134.680	3.014±0.877	11.875	4.391±0.631

approximate model selection for large scale problems. We finally verified the effectiveness of our approach on several benchmark datasets.

The application of our theoretical results and approach to real large problems will be one of major concerns. Besides, a new efficient model selection criterion directly dependent on kernel matrix approximation will be proposed in near future.

Acknowledgments

The authors would like to thank the anonymous reviewers for their detailed comments and suggestions that help to improve the quality of the paper. The work was supported in part by the National Natural Science Foundation of China under grant No. 61170019 and the Natural Science Foundation of Tianjin under grant No. 11JCYBJC00700.

References

G.C. Cawley and N.L.C. Talbot. Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. *Neural Networks*, 17:1467–1475, 2004.

G.C. Cawley and N.L.C. Talbot. Preventing over-fitting during model selection via Bayesian regularisation of the hyper-parameters. *Journal of Machine Learning Research*, 8:841–861, 2007.

G.C. Cawley and N.L.C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11:2079–2107, 2010.

O. Chapelle and V. Vapnik. Model selection for support vector machines. In *Advances in Neural Information Processing Systems 12*, pages 230–236. MIT Press, Cambridge, MA, 2000.

O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1):131–159, 2002.

H. Chen, P. Tino, and X. Yao. Probabilistic classification vector machines. *IEEE Transactions on Neural Networks*, 20(6):901–914, 2009.

C. Cortes, M. Mohri, and A. Talwalkar. On the impact of kernel approximation on learning accuracy. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, pages 113–120, Sardinia, Italy, 2010.

K. Crammer, J. Keshet, and Y. Singer. Kernel design using boosting. In *Advances in Neural Information Processing Systems 15*, pages 553–560. MIT Press, Cambridge, MA, 2003.

P.J. Davis. *Circulant Matrices*. John Wiley and Sons, New York, NY, 1979.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006. ISSN 1532-4435.

K. Duan, S.S. Keerthi, and A.N. Poo. Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing*, 51:41–59, 2003.

I. Guyon, A. Saffari, G. Dror, and G. Cawley. Model selection: Beyond the Bayesian/frequentist divide. *Journal of Machine Learning Research*, 11:61–87, 2010.

A. Luntz and V. Brailovsky. On estimation of characters obtained in statistical procedure of recognition (in Russian). *Technicheskaya Kibernetica*, 3, 1969.

C.A. Micchelli and M. Pontil. Learning the kernel function via regularization. *Journal of Machine Learning Research*, 6:1099–1125, 2006.

G. Rätsch, T. Onoda, and K.R. Müller. Soft margins for AdaBoost. *Machine Learning*, 42(3): 287–320, 2001.

G. Song. *Approximation of kernel matrices in machine learning*. PhD thesis, Syracuse University, 2010.

G. Song and Y. Xu. Approximation of kernel matrices by circulant matrices and its application in kernel selection methods. *Frontiers of mathematics in China*, 5(1):123–160, 2010a.

G. Song and Y. Xu. Approximation of high-dimensional kernel matrices by multilevel circulant matrices. *Journal of Complexity*, 26(4):375–405, 2010b.

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

E.E. Tyrtyshnikov. A unifying approach to some old and new theorems on distribution and clustering. *Linear Algebra and its Applications*, 232:1–43, 1996.

T. Van Gestel, J.A.K. Suykens, B. Baesens, S. Viaene, J. Vanthienen, G. Dedene, B. De Moor, and J. Vandewalle. Benchmarking least squares support vector machine classifiers. *Machine Learning*, 54(1):5–32, 2004.

V. Vapnik and O. Chapelle. Bounds on error expectation for support vector machines. *Neural Computation*, 12(9):2013–2036, 2000.

V.N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, NY, 1998.