Nyström Approximate Model Selection for LSSVM

Lizhong Ding and Shizhong Liao

School of Computer Science and Technology
Tianjin University, Tianjin 300072, China
szliao@tju.edu.cn

Abstract. Model selection is critical to least squares support vector machine (LSSVM). A major problem of existing model selection approaches is that a standard LSSVM needs to be solved with $O(n^3)$ complexity for each iteration, where n is the number of training examples. In this paper, we propose an approximate approach to model selection of LSSVM. We use Nyström method to approximate a given kernel matrix by a low rank representation of it. With such approximation, we first design an efficient LSSVM algorithm, and then theoretically analyze the effect of kernel matrix approximation on the decision function of LSSVM. Based on the matrix approximation error bound of Nyström method, we derive a model approximation error bound, which is a theoretical guarantee of approximate model selection. We finally present an approximate model selection scheme, whose complexity is lower than existing approaches. Experimental results on benchmark datasets demonstrate the effectiveness of approximate model selection.

Keywords: model selection, Nyström method, matrix approximation, least squares support vector machine.

1 Introduction

Support vector machine (SVM) [18] 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) [16] 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. 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 [4]. We usually refer to these parameters collectively as *hyperparameters*. Common model selection approaches mainly adopt a nested two-layer inference [11], where the inner layer trains the classifier for fixed hyperparameters and the outer layer tunes the hyperparameters to minimize the generalization

P.-N. Tan et al. (Eds.): PAKDD 2012, Part I, LNAI 7301, pp. 282–293, 2012.

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 [17,5].

The k-fold cross validation gives an excellent estimate of the generalization error [9] and the extreme form of cross validation, leave-one-out (LOO), provides an almost unbiased estimate of the generalization error [14]. 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 [17] and radius margin bound [5]. 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 [5,3], the 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. We argue that it is sufficient to calculate an approximate criterion that can discriminate the optimal hyperparameters 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 of LSSVM, we consider to approximate a kernel matrix by a "nice" matrix with a lower computational cost when calculating its inverse. The Nyström method is an effective technique for generating a low rank approximation for the given kernel matrix [19,13,8]. Using the low rank approximation, we design an efficient algorithm for solving LSSVM, whose complexity is lower than $O(n^3)$. We further derive a model approximation error bound to measure the effect of Nyström approximation on the decision function of LSSVM. Finally, we present an efficient approximate model selection scheme. It conforms to the two-layer iterative procedure, but the inner computation has been realized more efficiently. By rigorous experiments on several 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 Nyström approximation on the decision function of LSSVM. In Section 5, we present an approximate model selection scheme for LSSVM. In Section 6, 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

$$\mathcal{S} = ((\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)) \in (\mathcal{X} \times \mathcal{Y})^n .$$

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 $(\boldsymbol{w}, b)$ of the linear classifier are given by the minimizer of a regularized least-squares training function

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

where $\mu > 0$ is called regularization parameter. The basic training algorithm for LSSVM [16] views the regularized loss function (1) as a constrained minimization problem

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

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

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

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

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

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

If we let $\boldsymbol{K}_{\mu,n} = \boldsymbol{K} + \mu \boldsymbol{I}_n$, we can write the first row of Equation (4) as

$$\boldsymbol{K}_{\mu,n}(\boldsymbol{\alpha} + \boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{1}b) = \boldsymbol{y}. \tag{5}$$

Therefore, $\boldsymbol{\alpha} = \boldsymbol{K}_{\mu,n}^{-1}(\boldsymbol{y} - \boldsymbol{1}b)$. Replacing $\boldsymbol{\alpha}$ with $\boldsymbol{K}_{\mu,n}^{-1}(\boldsymbol{y} - \boldsymbol{1}b)$ in the second row of Equation (4), we can obtain

$$\boldsymbol{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{1}b = \boldsymbol{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{y}. \tag{6}$$

The system of linear equations (4) can then be rewritten as

$$\begin{bmatrix} \boldsymbol{K}_{\mu,n} & \boldsymbol{0} \\ \boldsymbol{0}^{\mathrm{T}} & \boldsymbol{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} + \boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{1}b \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{1}^{\mathrm{T}}\boldsymbol{K}_{\mu,n}^{-1}\boldsymbol{y} \end{bmatrix}. \tag{7}$$

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

Equation (7) can be solved as follows: we first solve

$$\boldsymbol{K}_{\mu,n}\boldsymbol{\rho} = \mathbf{1} \qquad \text{and} \qquad \boldsymbol{K}_{\mu,n}\boldsymbol{\nu} = \boldsymbol{y}. \tag{8}$$

The solution $(\boldsymbol{\alpha}, b)$ of Equation (4) are then given by

$$b = \frac{\mathbf{1}^{\mathrm{T}}\boldsymbol{\nu}}{\mathbf{1}^{\mathrm{T}}\boldsymbol{\rho}} \qquad \text{and} \qquad \boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho} b. \tag{9}$$

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

If Equation (8) is solved, we can easily obtain the solution of LSSVM. However, the complexity of calculating the inverse of the matrix $\boldsymbol{K}_{\mu,n}$ is $O(n^3)$. In the following, we will demonstrate that Nyström method can be used to speed up this process.

3 Approximating LSSVM Using Nyström Method

We first introduce a fundamental result of matrix computations [10]: for any matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and positive integer k, there exists a matrix $\boldsymbol{A}_k$ such that

$$\|\boldsymbol{A} - \boldsymbol{A}_k\|_{\xi} = \min_{\boldsymbol{D}\in\mathbb{R}^{m\times n}:\mathrm{rank}(\boldsymbol{D})\leq k} \|\boldsymbol{A} - \boldsymbol{D}\|_{\xi}$$

for $\xi = \mathrm{F}, 2$. $\|\cdot\|_{\mathrm{F}}$ and $\|\cdot\|_2$ denote the Frobenius norm and the spectral norm. Such $\boldsymbol{A}_k$ is called the optimal rank k approximation of the matrix $\boldsymbol{A}$. It can be computed through the singular value decomposition (SVD) of $\boldsymbol{A}$. If $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is symmetric positive semi-definite (SPSD), $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}$, where $\boldsymbol{U}$ is a unitary matrix and $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ is a real diagonal matrix with $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$. For $k \leq \mathrm{rank}(\boldsymbol{A})$, $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i \boldsymbol{U}^i \boldsymbol{U}^{i\mathrm{T}}$, where $\boldsymbol{U}^i$ is the ith column of $\boldsymbol{U}$.

We now briefly review the Nyström method [8,19]. Let $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ be an SPSD matrix. The Nyström method generates a low rank approximation of $\boldsymbol{K}$ using a subset of the columns of the matrix. Suppose we randomly sample c columns of $\boldsymbol{K}$ uniformly without replacement. Let $\boldsymbol{C}$ denote the $n \times c$ matrix formed by theses columns. Let $\boldsymbol{W}$ be the $c \times c$ matrix consisting of the intersection of these c columns with the corresponding c rows of $\boldsymbol{K}$. Without loss of generality, we can rearrange the columns and rows of $\boldsymbol{K}$ based on this sampling such that:

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{W} & \boldsymbol{K}_{21}^{\mathrm{T}} \\ \boldsymbol{K}_{21} & \boldsymbol{K}_{22} \end{pmatrix}, \qquad \boldsymbol{C} = \begin{pmatrix} \boldsymbol{W} \\ \boldsymbol{K}_{21} \end{pmatrix}. \tag{10}$$

Since $\boldsymbol{K}$ is SPSD, $\boldsymbol{W}$ is also SPSD. The Nyström method uses $\boldsymbol{W}$ and $\boldsymbol{C}$ from Equation (10) to construct a rank k approximation $\widetilde{\boldsymbol{K}}$ of $\boldsymbol{K}$ for $k \leq c$ defined by:

$$\widetilde{\boldsymbol{K}} = \boldsymbol{C}\boldsymbol{W}_k^{+}\boldsymbol{C}^{\mathrm{T}} \approx \boldsymbol{K}, \tag{11}$$

where $\boldsymbol{W}_k$ is the optimal rank k approximation to $\boldsymbol{W}$ and $\boldsymbol{W}_k^{+}$ is the Moore-Penrose generalized inverse of $\boldsymbol{W}_k$. Since $\boldsymbol{W}$ is SPSD, $\boldsymbol{W}_k = \sum_{i=1}^{k} \sigma_i \boldsymbol{U}^i \boldsymbol{U}^{i\mathrm{T}}$ and therefore $\boldsymbol{W}_k^{+} = \sum_{i=1}^{k} \sigma_i^{-1} \boldsymbol{U}^i \boldsymbol{U}^{i\mathrm{T}}$ for $k \leq \mathrm{rank}(\boldsymbol{W})$.

If we write the SVD of $\boldsymbol{W}$ as $\boldsymbol{W} = U_W \boldsymbol{\Sigma}_W \boldsymbol{U}_W^{\mathrm{T}}$, then

$$\boldsymbol{W}_k^+ = \boldsymbol{U}_{W,k} \boldsymbol{\Sigma}_{W,k}^+ \boldsymbol{U}_{W,k}^{\mathrm{T}}, \tag{12}$$

where $\Sigma_{W,k}$ and $\boldsymbol{U}_{W,k}$ correspond the top k singular values and singular vectors of $\boldsymbol{W}$. The diagonal elements of $\Sigma_{W,k}$ are all positive, since $\boldsymbol{W}$ is SPSD and $k \leq \mathrm{rank}(\boldsymbol{W})$.

If we plug Equation (12) into Equation (11), we can obtain

$$\begin{aligned} \widetilde{\boldsymbol{K}} &= \boldsymbol{C}\boldsymbol{U}_{W,k} \boldsymbol{\Sigma}_{W,k}^+ \boldsymbol{U}_{W,k}^{\mathrm{T}} \boldsymbol{C}^{\mathrm{T}} \\ &= \underbrace{\boldsymbol{C}\boldsymbol{U}_{W,k} \sqrt{\boldsymbol{\Sigma}_{W,k}^+}}_{\boldsymbol{V}} \underbrace{\left(\boldsymbol{C}\boldsymbol{U}_{W,k} \sqrt{\boldsymbol{\Sigma}_{W,k}^+}\right)^{\mathrm{T}}}_{\boldsymbol{V}^{\mathrm{T}}}, \end{aligned} \tag{13}$$

where we let $\boldsymbol{V} := \boldsymbol{C}\boldsymbol{U}_{W,k} \sqrt{\boldsymbol{\Sigma}_{W,k}^+} \in \mathbb{R}^{n \times k}$.

For LSSVM, we need to solve the inverse of $\boldsymbol{K} + \mu \boldsymbol{I}_n$. To reduce the computational cost, we intend to use the inverse of $\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n$ as an approximation of the inverse of $\boldsymbol{K} + \mu \boldsymbol{I}_n$. Since $\boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}$ is positive semi-definite, the invertibility of $\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_{\boldsymbol{n}}$ is guaranteed.

To efficiently calculate the inverse of $\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n$, we further introduce the Woodbury formula [12]

$$(\boldsymbol{A} + \boldsymbol{X}\boldsymbol{Y}\boldsymbol{Z})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{X}\left(\boldsymbol{Y}^{-1} + \boldsymbol{Z}\boldsymbol{A}^{-1}\boldsymbol{X}\right)^{-1} \boldsymbol{Z}\boldsymbol{A}^{-1}, \tag{14}$$

where $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, $\boldsymbol{X} \in \mathbb{R}^{n \times k}$, $\boldsymbol{Y} \in \mathbb{R}^{k \times k}$ and $\boldsymbol{Z} \in \mathbb{R}^{k \times n}$.

Now, we can obtain

$$\begin{aligned} &(\mu \boldsymbol{I}_n + \boldsymbol{K})^{-1} \\ &\approx \left(\mu \boldsymbol{I}_n + \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}\right)^{-1} \\ &= \frac{1}{\mu}\left(\boldsymbol{I}_n - \boldsymbol{V}\left(\mu \boldsymbol{I}_k + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\right)^{-1} \boldsymbol{V}^{\mathrm{T}}\right). \end{aligned} \tag{15}$$

The last equality of Equation (15) is directly derived from the Woodbury formula with $\boldsymbol{A} = \mu \boldsymbol{I}_n$, $\boldsymbol{X} = \boldsymbol{V}$, $\boldsymbol{Y} = \boldsymbol{I}_k$ and $\boldsymbol{Z} = \boldsymbol{V}^{\mathrm{T}}$.

The essential step of solving LSSVM is to solve Equation (8). If we let $\boldsymbol{u} = [\boldsymbol{\rho}, \ \boldsymbol{\nu}]$ and $\boldsymbol{z} = [\mathbf{1}, \ \boldsymbol{y}]$, Equation (8) is equivalent to

$$(\mu \boldsymbol{I}_n + \boldsymbol{K})\boldsymbol{u} = \boldsymbol{z}.$$

Using Equation (15) to replace $\mu \boldsymbol{I}_n + \boldsymbol{K}$ with $\mu \boldsymbol{I}_n + \widetilde{\boldsymbol{K}}$, we can obtain

$$\boldsymbol{u} = \frac{1}{\mu}\left(\boldsymbol{z} - \boldsymbol{V}\left(\mu \boldsymbol{I}_k + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\right)^{-1} \boldsymbol{V}^{\mathrm{T}}\boldsymbol{z}\right). \tag{16}$$

We further introduce a temporary variable $\boldsymbol{t}$ to efficiently solve Equation (16):

$$\begin{aligned} \boldsymbol{t} &: \ \left(\mu \boldsymbol{I}_k + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\right)\boldsymbol{t} = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{z}, \\ \boldsymbol{u} &= \frac{1}{\mu}(\boldsymbol{z} - \boldsymbol{V}\boldsymbol{t}). \end{aligned} \tag{17}$$

We now present an algorithm of solving LSSVM (Algorithm 1).

We estimate the computational complexity of Algorithm 1 in Theorem 1.

Algorithm 1. Approximating LSSVM using Nyström method

```
Input: $n \times n$ kernel matrix $\boldsymbol{K}$, label vector $\boldsymbol{y}$, $c < n$, $k < c$, $\mu$;
Output: $(\alpha, b)$;
1: Calculate $\boldsymbol{C}$, $\boldsymbol{U}_{W,k}$ and $\boldsymbol{\Sigma}^{+}_{W,k}$ according to (10) and (12) using Nyström method;
2: Calculate $\boldsymbol{V} = \boldsymbol{C}\boldsymbol{U}_{W,k}\sqrt{\boldsymbol{\Sigma}^{+}_{W,k}}$ according to (13);
3: Let $\boldsymbol{z} = [\boldsymbol{1}, \boldsymbol{y}]$ and solve the linear system $\left(\mu \boldsymbol{I}_k + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\right)\boldsymbol{t} = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{z}$ to obtain $\boldsymbol{t}$;
4: Calculate $\boldsymbol{u} = \frac{1}{\mu}(\boldsymbol{z} - \boldsymbol{V}\boldsymbol{t})$ and let $\boldsymbol{\rho}$, $\boldsymbol{\nu}$ be the first and second column of $\boldsymbol{u}$;
5: Calculate $b = \frac{\boldsymbol{1}^{\mathrm{T}}\boldsymbol{\nu}}{\boldsymbol{1}^{\mathrm{T}}\boldsymbol{\rho}}$ and $\boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho} b$ according to (9);
return $(\alpha, b)$;
```

Theorem 1. *The computational complexity of Algorithm 1 is $O(c^3 + nck)$.*

Proof. The computational complexity of step 1 is $O(c^3)$, since the main computational part of this step is the SVD on $\boldsymbol{W}$. In step 2, matrix multiplications are required, so its complexity is $O(kcn)$. In step 3, the inverse of $\left(\mu \boldsymbol{I}_k + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\right)$ is solved by computing Cholesky factorization of it with the complexity $O(k^3)$. The complexity of $\boldsymbol{V}^{\mathrm{T}}\boldsymbol{z}$ is $O(nk)$. The last matrix multiplication to obtain $\boldsymbol{t}$ requires $O(k^2)$. Therefore the total complexity of step 3 is $O(k^3 + nk)$. The complexity of step 4 is $O(nk)$. The complexity of step 5 is $O(n)$, since the multiplication and subtraction between two vectors need to be done. For Nyström approximation, we have $k < c < n$, so the total complexity of Algorithm 1 is $O(c^3 + nck)$. For large scale problems, we usually set $c \ll n$.

Compared to Related Work. Theorem 1 shows that if Nyström approximation is given, we can solve LSSVM in $O(k^3)$. Williams et al. [19] used Nyström method to speed up Gaussian Process (GP) regression. After Nyström approximation was given, they solved GP regression with $O(nk^2)$ complexity. Cortes et al. [6] scaled kernel ridge regression (KRR) using Nyström method. The complexity of their method is $O(n^2c)$ with Nyström approximation (Section 3.3 of [6]).

4 Error Analysis

In this section, we analyze the effect of Nyström approximation on the decision function of LSSVM.

We assume that approximation is only used in training. At testing time the true kernel function is used. This scenario has been considered by [6]. The decision function f derived with the exact kernel matrix K is defined by

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b = \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{k}_x \\ 1 \end{bmatrix},$$

where $\boldsymbol{k}_x = (K(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_n))^{\mathrm{T}}$. We define $\kappa > 0$ such that $K(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa$ and $\widetilde{K}(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa$.

We first consider the effect of Nyström approximation on $\boldsymbol{\rho}$ of Equation (8). Let $\boldsymbol{\rho}'$ denote the solution of $(\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n)\boldsymbol{\rho}' = \mathbf{1}$. We can write

$$\begin{aligned} \boldsymbol{\rho}' - \boldsymbol{\rho} &= (\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n)^{-1}\mathbf{1} - (\boldsymbol{K} + \mu \boldsymbol{I}_n)^{-1}\mathbf{1} \\ &= -\left[(\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n)^{-1}(\widetilde{\boldsymbol{K}} - \boldsymbol{K})(\boldsymbol{K} + \mu \boldsymbol{I}_n)^{-1}\right]\mathbf{1}. \end{aligned} \tag{18}$$

For last equality, we used the identity $\boldsymbol{A}^{-1} - \boldsymbol{B}^{-1} = -\boldsymbol{A}^{-1}(\boldsymbol{A} - \boldsymbol{B})\boldsymbol{B}^{-1}$ for any two invertible matrices $\boldsymbol{A}, \boldsymbol{B}$. Thus, $||\boldsymbol{\rho}' - \boldsymbol{\rho}||_2$ can be bounded as follows:

$$\begin{aligned} ||\boldsymbol{\rho}' - \boldsymbol{\rho}||_2 &\leq ||(\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n)^{-1}||_2 \; ||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2 \; ||(\boldsymbol{K} + \mu \boldsymbol{I}_n)^{-1}||_2 \; ||\mathbf{1}||_2 \\ &\leq \frac{||\mathbf{1}||_2}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2 = \frac{\sqrt{n}}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2. \end{aligned} \tag{19}$$

Since $\widetilde{\boldsymbol{K}}$ and $\boldsymbol{K}$ are positive semi-definite matrices, the eigenvalues of $\widetilde{\boldsymbol{K}}+\mu \boldsymbol{I}_n$ and $\boldsymbol{K}+\mu \boldsymbol{I}_n$ are larger than or equal to μ. Therefore the eigenvalues of $(\widetilde{\boldsymbol{K}} + \mu \boldsymbol{I}_n)^{-1}$ and $(\boldsymbol{K} + \mu \boldsymbol{I}_n)^{-1}$ are less than or equal to $1/\mu$.

We further consider $\boldsymbol{\nu}$ of Equation (8). Replacing $\mathbf{1}$ with $\boldsymbol{y}$, we can obtain the similar bound

$$||\boldsymbol{\nu}' - \boldsymbol{\nu}||_2 \leq \frac{||\boldsymbol{y}||_2}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2 = \frac{\sqrt{n}}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2. \tag{20}$$

As the assumptions, we use the true kernel function at testing time, so no approximation affects $\boldsymbol{k}_x$. For simplicity, we assume the offset b to be a constant ζ. Therefore, the approximate decision function f' is given by $f'(\boldsymbol{x}) = [\boldsymbol{\alpha}'; \zeta]^{\mathrm{T}}[\boldsymbol{k}_x; 1]$.

We can obtain

$$f'(\boldsymbol{x}) - f(\boldsymbol{x}) = \left(\begin{bmatrix}\boldsymbol{\alpha}' \\ \zeta\end{bmatrix}^{\mathrm{T}} - \begin{bmatrix}\boldsymbol{\alpha} \\ \zeta\end{bmatrix}^{\mathrm{T}}\right)\begin{bmatrix}\boldsymbol{k}_x \\ 1\end{bmatrix} = \begin{bmatrix}\boldsymbol{\alpha}' - \boldsymbol{\alpha} \\ 0\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\boldsymbol{k}_x \\ 1\end{bmatrix} = (\boldsymbol{\alpha}' - \boldsymbol{\alpha})^{\mathrm{T}}\boldsymbol{k}_x. \tag{21}$$

By Schwarz inequality,

$$|f'(\boldsymbol{x}) - f(\boldsymbol{x})| \leq ||\boldsymbol{\alpha}' - \boldsymbol{\alpha}||_2||\boldsymbol{k}_x||_2 = \sqrt{n}\kappa||\boldsymbol{\alpha}' - \boldsymbol{\alpha}||_2. \tag{22}$$

From Equation (9), we know that $\boldsymbol{\alpha} = \boldsymbol{\nu} - \boldsymbol{\rho}b = \boldsymbol{\nu} - \boldsymbol{\rho}\zeta$, so

$$\begin{aligned} ||\boldsymbol{\alpha}' - \boldsymbol{\alpha}||_2 &\leq ||\boldsymbol{\nu}' - \boldsymbol{\nu}||_2 + \zeta||\boldsymbol{\rho} - \boldsymbol{\rho}'||_2 \\ &\leq \frac{\sqrt{n}}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2 + \zeta\left(\frac{\sqrt{n}}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2\right) \\ &\leq (1 + \zeta)\frac{\sqrt{n}}{\mu^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2. \end{aligned} \tag{23}$$

We let $\mu_0 = \mu/n$. Substituting the upper bound of $||\boldsymbol{\alpha}' - \boldsymbol{\alpha}||_2$ into Equation (22), we can obtain

$$|f'(\boldsymbol{x}) - f(\boldsymbol{x})| \leq \sqrt{n}\kappa(1 + \zeta)\frac{\sqrt{n}}{n^2\mu_0^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2 = \frac{\kappa(1 + \zeta)}{n\mu_0^2}||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2. \tag{24}$$

We further introduce a kernel matrix approximation error bound of Nyström method [13] to upper bound $||\widetilde{\boldsymbol{K}} - \boldsymbol{K}||_2$.

Theorem 2. *Let* $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ *be an SPSD matrix. Assume that c columns of* $\boldsymbol{K}$ *are sampled uniformly at random without replacement, let* $\widetilde{\boldsymbol{K}}$ *be the rank-k Nyström approximation to* $\boldsymbol{K}$*, and let* $\boldsymbol{K}_k$ *be the best rank-k approximation to* $\boldsymbol{K}$*. For* $\epsilon > 0$*,* $\eta = \sqrt{\frac{\log(2/\delta)g(c,n-c)}{c}}$ *with* $g(a,s) = \frac{as}{a+s-1/2} \cdot \frac{1}{1-1/(2\max\{a,s\})}$*, if* $c \geq 64k/\epsilon^4$*, then with probability at least* $1-\delta$*,*

$$\|\boldsymbol{K} - \widetilde{\boldsymbol{K}}\|_F \leq \|\boldsymbol{K} - \boldsymbol{K}_k\|_F + \epsilon \left[\left(\frac{n}{c} \sum_{i \in D(c)} \boldsymbol{K}_{ii} \right) \left(\sqrt{n \sum_{i=1}^{n} \boldsymbol{K}_{ii}^2} + \eta \max(n\boldsymbol{K}_{ii}) \right) \right]^{\frac{1}{2}},$$

where $\sum_{i \in D(c)} \boldsymbol{K}_{ii}$ *is the sum of largest c diagonal entries of* $\boldsymbol{K}$*.*

Since $\|\boldsymbol{K} - \widetilde{\boldsymbol{K}}\|_2 \leq \|\boldsymbol{K} - \widetilde{\boldsymbol{K}}\|_F$, if we combine Equation (24) with Theorem 2, we can directly obtain the following theorem.

Theorem 3. *Let* $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ *be an SPSD matrix. Assume that c columns of* $\boldsymbol{K}$ *are sampled uniformly at random without replacement, let* $\widetilde{\boldsymbol{K}}$ *be the rank-k Nyström approximation to* $\boldsymbol{K}$*, and let* $\boldsymbol{K}_k$ *be the best rank-k approximation to* $\boldsymbol{K}$*. For* $\epsilon > 0$*,* $\eta = \sqrt{\frac{\log(2/\delta)g(c,n-c)}{c}}$ *with* $g(a,s) = \frac{as}{a+s-1/2} \cdot \frac{1}{1-1/(2\max\{a,s\})}$*, if* $c \geq 64k/\epsilon^4$*, then with probability at least* $1-\delta$*,*

$$|f'(\boldsymbol{x}) - f(\boldsymbol{x})| \leq \frac{\kappa(1+\zeta)}{n\mu_0^2} \left(\|\boldsymbol{K} - \boldsymbol{K}_k\|_F + \epsilon \left[\left(\frac{n}{c} \sum_{i \in D(c)} \boldsymbol{K}_{ii} \right) \left(\sqrt{n \sum_{i=1}^{n} \boldsymbol{K}_{ii}^2} + \eta \max(n\boldsymbol{K}_{ii}) \right) \right]^{\frac{1}{2}} \right),$$

where $\sum_{i \in D(c)} \boldsymbol{K}_{ii}$ *is the sum of largest c diagonal entries of* $\boldsymbol{K}$*.*

Theorem 3 measures the effect of kernel matrix approximation on the decision function of LSSVM. It enables us to bound the relative performance of LSSVM when the Nyström method is used to approximate the kernel matrix. We refer to the bound given in Theorem 3 as *a model approximation error bound.*

5 Approximate Model Selection for LSSVM

In order to find the hyperparameters that minimize the generalization error of LSSVM, many model selection approaches have been proposed, such as the cross validation, span bound [17], radius margin bound [5], PRESS criterion [1] and so on. However, when optimizing model selection criteria, all these approaches need to solve LSSVM completely in the inner layer for each iteration.

Here 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 hyperparameters from candidates. Theorem 3 shows that when Nyström approximation is used, the change of learning results of LSSVM is bounded, which is a theoretical support for approximate model selection. In the following, we present an approximate model selection scheme, as shown in Algorithm 2.

We use the RBF kernel $K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \exp\left(-\gamma\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2\right)$ to describe the scheme, but this scheme is also suitable for other kernel types.

Algorithm 2. Approximate Model Selection Scheme for LSSVM

```
Input: $\mathcal{S} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$;
Output: $(\gamma, \mu)_{\mathrm{opt}}$;
Initialize: $(\gamma, \mu) = (\gamma^0, \mu^0)$;
repeat
    1: Generate kernel matrix $\boldsymbol{K}$;
    2: Calculate $\boldsymbol{\alpha}$ and $b$ for LSSVM with $\boldsymbol{K}$ and $\mu$ using Algorithm 1;
    3: Calculate model selection criterion $T$ using $\boldsymbol{\alpha}$ and $b$;
    4: Update $(\gamma, \mu)$ to minimize $T$;
until the criterion T is minimized ;
return $(\gamma, \mu)_{\mathrm{opt}}$;
```

Let S denote the iteration steps of optimizing model selection criteria. The complexity of solving LSSVM by calculating the inverse of the exact kernel matrix is $O(n^3)$. For radius margin bound or span bound [5], a standard LSSVM needs to be solved in the inner layer for each iteration, so the total complexity of these two methods is $O(S n^3)$. For PRESS criterion [1], the inverse of kernel matrix also needs to be calculated for each iteration, so its complexity is $O(S n^3)$. From Theorem 1, we know that using Algorithm 1, we could solve LSSVM in $O(c^3 + nck)$. Therefore, if we use the above model selection criteria in the outer layer, the complexity of approximate model selection is $O(S(c^3 + nck))$. 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 $O(tS_\gamma S_\mu n^3)$. However, the complexity of approximate model selection using t-fold cross validation as outer layer criterion will be $O(tS_\gamma S_\mu(c^3 + nck))$.

6 Experiments

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

6.1 Experimental Scheme

The benchmark datasets in our experiments are introduced in [15], as shown in Table 1. For each dataset, there are 100 random training and test pre-defined partitions[1] (except 20 for the Image and Splice 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.

In Rätsch's experiment [15], 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

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

Table 1. Datasets used in experiments

Dataset	Features	Training	Test	Replications
Thyroid	5	140	75	100
Heart	13	170	100	100
Breast	9	200	77	100
Banana	2	400	4900	100
Ringnorm	20	400	7000	100
Twonorm	20	400	7000	100
Waveform	21	400	4600	100
Diabetes	8	468	300	100
Flare solar	9	666	400	100
German	20	700	300	100
Splice	60	1000	2175	20
Image	18	1300	1010	20

statistically "pure" since it has been used during model selection. Furthermore, the use of median of the hyperparameters would introduce an optimistic bias [3]. In our experiments, we perform model selection on the training set of each partition, then train the classifier with the obtained optimal hyperparameters still on the 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 and Splice dataset). The statistical analysis of these test error rates is conducted to assess the performance of the model selection approach. This experimental scheme is rigorous and can avoid the major flaws of the previous one [3]. All experiments are performed on a Core2 Quad PC, with 2.33GHz CPU and 4GB memory.

6.2 Effectiveness

Following the experimental setup in Section 6.1, 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}. We set $c = 0.1n$ and $k = 0.5c$ in Algorithm 1.

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 [2] 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 $z = (\bar{x} - \bar{y})/\sqrt{e_x^2 + e_y^2}$ and $z = 1.64$ corresponds to a 95% significance level. From Table 2, approximate 5-fold CV is significantly outperformed by 5-fold CV only on the Splice dataset, but the difference is just 2.5%. Besides, according

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	1.043	4.680±2.246	0.508	4.800±2.359
Heart	1.127	16.750±3.616	0.623	16.080±3.678
Breast	1.671	27.012±4.636	0.725	26.454±4.675
Banana	7.105	10.758±0.590	1.960	10.941±0.713
Ringnorm	7.601	2.044±0.358	2.058	2.872±3.895
Twonorm	7.097	2.528±0.234	2.213	2.446±0.163
Waveform	7.423	10.172±0.783	2.378	10.352±1.054
Diabetes	10.760	23.583±1.738	2.727	23.406±1.700
Flare solar	19.477	34.230±1.965	5.446	34.230±1.860
German	24.501	23.890±2.231	6.740	23.943±2.304
Splice	42.210	11.326±0.547	14.275	13.862±1.304
Image	141.792	2.876±0.725	28.743	4.628±0.944

to the Wilcoxon signed rank test [7], 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.

7 Conclusion

In this paper, Nyström method was 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 Nyström approximation. We designed an efficient algorithm for solving LSSVM and bounded the effect of kernel matrix approximation on the decision function of LSSVM. We derived a model approximation error bound, which is a theoretical support for approximate model selection. 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 approximate model selection for large scale problems. We finally verified the effectiveness of our approach by rigorous experiments on several benchmark datasets.

The application of our theoretical results and approach to practical 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 work is supported in part by the Natural Science Foundation of China under grant No. 61170019, and the Natural Science Foundation of Tianjin under grant No. 11JCYBJC00700.

References

1. Cawley, G.C., Talbot, N.L.C.: Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. Neural Networks 17(10), 1467–1475 (2004)
2. Cawley, G.C., Talbot, N.L.C.: Preventing over-fitting during model selection via Bayesian regularisation of the hyper-parameters. Journal of Machine Learning Research 8, 841–861 (2007)
3. Cawley, G.C., Talbot, N.L.C.: On over-fitting in model selection and subsequent selection bias in performance evaluation. Journal of Machine Learning Research 11, 2079–2107 (2010)
4. Chapelle, O., Vapnik, V.: Model selection for support vector machines. In: Advances in Neural Information Processing Systems, vol. 12, pp. 230–236. MIT Press, Cambridge (2000)
5. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. Machine Learning 46(1), 131–159 (2002)
6. Cortes, C., Mohri, M., Talwalkar, A.: On the impact of kernel approximation on learning accuracy. In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS), Sardinia, Italy, pp. 113–120 (2010)
7. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
8. Drineas, P., Mahoney, M.: On the Nyström method for approximating a Gram matrix for improved kernel-based learning. Journal of Machine Learning Research 6, 2153–2175 (2005)
9. Duan, K., Keerthi, S., Poo, A.: Evaluation of simple performance measures for tuning SVM hyperparameters. Neurocomputing 51, 41–59 (2003)
10. Golub, G., Van Loan, C.: Matrix Computations. Johns Hopkins University Press, Baltimore (1996)
11. Guyon, I., Saffari, A., Dror, G., Cawley, G.: Model selection: Beyond the Bayesian / frequentist divide. Journal of Machine Learning Research 11, 61–87 (2010)
12. Higham, N.: Accuracy and stability of numerical algorithms. SIAM, Philadelphia (2002)
13. Kumar, S., Mohri, M., Talwalkar, A.: Sampling techniques for the Nyström method. In: Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS), Clearwater, Florida, USA, pp. 304–311 (2009)
14. Luntz, A., Brailovsky, V.: On estimation of characters obtained in statistical procedure of recognition. Technicheskaya Kibernetica 3 (1969) (in Russian)
15. Rätsch, G., Onoda, T., Müller, K.: Soft margins for AdaBoost. Machine Learning 42(3), 287–320 (2001)
16. Suykens, J., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters 9(3), 293–300 (1999)
17. Vapnik, V., Chapelle, O.: Bounds on error expectation for support vector machines. Neural Computation 12(9), 2013–2036 (2000)
18. Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York (1998)
19. Williams, C., Seeger, M.: Using the Nyström method to speed up kernel machines. In: Advances in Neural Information Processing Systems 13, pp. 682–688. MIT Press, Cambridge (2001)