Scaling Up ROC-Optimizing Support Vector Machines

Gimun Bae[1] and Seung Jun Shin[1]

[1]Department of Statistics, Korea University, Seoul, 02841, Republic of Korea

Abstract

The ROC–SVM, originally proposed by Rakotomamonjy [2004], directly maximizes the area under the ROC curve (AUC) and has become an attractive alternative of the conventional binary classification under the presence of class imbalance. However, its practical use is limited by high computational cost, as training involves evaluating all $O(n^2)$ pairwise terms. To overcome this limitation, we develop a scalable variant of the ROC–SVM that leverages incomplete U-statistics, thereby substantially reducing computational complexity. We further extend the framework to nonlinear classification through a low-rank kernel approximation, enabling efficient training in reproducing kernel Hilbert spaces. Theoretical analysis establishes an error bound that justifies the proposed approximation, and empirical results on both synthetic and real datasets demonstrate that the proposed method achieves comparable AUC performance to the original ROC–SVM with drastically reduced training time.

Keywords: Imbalanced Classification, Incomplete U-statistics, Nyström Approximation, Receiver Operating Characteristic Curve, Support Vector Machines

1 Introduction

Binary classification is a fundamental problem in machine learning. Given a pair $(\mathbf{X}, Y)$, where $\mathbf{X}$ is a p-dimensional predictor and Y is a binary response taking values in $\{-1, 1\}$, the goal is to learn a decision function f of $\mathbf{X}$ that predicts Y by $\hat{Y} = \text{sign}\{f(\mathbf{X})\}$. A canonical approach is to choose f that minimizes the classification error, or equivalently, maximizes the accuracy. For instance, the support vector machine (SVM; Vapnik, 1999) determines the decision function by maximizing the geometric margin, which effectively aligns with maximizing accuracy [Lin, 2002].

However, in imbalanced settings where one class is substantially underrepresented, accuracy can be a misleading measure of performance. Even a trivial classifier that always predicts the majority class can achieve high accuracy while completely failing to detect samples from the minor class. As an alternative, the receiver operating characteristic (ROC) curve is widely used to evaluate classifier performance under class imbalance. By definition, the ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) to summarize classification performance, and the area under the ROC curve (AUC) serves as a popular scalar summary. A classifier with a larger AUC value is generally regarded as having better classification performance.

By defintion, the population AUC of a classifier f is given by

$$\text{AUC}(f) = \mathbb{P}\big(f(\mathbf{X}^+) > f(\mathbf{X}^-)\big), \tag{1}$$

where $\mathbf{X}^+$ and $\mathbf{X}^-$ denote predictor vectors from the positive and negative classes, respectively. Unlike accuracy, AUC is less sensitive to class proportions. This robustness has also been examined from a theoretical perspective. Cortes and Mohri [2003] was the first to theoretically investigate the relationship between classification error and AUC, showing that minimizing the error rate does not necessarily lead to the best AUC performance. Furthermore, Rosset [2004] demonstrated that AUC can serve as a more reliable criterion than empirical error, even when the objective is to minimize the misclassification rate.

This motivates us to directly maximize the AUC rather than accuracy. Suppose we are given data $(\mathbf{x}_i, y_i) \in \mathbb{R} \times \{-1, 1\}$ for $i = 1, \cdots, n$. Let $I_+ = \{i : y_i = 1\}$ and $I_- = \{i : y_i = -1\}$ denote the index sets

corresponding to the positive and negative classes, respectively, and define $n_+ = |I_+|$ and $n_- = |I_-|$ so that $n_+ + n_- = n$. Then, the empirical counterpart of (1) is given by

$$\frac{1}{n_+ n_-} \sum_{i \in I_+} \sum_{j \in I_-} \mathbb{1}\{f(\mathbf{x}_i) \leq f(\mathbf{x}_j)\}, \tag{2}$$

where $\mathbb{1}\{\cdot\}$ denotes the indicator function. However, the direct optimization of (2) is computationally intractable due to the iregularity of the zero–one loss. To overcome this, the zero–one loss can be replaced with a convex surrogate [Rakotomamonjy, 2004, Clémençon et al., 2013, Natole et al., 2018, Yang et al., 2020]. Among various choices, the hinge loss is a nautral choice due the popularity of the SVM. The ROC–SVM [Rakotomamonjy, 2004] solves

$$\min_{f \in \mathcal{H}_K} \frac{1}{n_+ n_-} \sum_{i \in I_+} \sum_{j \in I_-} [1 - \{f(\mathbf{x}_i) - f(\mathbf{x}_j)\}]_+ + \frac{\lambda}{2} \|f\|^2_{\mathcal{H}_K} \tag{3}$$

where $[z]_+ = \max\{z, 0\}$, $\mathcal{H}_K$ denotes the reproducing kernel Hilbert space (RKHS; Schölkopf and Smola, 2002) generated by a positive definite kernel $K(\mathbf{x}, \mathbf{x}')$, and $\lambda > 0$ is a regularization parameter controlling the complexity of f. The ROC–SVM closely resembles the standard SVM, which has led to a variety of extensions [Arzhaeva et al., 2006, Tian et al., 2011, Zhao et al., 2011, Ying et al., 2016, among many others].

Despite its conceptual simplicity, computing the ROC–SVM in (3) remains prohibitively intensive for large n, as it requires evaluating $n_+ \times n_-$ pairwise terms, which scales as $O(n^2)$. To address this computational bottleneck, several studies have proposed improvements to enhance the efficiency. For example, Kim and Shin [2020] developed a regularization path algorithm that substantially accelerates parameter tuning by exploiting the piecewise linearity of the ROC–SVM solution. However, this approach does not reduce the inherent computational complexity of the ROC–SVM itself.

In this article, we propose an efficient algorithm for solving (3), even when n is substantially large. We first note that the empirical AUC in (2) is a U-statistic, and we leverage the incomplete U-statistic approach of Clémençon et al. [2016], which significantly reduces the computational burden while keeping the approximation error within a controlled level. While the linear ROC–SVM naturally benefits from this incomplete U-statistic approximation, its kernel version is still computationally demanding due to the size of the kernel matrix which scales as $O(n^2)$. To address this, we further employ a low-rank approximation of the kernel matrix, which not only transforms the kernel ROC–SVM into a linear ROC–SVM in the induced feature space but also substantially reduces the memory requirement.

The remainder of this paper is organized as follows. Section 2 introduces incomplete U-statistics and their application to the linear ROC–SVM. Section 3 extends the proposed idea to the kernel ROC–SVM via a low-rank approximation of the kernel matrix and provides its approximation-error bound. Section 4 presents application to both synthetic and real data sets to illustrate the effectiveness of the proposed method, and Section 5 concludes the paper. Technical details are provided in Appendix.

2 ROC-SVM as an Incomplete U-statistic

2.1 Incomplete U-statistic

Let U_n denote the loss part of ROC-SVM in (3):

$$U_n = \frac{1}{n_+ n_-} \sum_{i \in I_+} \sum_{j \in I_-} [1 - \{f(\mathbf{x}_i) - f(\mathbf{x}_j)\}]_+ \tag{4}$$

We note that U_n takes the form of the generalized U-statistic defined as follows.

Definition 1. (Generalized U-statistic) *Suppose we have J independent population, and we let $\mathbf{x}_{i,j}$ denote the i-th i.i.d observations from the j-th population where $i = 1, 2, \cdots, n_j$ and $j = 1, 2, \cdots, J$. Let Λ_j denotes all possible combination of index $(1, 2, \cdots, n_j)$ of a given size d_j, where $j = 1, 2, \cdots, J$. Then, the generalized U-statistic with a kernel functinon $H : \mathbb{R}^p \times \cdots \times \mathbb{R}^p \mapsto \mathbb{R}$, is defined as*

$$U_n = \frac{1}{\prod_{j=1}^J \binom{n_j}{d_j}} \sum_{I_1 \in \Lambda_1} \cdots \sum_{I_J \in \Lambda_J} H\left(\mathbf{X}_{I_1,1}, \ldots, \mathbf{X}_{I_J,J}\right). \tag{5}$$

where $\mathbf{X}_{I_j,j} = \{\mathbf{x}_{i,j} \mid i \in I_j\}$ *for a given index set* $I_j = \{i_{j1}, \cdots, i_{jd_j}\} \subset \{1, 2, \cdots, n_j\}$ *for the* j*-th sample,* $j = 1, 2, \cdots, J$*, and* $n = \sum_{j=1}^{J} n_j$*.*

The generalized U-statistic (5) reduces to the conventional U-statistic when $J = 1$. A well-known example of a generalized U-statistic is the Mann–Whitney statistic [Lee, 2019] for two-sample location problem. Namely, the Mann–Whitney statistic is a form (upto constant multiplication) of (5) with $J = 2$ and $d_j = 1$ with a zero-one kernel $H(x_{i,1}, x_{i,2}) = \mathbb{1}\{x_{i,1} < x_{i,2}\}$. In fact, the AUC is equivalent to the Mann–Whitney statistic [Yan et al., 2003], and U_n in (4) is a convex relaxation of the sample 1− AUC that replaces the zero-one loss with the hinge loss function.

The generalized U-statistic involves sum of $\prod_{j=1}^{J} \binom{n_j}{d_j}$ terms, which becomes quickly intractable as n_j gets large, even with $d_j = 1, j = 1, 2, \cdots, J$. Notably, due to dependence among terms, a small subset can often approximate the full sum without significantly increasing the variance [Lee, 2019]. This motivates the use of incomplete U-statistics [Blom, 1976, Clémençon et al., 2016] defined in the following, and this greately reduces the computational cost of the U-statistics while preserving a desired level of accuracy.

Definition 2. (Incomplete Generalized U-statistic) *Let* Λ *denote the set of all index tuples* $(I_1, \ldots, I_J), \forall I_j \in \Lambda_j, j = 1, 2, \cdots J$*. Given a randomly chosen subset* $\mathcal{D}_B \subseteq \Lambda$ *of size* B*, the incomplete generalized U-statistic* (5) *is defined as*

$$\tilde{U}_B = \frac{1}{B} \sum_{(I_1,\ldots,I_J)\in\mathcal{D}_B} H\left(\mathbf{X}_{I_1,1}, \ldots, \mathbf{X}_{I_J,J}\right). \tag{6}$$

The cardinalty of $\mathcal{D}_B$, B is much smaller than that of Λ. The incomplete U-statistic subtantially reduces the computational cost while preserving the learning rate. Specifically, the incomplete U-statistic $\tilde{U}_B$ achieves the same convergence rate bound as U_n, $O_{\mathbb{P}}\left(\sqrt{\frac{\log(n)}{n}}\right)$, if $\mathcal{D}_B$ is randomly chosen index tuples from Λ with replacement and $B = O(n)$ [Clémençon et al., 2016].

2.2 Scaling-up Linear ROC-SVM via Incomplete U-statistic

Suppose the classification function is linear, that is, $f(\mathbf{x}) = \alpha + \boldsymbol{\beta}^T\mathbf{x}$. Then, the incomplete ROC–SVM solves

$$\min_{\boldsymbol{\beta}} \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} [1 - \boldsymbol{\beta}^T(\mathbf{x}_i - \mathbf{x}_j)]_+ + \frac{\lambda}{2}\boldsymbol{\beta}^T\boldsymbol{\beta}, \tag{7}$$

where $\mathcal{D}_B$ is a randomly selected subset (with replacement) of $\Lambda = \{(i, j) \mid \forall i \in I_+, j \in I_-\}$ with replacement. We remark that the intercept α is not identifiable, as it cancels out in the difference $f(\mathbf{x}_i) - f(\mathbf{x}_j)$. Nevertheless, it is still required for prediction, and one can determine α to achieve a desired sensitivity or specificity [Kim and Shin, 2020].

Regarding the choice of B, there exists a trade-off between computational efficiency and predictive performance. A smaller B reduces computational cost but increases variance, whereas a larger B yields more accurate estimates at the expense of higher computational burden. Clémençon et al. [2016] assume B increases at the rate $O(n)$ to obtain an approximation error bound of $O_{\mathbb{P}}\left(\sqrt{\frac{\log(n)}{n}}\right)$. Accordingly, we set $B = n$, the number of training samples, in subsequent applications in Section 4.

Finally, we employ the gradient descent algorithm (GDA) to solve (7). Let $\mathcal{L}(\boldsymbol{\beta})$ denote the objective function in (7). The GDA update for $\boldsymbol{\beta}$ is given by

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \eta_t \nabla\mathcal{L}(\boldsymbol{\beta}^{(t)}), \tag{8}$$

where

$$\nabla\mathcal{L}(\boldsymbol{\beta}) = \frac{\partial\mathcal{L}(\boldsymbol{\beta})}{\partial(\boldsymbol{\beta})} = -\frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} (\mathbf{x}_i - \mathbf{x}_j) \cdot \mathbb{1}\left\{1 - \boldsymbol{\beta}^T(\mathbf{x}_i - \mathbf{x}_j) > 0\right\} + \lambda\boldsymbol{\beta}.$$

The step size η_t in (8) can be chosen in several ways. A simple approach is to fix η_t as a small constant, such as 10^{-2} or 10^{-3}. For more efficient optimization, adaptive methods such as the ADAM algorithm

have been proposed in the context of stochastic gradient descent [Kingma and Ba, 2014]. Among these, we adopt ADAMAX, a variant of the ADAM algorithm based on the infinity norm. Although (8) describes a GDA algorithm rather than stochastic gradient descent, the ADAMAX update remains computationally efficient and yields stable solutions. We refer readers to Section 7 of Kingma and Ba [2014] for details on the ADAMAX update.

3 Scaling-up Kernel ROC-SVM

3.1 Low-Rank Linearization of the Kernel ROC-SVM

By the Representer Theorem [Kimeldorf and Wahba, 1971], the solution to (3) admits the following finite representation:

$$f(\mathbf{x}) = \alpha + \sum_{k=1}^{n} \theta_k K(\mathbf{x}, \mathbf{x}_k) \tag{9}$$

Plugging (9) into (3), the kernel ROC-SVM solves

$$\min_{\boldsymbol{\theta}} \frac{1}{n_+ n_-} \sum_{i \in I_+} \sum_{j \in I_-} [1 - \boldsymbol{\theta}^T (\mathbf{k}_i - \mathbf{k}_j)]_+ + \frac{\lambda}{2} \boldsymbol{\theta}^\top \mathbf{K} \boldsymbol{\theta} \tag{10}$$

where $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_n)^T$, $\mathbf{K}$ is the n-dimensional kernel matrix whose (l, m)th element is $K(\mathbf{x}_l, \mathbf{x}_m)$, and $\mathbf{k}_i$ is the ith column of $\mathbf{K}$.

Despite the use of the incomplete U-statistic, solving (10) remains challenging, particularly when n is excessively large, because the kernel matrix $\mathbf{K}$ in the penalty term requires $O(n^2)$ memory and computation. To address this issue, we employ the idea of low-rank linearization [Zhang et al., 2012]. Specifically, one can transform (10) into an equivalent but computationally simpler linear ROC–SVM.

Proposition 3. *For a given kernel matrix* $\mathbf{K}$*, suppose we have* $\mathbf{K} = \mathbf{V}\mathbf{V}^T$*, (10) is equivalent to solve the following linear ROC-SVM*

$$\min_{\boldsymbol{\beta}} \frac{1}{n_+ n_-} \sum_{i \in I_+} \sum_{j \in I_-} \left[1 - \boldsymbol{\beta}^T (\mathbf{v}_i - \mathbf{v}_j)\right]_+ + \frac{\lambda}{2} \boldsymbol{\beta}^\top \boldsymbol{\beta}$$

where $\mathbf{v}_i$ *denotes the* i*th row of* $\mathbf{V}$*.*

The proof of Proposition 3 is provided in the Appendix. A straightforward solution of $\mathbf{V}$ can be obtained from the eigen-decomposition of $\mathbf{K}$. Namely, we have

$$\mathbf{V} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}.$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{n \times n}$ is a diagonal matrix whose diagonal elements are eigenvalues of $\mathbf{K}$ arranged in descending order and $\mathbf{U} \in \mathbb{R}^{n \times n}$ is the orthogonal matrix whose columns are the corresponding eigenvectors of $\mathbf{K}$. However, this approach still requires computing the full kernel matrix $\mathbf{K}$, and the resulting $\mathbf{V} \in \mathbb{R}^{n \times n}$ provides no computational advantage.

To tackle this, we apply the Nyström approximation [Williams and Seeger, 2000] to obtain $\tilde{\mathbf{V}} \in \mathbb{R}^{n \times d}$, a low-rank approximation of $\mathbf{V}$. Nyström Approximation begins by selecting $d (\ll n)$ landmark samples from $\mathbf{x}_i, i = 1, 2, \cdots, n$, and then compute the corresponding kernel matrix denoted by $\mathbf{K}_{\text{selected}} \in \mathbb{R}^{d \times d}$. The Landmark samples can be chosen through different strategies: uniform random sampling is the simplest approach, whereas sophisticated methods such as K-means clustering have been proposed to reduce approximation error [Zhang and Kwok, 2010]. For the theoretical analysis in Section 3.2, we assume the uniform random sampling for simplicity. In our simulation studies, however, we employ the stratified random sampling to account for class imbalance, and in real-data applications we adopt the K-means–based selection strategy, as our results are not overly sensitive to the specific choice of landmark selection method.

Given d landmark samples, we construct

$$\tilde{\mathbf{V}} = \mathbf{C}(\mathbf{K}^{\dagger}_{\text{selected}})^{1/2}, \tag{11}$$

where $\mathbf{A}^{\dagger}$ denotes Moore-Penrose inverse of $\mathbf{A}$ and $\mathbf{C} \in \mathbb{R}^{n\times d}$ is the kernel matrix computed between all training samples $\mathbf{x}_i, i = 1, 2, \cdots, n$ and the d landmark samples. Compared with the exact formulation, this approximation in (11) only requires computing $d \times d$ kernel matrix $\mathbf{K}_{\text{selected}}$ (and its inverse) and an $n \times d$ matrix $\mathbf{C}$. Unless d is chosen exesslive large, this procedure is substantially more efficient.

The kernel ROC-SVM in (10) can therefore be approximated by the following d-dimensional linear ROC-SVM:

$$\min_{\boldsymbol{\beta}} \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} [1 - \boldsymbol{\beta}^T(\tilde{\mathbf{v}}_i - \tilde{\mathbf{v}}_j)]_+ + \frac{\lambda}{2}\boldsymbol{\beta}^\top\boldsymbol{\beta}, \tag{12}$$

where $\tilde{\mathbf{v}}_i$ denotes the ith row of $\tilde{\mathbf{V}}$.

For the prediction, we first compute the test feature matrix $\tilde{\mathbf{V}}^{\text{ts}} = \mathbf{C}^{\text{ts}}(\mathbf{K}^{\dagger}_{\text{selected}})^{1/2}$, where $\mathbf{C}^{\text{ts}} \in \mathbb{R}^{n^{\text{ts}}\times d}$ is defined analogously to $\mathbf{C}$ using the test samples $\mathbf{x}_i^{\text{ts},i=1,2,\ldots,n^{\text{ts}}}$. Then, the ith row $\tilde{\mathbf{v}}_i^{\text{ts}}$ of $\tilde{\mathbf{V}}^{\text{ts}}$ serves as the transformed feature vector for the ith test observation, whose predicted value is $\hat{y}_i^{\text{ts}} = \hat{\alpha} + \hat{\boldsymbol{\beta}}^T\mathbf{v}_i^{\text{ts}}$, where $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$ are the parameter estimates from the training data.

3.2 Approximation Error Bound

As seen in (4), the ROC–SVM loss can be expressed as a U-statistic with the following kernel:

$$H_f(\mathbf{x}_i, \mathbf{x}_j) = [1 - \{f(\mathbf{x}_i) - f(\mathbf{x}_j)\}]_+. \tag{13}$$

We approximate this loss by its incomplete counterpart and construct $f(\mathbf{x})$ using the Nyström approximation, which inevitably introduces an approximation error. In what follows, we show that the difference between the empirical losses—with and without the approximation—can be bounded with high probability.

We approximate the loss with the incomplete counterpart and construct $f(\mathbf{x})$ using the Nyström approximation, which results approximation error. In what follows, we show that the difference in empirical loss—with and without the approximation—can be bounded with high probability.

For clarity, we introduce some notation. Let $U_n(f)$ denote the full U-statistic-based loss function (4) with decision function f, and $\tilde{U}_B(f)$ be the corresponding incomplete U-statistic based on B randomly selected pairs. A baseline approach is to optimize the full U-statistic $U_n(f)$ without applying the Nyström approximation. That is, the method solves (3), where we write $\mathcal{H} := \mathcal{H}_K$ for simplicity and denote its solution by $\hat{f}$. In our approach, the model is formulated by solving the linearized problem (12). We remark that Proposition 3 remains valid even when the problem is defined using the incomplete U-statistic and the Nyström approximation, since its proof does not depend on either assumption. Thus, the problem (12) can be equivalently expressed as

$$\min_{f\in\tilde{\mathcal{H}}} \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} [1 - (f(\mathbf{x}_i) - f(\mathbf{x}_j))]_+ + \frac{\lambda}{2}\|f\|^2_{\tilde{\mathcal{H}}}, \tag{14}$$

where $\tilde{\mathcal{H}}$ is the RKHS induced by the Nyström-approximated kernel matrix $\tilde{\mathbf{K}}$ defined as $\tilde{\mathbf{K}} = \tilde{\mathbf{V}}\tilde{\mathbf{V}}^\top$. We note that $\tilde{\mathcal{H}}$ is well-defined because $\mathbf{K}$ is positive definite and $\tilde{\mathbf{K}}$ is positive semi-definite. We further denote its solution by $\tilde{f}_B$, which, by the Representer Theorem, has the same form as in (9), with $K(\mathbf{x}_i, \mathbf{x}_k)$ replaced by $\tilde{K}(\mathbf{x}_i, \mathbf{x}_k)$, the (i, k)-th entry of $\tilde{\mathbf{K}}$.

Let $\lambda_{\min}$ denote the minimum eigenvalue of $\mathbf{K}$, and let $\tilde{\lambda}^+_{\min}$ denote the smallest nonzero eigenvalue of $\tilde{\mathbf{K}}$. Define the kernel-approximation error by $\mathcal{E} := \mathbf{K} - \tilde{\mathbf{K}}$, and let $\mathcal{E}_2 := \|\mathcal{E}\|_2$ denote its spectral norm. We further define the random variable $N_{\max}$ as the maximum number of times a sample is selected in $\mathcal{D}_B$. For the RKHS norm, let $K_{\max}$ denote the supremum of $\|K(\cdot, \mathbf{x}_i)\|_{\mathcal{H}}$ over all observed data:

$$K_{\max} = \sup_{1\le i\le n} \|K(\cdot, \mathbf{x}_i)\|_{\mathcal{H}},$$

where the kernel $K(\mathbf{x}, \mathbf{x}')$ is assumed to be bounded. Let C and $\tilde{C}_B$ be upper bounds on the RKHS norms of the minimizers $\hat{f}$ and $\tilde{f}_B$, respectively; that is, $\|\hat{f}\|_{\mathcal{H}}^2 \leq C$ and $\|\tilde{f}_B\|_{\tilde{\mathcal{H}}}^2 \leq \tilde{C}_B$. We also denote $C_{\max}$ a finite constant related to C, whose precise definition will be specified in the proof.

We now establish the following theorem, which states that the approximation error is bounded with high probability. The proof is given in the Appendix.

Theorem 4. *Given the observed data, for all $\delta \in (0,1)$, with probability at least 1-δ,*

$$|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)| \leq A_1\sqrt{\frac{\log(4/\delta)}{B}} + \frac{N_{\max}}{B}\left(A_2\mathcal{E}_2\sqrt{n} + A_3\sqrt{\frac{N_{\max}\mathcal{E}_2 n^{3/2}}{B}}\right),$$

where

$$A_1 = \frac{1 + 2K_{\max}\sqrt{C_{\max}}}{\sqrt{2}}, \quad A_2 = \sqrt{\frac{\tilde{C}_B}{\tilde{\lambda}_{\min}^+}}, \quad A_3 = K_{\max}\sqrt{\tfrac{2}{\lambda}}\left(\sqrt{\tfrac{C}{\lambda_{\min}}} + \sqrt{\tfrac{\tilde{C}_B}{\tilde{\lambda}_{\min}^+}}\right)^{1/2}.$$

We make several remarks on Theorem 4. The first term corresponds to the error introduced by the incomplete U-statistic and converges to zero at the rate $O(1/\sqrt{B}) = O(1/\sqrt{n})$, since $B = O(n)$. The second term arises from the Nyström approximation. By Bernstein's inequality, one can show that $N_{\max} = O_{\mathbb{P}}(\log n)$ when $\min(n_+, n_-) = O(n)$; that is, when no class dominates the other. Furthermore, Drineas et al. [2005] showed that $\mathcal{E}_2 = O(n/\sqrt{d})$, where d is the target rank. Hence, if $d = o(n)$, then $\mathcal{E}_2 = o(n)$, and consequently the second term is of order $o(\sqrt{n}\log n)$. Specifically, if we set $d = n/\log n$, then $\mathcal{E}_2 = O(\sqrt{n\log n})$, and consequently the second term is of order $O((\log n)^{3/2})$. Although the upper bound diverges asymptotically, its growth rate $o(\sqrt{n}\log n)$ is subpolynomial, and therefore justifing the use of our approximation in practice.

4 Application

4.1 Synthetic Data

We consider two illustrative scenarios to evaluate the efficiency of the proposed method. The data are generated from the following binary classification model:

$$y_i = \text{sign}\{\alpha + f(\mathbf{x}_i) + \epsilon_i\}, \quad i = 1, 2, \cdots, n,$$

where the bivariate predictor $\mathbf{x}_i \sim N_2(\mathbf{0}, \mathbf{I})$ and the random error $\epsilon_i \sim N(0,1)$. The offset parameter α, which controls the class imbalance, is chosen so that approximately 80% of the responses y_i are negative.

We consider two different forms of the classification function f.

- Linear Model – $f(\mathbf{x}_i) = \boldsymbol{\beta}^T\mathbf{x}_i$, where $\boldsymbol{\beta} - (1,1)^T$.
- Radial Model – $f(\mathbf{x}_i) = \|\mathbf{x}_i\|^2$

We generate training samples of sizes $n_{\text{train}} \in \{5, 50, 100\} \times 10^3$ while fixing the test sample size at $_{\text{test}} = 25 \times \times 10^3$.

To evaluate the finite sample performance, we compute the empirical AUC as

$$\hat{\text{AUC}} = \frac{1}{n_+ n_-}\sum_{i \in I_+}\sum_{j \in I_-} \mathbb{1}(\hat{f}(\mathbf{x}_i) > \hat{f}(\mathbf{x}_j)),$$

where $\hat{f}(\mathbf{x})$ denotes the estimated classification function. As a reference, we also compute the population AUC via Monte Carlo approximation, which is feasible because the true function f is known in the simulation setting.

For the linear model, we train the ROC–SVM using both the full U-statistic risk and the incomplete U-statistic risk, hereafter referred to as the full and incomplete models, respectively. The regularization parameter is selected via cross-validation. We then compare their performance to evaluate the effect of the approximation.

Table 1: Comparison of averaged training times and sample AUCs (over 50 independent repetitions) between the full and incomplete models when the true classification function is linear.

	Computing Time (in sec.)		AUC(%)	
n_{train}	**Full**	**Incomplete**	**Full**	**Incomplete**
5,000	196.10 (7.75)	2.69 (0.15)	90.370 (0.88)	90.370 (1.14)
10,000	819.10 (31.41)	3.02 (0.21)	90.374 (0.37)	90.373 (0.54)
50,000	–	8.03 (0.90)	–	90.375 (0.22)
100,000	–	10.26 (0.24)	–	90.376 (0.10)

The population AUC (AUC_{true}) is 90.377%. The symbol "–" indicates that model fitting was computationally infeasible for the given sample size. Numbers in parentheses denote standard errors.

Table 1 reports the average computing time and sample AUC over 50 independent replications for the linear model, whose population AUC is 0.90377. The results show that the incomplete model substantially reduces computation time compared with the full model, supporting the validity of the incomplete approximation. This advantage becomes more evident as n increases: for instance, the full model fails to run when the training size exceeds 10,000 due to memory limitations, whereas the incomplete model scales to about 100,000 training samples within several seconds. Nevertheless, the incomplete model exhibits virtually no loss in sample AUC, clearly demonstrating the practical effectiveness of the proposed method.

For the nonlinear model, we train the ROC–SVM with a radial kernel. In this case, the full model cannot be computed because of its prohibitive cost for large n (e.g., $n = 500,000$). Therefore, we fit only the incomplete model and compare it with its Nyström-approximated counterpart ($d = 300$). This comparison allows us to assess the efficiency of the Nyström approximation. Table 2 summarizes the results.

Table 2: Comparison of averaged training times and sample AUCs (over 50 independent repetitions) between the incomplete models with and without the Nyström approximation for the nonlinear model.

	Computing Time (in sec.)		AUC(%)	
n_{train}	**Without Nyström**	**Nyström**	**Without Nyström**	**Nyström**
5,000	10.21 (0.30)	4.64 (0.25)	90.947 (0.53)	90.921 (0.69)
10,000	19.81 (2.84)	4.60 (0.52)	90.863 (2.27)	90.927 (0.64)
50,000	–	5.19 (0.14)	–	90.936 (0.61)
100,000	–	5.87 (0.14)	–	90.926 (0.64)

The population AUC (AUC_{true}) is 90.985%. The symbol "–" indicates that model fitting was computationally infeasible for the given sample size. Numbers in parentheses denote standard errors.

We observe that the incomplete U-statistic method alone cannot handle training sets larger than 10,000 samples, whereas the Nyström approximation drastically reduces computational burden and enables model fitting with training sizes up to about 100,000 within a few seconds. In terms of AUC, the Nyström approximation introduces only a minor loss, slightly increasing the gap from the population AUC (0.90985); nevertheless, the empirical AUC remains close to the population value, indicating that predictive performance is well preserved.

4.2 Real Data

We further evaluate the performance of our methods on three datasets for imbalanced classification. The first dataset is the Skin Segmentation dataset [Bhatt and Dhall, 2009], which contains 245,057 observations with three predictors and a positive-class proportion of 20.8%. Because the dataset is large and the class ratio (n_+/n) is not extremely small, fitting the model on the full data is infeasible on a personal computer due to memory constraints. Therefore, we randomly sample 100,000 observations while preserving the original class distribution. The second dataset is the Bank Marketing dataset [Moro et al., 2014], which includes 41,188 observations with 19 predictors and a positive-class proportion of 11.3%. The third dataset is the

Accelerometer Gyro Mobile Phone dataset [AlSahly, 2022], containing 31,991 observations with six predictors and a positive-class proportion of 1.8%.

For the real-data experiments, each dataset is randomly split into training and test sets. We then apply the proposed linear and kernel ROC–SVM models using the incomplete U-statistic approximation. For the kernel model, we further employ the Nyström approximation with $d = 300$. The entire procedure is repeated 30 times, and we record the training time and empirical AUC on the test sets. The regularization parameter λ is selected via cross-validation. Table 3 reports the average training time and test AUC obtained at the optimal value of λ. Both proposed models train within seconds while achieving high test AUC. In particular, the kernel model consistently attains higher AUC than the linear model, and for the Skin Segmentation and Accelerometer datasets it even trains faster. Although the kernel model trains somewhat more slowly on the Bank Marketing dataset and exhibits a larger standard error in AUC for the Accelerometer dataset, the overall computational cost remains practical and the classification performance is strong. In summary, the original (non-approximated) ROC–SVM is computationally infeasible—requiring hundreds of seconds even for smaller subsamples—whereas our proposed approach achieves competitive accuracy within a fraction of the time.

Table 3: Training time and test AUC on the three real datasets, averaged over 30 random train–test splits. The regularization parameter λ is selected via cross-validation.

Dataset	n	p	$n_+/n(\%)$	**Kernel**	**Time**	**AUC**(%)
Skin Segmentation	100,000	3	20.8	Linear	8.38 (2.09)	94.64 (1.96)
				Radial	5.82 (0.15)	98.53 (1.40)
Bank Marketing	41,188	18	11.3	Linear	0.43 (0.04)	74.71 (9.25)
				Radial	46.33 (9.61)	76.04 (7.94)
Accelerometer	31,991	6	1.8	Linear	3.54 (0.12)	88.53 (9.15)
				Radial	1.60 (0.38)	89.92 (21.61)

Numbers in parentheses denote standard errors. AUC values are reported as percentages, with standard deviations shown without the 10^{-3} factor.

5 Concluding Remarks

In this work, we addressed the challenge of scaling ROC–SVM to large datasets, where training involves evaluating $O(n^2)$ pairwise terms, rendering the standard approach impractical. By leveraging incomplete U-statistics and a low-rank kernel approximation, we proposed a computationally efficient algorithm that reduces the overall complexity to $O(n)$. Beyond algorithmic development, we also provided theoretical justification by deriving an approximation bound for the proposed approach.

Empirical results on simulated and real datasets demonstrate that the proposed method consistently achieves competitive AUC compared with the original ROC–SVM, while substantially reducing computational cost. These findings confirm the feasibility of efficient ROC-optimizing classification, a task that has traditionally been constrained by computational limitations.

Nevertheless, our approach has room for further improvement. For instance, extending the theoretical analysis beyond the hinge loss remains challenging, as many surrogate losses fail to satisfy global Lipschitz continuity—a key property underlying our proofs. In addition, our method is currently a batch algorithm and therefore not well suited for streaming data. An interesting direction for future work is to extend our framework to an online setting, for example, by incorporating stochastic gradient descent. Such an extension is particularly challenging for the kernel version but warrants further investigation.

References

Alain Rakotomamonjy. Optimizing area under roc curve with svms. In *ROCAI*, pages 71–80, 2004.

Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer: New York, 1999.

Yi Lin. Support vector machines and the bayes rule in classification. *Data Mining and Knowledge Discovery*, 6(3):259–275, 2002.

Corinna Cortes and Mehryar Mohri. Auc optimization vs. error rate minimization. *Advances in neural information processing systems*, 16, 2003.

Saharon Rosset. Model selection via the auc. In *Proceedings of the twenty-first international conference on Machine learning*, page 89, 2004.

Stéphan Clémençon, Marine Depecker, and Nicolas Vayatis. An empirical comparison of learning algorithms for nonparametric scoring: the treerank algorithm and other methods. *Pattern Analysis and Applications*, 16(4):475–496, 2013.

Michael Natole, Yiming Ying, and Siwei Lyu. Stochastic proximal algorithms for auc maximization. In *International Conference on Machine Learning*, pages 3710–3719. PMLR, 2018.

Zhenhuan Yang, Wei Shen, Yiming Ying, and Xiaoming Yuan. Stochastic auc optimization with general loss. *Communications on Pure & Applied Analysis*, 19(8), 2020.

Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Y Arzhaeva, Robert PW Duin, and DMJ Tax. Linear model combining by optimizing the area under the roc curve. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 4, pages 119–122. IEEE, 2006.

Yingjie Tian, Yong Shi, Xiaojun Chen, and Wenjing Chen. Auc maximizing support vector machines with feature selection. *Procedia Computer Science*, 4:1691–1698, 2011.

Peilin Zhao, Steven C. H. Hoi, Rong Jin, and Tianbao Yang. Online auc maximization. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 233–240, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online auc maximization. *Advances in neural information processing systems*, 29, 2016.

Dohyun Kim and Seung Jun Shin. The regularization paths for the roc-optimizing support vector machines. *Journal of the Korean Statistical Society*, 49(1):264–275, 2020.

Stephan Clémençon, Igor Colin, and Aurélien Bellet. Scaling-up empirical risk minimization: optimization of incomplete u-statistics. *Journal of Machine Learning Research*, 17(76):1–36, 2016.

A J Lee. *U-statistics: Theory and Practice*. Routledge, 2019.

Lian Yan, Robert H Dodier, Michael Mozer, and Richard H Wolniewicz. Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 848–855, 2003.

Gunnar Blom. Some properties of incomplete u-statistics. *Biometrika*, pages 573–580, 1976.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95, 1971.

Kai Zhang, Liang Lan, Zhuang Wang, and Fabian Moerchen. Scaling up kernel svm on limited resources: A low-rank linearization approach. In *Artificial intelligence and statistics*, pages 1425–1434. PMLR, 2012.

Christopher Williams and Matthias Seeger. Using the nyström method to speed up kernel machines. *Advances in neural information processing systems*, 13, 2000.

Kai Zhang and James T Kwok. Clustered nyström method for large scale manifold learning and dimension reduction. *IEEE Transactions on Neural Networks*, 21(10):1576–1587, 2010.

Petros Drineas, Michael W Mahoney, and Nello Cristianini. On the nyström method for approximating a gram matrix for improved kernel-based learning. *journal of machine learning research*, 6(12), 2005.

Rajen Bhatt and Abhinav Dhall. Skin Segmentation. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C5T30C.

Sérgio Moro, P. Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decis. Support Syst.*, 62:22–31, 2014. URL https://api.semanticscholar.org/CorpusID:14181100.

Abdullah AlSahly. Accelerometer Gyro Mobile Phone. UCI Machine Learning Repository, 2022. DOI: https://doi.org/10.3390/s22176513.

Appendix

The Appendix provides the proofs of Proposition 3 and Theorem 4 presented in the main article.

A.1 Proof of the proposition 3

Proof. The loss function $[1 - \boldsymbol{\theta}^\top \{\mathbf{k}_i - \mathbf{k}_j\}]_+$ in (10) can be equivalently written as $[1 - \boldsymbol{\theta}^\top \mathbf{V}(\mathbf{v}_i - \mathbf{v}_j)]_+$, where $\mathbf{v}_i$ denotes the ith row of $\mathbf{V}$. Similarly, the penalty term $\frac{\lambda}{2}\boldsymbol{\theta}^\top \mathbf{K}\boldsymbol{\theta}$ can be expressed as

$$\frac{\lambda}{2}\boldsymbol{\theta}^\top \mathbf{V}\mathbf{V}^\top \boldsymbol{\theta} = \frac{\lambda}{2}(\mathbf{V}^\top \boldsymbol{\theta})^\top (\mathbf{V}^\top \boldsymbol{\theta}).$$

By reparametrizing as $\boldsymbol{\beta} = \mathbf{V}^\top \boldsymbol{\theta} \in \mathbb{R}^{d\times 1}$, the objective function can be rewritten as

$$\frac{1}{n_+ n_-} \sum_{i\in I_+} \sum_{j\in I_-} [1 - \boldsymbol{\theta}^\top \mathbf{V}(\mathbf{v}_i - \mathbf{v}_j)]_+ + \frac{\lambda}{2}(\mathbf{V}^\top \boldsymbol{\theta})^\top (\mathbf{V}^\top \boldsymbol{\theta}) = \frac{1}{n_+ n_-} \sum_{i\in I_+} \sum_{j\in I_-} [1 - \boldsymbol{\beta}^\top (\mathbf{v}_i - \mathbf{v}_j)]_+ + \frac{\lambda}{2}\boldsymbol{\beta}^\top \boldsymbol{\beta}.$$

This coincides with the objective function of a linear ROC-SVM with respect to the transformed features $\mathbf{v}_i$. □

A.2 Proof of the theorem 4

Proof. We begin by introducing the parameter vectors $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_n)^\top$ and $\tilde{\boldsymbol{\theta}}_B = (\tilde{\theta}_{B,1}, \ldots, \tilde{\theta}_{B,n})^\top$ from $\hat{f}$ and $\tilde{f}_B$. By the Representer Theorem, $\hat{f}$ and $\tilde{f}_B$ can be represented in terms of $\hat{\boldsymbol{\theta}}$ and $\tilde{\boldsymbol{\theta}}_B$:

$$\hat{f}(\mathbf{x}) = \sum_{k=1}^{n} \hat{\theta}_k K(\mathbf{x}, \mathbf{x}_k), \quad \tilde{f}_B(\mathbf{x}) = \sum_{k=1}^{n} \tilde{\theta}_{B,k} \tilde{K}(\mathbf{x}, \mathbf{x}_k).$$

Here, we set the intercept term to zero without loss of generality, since it cancels out in pairwise difference.

We next establish upper bounds on the ℓ_2 norms of $\hat{\boldsymbol{\theta}}$ and $\tilde{\boldsymbol{\theta}}_B$ by first deriving their RKHS norm bounds. These RKHS norm bounds follow directly from the definitions of $\hat{f}$ and $\tilde{f}_B$, the minimizers of the regularized loss function. Note that these optimization problems can be equivalently expressed in constrained form:

$$\hat{f} = \underset{f\in\mathcal{H}}{\operatorname{argmin}} \frac{1}{n_+ n_-} \sum_{i\in I_+} \sum_{j\in I_-} [1 - \{f(\mathbf{x}_i) - f(\mathbf{x}_j)\}]_+ \quad \text{subject to} \quad \|f\|_{\mathcal{H}}^2 \le C,$$

$$\tilde{f}_B = \underset{f\in\tilde{\mathcal{H}}}{\operatorname{argmin}} \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} [1 - \{f(\mathbf{x}_i) - f(\mathbf{x}_j)\}]_+ \quad \text{subject to} \quad \|f\|_{\tilde{\mathcal{H}}}^2 \le \tilde{C}.$$

From these constraints, it follows that the minimizers satisfy

$$\|\hat{f}\|_{\mathcal{H}}^2 \leq C, \qquad \|\tilde{f}_B\|_{\tilde{\mathcal{H}}}^2 \leq \tilde{C}_B,$$

where, by the reproducing property,

$$\|\hat{f}\|_{\mathcal{H}}^2 = \|\hat{\boldsymbol{\theta}}\|_{\mathcal{H}}^2 = \hat{\boldsymbol{\theta}}^\top \mathbf{K}\hat{\boldsymbol{\theta}}, \qquad \|\tilde{f}_B\|_{\tilde{\mathcal{H}}}^2 = \|\tilde{\boldsymbol{\theta}}_B\|_{\tilde{\mathcal{H}}}^2 = \tilde{\boldsymbol{\theta}}_B^\top \tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B.$$

Therefore, $\|\hat{\boldsymbol{\theta}}\|_{\mathcal{H}}^2$ and $\|\tilde{\boldsymbol{\theta}}_B\|_{\tilde{\mathcal{H}}}^2$ are bounded by C and $\tilde{C}_B$, respectively. From these RKHS norm bounds and the Rayleigh quotient inequality, we have

$$\|\hat{\boldsymbol{\theta}}\|_2 \;\leq\; \sqrt{\frac{\|\hat{\boldsymbol{\theta}}\|_{\mathcal{H}}^2}{\lambda_{\min}}} \;\leq\; \sqrt{\frac{C}{\lambda_{\min}}},$$

since $\mathbf{K}$ is symmetric positive definite. Although $\tilde{\mathbf{K}}$ is only positive semi-definite, the Rayleigh quotient can still be applied if we assume that $\tilde{\boldsymbol{\theta}}_B \perp \text{null}(\tilde{\mathbf{K}})$. This assumption is trivial, because any component of $\tilde{\boldsymbol{\theta}}_B$ lying in null($\tilde{\mathbf{K}}$) (denote it by $\tilde{\boldsymbol{\theta}}_{B,\perp}$) does not affect the solution: it vanishes under multiplication with $\tilde{\mathbf{K}}$ in the decision function $\tilde{f}_B$. Therefore, the Rayleigh quotient bound on Range($\tilde{\mathbf{K}}$) yields

$$\|\tilde{\boldsymbol{\theta}}_B\|_2 \;\leq\; \sqrt{\frac{\|\tilde{\boldsymbol{\theta}}_B\|_{\tilde{\mathcal{H}}}^2}{\tilde{\lambda}_{\min}^+}} \;\leq\; \sqrt{\frac{\tilde{C}_B}{\tilde{\lambda}_{\min}^+}}.$$

We now turn to bounding $|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)|$, which is the main goal of this proof. By the triangle inequality,

$$|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)| \leq |U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B)| + |\tilde{U}_B(\hat{f}_B) - \tilde{U}_B(\tilde{f}_B)|,$$

where $\hat{f}_B$ denotes the incomplete counterpart of $\hat{f}$. That is,

$$\hat{f}_B = \operatorname*{argmin}_{f\in\mathcal{H}} \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} [1 - (f(\mathbf{x}_i) - f(\mathbf{x}_j))]_+ + \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2. \tag{15}$$

Similarly to $\hat{f}$, $\|\hat{f}_B\|_{\mathcal{H}}$ is bounded by a constant $\sqrt{C_B}$, and we define $C_{\max} = \max(C, C_B)$. The corresponding $\hat{\boldsymbol{\theta}}_B$ can also be obtained from

$$\hat{f}_B(\mathbf{x}) = \sum_{k=1}^{n} \hat{\theta}_{B,k} K(\mathbf{x}, \mathbf{x}_k). \tag{16}$$

For the first term $|U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B)|$, since $\hat{f}$ is the minimizer of $U_n(f)$ over $\mathcal{H}$ and $\hat{f}_B \in \mathcal{H}$, we have $U_n(\hat{f}) \leq U_n(\hat{f}_B)$. Thus, if $\tilde{U}_B(\hat{f}_B) < U_n(\hat{f})$, then

$$|U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B)| = U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B) \leq U_n(\hat{f}_B) - \tilde{U}_B(\hat{f}_B) = |U_n(\hat{f}_B) - \tilde{U}_B(\hat{f}_B)|.$$

If instead $U_n(\hat{f}) \leq \tilde{U}_B(\hat{f}_B)$, then

$$|U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B)| = \tilde{U}_B(\hat{f}_B) - U_n(\hat{f}) \leq \tilde{U}_B(\hat{f}) - U_n(\hat{f}) = |\tilde{U}_B(\hat{f}) - U_n(\hat{f})|,$$

since $\hat{f}_B$ is the minimizer of $\tilde{U}_B(f)$ over $\mathcal{H}$ and thus $\tilde{U}_B(\hat{f}_B) \leq \tilde{U}_B(\hat{f})$. Therefore,

$$|U_n(\hat{f}) - \tilde{U}_B(\hat{f}_B)| \leq \sup_{f\in\{\hat{f},\hat{f}_B\}} |\tilde{U}_B(f) - U_n(f)|. \tag{17}$$

To derive a probability bound for $\sup_{f\in\{\hat{f},\hat{f}_B\}} |\tilde{U}_B(f) - U_n(f)|$, we introduce the random sequence, $((\zeta_k(I))_{I\in\Lambda})_{1\leq k\leq B}$, where Λ is defined as $\{(i,j)|i \in I_+, j \in I_-\}$ and $\zeta_k(I)$ is equal to 1 if $I = (I_1, I_2) \in \Lambda$ has been selected at

the k-th draw and 0 otherwise: For $k \in \{1, 2, \ldots, B\}$, each $(\zeta_k(I))_{I\in\Lambda}$ are i.i.d. random vectors following a Multinomial$\left(1; \frac{1}{|\Lambda|}, \ldots, \frac{1}{|\Lambda|}\right)$ distribution. Then,

$$\tilde{U}_B(f) = \frac{1}{B}\sum_{k=1}^{B}\sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}), \quad U_n(f) = \frac{1}{|\Lambda|}\sum_{I\in\Lambda}H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}),$$

where $H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2})$ denotes the kernel function (13). Given the observed data,

$$\begin{aligned} &P\left(\sup_{f\in\{\hat{f},\hat{f}_B\}}|\tilde{U}_B(f) - U_n(f)| > \eta \,\middle|\, (\mathbf{x}_{I_1}, \mathbf{x}_{I_2})_{(I_1,I_2)\in\Lambda}\right) \\ &\le \sum_{f\in\{\hat{f},\hat{f}_B\}} P\left(\left|\frac{1}{B}\sum_{k=1}^{B}\sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}) - \frac{1}{|\Lambda|}\sum_{I\in\Lambda}H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2})\right| > \eta \,\middle|\, (\mathbf{x}_{I_1}, \mathbf{x}_{I_2})_{(I_1,I_2)\in\Lambda}\right), \end{aligned}$$

where

$$E_\zeta\left[\frac{1}{B}\sum_{k=1}^{B}\sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}) \,\middle|\, (\mathbf{x}_{I_1}, \mathbf{x}_{I_2})_{(I_1,I_2)\in\Lambda}\right] = \frac{1}{|\Lambda|}\sum_{I\in\Lambda}H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}).$$

Note that for each k,

$$\sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}) \le \sup_{f\in\{\hat{f},\hat{f}_B\}}\sup_{\mathbf{x}_1,\mathbf{x}_2}[1 - \{f(\mathbf{x}_1) - f(\mathbf{x}_2)\}]_+ \le 1 + 2\sup_{f\in\{\hat{f},\hat{f}_B\}}\sup_{1\le i\le n}|f(\mathbf{x}_i)|.$$

Since $\hat{f}, \hat{f}_B \in \mathcal{H}$, the reproducing property and the Cauchy–Schwarz inequality yield

$$\sup_{f\in\{\hat{f},\hat{f}_B\}}\sup_{1\le i\le n}|f(\mathbf{x}_i)| = \sup_{f\in\{\hat{f},\hat{f}_B\}}\sup_{1\le i\le n}|\langle f, K(\cdot, \mathbf{x}_i)\rangle_{\mathcal{H}}| \le \sup_{f\in\{\hat{f},\hat{f}_B\}}\|f\|_{\mathcal{H}}\sup_{1\le i\le n}\|K(\cdot, \mathbf{x}_i)\|_{\mathcal{H}} \le K_{\max}\sqrt{C_{\max}},$$

which gives

$$0 \;\le\; \sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}) \;\le\; 1 + 2K_{\max}\sqrt{C_{\max}}.$$

Therefore, Hoeffding's inequality yields

$$P\left(\left|\frac{1}{B}\sum_{k=1}^{B}\sum_{I\in\Lambda}\zeta_k(I)H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2}) - \frac{1}{|\Lambda|}\sum_{I\in\Lambda}H_f(\mathbf{x}_{I_1}, \mathbf{x}_{I_2})\right| > \eta \,\middle|\, (\mathbf{x}_{I_1}, \mathbf{x}_{I_2})_{(I_1,I_2)\in\Lambda}\right) \le 2\exp\left(-\frac{2B\eta^2}{(1 + 2K_{\max}\sqrt{C_{\max}})^2}\right).$$

Consequently,

$$P\left(\sup_{f\in\{\hat{f},\hat{f}_B\}}|\tilde{U}_B(f) - U_n(f)| > \eta \,\middle|\, (\mathbf{x}_{I_1}, \mathbf{x}_{I_2})_{(I_1,I_2)\in\Lambda}\right) \le 4\exp\left(-\frac{2B\eta^2}{(1 + 2K_{\max}\sqrt{C_{\max}})^2}\right).$$

That is, for all $\delta \in (0, 1)$, with probability at least $1 - \delta$,

$$\sup_{f\in\{\hat{f},\hat{f}_B\}}|\tilde{U}_B(f) - U_n(f)| \le (1 + 2K_{\max}\sqrt{C_{\max}})\sqrt{\frac{\log(4/\delta)}{2B}}.$$

Therefore, combining with (17), we obtain that for all $\delta \in (0, 1)$, with probability at least $1 - \delta$,

$$|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)| \le (1 + 2K_{\max}\sqrt{C_{\max}})\sqrt{\frac{\log(4/\delta)}{2B}}.$$

For the second term $|\tilde{U}_B(\hat{f}_B) - \tilde{U}_B(\tilde{f}_B)|$,

$$\begin{aligned}
|\tilde{U}_B(\hat{f}_B) - \tilde{U}_B(\tilde{f}_B)| &\leq \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left| \left[1 - \{\hat{f}_B(\mathbf{x}_i) - \hat{f}_B(\mathbf{x}_j)\}\right]_+ - \left[1 - \{\tilde{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_j)\}\right]_+ \right| \\
&\leq \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left| \hat{f}_B(\mathbf{x}_i) - \hat{f}_B(\mathbf{x}_j) - \left(\tilde{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_j)\right) \right| \quad \text{(1-Lipschitz property of hinge loss)} \\
&\leq \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left(|\hat{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_i)| + |\hat{f}_B(\mathbf{x}_j) - \tilde{f}_B(\mathbf{x}_j)| \right)
\end{aligned}$$

Here, we introduce a random variable N_i $(1 \leq i \leq n)$, which represents the number of times the ith sample is selected in $\mathcal{D}_B$ and let $N_{\max} = \max_{1\leq i\leq n} N_i$. Then,

$$\begin{aligned}
\frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left(|\hat{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_i)| + |\hat{f}_B(\mathbf{x}_j) - \tilde{f}_B(\mathbf{x}_j)| \right) &= \frac{1}{B} \sum_{1\leq i\leq n} N_i |\hat{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_i)| \\
&\leq \frac{N_{\max}}{B} \sum_{1\leq i\leq n} |\hat{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_i)| \\
&\leq \frac{N_{\max}}{B} \sum_{1\leq i\leq n} \left(|(\mathbf{K}\hat{\boldsymbol{\theta}}_B)_i - (\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i| + |(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i - \tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_i| \right)
\end{aligned}$$

Define $\Delta\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}_B - \tilde{\boldsymbol{\theta}}_B$ and $\mathcal{E} = \mathbf{K} - \tilde{\mathbf{K}}$. Then,

$$\begin{aligned}
\frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left(|\hat{f}_B(\mathbf{x}_i) - \tilde{f}_B(\mathbf{x}_i)| + |\hat{f}_B(\mathbf{x}_j) - \tilde{f}_B(\mathbf{x}_j)| \right) &\leq \frac{N_{\max}}{B} (\|\mathbf{K}\Delta\boldsymbol{\theta}\|_1 + \|\mathcal{E}\tilde{\boldsymbol{\theta}}_B\|_1) \\
&\leq \frac{N_{\max}}{B} (n\|\mathbf{K}\Delta\boldsymbol{\theta}\|_\infty + \sqrt{n}\|\mathcal{E}\tilde{\boldsymbol{\theta}}_B\|_2) \\
&\leq \frac{N_{\max}}{B} (n\|\mathbf{K}\Delta\boldsymbol{\theta}\|_\infty + \sqrt{n}\|\mathcal{E}\|_2\|\tilde{\boldsymbol{\theta}}_B\|_2)
\end{aligned}$$

where the last inequality follows from the Cauchy–Schwarz inequality. In order to bound $\|\mathbf{K}\Delta\boldsymbol{\theta}\|_\infty$, define

$$f_\Delta(\mathbf{x}) = \sum_{k=1}^{n} \Delta\boldsymbol{\theta}_k \, K(\mathbf{x}, \mathbf{x}_k),$$

so that $f_\Delta \in \mathcal{H}$ and $f_\Delta(\mathbf{x}_i) = (\mathbf{K}\Delta\boldsymbol{\theta})_i$ for $1 \leq i \leq n$. By the reproducing property and the Cauchy–Schwarz inequality,

$$\|\mathbf{K}\Delta\boldsymbol{\theta}\|_\infty = \sup_{1\leq i\leq n} |f_\Delta(\mathbf{x}_i)| = \sup_{1\leq i\leq n} |\langle f_\Delta, K(\cdot, \mathbf{x}_i)\rangle_{\mathcal{H}}| \leq \|f_\Delta\|_{\mathcal{H}} \sup_{1\leq i\leq n} \|K(\cdot, \mathbf{x}_i)\|_{\mathcal{H}} = K_{\max} \|f_\Delta\|_{\mathcal{H}}.$$

Thus, we have

$$|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)| \leq \frac{N_{\max}}{B} (nK_{\max} \|f_\Delta\|_{\mathcal{H}} + \sqrt{n}\|\mathcal{E}\|_2\|\hat{\boldsymbol{\theta}}_B\|_2). \tag{18}$$

To bound $\|f_\Delta\|_{\mathcal{H}} = \|\Delta\boldsymbol{\theta}\|_{\mathcal{H}} = \sqrt{\Delta\boldsymbol{\theta}^\top \mathbf{K}\Delta\boldsymbol{\theta}}$, consider the problem (15). As the problem (3) can be reformulated into problem (10), the objective function in (15) can likewise be expressed in terms of $\boldsymbol{\theta}$:

$$F_{B,\mathbf{k}}(\boldsymbol{\theta}) = L_{B,\mathbf{k}}(\boldsymbol{\theta}) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_{\mathcal{H}}^2,$$

where

$$L_{B,\mathbf{k}}(\boldsymbol{\theta}) = \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \left[1 - \boldsymbol{\theta}^\top(\mathbf{k}_i - \mathbf{k}_j)\right]_+.$$

The function $L_{B,\mathbf{k}}(\boldsymbol{\theta})$ is convex and the regularization term $\frac{\lambda}{2}\|\boldsymbol{\theta}\|_{\mathcal{H}}^2$ is λ-strongly convex, which implies $F_{B,\mathbf{k}}(\boldsymbol{\theta})$ is λ-strongly convex. Since $\hat{f}_B$ is the solution of (15), $\hat{\boldsymbol{\theta}}_B$ obtained from (16) minimizes $F_{B,\mathbf{k}}(\boldsymbol{\theta})$. Therefore,

$$\frac{\lambda}{2}\|\Delta\boldsymbol{\theta}\|_{\mathcal{H}}^2 \leq F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B). \tag{19}$$

To proceed, we decompose the right-hand side as

$$F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B) = \big[F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\tilde{\mathbf{k}}}(\tilde{\boldsymbol{\theta}}_B)\big] + \big[F_{B,\tilde{\mathbf{k}}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\tilde{\mathbf{k}}}(\hat{\boldsymbol{\theta}}_B)\big] + \big[F_{B,\tilde{\mathbf{k}}}(\hat{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B)\big],$$

where $F_{B,\tilde{\mathbf{k}}}(\boldsymbol{\theta})$ and $L_{B,\tilde{\mathbf{k}}}(\boldsymbol{\theta})$ are defined analogously to $F_{B,\mathbf{k}}(\boldsymbol{\theta})$ and $L_{B,\mathbf{k}}(\boldsymbol{\theta})$, with $\mathbf{K}, \mathbf{k}$ replaced by their Nyström approximations $\tilde{\mathbf{K}}, \tilde{\mathbf{k}}$. That is, $F_{B,\tilde{\mathbf{k}}}(\boldsymbol{\theta})$ is the objective function of (14) written in terms of $\boldsymbol{\theta}$, obtained by substituting $f(\mathbf{x}) = \sum_{k=1}^{n} \theta_k \tilde{K}(\mathbf{x}, \mathbf{x}_k)$ into (14). Since $\tilde{f}_B$ is the solution of (14) and thus the corresponding $\tilde{\boldsymbol{\theta}}_B$ minimizes $F_{B,\tilde{\mathbf{k}}}(\boldsymbol{\theta})$, it follows that

$$F_{B,\tilde{\mathbf{k}}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\tilde{\mathbf{k}}}(\hat{\boldsymbol{\theta}}_B) \leq 0.$$

Therefore,

$$\begin{aligned}
F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B) &\leq \big[F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\tilde{\mathbf{k}}}(\tilde{\boldsymbol{\theta}}_B)\big] + \big[F_{B,\tilde{\mathbf{k}}}(\hat{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B)\big] \\
&= \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} \big\{\big[1 - \tilde{\boldsymbol{\theta}}_B^\top(\mathbf{k}_i - \mathbf{k}_j)\big]_+ - \big[1 - \tilde{\boldsymbol{\theta}}_B^\top(\tilde{\mathbf{k}}_i - \tilde{\mathbf{k}}_j)\big]_+\big\} \\
&\quad + \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} \big\{\big[1 - \hat{\boldsymbol{\theta}}_B^\top(\tilde{\mathbf{k}}_i - \tilde{\mathbf{k}}_j)\big]_+ - \big[1 - \hat{\boldsymbol{\theta}}_B^\top(\mathbf{k}_i - \mathbf{k}_j)\big]_+\big\}.
\end{aligned}$$

Using the 1-Lipschitz property of the hinge loss,

$$\begin{aligned}
F_{B,\mathbf{k}}(\tilde{\boldsymbol{\theta}}_B) - F_{B,\mathbf{k}}(\hat{\boldsymbol{\theta}}_B) &\leq \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} \{|\tilde{\boldsymbol{\theta}}_B^\top(\mathbf{k}_i - \mathbf{k}_j) - \tilde{\boldsymbol{\theta}}_B^\top(\tilde{\mathbf{k}}_i - \tilde{\mathbf{k}}_j)| + |\hat{\boldsymbol{\theta}}_B^\top(\tilde{\mathbf{k}}_i - \tilde{\mathbf{k}}_j) - \hat{\boldsymbol{\theta}}_B^\top(\mathbf{k}_i - \mathbf{k}_j)|\} \\
&= \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} |(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i - (\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_j - (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_i + (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_j| \\
&\quad + \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} |(\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_i - (\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_j - (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_i + (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_j| \\
&\leq \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} \{|(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i - (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_i| + |(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_j - (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_j|\} \\
&\quad + \frac{1}{B}\sum_{(i,j)\in\mathcal{D}_B} \{|(\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_i - (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_i| + |(\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_j - (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_j|\} \\
&= \frac{1}{B}\sum_{l\leq i\leq n} N_i\{|(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i - (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_i| + |(\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_i - (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_i|\} \\
&\leq \frac{N_{\max}}{B}\sum_{l\leq i\leq n} \{|(\mathbf{K}\tilde{\boldsymbol{\theta}}_B)_i - (\tilde{\mathbf{K}}\tilde{\boldsymbol{\theta}}_B)_i| + |(\tilde{\mathbf{K}}\hat{\boldsymbol{\theta}}_B)_i - (\mathbf{K}\hat{\boldsymbol{\theta}}_B)_i|\} \\
&\leq \frac{N_{\max}}{B}\sum_{l\leq i\leq n} \{|(\mathcal{E}\tilde{\boldsymbol{\theta}}_B)_i| + |(\mathcal{E}\hat{\boldsymbol{\theta}}_B)_i|\} \\
&\leq \frac{N_{\max}\sqrt{n}}{B}(\|\mathcal{E}\tilde{\boldsymbol{\theta}}_B\|_2 + \|\mathcal{E}\hat{\boldsymbol{\theta}}_B\|_2) \\
&\leq \frac{N_{\max}\|\mathcal{E}\|_2\sqrt{n}}{B}(\|\tilde{\boldsymbol{\theta}}_B\|_2 + \|\hat{\boldsymbol{\theta}}_B\|_2)
\end{aligned} \tag{20}$$

Applying (20) into (19),

$$\frac{\lambda}{2}\|\Delta\boldsymbol{\theta}\|_{\mathcal{H}}^2 \leq \frac{N_{\max}\|\mathcal{E}\|_2\sqrt{n}}{B}(\|\tilde{\boldsymbol{\theta}}_B\|_2 + \|\hat{\boldsymbol{\theta}}_B\|_2),$$

and therefore

$$\|\Delta\boldsymbol{\theta}\|_{\mathcal{H}} \leq \sqrt{\frac{2N_{\max}\|\mathcal{E}\|_2\sqrt{n}}{B\lambda}(\|\tilde{\boldsymbol{\theta}}_B\|_2 + \|\hat{\boldsymbol{\theta}}_B\|_2)}. \tag{21}$$

Substituting (21) into (18),

$$|\tilde{U}_B(\hat{f}_B) - \tilde{U}_B(\tilde{f}_B)| \leq \frac{N_{\max}}{B}\left(\|\mathcal{E}\|_2\|\tilde{\boldsymbol{\theta}}_B\|_2\sqrt{n} + K_{\max}\sqrt{\frac{2N_{\max}\|\mathcal{E}\|_2 n^{3/2}}{B\lambda}(\|\tilde{\boldsymbol{\theta}}_B\|_2 + \|\hat{\boldsymbol{\theta}}_B\|_2)}\right)$$

Plugging in $\|\hat{\boldsymbol{\theta}}\|_2 \leq \sqrt{\frac{C}{\lambda_{\min}}}$ and $\|\tilde{\boldsymbol{\theta}}_B\|_2 \leq \sqrt{\frac{\tilde{C}}{\tilde{\lambda}^+_{\min}}}$, we have

$$|\tilde{U}_B(\hat{f}_B) - \tilde{U}_B(\tilde{f}_B)| \leq \frac{N_{\max}}{B}\left(\|\mathcal{E}\|_2\sqrt{\frac{n\tilde{C}}{\tilde{\lambda}^+_{\min}}} + K_{\max}\sqrt{\frac{2N_{\max}\|\mathcal{E}\|_2 n^{3/2}}{B\lambda}\left(\sqrt{\frac{C}{\lambda_{\min}}} + \sqrt{\frac{\tilde{C}}{\tilde{\lambda}^+_{\min}}}\right)}\right).$$

In conclusion, for all $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\begin{aligned}|U_n(\hat{f}) - \tilde{U}_B(\tilde{f}_B)| &\leq (1 + 2K_{\max}\sqrt{C_{\max}})\sqrt{\frac{\log(4/\delta)}{2B}} \\ &+ \frac{N_{\max}}{B}\left(\|\mathcal{E}\|_2\sqrt{\frac{n\tilde{C}}{\tilde{\lambda}^+_{\min}}} + K_{\max}\sqrt{\frac{2N_{\max}\|\mathcal{E}\|_2 n^{3/2}}{B\lambda}\left(\sqrt{\frac{C}{\lambda_{\min}}} + \sqrt{\frac{\tilde{C}}{\tilde{\lambda}^+_{\min}}}\right)}\right).\end{aligned}$$

□