Multi-scale Nyström Method

Woosang Lim$^1$, Rundong Du$^1$, Bo Dai$^1$, Kyomin Jung$^2$, Le Song$^{1,3}$, Haesun Park$^1$
Georgia Institute of Technology$^1$, Seoul National University$^2$, Ant Financial$^3$
woosang.lim@cc.gatech.edu, {rdu, bodai}@gatech.edu, kjung@snu.ac.kr, {lsong, hpark}@cc.gatech.edu

Abstract

Kernel methods are powerful tools for modeling nonlinear data. However, the amount of computation and memory required for kernel methods becomes the bottleneck when dealing with large-scale problems. In this paper, we propose Nested Nyström Method (NNM) which achieves a delicate balance between the approximation accuracy and computational efficiency by exploiting the multilayer structure and multiple compressions. Even when the size of the kernel matrix is very large, NNM consistently decomposes very small matrices to update the eigen-decomposition of the kernel matrix. We theoretically show that NNM implicitly updates the principal subspace through the multiple layers, and also prove that its corresponding errors of rank-$k$ PSD matrix approximation and kernel PCA (KPCA) are decreased by using additional sublayers before the final layer. Finally, we empirically demonstrate the decreasing property of errors of NNM with the additional sublayers through the experiments on the constructed kernel matrices of real data sets, and show that NNM effectively controls the efficiency both for rank-$k$ PSD matrix approximation and KPCA.

1 Introduction

The scalability of kernel methods is the major bottleneck for applying them to large-scale problems due to the computational and memory cost caused by the large dense kernel matrices. Nyström method is one of the effective methods for accelerating the kernel methods by low-rank approximation of the kernel matrix, \( \mathbf{K} \in \mathbb{R}^{n \times n} \). There has been a large body of work that further improves the approximation quality and computational efficiency via adopting various sampling methods [5, 15, 4, 8, 13, 3, 11, 23, 10] and refining approximation formula [7, 5, 12, 20, 13, 18]. Especially, for rank-$k$ spectral decomposition of \( \mathbf{K} \), there are two basic rank-$k$ Nyström methods which are rank-$k$ Standard Nyström Method (SNM) [5] and orthogonal Nyström method (ONM) [7]. Recently, their efficient versions which are SNM using Randomized SVD (SNM+Rand.SVD) [12] and Double Nyström Method (DNM) [13] were proposed. All these four methods implicitly approximate the first \( k \) principal directions \( \mathbf{U}_{\mathbf{Y},k} \) of \( n \) mapped data points \( \mathbf{Y} \) in the feature space to compute the rank-$k$ spectral decomposition of \( \mathbf{K} = \mathbf{Y}^\top \mathbf{Y} \) with distinct schemes based on different motivations [13]. Rank-$k$ SNM [5] actually computes the first \( k \) principal directions \( \mathbf{U}_{\mathbf{S},k} \) of \( s \) sample mapped points \( \mathbf{S} \) in the feature space, and SNM+Rand.SVD [12] uses randomized SVD to improve efficiency for computing the principal directions of sample mapped points. That is, rank-$k$ SNM and SNM+Rand.SVD approximate \( \mathbf{U}_{\mathbf{Y},k} \) via \( \mathbf{U}_{\mathbf{S},k} \), which is computed by a particular form. However, it is known that both these two approximations lose the orthogonality and are biased to the sample subspace which is \( \text{range}(\mathbf{S}) \). On the other hand, the ONM computes the best \( k \) approximate principal orthogonal direction in the sample subspace \( \text{range}(\mathbf{S}) \) in the sense of minimize the KPCA reconstruction error [13]. However, such approximation requires extra computation, resulting higher time complexity \( O(s^2 n) \) compared to the time complexity of rank-$k$ SNM which is \( O(k sn + k^3) \). To further accelerate ONM, DNM [13] uses ONM twice in different scales, so that to compress the sample subspace \( \text{range}(\mathbf{S}) \) for reducing the dimension of possible solution space for efficient computing of \( \mathbf{U}_{\mathbf{Y},k} \). Although the algorithm performs well in practice, there is no analysis about how its rank-$k$ approximation error varies after compression of sample subspace, and it is not clear whether the double scales are enough in terms of the balance between approximation accuracy and computation efficiency.

To achieve a better trade-off between these two factors,
we extend the DNM to a multi-scale Nyström method. Accelerating the algorithms by exploiting multi-scale structures has been studied for the various methods including FEM [6], Bayesian optimization [21] and neural network [1] to solve the nonlinear problems, and there are also a number of applications such as multi-scale stable kernel construction [17], manifold learning [19], dictionary learning [16], and object detection [2, 14]. Among them, feature pyramid networks [14] successfully achieves both efficient and accurate object detection. Inspired by the multi-scale approximation, we propose a multi-scale Nyström method, Nested Nyström Method (NNM), for both efficient and accurate multi-layer structure which consists of t sublayers and the final layer to efficiently and accurately updates the first k principal direction \( \mathbf{U}_{\mathbf{Y},k} \) for computing a rank-k spectral decomposition of \( \mathbf{K} \). We note that NNM is a general multi-scale framework which can be combined with any other column sampling, and our contribution is orthogonal to the column samplings. Interestingly, it can be viewed as t fully connected neural networks in the structure of NNM as described in Fig 1. We first briefly introduce the rank-k Nyström algorithms in Section 2. Then, we describe the nested Nyström method and provide an error analysis of NNM accordingly in Section 3. In Section 5, we will demonstrate our theoretical analysis of NNM and show that NNM is efficient for both rank-k PSD matrix approximation and KPCA on several benchmarks.

## 2 Rank-k Nyström Methods and Their Implicit Equations

In this section, we briefly introduce the notations and discuss the Nyström methods with the implicit equations regarding to approximating principal directions \( \mathbf{U}_{\mathbf{Y},k} \) of \( n \) mapped data points \( \mathbf{Y} \) in the feature space.

By the spectral theorem, for any \( n \times n \) PSD matrix \( \mathbf{K} \), there exists a matrix \( \mathbf{Y} \in \mathbb{R}^{d \times n} \) which can be considered as \( n \) mapped data points so that \( \mathbf{K} = \mathbf{Y}^\top \mathbf{Y} \) without loss of generality, where \( d \) is finite [5]. Even if \( \mathbf{K} \) is generated by RBF kernel, the unknown mapped data in feature space can be isomorphically represented as \( \mathbf{Y} \) s.t. \( \mathbf{K} = \mathbf{Y}^\top \mathbf{Y} \). Then, let \( \mathbf{S} \) be the \( d \times s \) sample matrix which consists of \( s \) sample columns of \( d \times n \) matrix \( \mathbf{Y} \) corresponding to the column index \( \mathcal{J} \), and let \( \mathbf{C} = \mathbf{Y}^\top \mathbf{S} \) be the \( n \times s \) submatrix of PSD matrix \( \mathbf{K} \), which can be regarded as the inner product matrix of the whole data instances and the samples in the feature space. For kernel methods, \( \mathbf{C} \) can be computed by using the kernel function \( \kappa \), i.e., \( \mathbf{C}_{(i,j)} = \kappa(\mathbf{x}_i, \mathbf{x}_j) \) where \( t \in \mathcal{J} \) is the \( j \)-th sample index among \( s \) sample indices. Let \( \mathbf{K}_\mathcal{S} \) be the \( s \times s \) principal submatrix of \( \mathbf{K} \) s.t. \( \mathbf{K}_\mathcal{S} = \mathbf{S}^\top \mathbf{S} \). For kernel methods using PSD kernel function, without loss of generality, we can also apply these implicit equations \( \mathbf{C} = \mathbf{Y}^\top \mathbf{S} \) and \( \mathbf{K}_\mathcal{S} = \mathbf{S}^\top \mathbf{S} \). A summary of notations is displayed in Tbl 1.

Now, we are going to discuss rank-k Nyström methods which are rank-k standard Nyström method (SNM) [5], SNM using randomized SVD (SNM+Rand.SVD) [12], orthogonal Nyström method (ONM) [7], and double Nyström method (DNM) [13]. These four methods implicitly approximate the first k principal directions \( \mathbf{U}_{\mathbf{Y},k} \) of the subspace spanned by mapped data points \( \mathbf{Y} \) in the feature space to compute the rank-k spectral

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>the number of instances</td>
</tr>
<tr>
<td>( s )</td>
<td>the number of samples</td>
</tr>
<tr>
<td>( \mathbf{K} )</td>
<td>( n \times n ) PSD matrix</td>
</tr>
<tr>
<td>( \mathbf{Y} )</td>
<td>( d \times n ) matrix</td>
</tr>
<tr>
<td>( \mathbf{S} )</td>
<td>( d \times s ) sample matrix of ( \mathbf{Y} )</td>
</tr>
<tr>
<td>( \mathbf{C} )</td>
<td>( n \times s ) sample matrix of ( \mathbf{K} ), ( \mathbf{C} = \mathbf{Y}^\top \mathbf{S} )</td>
</tr>
<tr>
<td>( \mathbf{K}_A )</td>
<td>( \mathbf{A}^\top \mathbf{A} ) for all ( \mathbf{A} )</td>
</tr>
<tr>
<td>( \mathbf{A}' )</td>
<td>compressed matrix for ( \mathbf{A} )</td>
</tr>
<tr>
<td>( \mathbf{A}_k )</td>
<td>approximate compact SVD for ( \forall \mathbf{A} )</td>
</tr>
<tr>
<td>( \mathbf{A}_k )</td>
<td>rank-k SVD for ( \forall \mathbf{A} )</td>
</tr>
<tr>
<td>( \mathbf{A}_k )</td>
<td>approximate rank-k SVD for ( \forall \mathbf{A} )</td>
</tr>
<tr>
<td>( \mathbf{A}_k )</td>
<td>pseudo inverse for ( \forall \mathbf{A} )</td>
</tr>
</tbody>
</table>
decomposition of $K$ as
\[
K_k \approx Y\tilde{U}_{Y,k}^T(Y\tilde{U}_{Y,k})^T Y^T ,
\]
where $K = Y^TY$ without loss of generality, the rank-$k$ SVD of $Y$ is $Y_k = U_{Y,k}\Sigma_{Y,k}(V_{Y,k})^T$, $U_{Y,k}$ is approximation of $U_{Y,k}$, and $K_k = YU_{Y,k}(U_{Y,k})^TY^T$ is the best rank-$k$ approximation of $K$ computed by SVD [13]. The explicit equations of ONM for rank-$k$ is
\[
\tilde{U}_{Y,k}^T = U_{S,k}, \quad \tilde{K}_{snm} = Y(U_{S,k})(U_{S,k})^TY^T ,
\]
where columns of $U_{S,k}$ are the first $k$ principal directions of $S$ s.t. $S_k = U_{S,k}\Sigma_{S,k}V_{S,k}^T$. We note that $\tilde{K}_{snm} = Y(U_{S,k})(U_{S,k})^TY^T = CK_k^S C^T$. Since SNM approximates $U_{Y,k}$ as $U_{S,k}$ Eqn (2) is biased to the sample subspace which is range($S$). To reduce time complexity $O(k s n + s^3)$ to $O(k s n)$, SNM+Rand.SVD [12] uses randomized SVD to quickly decompose $s \times s$ sample matrix $K_S$, and implicitly compute $U_{S,k}$. Thus, we can consider the implicit equations of SNM+Rand.SVD by replacing $U_{S,k}$ in Eqn (2) as $U_{S,k}$.

To efficiently obtain accurate orthonormal eigenvectors in one-shot, orthogonal Nyström method (ONM) was proposed [7]. In fact, rank-$k$ SNM and ONM are identically the same when $k = rank(S)$. However, for $k < rank(S)$, rank-$k$ SNM and ONM can be distinguished by using the modified approximation formula [13]. The explicit equations of ONM for rank-$k$ approximation are
\[
\tilde{K}_{snm} = Y(U_{S,k})(U_{S,k})^TY^T = CK_k^S C^T ,
\]
where $G = CV_S\Sigma_{S}^{-1}$, $\tilde{K}_S = V_S\Sigma_{S}^2 V_S^T$ and $G$ has rank $k$ SVD $G_k = U_{G,k}\Sigma_{G,k}V_{G,k}^T$. The implicit equations of ONM are
\[
\tilde{U}_{Y,k} = U_S G_k, \quad \tilde{K}_{snm} = Y^T U_{Y,k} (U_{Y,k})^T Y ,
\]
where $G = CV_S\Sigma_{S}^{-1} = Y^TY_S$, $\tilde{U}_{Y,k} = GV_{G,k}\Sigma_{G,k}^{-1} = Y^TY_S V_{G,k}(\Sigma_{G,k})^{-1}$. It is straightforward to verify that $U_{S,k}$ and $U_S V_{G,k}$ are different when $k < rank(S)$, consequently two approximations of the first $k$ principal directions $U_{Y,k}$ are different i.e., $U_{Y,km} \neq \tilde{U}_{Y,km}$. Thus, it is again trivial to show that the results of Eqn (2) and Eqn (4) are different when $k < rank(S)$.

The other benefit of ONM is that it solves the sample-based kernel PCA problem (Lem 1) [13]. That is, given sample matrix $S$, $\tilde{U}_{Y,k}$ minimizes the reconstruction error of kernel PCA with the constraint that approximate principal directions are in the range($S$). Columns of $\tilde{V}_{Y,k}^{onm} \Sigma_{Y,k}^{onm}$ are the corresponding principal components.

**Lemma 1** [13] Sample-based kernel PCA is defined as kernel PCA with an additional subspace constraint of range($U_{Y,k}$) $\subseteq$ range($S$). Then, ONM is the optimal sample-based kernel PCA.

Based on Lem 1, double Nyström method (DNM) proposed sample subspace compression, and it uses ONM twice [13]. However, there is no further error analysis regarding a multilayer structure in [13]. We provide Lemma 2, a refined version of Lem 1, which states that ONM computes rank-$k$ SVD of $Y$ such that the computed $k$ principal directions $\tilde{U}_{Y,k}$ minimize reconstruction error of $Y$ regardless of whether $Y$ is mean centered or not.

**Lemma 2** Regardless of the condition of mean centering on $Y$, given sample matrix $S$, ONM computes first $k$ approximate principal directions $\tilde{U}_{Y,k}$ which minimize reconstruction error of $Y$ with range($\tilde{U}_{Y,k}$) $\subseteq$ range($S$), where the reconstruction error of $Y$ is defined as $RE(\tilde{U}_{Y,k}) = \|Y - \tilde{U}_{Y,k}\|_F$.

We will use Lem 2 for the analysis of our method.

### 3 Nested Nyström Method

If the data size $n$ is large, rank-$k$ SNM and ONM need a relatively larger number of samples $s$ to get accurate spectral decomposition, and the approximation will take longer. We propose NNK which consistently decomposes very small matrices to efficiently update the first $k$ principal directions of $Y$ and eigen-decomposition of $K$ even though $n$ is large. NNK is described in Alg 1, and its example is displayed in Fig 2, and the detailed description is as follows.

The multilayer architecture of NNK is based on the following three parts: subsampling part, Nyström method part, and compression part. First, we run subsampling part which constructs a nested sequence of subsample matrices and stacks multiple layers with it. Then, we run both Nyström method and compression parts with the nested sequence of subsample matrices on the $t$ sublayers until the final layer. Specifically, NNK updates...
the principal subspace and compresses sample matrices by using the small-dimensional subspace which is compressed and transformed by computed eigenvectors on each sublayer. At the final layer of NNM, we compute the rank-$k$ spectral decomposition by using ONM with the compressed sample matrices, since ONM computes the best principal directions which minimizes reconstruction error given the samples (Lem 1, Lem 2). We note that NNM computes the true rank-$k$ spectral decomposition when the range of compressed samples includes the true rank-$k$ principal subspace, i.e., range($U_k$) $\subseteq$ range($S'$).

**Subsampling part:** Given indices set $J$ of $s$ samples and the corresponding sample matrices $S$ and $K_S$, we construct a nested index sets $J \supseteq J_1 \supseteq \ldots \supseteq J_t$ and the corresponding nested sequence of submatrices as Eqn (5).

\[ S \supseteq S_1 \supseteq S_2 \supseteq \ldots \supseteq S_t, \tag{5} \]

\[ C \supseteq K_S \supseteq C_1 \supseteq K_{S_1} \supseteq C_2 \supseteq K_{S_2} \supseteq \ldots \supseteq C_t \supseteq K_{S_t}, \]

where $|J_i| = s_i$, and $s \gg s_1 \gg \ldots \gg s_t$. Especially, we can understand $(s_i-1) \times s_i$ matrix $C_i$ and $s_i \times s_i$ matrix $K_S$ with implicit equations as $C_i = S_{i-1}^\top S_i$ and $K_S = S_i^\top S_i$ for $1 \leq i \leq t$, where $S_i$ is $d \times s_i$ and $S_0 = S$.

**Rank-$s_t$ Nyström method part:** In this part, we compute the approximate eigenvectors $\hat{V}_S$ of $K_S$ by using compressed submatrices $C_{S_{t-1},i}$ and $K_{S_{t-1},i}$. From the 1st to the $(t-1)$-th sublayer: We compute the first $s_i$ approximate eigenvectors $\hat{V}_{S_{t-1},i}$ of $K_{S_{t-1}}$ by using compressed submatrices $C_{i+1}'$ and $K_{S_{i+1}}'$ on the $(t-i)$-th layer, where $i \in \{1, 2, \ldots, (t-1)\}$ and $C_i' = C_i$ and $K_S' = K_S$. On the $t$-th sublayer: We compute the first $s_t$ approximate eigenvectors $\hat{V}_{S_{t-1},t}$ of $K_S$ using $C_{S_{t-1},t}$ and $K_{S_{t-1},t}$ of ONM with the compressed sample matrices, since $S_{t-1} \supseteq \ldots \supseteq S_t \supseteq S_{t-1}$. From 1st to the $t$-th sublayer: We compute the rank-$s_t$ Nyström method part with ONM at $i$-th layer is $O(s_t^2 s_{t-1}^{-1})$ for $i \in \{1, 2, \ldots, (t-1)\}$, and the time complexity of rank-$s_t$ Nyström method part with ONM at $t$-th layer is $O(s_t^2 s_t)$. These time complexities are very small, since $n \gg s \gg s_1 \gg \ldots \gg s_t \geq t \geq k$.

**Compression part:** In this part, we compress sample matrices by using the approximate eigenvectors. From the 1st to the $(t-1)$-th sublayer: we compress sample matrices $C_i$ and $K_S$ by using $\hat{V}_{S_{t-1},i}$ as

\[ C_i' = C_i \hat{V}_{S_{t-1},i}, \quad K_S' = (\hat{V}_{S_{t-1},i})^\top K_S \hat{V}_{S_{t-1},i}, \tag{6} \]

where $\hat{V}_{S_{t-1},i}$ is computed at $(t-i)$-th layer, and $i \in \{1, 2, \ldots, (t-1)\}$. On the $t$-th sublayer: we compress sample matrices $C_i$ and $K_S$ by using $\hat{V}_{S_{t-1},t}$ with $k \leq t \leq s_t$

\[ C' = C \hat{V}_{S_{t-1},t}, \quad K_S' = (\hat{V}_{S_{t-1},t})^\top K_S \hat{V}_{S_{t-1},t}. \tag{7} \]

We can connect the compression of sample matrices to the compression of sample subspace with implicit
and space complexities of NNM are sequence of subsamples with a smaller dimension instead of using $s_t$, since the principal subspace of $s_t$ samples is more closer to the rank-$k$ principal subspace of $n$ nodes. That is, we can use $V_{S,t}$ for compression of $S$ instead of using $V_{S,s_t}$, where $k \leq \ell \leq s_t$.

The time complexity of compression part at $(t-i)$-th layer is $O(s_t s_i s_{i-1})$ for $i \in \{1, 2, ..., (t-1)\}$, and the time complexity of compression part at $t$-th layer is $O(t n s)$, where $s_0 = s$. We note that all performances in the compression parts are only matrix multiplications.

**Time complexity analysis:** Suppose that we construct the nested sequence of subsamples with $\sum_{j=1}^{t} s_j = O(s)$, e.g., $\sum_{j=1}^{t} s_j \leq s$, where $s \gg s_1 \gg ... \gg s_t \geq \ell \geq k$. Then, we provide Proposition 1 which states the time and space complexities of NNM.

**Proposition 1** Suppose that we use ONM for rank-$s_t$ Nyström method parts in NNM, and set the nested sequence of subsamples with $\sum_{j=1}^{t} s_j = O(s)$, where $s \gg s_1 \gg ... \gg s_t \geq \ell \geq k$. Then, the total time and space complexities of NNM are $O(\ell n s + s_t s_i s)$ and $O(sn)$, respectively.

The detailed proof of Proposition 1 is in the Appendix. A large portion of the total time complexity $O(\ell n s + s_t s_i s)$ is $O(t s n)$ corresponding to the simple matrix multiplications in the compression parts. In Section 5, we will show that the running time of NNM is linear for $s$.

**Selecting number of subsamples and sublayers:** We can select the number of subsamples based on the condition $\sum_{j=1}^{t} s_j = O(s)$ of Proposition 1. We first set $M$ for $M \cdot s \geq \sum_{j=1}^{t} s_j$. The simple choice of $M$ is 1 or 2, then we have $s \geq \sum_{j=1}^{t} s_j$ or $2s \geq \sum_{j=1}^{t} s_j$. We then define a relation among the numbers of subsamples. One of the simplest way is to set $s_i = a s_{i-1}$ for $i \in \{1, 2, ..., t\}$, where $s_0 = s$ and $0 < a \leq 1$. A smaller $a$ leads to a shorter running time, but a larger $a$ is better to obtain a small approximation error. To attain both efficiency and accuracy, we can set $a = \frac{1}{2}$, then $\sum_{j=1}^{t} s_j = \sum_{j=1}^{t} \frac{1}{2^k} s \leq s$. For example, we can set $s_1 = 1000$, $s_2 = 500$, and $s_3 = 250$ when $s = 2000$ and $t = 3$. We note that $s_t$ and $t$ are tuning parameter, where $s \gg s_1 \gg ... \gg s_t \geq \ell \geq k$. Finally, we note that the proper number of sublayer $t$ should satisfy $M \cdot s \geq \sum_{j=1}^{t} s_j$ and $s \geq s_j$. For example, suppose that we set $a = \frac{1}{2}$ and $M = 2$, then we have $2 \cdot s \geq \sum_{j=1}^{t} s_j$ and $s \geq 2 t^{-j} s_t$. Then, for $s_t = 250$ and $s \in \{2000, 5000\}$, the proper number of sublayer $t$ is between 1 and 4, since we have $2 \cdot s \geq \sum_{j=1}^{4} 2^{t-j} s_t$ and $s \geq 2 t^{-j} s_t$ when $t \in \{1, 2, 3, 4\}$.

**Several properties of NNM:** We note that NNM is a generalized multilayer architecture, not a simple approximation version. For example, NNM with no sublayer is equivalent to ONM, and NNM with one sublayer is equivalent to the DNN. The main difference is that the upper bound of NNM further decreases when we decompose the same sized sample matrix with additional layers. That is, we can compute more accurate rank-$k$ decomposition within the same short time. We will show it in Section 5.

We can use any sampling method along with NNM, since NNM does not need any assumption for properties of sample matrices $C$ and $K_S$. Thus, for kernel methods, we can apply any sampling method both for constructing sample matrices and a nested set of subsamples.

We note that it is possible to replace ONM in the rank-$s_t$ Nyström method part with other eigen-decomposition methods. However, if we use the ONM in the rank-$s_t$ Nyström method part, then the benefit will be small time complexity, low errors, and easy implementation. Furthermore, it guarantees that the upper error bound of NNM decreases when we use an additional sublayer. We will prove it as Thm 1 in Section 4.

We do not consider rank-$k$ SNM at any layers instead of ONM. Since, if we use the SNM at any layers, we can not guarantee that the error decreases even we use additional sublayers or increase $\ell$. We provide a formal statement as Proposition 3 in Section 4.

### 3.1 Extension of NNM

In this section, we discuss the extension of NNM which is described in Alg 2. Suppose that, given the $s$ samples and NNM with $t$ sublayers, we want to compute $s_0$ additional samples to update the spectral decomposition of $n \times n$ PSD matrix $K$ by using the $s_0$ extended
Algorithm 2 Extension of NNM with Additional Samples

Require: the number of additional samples $s_b$. NNM with $t$ sublayers and its inputs and outputs

Ensure: rank-$k$ spectral decomposition of $K$

NNM with $(t + 1)$ sublayers, additional $s_b$ samples (total $s_a = (s + s_b)$ samples), appended sample matrices $C_a$ and $K_{S_a}$

1: **Additional sampling:**

- Sampling additional $s_b$ points
- Constructing appended sample matrices: $n \times s_a$ matrix $C_a$ and $s_a \times s_a$ matrix $K_{S_a}$

2: $(t + 1)$-th sublayer:

- Rank-$s_t$ Nyström method: Compute $\tilde{V}_{S_a,t}$ of $K_{S_a}$ by using the sample matrices of $K_{S_a}$ compressed by $V_{S_a,t}$
- Compression: Compress sample matrices $C_a$ and $K_{S_a}$ as $C'_a$ and $K'_{S_a}$ by using $V_{S_a,t}$ (Eqn (7))

3: **Final layer:**

- Run ONM (Eqn (3)) with $C'_a$ and $K'_{S_a}$

samples, where $s_a = (s + s_b)$. Then, by extending the multilayer structure of NNM, we can efficiently update the spectral decomposition. The extension of NNM which consists of three components: additional sampling, $(t + 1)$-th sublayer, and the final layer. The following descriptions are known.

**Additional sampling:** We can use either uniform or non-uniform sampling. For non-uniform sampling, we can efficiently compute $s_b$ additional samples by using the rank-$k$ spectral decomposition obtained from NNM with $t$ layers, e.g., approximate column norm sampling [5], approximate leverage score sampling [13], and adaptive partial sampling [11]. The implicit equation of $s_a$ samples is $S_a = [S \quad S_a]$, and the implicit equations of appended $n \times s_a$ sample matrix $C_a$ and $s_a \times s_a$ sample matrix $K_{S_a}$ are $C_a = Y^T S_a$ and $K_{S_a} = S_a^T S_a$, respectively.

$(t + 1)$-th sublayer: In the rank-$s_t$ Nyström method part, we efficiently compute $V_{S_a,t}$ of $K_{S_a}$. Let $C_0$ and $K_S$ be the $s_a \times s$ and $s \times s$ sample matrices of $K_{S_a}$, respectively. Then, we compress $C_0$ and $K_S$ as $C'_0$ and $K'_S$ by using $V_{S_a,t}$, which was computed at the $t$ sublayer of NNM, and compute $V_{S_a,t}$ of $K_{S_a}$ by using ONM and the compressed sample matrices $C'_0$ and $K'_S$. In the compression part, we compress sample matrices $C_a$ and $K_{S_a}$ as $C'_a$ and $K'_{S_a}$ by using $V_{S_a,t}$

**Final layer:** We compute $\tilde{V}_k$ by using $C'_a$, $K'_{S_a}$ and ONM.

The time complexity of rank-$s_t$ Nyström method part using ONM in $(t + 1)$-th sublayer is $O(s_t^2 s_a)$, and the time complexity of compression part is $O(\ell^2 n)$. At the final layer, the time complexity of ONM with $C'_a$ and $K'_{S_a}$ is $O(\ell^2 n)$.

We can combine different sampling strategies for computing spectral decomposition of PSD matrices by using the extension of NNM. For example, we can easily combine uniform and approximate leverage score sampling, since the time complexity of computing approximate leverage scores using $\tilde{V}_k$ computed by NNM with $t$ sublayers is $O(kn)$. Then, the total time complexity of NNM with $(t + 1)$ sublayers is $O((s_a n + s_t s_a) \ell)$ which is linear for $s_a$ when $(s + \sum_{j=1}^{t} s_j) = O(s_a)$. In Section 5, we will compare the experimental results of NNM by using uniform sampling and uniform + approximate leverage score sampling. Finally, we note that we can use multi-sublayers between $t$-th sublayer and the final layer, if $s_a/s$ is too large.

4 Error Analysis of NNM

In this section, we provide an error analysis of NNM. First, we provide the implicit representations of compressed sample subspaces to analyze the error of NNM. Next, we show the upper error bounds of NNM, and prove that the upper error bounds decreases when we use additional sublayers.

4.1 Representations of Compressed Sample Subspaces

NNM efficiently and accurately updates the compressed sample matrix $S'_i$ so that $\text{range}(S'_i)$ closely approximates the true principal subspace based on Eqn (8) until the final layer. That is, we want to compute $S'_i$ s.t. $\text{range}(U_k) \subset \text{range}(S'_i)$. Consequently, we need to analyze how compressed sample subspace varies $\text{range}(S'_i)$ through the multilayer structure to analyze NNM.

First, we provide the implicit representation of the principal subspace as $\text{range}(U_k) = \text{range}(U_k \Sigma YV_{Y,k})$. Similarly, we can give the implicit representations of compressed sample subspaces $\text{range}(S'_i)$ and $\text{range}(S'_i) = \text{range}(YV_{Y,s_t})$ and $\text{range}(YV_{Y,t})$, respectively. Lem 3 formally states their implicit representations.

**Lemma 3** Given the multilayer Nyström structure of NNM with $t$ sublayers, let $S = YP$ and $S = S_{t-1}P_1$, where $P_1 P_1 = P_1$ for column sampling, $P_0 = P$, and $S_0 = S$. Then, we have $S_i = YZ_i$ with $Z_i = PP_1 \cdots P_i$ and $S'_i = YV_{Y,t,s_i}$ on the $(t-i)$-th layer, and $S'_i = YV_{Y,t,s_i}$ on the $t$-th layer, where $V_{Y,s_t} = Z_i V_{S_{t-1}s_t}$ and $V_{Y,t} = PV_{S_{t-1}}$. By Lem 3, we note that if $\text{range}(YV_{Y,k}) \subset \text{range}(YV_{Y,t})$, then NNM computes the rank-$k$ spectral decomposition with the optimal error.
4.2 Decrease of Upper Error Bounds of NNM

For the case of using linear combination input $S = YY^T$, with $V_{Y,s}^T V_{Y,s} = I$, the generalized upper error bounds of ONM have been proven [13]. Since the input of the final layer of NNM can be considered as $S' = YV_{Y,t}$ by Lem 3, we provide Lem 4 which states the upper bounds of the final error of NNM.

Lemma 4 Suppose that $S' = YV_{Y,t}$ is the compressed samples as an input of the final layer of NNM, where $V_{Y,t} V_{Y,t} = I$ and $k \leq t \leq s_i$. Then, upper error bounds of NNM for kernel PCA and rank-k matrix approximation are

$$
\text{NRE}(\tilde{U}_{Y,k}) \leq \text{NRE}(U_{Y,k}) + \frac{2c_2(V_{Y,k})}{\gamma_k} \sqrt{\text{tr}(K)} \tag{9}
$$

$$
\text{MRE}(\tilde{K}_k) \leq \text{MRE}(K_k) + \frac{2c_2(V_{Y,k})}{\gamma_k} \text{tr}(K),
$$

where $V_{Y,k}$ is any submatrix consisting of $k$ columns of $V_{Y,t}$, $U_{Y,k}$ consists of the first $k$ approximate principal directions which are implicitly generated by kernel PCA, $\text{NRE}(U_{Y,k}) = \|Y - U_{Y,k} U_{Y,k}^T Y\|_F^2/\|Y\|_F^2$ is the normalized reconstruction error (NRE) of kernel PCA, $\text{MRE}(K_k) = \|K - K_k\|_F$ is the reconstruction error of rank-$k$ PSD matrix approximation, $\gamma_k$ is the $k$-th eigengap, $c_2(V_{Y,k})$ is the sum of errors of eigenvalues from $V_{Y,k}$ with $V_{Y,k} V_{Y,k} = I$ s.t. $c_2(V_{Y,k}) = \text{tr}(V_{Y,k}^T Y^T Y V_{Y,k}) - \text{tr}(V_{Y,k} Y^T Y V_{Y,k})$.

These upper error bounds in Lem 4 only depend on $c_2(V_{Y,k})$, since $\gamma_k$ and $\text{tr}(K)$ are constant given the $K$. The approximation errors in Eqn (9) go to the optimal errors which are $\text{NRE}(U_{Y,k})$ and $\text{MRE}(K_k)$ as $c_2(V_{Y,k})$ goes to 0, and $c_2(V_{Y,k}) = 0$ when range($V_{Y,k}$) $\subset$ range($V_{Y,t}$). Since reducing $c_2(V_{Y,k})$ is important, we need to show how $c_2(V_{Y,k})$ varies through the sublayers.

Suppose that, given the mutilayer structure of NNM with $t$ sublayers, we use ONM for kernel PCA parts and only $i$ sublayers which are from the first sublayer to $i$-th sublayer until the final layer, where $i \in \{1, 2, ..., (t-1)\}$. Then, the $i$-th sublayer becomes the last sublayer before the final layer, and we have $S'_j = Y(Z_{j} V_{S_{j,t}})$ as the input of the final layer with $j = (t - i)$ by Lem 3 and Lem 4. Thus, we can select $V_{Y,t} = Z_j V_{S_{j,t}}$ for $c_2(V_{Y,k})$, i.e., $c_2(Z_j V_{S_{j,t}})$ since $V_{Y,t} = Z_j V_{S_{j,t}}$.

Proposition 2 states that the sum of eigenvalue error $c_2(V_{Y,k})$ decreases as we use additional sublayers.

Proposition 2 Suppose that we use ONM for kernel PCA parts in NNM. Then, we have $c_2(Z_j V_{S_{j,t}})$ and $c_2(Z_j V_{S_{j,t}})$.

Table 2: The summary of 4 real data sets. $n$ is the number of instances and $m$ is the dimension of the original data.

<table>
<thead>
<tr>
<th>data set</th>
<th>Letter</th>
<th>MNIST</th>
<th>MiniBooNE</th>
<th>Covertype</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>m</td>
<td>20000</td>
<td>60000</td>
<td>130064</td>
</tr>
<tr>
<td>m</td>
<td>16</td>
<td>784</td>
<td>50</td>
<td>54</td>
</tr>
</tbody>
</table>

Figure 3: Performance comparison of SNM, SNM+Rand.SVD, and NNM. The upper left figure displays the results for rank-k kernel matrix approximation, and the upper right figure displays their running time. The results show that the error of NNM decreases as we use additional sublayers within the short time.

$\epsilon_2(Z_{j-1} V_{S_{j-1,t}}^\text{onm})$ for NNM using $i$ and $(i+1)$ sublayers respectively, and $\epsilon_2(Z_{j-1} V_{S_{j-1,t}}^\text{onm}) \leq \epsilon_2(Z_{j} V_{S_{j,t}}^\text{onm})$.

Due to the page limit, the proofs are in the Appendix. By Lem 4 and Proposition 2, we provide our main theoretical result Thm 1 which states that the quality of compressed input at the final layer is important, and we can increase accuracy by using more sublayers.

Theorem 1 Suppose that we use ONM for the kernel PCA parts in the sublayers. Then, the upper error bound of NNM in Lem 4 decreases when we use additional sublayers.

Finally, Proposition 3 states that rank-k SNM does not have the benefit for using additional sublayers or increasing $t$ parameter.

Proposition 3 Suppose that we compress $C$ and $K_S$ as $C'$ and $K_S'$ with $V_{S_{t,t}} = V_{S_{t,t}}$ at the $t$-th layer, and we run the rank-k SNM with $C'$ and $K_S'$ at the final layer. Then $K_k^\text{onm}$ are the same regardless of values of $t$, where $t \geq k$. That is, the rank-k spectral decomposition using rank-k SNM is biased by $s$ samples.

5 Experiments

In this section, we present experimental results that demonstrate our theoretical work. We compare rank-k Nyström methods to the rank-k kernel matrix approximation and KPCA. The three error measures which we used are matrix reconstruction error (MRE($K_k$) =
Multi-scale Nyström Method

Figure 4: Performance comparison for rank-$k$ kernel matrix approximation among the NNM with 1, 2, 3, 4 sublayers. NNM ($t = 4$) is more efficient than NNM ($t = 1, 2, 3$).

Figure 5: Comparison of MRE($\hat{K}_k$) for rank-$k$ kernel matrix approximation among the four representative methods with: SNM, SNM+Rand.SVD, ONM, NNM (ours). NNM is more efficient than other state-of-the-art Nyström methods given the short time.

Figure 6: Comparison of convergence to the optimal error with rMRE($\hat{K}_k$) for rank-$k$ kernel matrix approximation. It shows that error of NNM rapidly decreases compared to other Nyström methods.

Figure 7: Comparison of convergence to the optimal error with rNRE($\hat{U}_k$) for KPCA. It shows that reconstruction error of KPCA of NNM rapidly decreases compared to other Nyström methods.

$\|K - \hat{K}_k\|_F$, relative matrix reconstruction error ($\text{rMRE}(\hat{K}_k) = \frac{\|K - \hat{K}_k\|_F}{\|K - \hat{K}_k\|_F} \in [1, \infty)$), and relative KPCA reconstruction error ($\text{rNRE}(\hat{U}_k) = \frac{\|K - \hat{K}_k\|_F}{\|K - \hat{K}_k\|_F} \in [1, \infty)$), where $\|K - \hat{K}_k\|_F$ and NRE($\hat{U}_k$) are the optimal error which comes from SVD. The optimum of rMRE and rNRE is 1. To construct PSD matrix $K$, we use RBF kernel which is defined as

$$\kappa(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right),$$

where $\sigma$ is a kernel parameter. We select 4 real data sets for evaluating performances, and summarize them in Tbl 2.

We empirically compare the NNM described in Alg 1 with four representative Nyström methods: SNM [5], SNM+Rand.SVD [12], ONM [7], and DNM [13]. We abbreviate NNM with $i$ sublayers to NNM ($t = i$) for convenience, and DNM is the same with NNM ($t = 1$). For all methods except NNM ($t = 4$), we set $s = 500j$ with $j = 4, 5, ..., 10$ in Fig 3. We set $s = 3000, 3500, ..., 5000$ for NNM ($t = 4$). We use the same amount of $sn$ kernel matrix elements for all methods.

We set $p$ parameter as 100, 500 for SNM+Rand.SVD, since the error of SNM+Rand.SVD decreases to the error of SNM when $p = 500$ regardless of data sets and experiment settings. We report the additional experimental results with $p = 500, 1000, 2000$ in the Appendix. To compare NNM with SNM+Rand.SVD, we use the following parameters: $s_1 = (1000 + 50j)$, $s_2 = (500 + 25j)$, $s_3 = (250 + 25j)$, $t = (k + 150 + 5j)$ for $s = 2000, 2500, and s_1 = (2000 + 50j)$, $s_2 = (1000 + 50j)$, $s_3 = (500 + 25j)$, $t = (k + 150 + 5j)$ for $s \geq 3000$. With these parameters, we have $s_j \approx 2^{-j}s_1$, and set the maximum number of sublayers $t$ as 4 to satisfy $2 \cdot s \geq \sum_{j=1}^{t} 2^{t-j}s_1$ and $s \geq 2^{t-j}s_t$ (see the paragraph of selecting $t$ in Section 3).

In Fig 3, we can see that the errors of NNM are smaller than SNM and SNM+Rand.SVD, and the errors of NNM further decrease as we use additional sublayers within the short time. We also run NNM by combining uniform sampling (Unif) and approximate leverage score sampling (ALev) based on the extension of NNM. For example, $(t = 2 + 1, \text{Unif+ALev})$ means that we run NNM with 2 sublayers by using Unif, and extend NNM with 1 sublayers by using ALev. In Fig 3, although the error of NNM ($t = 2 + 1, \text{Unif+ALev}$) is higher than NNM ($t = 3$), the error of NNM ($t = 3 + 1, \text{Unif+ALev}$) is smaller than the error of NNM ($t = 4$), since the accuracy of approximate leverage scores computed by NNM increases as we use more sublayers. Fig 4 shows that the error of NNM decreases as we use additional sublayers regardless of data sets. We can see that NNM ($t = 3, 4$) sublayers are more accurate than NNM ($t = 1, 2$) within the same short time. Fig 5 shows that the errors of NNM are smaller than errors of other state-of-the-art Nyström methods within the same short time. Fig 6 and Fig 7 show that the errors of NNM both for
rank-$k$ kernel matrix approximation and KPCA rapidly decrease compared to other rank-$k$ Nyström methods.

6 Conclusion

In this paper, we presented a multi-scale Nyström architecture, called nested Nyström Method (NNM), which efficiently and accurately updates the rank-$k$ spectral decomposition of PSD matrix on the multilayer structure with the nested sequence of subsamples and subspace compression. Both theoretically and empirically, we demonstrated that the error of NNM decreases as we use additional layers. Finally, we showed that NNM is more efficient than other rank-$k$ Nyström methods.

7 Acknowledgments

W. Lim acknowledges support from 2016-2017 Google Ph.D. Fellowship in machine learning. W. Lim and R. Du acknowledge Qingqing Liu who is a Ph.D. student in School of Mathematics at Georgia Tech and verified the proofs of theorem in our paper. K. Jung was supported by the National Research Foundation of Korea(NRF)(No.2016R1A2B2009759). L. Song and B. Dai were supported in part by NSF CAREER IIS-1639792 EAGER, NSF CNS-1704701, was supported by the National Research Foundation of Korea(NRF)(No.2016R1A2B2009759). L. Song and B. Dai were supported in part by NSF CAREER IIS-1350983, NSF IIS-1639792 EAGER, NSF CNS-1704701, ONR N00014-15-1-2340, Intel ISTC, NVIDIA and Amazon AWS.

References

A Additional Experimental Results

In this section, we provide the experimental results with various setting: different sigma, different rank-k, different \( \ell \), and etc. We empirically compare the NNM described in Alg 1 with four representative Nyström methods: standard Nyström method (SNM) [22], standard Nyström method using randomized SVD (SNM+Rand.SVD) [12], one-shot Nyström method (ONM) [7], and double Nyström method (DNM) [13]. We note that DNM is equivalent to NNM with 1 sub-layer. For convenience, we abbreviate NNM with \( i \) sublayers to NNM \( (t = i) \). If we consider the final layer, then NNM \( (t = i) \) has \((i + 1)\) total layers. For all methods, we set \( s = 500j \), where \( j = 4, 5, \ldots, 10 \). Thus, there are 7 episodes for each test and the corresponding 7 points on the each line in the figures. We use the same amount of \( sn \) kernel matrix elements for all methods. We report only the decomposition time in this section, since we report the running time including kernel construction time in the main section (Section 5).

We tested the value of \( p \) parameter of SNM+Rand.SVD from 5 to 2000, since the errors of SNM+Rand.SVD decrease to the errors of SNM as we increase the value \( p \) parameter. We report the experimental results of SNM+Rand.SVD with \( p = 500 \), since the errors of SNM+Rand.SVD decrease close to the errors of SNM regardless of data sets and experiment settings when we increase \( p \) as 500. We observed that the errors of SNM+Rand.SVD did not further decrease even if we increased the value of \( p \) parameter over 500 in the experiments, e.g., \( p = 1000, 2000 \), since the errors of SNM+Rand.SVD converge to the errors of SNM as we increase \( p \) parameter. To compare NNM with SNM+Rand.SVD, we use the following parameters for Fig 8 Fig 9 and Fig 10: For NNM with \( t = 3 \), we set \( s_1 = (1000 + 50j) \), \( s_2 = (500 + 25j) \), \( s_3 = (250 + 25j) \), \( \ell = (k + 80 + 5j) \), where \( j = 4, 5, \ldots, 10 \). For NNM with \( t = 2 \), we set \( s_1 = (500 + 25j) \), \( s_2 = (250 + 25j) \), \( \ell = (k + 80 + 5j) \). For NNM with \( t = 1 \), we set \( s_1 = (250 + 25j) \), \( \ell = (k + 80 + 5j) \). In Fig 8 Fig 9 and Fig 10, we exclude the kernel construction time for report the running time. Meanwhile, in Fig 3 and Fig 4 in Section 5, we reported the running time including kernel construction time.

Fig 8 displays the experimental results on 4 different real data sets. In this experiment, we set \( k = 20 \) instead of \( k = 50 \). We set \( \sigma \) for 4 data sets as follows: \( \sigma = 1.0 \) for Letter, \( \sigma = 5.0 \) for MNIST, \( \sigma = 1.0 \) for MiniBooNE, and \( \sigma = 1.0 \) for Covertype. For our method, we display the experimental results of NNM with 3 sublayers. Fig 8 shows that NNM computes more accurate rank-k matrix approximation compared to the other Nyström methods within the same time. Especially, for Letter and MNIST data sets, we can notice that the error of NNM rapidly decreases to the optimal error. Although SNM+Rand.SVD is slightly

![Figure 8](image-url)
Figure 9: Performance comparison with the different sigma values of kernel function. In this experiment, we display the experimental results of NNM with 3 sublayers. The results show that the errors of NNM is smaller than other Nyström methods within the same short time regardless of the sigma values.

faster, the errors of NNM are much smaller than the errors of SNM+Rand.SVD. ONM is the accurate, but it takes much longer. Fig 9 also shows that NNM is both accurate and efficient regardless of the sigma values of kernel function.

Fig 10 shows that the errors of SNM+Rand.SVD do not further decrease even if we increase the value of $p$ parameter over 500 in the experiments, e.g., $p = 1000, 2000$. We can also notice that the errors of SNM+Rand.SVD converge to the errors of SNM.

B Proof of Proposition 1

Proof 1 For the proof of Proposition 1, we use $\sum_{j=1}^t s_j = O(s)$ and properties of big $O$ notation. The time complexity of kernel PCA part with ONM at $i$-th layer is $O(s_i^2 s_{i-1})$ for $i \in \{1, 2, ..., (t-1)\}$, and the time complexity of kernel PCA part with ONM at $t$-th layer is $O(s_t^2 s)$. Then, the sum of time complexities in kernel PCA parts is bounded as

$$s_t^2 s + \sum_{i=1}^{t-1} s_i^2 s_{i-1} \leq (M+1)s_t^2 s,$$

where $\sum_{j=1}^t s_j \leq M \cdot s$ and we usually set $M \leq 2$ in the experiments. Thus, by the properties of big O notation, the total time complexity of kernel PCA parts using ONM is $O(s_t^2 s)$.

The time complexity of compression part at $(t-i)$-th layer is $O(s_is_t s_{i-1})$ for $i \in \{1, 2, ..., (t-1)\}$, and the time complexity of compression part at $t$-th layer is $O(sn)$, where $s_0 = s$. Then, the sum of time complexities in compression parts is bounded as

$$\ell sn + \sum_{i=1}^{t-1} s_is_{i-1} = \ell sn + s_is_1s + \sum_{i=1}^{t-2} s_is_{i+1}s_i \leq \ell sn + (M+1)s_is_1s,$$

where $s_0 = s$ and $\sum_{j=1}^t s_j \leq M \cdot s$. Thus, the total time complexity of kernel PCA parts using ONM is $O(\ell sn + s_is_1s)$.

Finally, at the final layer, the time complexity of ONM with $C'$ and $K_S$ is $O(\ell^2 n)$, since $C' = n \times \ell$ and $K_S$ is $\ell \times \ell$. Therefore, the total time complexity of NNM is $O(\ell sn + s_is_1s)$.

C Proof of Lem 3

Lem 3 formally states implicit representations of compressed subspaces.

Lemma 3: Given the mutliayer Nyström structure of NNM with $t$ sublayers, let $S = YP$ and $S_i = S_{i-1}P_i$, where $P_i^TP_i = I$ for column sampling, $P_0 = P$, and $S_0 = S$. Then, we have $S_i = YZ_i$ with $Z_i = PP_1 \cdots P_{i-1}$ and $S_i' = YV_{Y,s}$ on the $(t-i)$-th layer, and $S_i' = YV_{Y,t}$ on the $t$-th layer, where $V_{Y,s,t} = Z_i'V_{S,s,t}$. Therefore, the total time complexity of NNM is $O(\ell sn + s_is_1s)$.

Proof 2 For a general case, we assume that a multi-layer Nyström structure of NNM has $t$ sublayers with the nested sequence of samples in Eqn (5). Without loss of generality, we can define $P_j$ for column sampling s.t. $S = YP$ and $S_i = S_{i-1}P_i$ with $(P_i)^TP_i = I$, where
\[ P_0 = P, \text{ and } S_0 = S. \text{ Then, we have } S_i = YZ_i, \text{ where } Z_j = PP_1 \cdots P_j \text{ and } Z_j = P. \]

By Eqn (6), Eqn (7) and Eqn (8), we have \( S_i' = S_i \tilde{V}_{S_i,s_i} \) on the \((t-i)\)-th sublayer for \( i \in \{1, \ldots, (t-1)\} \) and \( S' = S'_{S,t} \) on the \( t \)-th sublayer. Thus, if we apply \( S_i = YZ_i \) to the \( S_i' = S_i \tilde{V}_{S_i,s_i} \) and \( S' = S'_{S,t} \), then we have \( S_i' = YZ_i \tilde{V}_{S_i,s_i} \) on the \((t-i)\)-th layer, and \( S' = YP' \tilde{V}_{S,t} \) on the \( t \)-th layer. Since \( (Z_i \tilde{V}_{S_i,s_i})^\top Z_i \tilde{V}_{S_i,s_i} = I \) and \( (P' \tilde{V}_{S,t})^\top P' \tilde{V}_{S,t} = I \), we can think that \( S_i' = Y \tilde{V}_{Y,s_i} \) on the \((t-i)\)-th layer with \( \tilde{V}_{Y,s_i} = Z_i \tilde{V}_{S_i,s_i} \), and \( S' = Y \tilde{V}_{Y,t} \) on the \( t \)-th layer with \( \tilde{V}_{Y,t} = P' \tilde{V}_{S,t} \).

By Lem 3, we note that if \( \text{range}(Y \tilde{V}_{Y,k}) \subset \text{range}(Y \tilde{V}_{Y,t}) = \text{range}(S') \), then NNM computes the rank-\( k \) spectral decomposition with the optimal error.
D Proof of Lem 4

For the case of using linear combination input $S = YY_V s$ with $V_Y V s = I$, the generalized upper error bounds of ONM have been proven as Proposition 4 [13].

**Proposition 4** [13] If we set $S = YY_V s$ with $V_Y V s = I$, then errors of kernel PCA and rank-$k$ matrix approximation using ONM are bounded as follows:

$$\text{NRE}(\hat{U}_{Y,k}) \leq \text{NRE}(U_{Y,k}) + \sqrt{\frac{2e_2(V_{Y,k})}{\gamma_k}}$$
$$\text{MRE}(\hat{K}_k) \leq \text{MRE}(K_k) + \sqrt{\frac{2e_2(V_{Y,k})}{\gamma_k} \text{tr}(K)},$$

where $\hat{U}_{Y,k}$ consists of the first $k$ approximate principal directions which are implicitly generated by kernel PCA, $\text{NRE}(U_{Y,k}) = \|Y - U_{Y,k} U_{Y,k}^T Y\|_F/\|Y\|_F$ is the normalized error (NRE) of kernel PCA, $\text{MRE}(K_k) = \|K - K_k\|_F$ is the error of rank-$k$ PSD matrix approximation, $\gamma_k$ is the $k$-th eigengap, $e_2(V_{Y,k})$ is the sum of errors of eigenvalues from $V_{Y,k}$ s.t. $e_2(V_{Y,k}) = \text{tr}(V_{Y,k}^T Y^T Y V_{Y,k}) - \text{tr}(V_{Y,k}^T Y^T Y V_{Y,k})$, and $V_{Y,k}$ is any submatrix consisting of $k$ columns of $V_Y$.

The input of the final layer of NNM can be considered as $S' = YY_V \ell$ by Lem 3, then we can consider $s$ samples, $V_Y s$, and $S$ in Proposition 4 as compressed $\ell$ samples, $V_Y \ell$, and $S'$, respectively. Since the input of the final layer of NNM satisfy the condition of input in Proposition 4, we can provide Lem 4 which states the upper bounds of the final errors of NNM.

**Lemma 4:** Suppose that $S' = YY_V \ell$ is the compressed samples as an input of the final layer of NNM, where $V_Y \ell V_Y \ell = I$ and $k \leq \ell \leq s$. Then, upper error bounds of NNM for kernel PCA and rank-$k$ matrix approximation are

$$\text{NRE}(\hat{U}_{Y,k}) \leq \text{NRE}(U_{Y,k}) + \sqrt{\frac{2e_2(V_{Y,k})}{\gamma_k}}$$
$$\text{MRE}(\hat{K}_k) \leq \text{MRE}(K_k) + \sqrt{\frac{2e_2(V_{Y,k})}{\gamma_k} \text{tr}(K)},$$

where $V_{Y,k}$ is any submatrix consisting of $k$ columns of $V_Y \ell$, $U_{Y,k}$ consists of the first $k$ approximate principal directions which are implicitly generated by kernel PCA, $\text{NRE}(U_{Y,k}) = \|Y - U_{Y,k} U_{Y,k}^T Y\|_F/\|Y\|_F$ is the normalized reconstruction error (NRE) of kernel PCA, $\text{MRE}(K_k) = \|K - K_k\|_F$ is the reconstruction error of rank-$k$ PSD matrix approximation, $\gamma_k$ is the $k$-th eigengap, $e_2(V_{Y,k})$ is the sum of errors of eigenvalues from $V_{Y,k}$ with $V_{Y,k}^T V_{Y,k} = I$ s.t. $e_2(V_{Y,k}) = \text{tr}(V_{Y,k}^T Y^T Y V_{Y,k}) - \text{tr}(V_{Y,k}^T Y^T Y V_{Y,k})$.

E Proof of Prop 2

We provide Proposition 2 which states that the eigenvalue error of $K$ from $V_{Y,k}$ decreases as we use additional sublayers.

**Proposition 2** Suppose that we use ONM for kernel PCA parts in NNM. Then, we have $e_2(Z_j V_{S_{j,k}})$ and $e_2(Z_j V_{S_{j,k}}^{onm})$ for $S_j$ using $i$ and $(i + 1)$ sublayers respectively, and $e_2(Z_j V_{S_{j,k}}^{onm}) \leq e_2(Z_j V_{S_{j,k}})$.

**Proof 3** To prove Proposition 2, we need Cor 1, Lem 7, Lem 6 and Lem 5, and their statements and proofs are in the following subsections.

Suppose that, given the multilayer structure of NNM with $t$ sublayers, we use ONM for kernel PCA parts and only $i$ sublayers which are from the first sublayer to $i$-th sublayer until the final layer, where $i \in \{1, 2, \ldots, (t - 1)\}$. Then, the $i$-th sublayer becomes the last sublayer before the final layer, and we have $S_j = Y(Z_j V_{S_{j,k}})$ as the input of the final layer with $j = (t - i)$ by Lem 3 and Lem 4. Since $V_{Y,t} = Z_j V_{S_{j,k}}^{onm}$, we can select $V_{Y,k} = Z_j V_{S_{j,k}}^{onm}$ which is the $k$ columns of $Z_j V_{S_{j,k}}^{onm}$ for $e_2(V_{Y,k})$.

Similarly, suppose that we use only $(i + 1)$ sublayers from the first sublayer to $(i + 1)$-th sublayer until the final layer, where $i \in \{1, 2, \ldots, (t - 1)\}$. Then, we have $S_{j-1} = Y Z_{j-1} V_{S_{j-1,k}}^{onm}$ as the input of the final layer with $j = (t - i)$, and we can select $V_{Y,k} = Z_{j-1} V_{S_{j-1,k}}^{onm}$ for $e_2(V_{Y,k})$ since $V_{Y,t} = Z_{j-1} V_{S_{j-1,k}}^{onm}$.

Thus, if we apply $V_{Y,k} = Z_j V_{S_{j,k}}^{onm}$ to $e_2(V_{Y,k})$ for using $i$ sublayer and $V_{Y,k} = Z_{j-1} V_{S_{j-1,k}}^{onm}$ to $e_2(V_{Y,k})$ for using $(i + 1)$ sublayers, then we have $e_2(Z_j V_{S_{j,k}}^{onm})$ and $e_2(Z_{j-1} V_{S_{j-1,k}}^{onm})$, respectively.

Next, we need to prove Eqn (11)

$$e_2(Z_j V_{S_{j-1,k}}^{onm}) \leq e_2(Z_j V_{S_{j,k}}^{onm}),$$

where $j = (t - i)$ and $i \in \{1, 2, \ldots, (t - 1)\}$. Eqn (11) is equivalent to Eqn (12) by Lem 5.

$$\text{tr}(V_{S_{j,k}}^{onm})^T S_j^T S_j V_{S_{j,k}}^{onm} \leq \text{tr}(V_{S_{j-1,k}}^{onm})^T S_{j-1}^T S_{j-1} V_{S_{j-1,k}}^{onm},$$

where $j = (t - i)$ and $i \in \{1, 2, \ldots, (t - 1)\}$.

By Lem 6, we have

$$\text{tr}(V_{S_{j-1,k}}^{onm})^T S_j^T S_j V_{S_{j,k}}^{onm} \leq \text{tr}(((S_j^T)^T S_j^T S_j V_{S_{j,k}}^{onm}))), \text{tr}(K_{S_{j,k}})$$
where \((\cdot)_k\) stands for the best rank \(k\) approximation, and \(V_{S_{j},k}^\text{rom}\) consists of the first \(k\) columns of \(V_{S_{j},l}^\text{rom}\).

Without loss of generalization, we can define \(P_j\) to satisfy \(S_j = S_{j-1} P_j\), then \((P_j)^{\top} P_j = I\). Now, we have
\[
S'_{*} = S_j V_{S_{j},l}^\text{rom} \quad \text{and} \quad P'_{*} = S_{j-1} P_j V_{S_{j},l}^\text{rom} = S_{j-1} P'_{*},
\]
where \(P'_{*}\) is due to the cases in Thm 1. By using compact SVD of \(S'_{j}\) s.t. \(S'_{j} = U_{S_{j}} \Sigma_{S_{j}} V_{S_{j}}^{\top}\), then we have
\[
\text{tr}((S'_{j})^{\top} S'_{j}) = \text{tr}(U_{S_{j}}^{\top} S_{j}^{\top} U_{S_{j}}(S_{j}^{\top} U_{S_{j}})) \leq \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}).
\]
The last inequality holds because of Lem 6 and the fact that \((P'_{j})^{\top} P'_{j} = I\).

So far, we have shown
\[
\text{tr}(V_{S_{j},k}^\text{rom})^{\top} S_{j}^{\top} S_{j} V_{S_{j},k}^\text{rom} \leq \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}).
\]

Now, we want to argue that
\[
\text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}) \leq \text{tr}((U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}})_{k}).
\]

We note that \(U_{S_{j},k}^\text{rom}\) and \(U_{S_{j},k}^\text{rom}\) are in the range of \(S'_{j}\).

Then, by Cor 1,
\[
\text{tr}((U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}})_{k}) = \max \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k}) \quad \text{subject to} \quad U_{S_{j}}^{\top} U_{S_{j}} = I.
\]

Thus, we have
\[
\text{tr}(V_{S_{j},k}^\text{rom})^{\top} S_{j}^{\top} S_{j} V_{S_{j},k}^\text{rom} \leq \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k}).
\]

Next, by Lem 7, we have
\[
\text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k}) \leq \text{tr}(V_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} V_{S_{j}}_{k}).
\]

Consequently, we have
\[
\text{tr}(V_{S_{j},k}^\text{rom})^{\top} S_{j}^{\top} S_{j} V_{S_{j},k}^\text{rom} \leq \text{tr}(V_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} V_{S_{j}}_{k}).
\]

E.1 Lemma 5

Lemma 5 Given the \(S_j = YZ_j\), if we consider \(V_{S_{j},k} = Z_j V_{S_{j}}_{k}\), then the followings are equivalent
\[
\text{maximize} \text{tr}(V_{S_{j},k}^{\top} S_{j}^{\top} S_{j} V_{S_{j},k}) \quad \text{or} \quad \text{minimize} \epsilon_2(Z_j V_{S_{j}}_{k}).
\]

Proof 4 Let us remind that the sum of eigenvalue errors from \(V_{S_{j},k}\) is defined as
\[
\epsilon_2(V_{S_{j},k}) = \text{tr}(V_{S_{j},k}^{\top} Y^{\top} Y V_{S_{j},k}) - \text{tr}(V_{S_{j},k}^{\top} Y^{\top} Y V_{S_{j},k}),
\]
where \(K = Y^{\top} Y, \quad Y = U_{S_{j}} \Sigma_{S_{j}} V_{S_{j}}^{\top}\), \(Y_k = U_{S_{j},k} \Sigma_{S_{j},k} V_{S_{j},k}^{\top}\), and \((V_{S_{j},k})^{\top} V_{S_{j},k} = I\).

If we consider \(V_{S_{j},k} = Z_j V_{S_{j}}_{k}\), then we have
\[
\epsilon_2(Z_j V_{S_{j}}_{k}) = \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k}) \leq \text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k}).
\]

Since \(\text{tr}(U_{S_{j}}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j}}_{k})\) is constant given the rank-\(k\) and \(K\), we complete the proof.

E.2 Lemma 6

Lemma 6 [9] Let \(A \in \mathbb{R}^{m \times n}\) have singular values \(\sigma_1(A) \geq \cdots \geq \sigma_q(A) \geq 0\), where \(q = \min\{m, n\}\). For each \(k = 1, \ldots, q\) we have
\[
\sum_{i=1}^{k} \sigma_i(A) = \max \{|\text{tr}(X^{\top} AX)| : X \in \mathbb{R}^{m \times k}, \quad Y \in \mathbb{R}^{n \times k}, \quad X^{\top} Y = X^{\top} = Y = I}\).

E.3 Corollary 1

Corollary 1 For a multilayer Nyström architecture of NNM, we have
\[
U_{S_{j-1},k}^{\text{rom}} = \arg\max_{U_{S_{j-1},k}} \text{tr}(U_{S_{j-1},k}^{\top} S_{j-1}^{\top} S_{j-1}^{\top} U_{S_{j-1},k}) \quad \text{subject to} \quad U_{S_{j-1},k}^{\top} U_{S_{j-1},k} = I.
\]

That is, given the subsample matrices \(S_j\) and \(S_{j-1}\), ONM minimizes the sum of eigenvalue errors of \(K_{S_{j-1}}\) under the formula of Eqn (13).

Proof 5 Since \(S_{j}' = S_j V_{S_{j},l}^{\text{rom}} = S_{j-1} P_j V_{S_{j},l}^{\text{rom}} = S_{j-1} P_{j}'\), we note that \(U_{S_{j-1},k}^{\text{rom}}\) and \(U_{S_{j},k}^{\top}\) are in the range of \(S_{j}'\). Then, we can easily derive Cor 1 from Lem 1 and Lem 2.
Therefore, \( \Sigma D \) consisting of diagonal elements of \( \Sigma \).

**Proof 6**  First we have \( \sum B_j \Sigma_j \sum_j B_j \Sigma_j = (\Sigma^2 - \Sigma D)^2 \). Next, suppose that we consider full SVD of \( S_j \) s.t. \( S_j = U S_j \Sigma_j V_{S_j}^\top \), then we have \( \sum U_s \Sigma_s \sum_s = \Sigma_s \). By Cauchy-Schwarz inequality,

\[
\sum_{j=1}^m \sigma_j^2 b_{ji} = \left( \sum_{j=1}^m \sigma_j^2 b_{ji} \right)^2 = \sum_{j=1}^m \sigma_j^2 b_{ji}^2 = d_i^2.
\]

Therefore, \( \sum_{j=1}^m \sigma_j^2 b_{ji} / d_i \geq d_i \). Thus, we have

\[
\sum D \geq \sum D = \sum D.
\]

F  Proof of Thm 1

To prove Thm 1, we need Lem 3, Lem 5, Proposition 2, and Lem 4, whose statements and proofs are in the Appendix.

**Theorem 1** Suppose that we use ONM for the kernel PCA parts in the sublayers. Then, the upper error bound of NNM in Lem 4 decreases when we use additional sublayers. That is, if we add an additional sublayer to the NNM structure with \( i \) sublayers as \( (i + 1) \)-th sublayer, then upper error bounds further decrease as \( \epsilon^2 (V_{Y, \ell}^\top) \).

**Proof 8** Suppose that, given the multilayer structure of NNM with \( t \) sublayers, we use ONM for kernel PCA parts and only \( i \) sublayers which are from the first sublayer to \( i \)-th sublayer until the final layer, where \( i \in \{1, 2, \ldots, (t - 1)\} \). Then, the \( i \)-th sublayer becomes the last sublayer before the final layer, and we have \( S_j = Y Z_j \sum_j \) as the input of the final layer with \( j = (t - i) \) by Lem 3 and Lem 4. Since \( \sum V_{Y, \ell} = Z_j \sum_j \), we can select \( \sum V_{Y, \ell} = Z_j \sum_j \), which is the first \( k \) columns of \( Z_j \sum_j \) for \( \epsilon^2 (V_{Y, \ell}^\top) \).

Similarly, suppose that we use only \( (i + 1) \) sublayers from the first sublayer to \( (i + 1) \)-th sublayer until the final layer, where \( i \in \{1, 2, \ldots, (t - 1)\} \). Then, we have \( S_j = Y Z_j \sum_j \) as the input of the final layer with \( j = (t - i) \) by Lem 3 and Lem 4. Since \( \sum V_{Y, \ell} = Z_j \sum_j \), we can select \( \sum V_{Y, \ell} = Z_j \sum_j \), which is the first \( k \) columns of \( Z_j \sum_j \) for \( \epsilon^2 (V_{Y, \ell}^\top) \).

Thus, if we apply \( \sum V_{Y, \ell} = Z_j \sum_j \) to \( \epsilon^2 (V_{Y, \ell}^\top) \) for using \( i \) sublayer and \( \sum V_{Y, \ell} = Z_j \sum_j \) to \( \epsilon^2 (V_{Y, \ell}^\top) \) for using \( (i + 1) \) sublayers, then we have \( \epsilon^2 (Z_j \sum_j \) and \( \epsilon^2 (Z_j \sum_j \) for \( \sum V_{Y, \ell} = Z_j \sum_j \).

We also note that, given the \( t \) column vectors of \( Z_j \sum_j \) \( \epsilon^2 (Z_j \sum_j \) is the minimum among \( \epsilon^2 (Z_j \sum_j \) where \( \sum V_{Y, \ell} = Z_j \sum_j \) is any submatrix consisting of \( k \) column vectors of \( Z_j \sum_j \)$ and $\ell = (t - i)$ for using \( i \) sublayer. We can easily prove it by considering Lem 1, Lem 2 and the definition of \( \sum V_{Y, \ell} = Z_j \sum_j \) which is the first \( k \) column vectors of \( \sum V_{Y, \ell} = Z_j \sum_j \)$ s.t.

\[
\sum V_{Y, \ell} = \arg\max \{ \text{tr}(\sum V_{Y, \ell}) S_j S_j \sum V_{S_j, k} \}
\]

subject to \( \text{range}(\sum V_{S_j, k} = \text{range}(S_j) \) and \( \sum V_{S_j, k} \sum V_{S_j, k} = I \),

where we compute \( \sum V_{S_j, k} \) by using ONM and the compressed sample matrices \( S_j = \sum_j S_j + 1 \) and \( K_{S_j} = \sum_j S_j + 1 \). Since maximizing \( \text{tr}(\sum V_{S_j, k} S_j S_j \sum V_{S_j, k}) \)
is equivalent to minimizing $\epsilon^2(Z_j\tilde{V}_{S,j,k})$ by Lem 5, $\epsilon^2(Z_j\tilde{V}_{S,j,k}^{nm})$ is the minimum among $\epsilon^2(V_{Y,k})$, where $V_{Y,k}$ is any submatrix consisting of $k$ column vectors of $Z_j\tilde{V}_{S,j}$. 

Thus, we complete the proof, since the minimum of sum of eigenvalue errors $\epsilon^2(V_{Y,k})$ in Lem 4 decreases as we use additional sublayers until the final layer.

G Proof of Proposition 3

We provide Proposition 3 which states why the SNM should not be utilized at the final layer of NNM.

Proposition 3 Suppose that we compress $C$ and $K_S$ as $C'$ and $K_S$ with $\tilde{V}_{S,\ell} = V_{S,\ell}$ at the $t$-th layer, and we run the SNM with $C'$ and $K_S'$ at the final layer. Then $K_{k}^{snm}$ are the same regardless of values of $\ell$, where $\ell \geq k$. That is, the rank-$k$ spectral decomposition using rank-$k$ SNM is biased by $s$ samples.

Proof 9 Suppose that $\tilde{V}_{Y,\ell} = V_{S,\ell}$ for some $\ell \geq k$, and $\sigma_i(S) \neq 0$ for $i = 1,...,\ell$. Then, we have $C' = CV_{S,\ell} = Y^T S V_{S,\ell}$ and $K_S' = S'^T S' = V_{S,\ell} S^T S V_{S,\ell} = \Sigma_{S,\ell}^2$. Consequently, $K_{S',k} = \Sigma_{S,k}^{-2}$. If we run the standard Nyström method at the final layer with $C'$ and $K_S'$, then we have

$$K_{k}^{snm} = C'K_{S',k}C'^\top = Y^T S V_{S,\ell} \Sigma_{S,k}^{-2} V_{S,\ell}^T Y$$
$$= Y^T U_{S,\ell} \Sigma_{S,\ell} \Sigma_{S,k}^{-2} \Sigma_{S,\ell} U_{S,\ell}^T Y$$
$$= Y^T U_{S,k} U_{S,k}^T Y.$$