Ensemble Nyström

Sanjiv Kumar, Mehryar Mohri and Ameet Talwalkar

A common problem in many areas of large-scale machine learning involves manipulation of a large matrix. This matrix may be a kernel matrix arising in Support Vector Machines [9, 15], Kernel Principal Component Analysis [47] or manifold learning [43, 51]. Large matrices also naturally arise in other applications, e.g., clustering, collaborative filtering, matrix completion, and robust PCA. For these large-scale problems, the number of matrix entries can easily be in the order of billions or more, making them hard to process or even store. An attractive solution to this problem involves the Nyström method, in which one samples a small number of columns from the original matrix and generates its low-rank approximation using the sampled columns [53]. The accuracy of the Nyström method depends on the number columns sampled from the original matrix. Larger the number of samples, higher the accuracy but slower the method.

In the Nyström method, one needs to perform SVD on a $l \times l$ matrix where l is the number of columns sampled from the original matrix. This SVD operation is typically carried out on a single machine. Thus, the maximum value of l used for an application is limited by the capacity of the machine. That is why in practice, one restricts l to be less than $20K$ or $30K$, even when the size of matrix is in millions. This restricts the accuracy of the Nyström method in very large-scale settings.

This chapter describes a family of algorithms based on mixtures of Nyström approximations called, *Ensemble Nyström algorithms*, which yields more accurate low-rank approximations than the standard Nyström method. The core idea of Ensemble Nyström is to sample many subsets of columns from the original matrix, each containing a relatively small number of columns. Then, Nyström method is

Sanjiv Kumar
Google Research, New York, NY, USA e-mail: sanjivk@google.com

Mehryar Mohri
Courant Institute, New York, NY, USA e-mail: mohri@cs.nyu.edu

Ameet Talwalkar
Division of Computer Science, University of California, Berkeley, CA, USA e-mail: ameet@eecs.berkeley.edu

performed on each group independently in parallel, and the results are combined yielding high accuracy. These ensemble algorithms naturally fit within distributed computing environments where their computational costs are roughly the same as that of the standard Nyström method. This issue is of great practical significance given the prevalence of distributed computing frameworks to handle large-scale learning problems. Several variants of these algorithms are described, including one based on simple averaging of p Nyström solutions, an exponential weighting method, and a regression based method which consists of estimating the mixture parameters using a few sampled columns.

In Sect. 1, we first introduce the notation and basic concepts of low-rank matrix approximation. The standard Nyström method is also described. Then, we present a number of Ensemble Nyström algorithms in Sect. 1.2. In many applications, one needs inverse of a large matrix e.g., SVM and Gaussian Processes. Deriving approximate inverse using the standard Nyström method is easy but not so for the Ensemble Nyström. We further show in Sect. 1.3 how one can efficiently use Woodbury's approximation with Ensemble Nyström to generate approximate inverses.

Another interesting aspect of the Ensemble Nyström methods is their theoretical properties that give explicit bounds for the reconstruction error for both the Frobenius norm and the spectral norm. In Sect. 2, we give a derivation of these bounds. These arise by developing a different bound for the standard Nyström method as used in practice, i.e., using uniform random sampling of columns without replacement. These novel generalization bounds guarantee a better convergence rate for Ensemble Nyström algorithms in comparison to the standard Nyström method.

Sect. 3 demonstrates the results from Ensemble Nyström algorithms on multiple data sets. A comprehensive comparison against other methods shows clear performance gains over the standard Nyström method. Sect. 3.2 describes a large-scale experiment with $1M$ points leading to a matrix of size $1M \times 1M$. This is a huge dense matrix, containing 1 trillion entries and its explicit storage would require 4TB space. We show that sampling based methods can easily handle such matrices and the proposed Ensemble Nyström outperforms other state-of-the-art methods for a fixed computational budget.

To conclude, we provide a summary of the chapter and discuss several open questions in Sect. 4. Further, related work is mentioned in Sect. 5.

1 Algorithms

Let $\mathbf{T} \in \mathbb{R}^{a \times b}$ be an arbitrary matrix. We define $\mathbf{T}^{(j)}$, $j = 1 \ldots b$, as the jth column vector of $\mathbf{T}$, $\mathbf{T}_{(i)}$, $i = 1 \ldots a$, as the ith row vector of $\mathbf{T}$ and $\|\cdot\|$ the l_2 norm of a vector. Furthermore, $\mathbf{T}^{(i:j)}$ refers to the ith through jth columns of $\mathbf{T}$ and $\mathbf{T}_{(i:j)}$ refers to the ith through jth rows of $\mathbf{T}$. If $\mathrm{rank}(\mathbf{T}) = r$, we can write the thin Singular Value Decomposition (SVD) of this matrix as $\mathbf{T} = \mathbf{U}_T \Sigma_T \mathbf{V}_T^\top$ where $\Sigma_T \in \mathbb{R}^{r \times r}$ is diagonal and contains the singular values of $\mathbf{T}$ sorted in decreasing order and $\mathbf{U}_T \in \mathbb{R}^{a \times r}$ and $\mathbf{V}_T \in \mathbb{R}^{b \times r}$ have orthogonal columns that contain the left and right singular

vectors of $\mathbf{T}$ corresponding to its singular values. We denote by $\mathbf{T}_k$ the 'best' rank-k approximation to $\mathbf{T}$, i.e., $\mathbf{T}_k = \text{argmin}_{\mathbf{V}\in\mathbb{R}^{a\times b},\text{rank}(\mathbf{V})=k}\|\mathbf{T}-\mathbf{V}\|_\xi$, where $\xi \in \{2,F\}$ and $\|\cdot\|_2$ denotes the spectral norm and $\|\cdot\|_F$ the Frobenius norm of a matrix. We can describe this matrix in terms of its SVD as $\mathbf{T}_k = \mathbf{U}_{T,k}\Sigma_{T,k}\mathbf{V}_{T,k}^\top$ where $\Sigma_{T,k}$ is a diagonal matrix of the top k singular values of $\mathbf{T}$ and $\mathbf{U}_{T,k}$ and $\mathbf{V}_{T,k}$ are the associated left and right singular vectors.

Now let $\mathbf{K} \in \mathbb{R}^{n\times n}$ be a symmetric positive semidefinite (SPSD) kernel or Gram matrix with $\text{rank}(\mathbf{K}) = r \leq n$, i.e. a symmetric matrix for which there exists an $\mathbf{X} \in \mathbb{R}^{N\times n}$ such that $\mathbf{K} = \mathbf{X}^\top\mathbf{X}$. We will write the SVD of $\mathbf{K}$ as $\mathbf{K} = \mathbf{U}\Sigma\mathbf{U}^\top$, where the columns of $\mathbf{U}$ are orthogonal and $\Sigma = \text{diag}(\sigma_1,\ldots,\sigma_r)$ is diagonal. The pseudo-inverse of $\mathbf{K}$ is defined as $\mathbf{K}^+ = \sum_{t=1}^r \sigma_t^{-1}\mathbf{U}^{(t)}{\mathbf{U}^{(t)}}^\top$, and $\mathbf{K}^+ = \mathbf{K}^{-1}$ when $\mathbf{K}$ is full rank. For $k < r$, $\mathbf{K}_k = \sum_{t=1}^k \sigma_t\mathbf{U}^{(t)}{\mathbf{U}^{(t)}}^\top = \mathbf{U}_k\Sigma_k\mathbf{U}_k^\top$ is the 'best' rank-k approximation to $\mathbf{K}$, i.e., $\mathbf{K}_k = \text{argmin}_{\mathbf{K}'\in\mathbb{R}^{n\times n},\text{rank}(\mathbf{K}')=k}\|\mathbf{K}-\mathbf{K}'\|_{\xi\in\{2,F\}}$, with $\|\mathbf{K}-\mathbf{K}_k\|_2 = \sigma_{k+1}$ and $\|\mathbf{K}-\mathbf{K}_k\|_F = \sqrt{\sum_{t=k+1}^r \sigma_t^2}$ [23].

We will be focusing on generating an approximation $\widetilde{\mathbf{K}}$ of $\mathbf{K}$ based on a sample of $l \ll n$ of its columns. We assume that l columns are sampled from $\mathbf{K}$ uniformly without replacement. Let $\mathbf{C}$ denote the $n \times l$ matrix formed by these columns and $\mathbf{W}$ the $l \times l$ matrix consisting of the intersection of these l columns with the corresponding l rows of $\mathbf{K}$. Note that $\mathbf{W}$ is SPSD since $\mathbf{K}$ is SPSD. Without loss of generality, the columns and rows of $\mathbf{K}$ can be rearranged based on this sampling so that $\mathbf{K}$ and $\mathbf{C}$ can be written as follows:

$$\mathbf{K} = \begin{bmatrix} \mathbf{W} & \mathbf{K}_{21}^\top \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} \mathbf{W} \\ \mathbf{K}_{21} \end{bmatrix}. \tag{1}$$

1.1 Standard Nyström method

The Nyström method uses $\mathbf{W}$ and $\mathbf{C}$ from (1) to approximate $\mathbf{K}$. Assuming a uniform sampling of the columns, the Nyström method generates a rank-k approximation $\widetilde{\mathbf{K}}$ of $\mathbf{K}$ for $k < n$ defined by:

$$\widetilde{\mathbf{K}}_k^{nys} = \mathbf{C}\mathbf{W}_k^+\mathbf{C}^\top \approx \mathbf{K}, \tag{2}$$

where $\mathbf{W}_k$ is the best k-rank approximation of $\mathbf{W}$ with respect to the spectral or Frobenius norm and $\mathbf{W}_k^+$ denotes the pseudo-inverse of $\mathbf{W}_k$. The Nyström method thus approximates the top k singular values (Σ_k) and singular vectors ($\mathbf{U}_k$) of $\mathbf{K}$ as:

$$\widetilde{\Sigma}_k^{nys} = \left(\frac{n}{l}\right)\Sigma_{W,k} \quad \text{and} \quad \widetilde{\mathbf{U}}_k^{nys} = \sqrt{\frac{l}{n}}\mathbf{C}\mathbf{U}_{W,k}\Sigma_{W,k}^+, \tag{3}$$

where $\Sigma_{W,k}$ contains the top k singular values of $\mathbf{W}$, and $\mathbf{U}_{W,k}$ contains the corresponding singular vectors. When $k = l$ (or more generally, whenever $k \geq \text{rank}(\mathbf{C})$),

this approximation perfectly reconstructs three blocks of $\mathbf{K}$, and $\mathbf{K}_{22}$ is approximated by the Schur Complement of $\mathbf{W}$ in $\mathbf{K}$:

$$\widetilde{\mathbf{K}}_l^{nys} = \mathbf{C}\mathbf{W}^+\mathbf{C}^\top = \begin{bmatrix} \mathbf{W} & \mathbf{K}_{21}^\top \\ \mathbf{K}_{21} & \mathbf{K}_{21}\mathbf{W}^+\mathbf{K}_{21} \end{bmatrix}. \tag{4}$$

The time complexity of SVD on $\mathbf{W}$ to get top k singular values and vectors is $O(kl^2)$ and matrix multiplication with $\mathbf{C}$ takes $O(kln)$. Hence, the total computational complexity of the Nyström approximation is $O(kln)$ since $n >> l$.

1.2 Ensemble Nyström

In this section, we discuss a meta algorithm called the Ensemble Nyström algorithm. We treat each approximation generated by the Nyström method for a sample of l columns as an *expert* and combine $p \geq 1$ such experts to derive an improved hypothesis, typically more accurate than any of the original experts.

The learning set-up is defined as follows. We assume a fixed kernel function $K\colon \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ that can be used to generate the entries of a kernel matrix $\mathbf{K}$. The learner receives a set S of lp columns randomly selected from matrix $\mathbf{K}$ uniformly without replacement. S is decomposed into p subsets $S_1, \ldots, S_p$. Each subset S_r, $r \in [1, p]$, contains l columns and is used to define a rank-k Nyström approximation $\widetilde{\mathbf{K}}_r$. Dropping the rank subscript k in favor of the sample index r, $\widetilde{\mathbf{K}}_r$ can be written as $\widetilde{\mathbf{K}}_r = \mathbf{C}_r\mathbf{W}_r^+\mathbf{C}_r^\top$, where $\mathbf{C}_r$ and $\mathbf{W}_r$ denote the matrices formed from the columns of S_r and $\mathbf{W}_r^+$ is the pseudo-inverse of the rank-k approximation of $\mathbf{W}_r$. The learner further receives a sample V of s columns used to determine the weight $\mu_r \in \mathbb{R}$ attributed to each expert $\widetilde{\mathbf{K}}_r$. Thus, the general form of the approximation, $\mathbf{K}^{ens}$, generated by the Ensemble Nyström algorithm, with $k \leq \operatorname{rank}(\mathbf{K}^{ens}) \leq pk$, is

$$\widetilde{\mathbf{K}}^{ens} = \sum_{r=1}^{p} \mu_r \widetilde{\mathbf{K}}_r \tag{5}$$

$$= \begin{bmatrix} \mathbf{C}_1 & & \\ & \ddots & \\ & & \mathbf{C}_p \end{bmatrix} \begin{bmatrix} \mu_1\mathbf{W}_1^+ & & \\ & \ddots & \\ & & \mu_p\mathbf{W}_p^+ \end{bmatrix} \begin{bmatrix} \mathbf{C}_1 & & \\ & \ddots & \\ & & \mathbf{C}_p \end{bmatrix}^\top. \tag{6}$$

As noted by [36], (6) provides an alternative description of the Ensemble Nyström method as a block diagonal approximation of $\mathbf{W}_{ens}^+$, where $\mathbf{W}_{ens}$ is the $lp \times lp$ SPSD matrix associated with the lp sampled columns.

The mixture weights μ_r can be defined in many ways. The most straightforward choice consists of assigning equal weight to each expert, $\mu_r = 1/p$, $r \in [1, p]$. This choice does not require the additional sample V, but it ignores the relative quality of each Nyström approximation. Nevertheless, this simple *uniform method* already

generates a solution superior to any one of the approximations $\widetilde{\mathbf{K}}_r$ used in the combination, as we shall see in the experimental section.

Another method, the *exponential weight method*, consists of measuring the reconstruction error $\hat{\varepsilon}_r$ of each expert $\widetilde{\mathbf{K}}_r$ over the validation sample V and defining the mixture weight as $\mu_r = \exp(-\eta\hat{\varepsilon}_r)/Z$, where $\eta > 0$ is a parameter of the algorithm and Z a normalization factor ensuring that the vector $\mu = (\mu_1, \ldots, \mu_p)$ belongs to the simplex Δ of $\mathbb{R}^p$: $\Delta = \{\mu \in \mathbb{R}^p : \mu \geq 0 \wedge \sum_{r=1}^{p} \mu_r = 1\}$. The choice of the mixture weights here is similar to that used in the Weighted Majority algorithm [38]. Let $\mathbf{K}_V$ denote the matrix formed by using the samples from V as its columns and let $\widetilde{\mathbf{K}}_r^V$ denote the submatrix of $\widetilde{\mathbf{K}}_r$ containing the columns corresponding to the columns in V. The reconstruction error $\hat{\varepsilon}_r = \|\widetilde{\mathbf{K}}_r^V - \mathbf{K}_V\|$ can be directly computed from these matrices.

A more general class of methods consists of using the sample V to train the mixture weights μ_r to optimize a regression objective function such as the following:

$$\min_{\mu} \lambda\|\mu\|_2^2 + \|\sum_{r=1}^{p} \mu_r \widetilde{\mathbf{K}}_r^V - \mathbf{K}_V\|_F^2, \tag{7}$$

where $\mathbf{K}_V$ denotes the matrix formed by the columns of the samples V and $\lambda > 0$. This can be viewed as a ridge regression objective function and admits a closed form solution. We will refer to this method as the *ridge regression method*. Note that to ensure that the resulting matrix is SPSD for use in subsequent kernel-based algorithms, the optimization problem must be augmented with standard non-negativity constraints. This is not necessary however for reducing the reconstruction error, as in our experiments. Also, clearly, a variety of other regression algorithms such as Lasso can be used here instead.

The total complexity of the Ensemble Nyström algorithm is $O(pl^3 + plkn + C_\mu)$, where C_μ is the cost of computing the mixture weights, μ, used to combine the p Nyström approximations. In general, the cubic term dominates the complexity since the mixture weights can be computed in constant time for the uniform method, in $O(psn)$ for the exponential weight method, or in $O(p^3 + p^2ns)$ for the ridge regression method where $O(p^2ns)$ time is required to compute a $p \times p$ matrix and $O(p^3)$ time to invert it. Furthermore, although the Ensemble Nyström algorithm requires p times more space and CPU cycles than the standard Nyström method, these additional requirements are quite reasonable in practice. The space requirement is still manageable for even large-scale applications given that p is typically O(1) and l is usually a very small percentage of n (see Section 3 for further details). In terms of CPU requirements, we note that this algorithm can be easily parallelized, as all p experts can be computed simultaneously. Thus, with a cluster of p machines, the running time complexity of this algorithm is nearly equal to that of the standard Nyström algorithm with l samples.

1.3 Ensemble Woodbury approximation

In many applications, one needs to invert a matrix $(\mathbf{K}+\lambda\mathbf{I})$, where λ is a positive scalar and $\mathbf{I}$ is the identity matrix. The Woodbury approximation is a useful tool to use alongside low-rank approximations to efficiently (and approximately) invert kernel matrices. We are able to apply the Woodbury approximation since the Nyström method represents $\widetilde{\mathbf{K}}$ as the product of low-rank matrices. This is clear from the definition of the Woodbury approximation:

$$(\mathbf{A}+\mathbf{BC}\mathrm{d})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1}+\mathrm{d}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathrm{d}\mathbf{A}^{-1}, \tag{8}$$

where $\mathbf{A}=\lambda\mathbf{I}$ and $\widetilde{\mathbf{K}}=\mathbf{BC}\mathrm{d}$ in the context of the Nyström method. In contrast, the Ensemble Nyström method represents $\widetilde{\mathbf{K}}$ as the sum of products of low-rank matrices, where each of the p terms corresponds to a base learner. Hence, we cannot directly apply the Woodbury approximation as presented above. There is however, a natural extension of the Woodbury approximation in this setting, which at the simplest level involves running the approximation p times. Starting with p base learners with their associated weights, i.e., $\widetilde{\mathbf{K}}_r$ and μ_r for $r\in[1,p]$, and defining $\mathbf{T}_0=\lambda\mathbf{I}$, we perform the following series of calculations:

$$\begin{aligned}
\mathbf{T}_1^{-1} &= (\mathbf{T}_0+\mu_1\widetilde{\mathbf{K}}_1)^{-1}\\
\mathbf{T}_2^{-1} &= (\mathbf{T}_1+\mu_2\widetilde{\mathbf{K}}_2)^{-1}\\
&\cdots\\
\mathbf{T}_p^{-1} &= (\mathbf{T}_{p-1}+\mu_p\widetilde{\mathbf{K}}_p)^{-1}.
\end{aligned}$$

To compute $\mathbf{T}_1^{-1}$, notice that we can use Woodbury approximation as stated in (8) since we can express $\mu_1\widetilde{\mathbf{K}}_1$ as the product of low-rank matrices and we know that $T_0^{-1}=\frac{1}{\lambda}\mathbf{I}$. More generally, for $1\le i\le p$, given an expression of T_{i-1}^{-1} as a product of low-rank matrices, we can efficiently compute T_i^{-1} using the Woodbury approximation (we use the low-rank structure to avoid ever computing or storing a full $n\times n$ matrix). Hence, after performing this series of p calculations, we are left with the inverse of $\mathbf{T}_p$, which is exactly the quantity of interest since $\mathbf{T}_p=\lambda\mathbf{I}+\sum_{r=1}^{p}\mu_r\widetilde{\mathbf{K}}_r$. Although this algorithm requires p iterations of the Woodbury approximation, these iterations can be parallelized in a tree-like fashion. Hence, when working on a cluster, using an Ensemble Nyström approximation along with the Woodbury approximation requires only $\log_2(p)$ more time than using the standard Nyström method.

2 Theoretical Analysis

We now present theoretical results that compare the quality of the Nyström approximation to the 'best' low-rank approximation, i.e., the approximation constructed from the top singular values and singular vectors of $\mathbf{K}$. This work, related to [18],

provides performance bounds for the Nyström method as used in practice, i.e., using uniform sampling without replacement. It holds for both the standard Nyström method as well as the Ensemble Nyström method discussed in Section 1.2.

Our theoretical analysis of the Nyström method uses some results previously shown by [18] as well as the following generalization of McDiarmid's concentration bound to sampling without replacement [13].

Theorem 1. *Let $Z_1,\ldots,Z_l$ be a sequence of random variables sampled uniformly without replacement from a fixed set of $l+u$ elements Z, and let $\phi\colon Z^l \to \mathbb{R}$ be a symmetric function such that for all $i\in[1,l]$ and for all $z_1,\ldots,z_l\in Z$ and $z'_1,\ldots,z'_l\in Z$, $|\phi(z_1,\ldots,z_l)-\phi(z_1,\ldots,z_{i-1},z'_i,z_{i+1},\ldots,z_l)|\leq c$. Then, for all $\varepsilon>0$, the following inequality holds:*

$$\Pr\left[\phi - \mathrm{e}[\phi] \geq \varepsilon\right] \leq \exp\left[\frac{-2\varepsilon^2}{\alpha(l,u)c^2}\right], \tag{9}$$

where $\alpha(l,u) = \frac{lu}{l+u-1/2}\,\frac{1}{1-1/(2\max\{l,u\})}$.

We define the *selection matrix* corresponding to a sample of l columns as the matrix $\mathbf{S}\in\mathbb{R}^{n\times l}$ defined by $\mathbf{S}_{ii}=1$ if the ith column of $\mathbf{K}$ is among those sampled, $\mathbf{S}_{ij}=0$ otherwise. Thus, $\mathbf{C}=\mathbf{KS}$ is the matrix formed by the columns sampled. Since $\mathbf{K}$ is SPSD, there exists $\mathbf{X}\in\mathbb{R}^{N\times n}$ such that $\mathbf{K}=\mathbf{X}^\top\mathbf{X}$. We shall denote by $\mathbf{K}_{\max}$ the maximum diagonal entry of $\mathbf{K}$, $\mathbf{K}_{\max}=\max_i \mathbf{K}_{ii}$, and by $d^{\mathbf{K}}_{\max}$ the distance $\max_{ij}\sqrt{\mathbf{K}_{ii}+\mathbf{K}_{jj}-2\mathbf{K}_{ij}}$.

2.1 Standard Nyström method

The following theorem gives an upper bound on the norm-2 error of the Nyström approximation of the form $\|\mathbf{K}-\widetilde{\mathbf{K}}\|_2/\|\mathbf{K}\|_2 \leq \|\mathbf{K}-\mathbf{K}_k\|_2/\|\mathbf{K}\|_2 + O(1/\sqrt{l})$ and an upper bound on the Frobenius error of the Nyström approximation of the form $\|\mathbf{K}-\widetilde{\mathbf{K}}\|_F/\|\mathbf{K}\|_F \leq \|\mathbf{K}-\mathbf{K}_k\|_F/\|\mathbf{K}\|_F + O(1/l^{\frac{1}{4}})$.

Theorem 2. *Let $\widetilde{\mathbf{K}}$ denote the rank-k Nyström approximation of $\mathbf{K}$ based on l columns sampled uniformly at random without replacement from $\mathbf{K}$, and $\mathbf{K}_k$ the best rank-k approximation of $\mathbf{K}$. Then, with probability at least $1-\delta$, the following inequalities hold for any sample of size l:*

$$\|\mathbf{K}-\widetilde{\mathbf{K}}\|_2 \leq \|\mathbf{K}-\mathbf{K}_k\|_2 + \tfrac{2n}{\sqrt{l}}\mathbf{K}_{\max}\left[1+\sqrt{\tfrac{n-l}{n-1/2}\,\tfrac{1}{\beta(l,n)}\log\tfrac{1}{\delta}}\;d^{\mathbf{K}}_{\max}/\mathbf{K}^{\frac{1}{2}}_{\max}\right]$$

$$\|\mathbf{K}-\widetilde{\mathbf{K}}\|_F \leq \|\mathbf{K}-\mathbf{K}_k\|_F + \left[\tfrac{64k}{l}\right]^{\frac{1}{4}} n\mathbf{K}_{\max}\left[1+\sqrt{\tfrac{n-l}{n-1/2}\,\tfrac{1}{\beta(l,n)}\log\tfrac{1}{\delta}}\;d^{\mathbf{K}}_{\max}/\mathbf{K}^{\frac{1}{2}}_{\max}\right]^{\frac{1}{2}},$$

where $\beta(l,n) = 1-\frac{1}{2\max\{l,n-l\}}$.

Proof. To bound the norm-2 error of the Nyström method in the scenario of sampling without replacement, we start with the following general inequality given

by [18][proof of Lemma 4]:

$$\|\mathbf{K}-\widetilde{\mathbf{K}}\|_2 \leq \|\mathbf{K}-\mathbf{K}_k\|_2 + 2\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_2, \tag{10}$$

where $\mathbf{Z} = \sqrt{\frac{n}{l}}\mathbf{X}\mathbf{S}$. We then apply the McDiarmid-type inequality of Theorem 1 to $\phi(\mathbf{S}) = \|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_2$. Let $\mathbf{S}'$ be a sampling matrix selecting the same columns as $\mathbf{S}$ except for one, and let $\mathbf{Z}'$ denote $\sqrt{\frac{n}{l}}\mathbf{X}\mathbf{S}'$. Let $\mathbf{z}$ and $\mathbf{z}'$ denote the only differing columns of $\mathbf{Z}$ and $\mathbf{Z}'$, then

$$|\phi(\mathbf{S}') - \phi(\mathbf{S})| \leq \|\mathbf{z}'\mathbf{z}'^\top - \mathbf{z}\mathbf{z}^\top\|_2 = \|(\mathbf{z}'-\mathbf{z})\mathbf{z}'^\top + \mathbf{z}(\mathbf{z}'-\mathbf{z})^\top\|_2 \tag{11}$$

$$\leq 2\|\mathbf{z}'-\mathbf{z}\|_2 \max\{\|\mathbf{z}\|_2, \|\mathbf{z}'\|_2\}. \tag{12}$$

Columns of $\mathbf{Z}$ are those of $\mathbf{X}$ scaled by $\sqrt{n/l}$. The norm of the difference of two columns of $\mathbf{X}$ can be viewed as the norm of the difference of two feature vectors associated to $\mathbf{K}$ and thus can be bounded by $d_{\mathbf{K}}$. Similarly, the norm of a single column of $\mathbf{X}$ is bounded by $\mathbf{K}_{\max}^{\frac{1}{2}}$. This leads to the following inequality:

$$|\phi(\mathbf{S}') - \phi(\mathbf{S})| \leq \frac{2n}{l} d_{\max}^{\mathbf{K}} \mathbf{K}_{\max}^{\frac{1}{2}}. \tag{13}$$

The expectation of ϕ can be bounded as follows:

$$\mathrm{e}[\Phi] = \mathrm{e}[\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_2] \leq \mathrm{e}[\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_F] \leq \frac{n}{\sqrt{l}}\mathbf{K}_{\max}, \tag{14}$$

where the last inequality follows Corollary 2 of [34]. The inequalities (13) and (14) combined with Theorem 1 give a bound on $\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_2$ and yield the statement of the theorem.

The following general inequality holds for the Frobenius error of the Nyström method [18]:

$$\|\mathbf{K}-\widetilde{\mathbf{K}}\|_F^2 \leq \|\mathbf{K}-\mathbf{K}_k\|_F^2 + \sqrt{64k}\,\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_F^2\, n\mathbf{K}_{ii}^{\max}. \tag{15}$$

Bounding the term $\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}\mathbf{Z}^\top\|_F^2$ as in the norm-2 case and using the concentration bound of Theorem 1 yields the result of the theorem.

2.2 Ensemble Nyström method

The following error bounds hold for Ensemble Nyström methods based on a convex combination of Nyström approximations.

Theorem 3. *Let S be a sample of pl columns drawn uniformly at random without replacement from $\mathbf{K}$, decomposed into p subsamples of size l, $S_1, \ldots, S_p$. For $r \in [1, p]$, let $\widetilde{\mathbf{K}}_r$ denote the rank-k Nyström approximation of $\mathbf{K}$ based on the sample S_r, and let $\mathbf{K}_k$ denote the best rank-k approximation of $\mathbf{K}$. Then, with probability at*

least $1-\delta$*, the following inequalities hold for any sample* S *of size* pl *and for any* μ *in the simplex* Δ *and* $\widetilde{\mathbf{K}}^{ens} = \sum_{r=1}^{p} \mu_r \widetilde{\mathbf{K}}_r$*:*

$$\|\mathbf{K} - \widetilde{\mathbf{K}}^{ens}\|_2 \leq \|\mathbf{K} - \mathbf{K}_k\|_2 + \frac{2n}{\sqrt{l}} \mathbf{K}_{\max} \left[1 + \mu_{\max} p^{\frac{1}{2}} \sqrt{\frac{n-pl}{n-1/2} \frac{1}{\beta(pl,n)} \log \frac{1}{\delta}}\, d^{\mathbf{K}}_{\max} / \mathbf{K}^{\frac{1}{2}}_{\max}\right]$$

$$\|\mathbf{K} - \widetilde{\mathbf{K}}^{ens}\|_F \leq \|\mathbf{K} - \mathbf{K}_k\|_F + \left[\frac{64k}{l}\right]^{\frac{1}{4}} n \mathbf{K}_{\max} \left[1 + \mu_{\max} p^{\frac{1}{2}} \sqrt{\frac{n-pl}{n-1/2} \frac{1}{\beta(pl,n)} \log \frac{1}{\delta}}\, d^{\mathbf{K}}_{\max} / \mathbf{K}^{\frac{1}{2}}_{\max}\right]^{\frac{1}{2}},$$

where $\beta(pl,n) = 1 - \frac{1}{2\max\{pl, n-pl\}}$ *and* $\mu_{\max} = \max_{r=1}^{p} \mu_r$*.*

Proof. For $r \in [1,p]$, let $\mathbf{Z}_r = \sqrt{n/l}\,\mathbf{X}\mathbf{S}_r$, where $\mathbf{S}_r$ denotes the selection matrix corresponding to the sample S_r. By definition of $\widetilde{\mathbf{K}}^{ens}$ and the upper bound on $\|\mathbf{K} - \widetilde{\mathbf{K}}_r\|_2$ already used in the proof of theorem 2, the following holds:

$$\|\mathbf{K} - \widetilde{\mathbf{K}}^{ens}\|_2 = \Big\| \sum_{r=1}^{p} \mu_r (\mathbf{K} - \widetilde{\mathbf{K}}_r) \Big\|_2 \leq \sum_{r=1}^{p} \mu_r \|\mathbf{K} - \widetilde{\mathbf{K}}_r\|_2 \tag{16}$$

$$\leq \sum_{r=1}^{p} \mu_r \big(\|\mathbf{K} - \mathbf{K}_k\|_2 + 2\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2\big) \tag{17}$$

$$= \|\mathbf{K} - \mathbf{K}_k\|_2 + 2\sum_{r=1}^{p} \mu_r \|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2. \tag{18}$$

We apply Theorem 1 to $\phi(S) = \sum_{r=1}^{p} \mu_r \|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2$. Let S' be a sample differing from S by only one column. Observe that changing one column of the full sample S changes only one subsample S_r and thus only one term $\mu_r \|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2$. Thus, in view of the bound (13) on the change to $\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2$, the following holds:

$$|\phi(S') - \phi(S)| \leq \frac{2n}{l} \mu_{\max} d^{\mathbf{K}}_{\max} \mathbf{K}^{\frac{1}{2}}_{\max}, \tag{19}$$

The expectation of Φ can be straightforwardly bounded by:

$$\mathrm{e}[\Phi(S)] = \sum_{r=1}^{p} \mu_r \mathrm{e}[\|\mathbf{X}\mathbf{X}^\top - \mathbf{Z}_r\mathbf{Z}_r^\top\|_2] \leq \sum_{r=1}^{p} \mu_r \frac{n}{\sqrt{l}} \mathbf{K}_{\max} = \frac{n}{\sqrt{l}} \mathbf{K}_{\max}$$

using the bound (14) for a single expert. Plugging in this upper bound and the Lipschitz bound (19) in Theorem 1 yields our norm-2 bound for the Ensemble Nyström method.

For the Frobenius error bound, using the convexity of the Frobenius norm square $\|\cdot\|_F^2$ and the general inequality (15), we can write

Dataset	Type of data	# Points (n)	# Features (d)	Kernel
PIE-2.7K	face images	2731	2304	linear
MNIST	digit images	4000	784	linear
ESS	proteins	4728	16	RBF
AB-S	abalones	4177	8	RBF
DEXT	bag of words	2000	20000	linear
SIFT-1M	Image features	1M	128	RBF

Table 1 Description of the datasets used in our Ensemble Nyström experiments [3, 27, 35, 39, 48].

$$\|\mathbf{K}-\widetilde{\mathbf{K}}^{ens}\|_F^2 = \Big\|\sum_{r=1}^{p}\mu_r(\mathbf{K}-\widetilde{\mathbf{K}}_r)\Big\|_F^2 \leq \sum_{r=1}^{p}\mu_r\|\mathbf{K}-\widetilde{\mathbf{K}}_r\|_F^2 \tag{20}$$

$$\leq \sum_{r=1}^{p}\mu_r\Big[\|\mathbf{K}-\mathbf{K}_k\|_F^2+\sqrt{64k}\,\|\mathbf{X}\mathbf{X}^\top-\mathbf{Z}_r\mathbf{Z}_r^\top\|_F\, n\mathbf{K}_{ii}^{\max}\Big]. \tag{21}$$

$$= \|\mathbf{K}-\mathbf{K}_k\|_F^2+\sqrt{64k}\sum_{r=1}^{p}\mu_r\|\mathbf{X}\mathbf{X}^\top-\mathbf{Z}_r\mathbf{Z}_r^\top\|_F\, n\mathbf{K}_{ii}^{\max}. \tag{22}$$

The result follows by the application of Theorem 1 to $\psi(S)=\sum_{r=1}^{p}\mu_r\|\mathbf{X}\mathbf{X}^\top-\mathbf{Z}_r\mathbf{Z}_r^\top\|_F$ in a way similar to the norm-2 case.

The bounds of Theorem 3 are similar in form to those of Theorem 2. However, the bounds for the Ensemble Nyström are tighter than those for any Nyström expert based on a single sample of size l even for a uniform weighting. In particular, for $\mu_i=1/p$ for all i, the last term of the ensemble bound for norm-2 is smaller by a factor larger than $\mu_{\max}p^{\frac{1}{2}}=1/\sqrt{p}$.

3 Experiments

In this section, we present experimental results that illustrate the performance of the Ensemble Nyström method. We work with the data sets listed in Table 1, and compare the performance of various methods for calculating the mixture weights (μ_r). Throughout our experiments, we measure the accuracy of a low-rank approximation $\widetilde{\mathbf{K}}$ by calculating the relative error in Frobenius and spectral norms, that is, if we let $\xi=\{2,F\}$, then we calculate the following quantity:

$$\%\,\text{error} = \frac{\|\mathbf{K}-\widetilde{\mathbf{K}}\|_\xi}{\|\mathbf{K}\|_\xi}\times 100. \tag{23}$$

3.1 Ensemble Nyström with various mixture weights

In this set of experiments, we show results for our Ensemble Nyström method using different techniques to choose the mixture weights as previously discussed. We first experimented with the first five datasets shown in Table 1. For each dataset, we fixed the reduced rank to $k = 50$, and set the number of sampled columns to $l = 3\% \times n$.[1] Furthermore, for the exponential and the ridge regression variants, we sampled a set of $s = 20$ columns and used an additional 20 columns (s') as a hold-out set for selecting the optimal values of η and λ. The number of approximations, p, was varied from 2 to 30. As a baseline, we also measured the minimum and the mean percent error across the p Nyström approximations used to construct $\widetilde{\mathbf{K}}^{ens}$. For the Frobenius norm, we also calculated the performance when using the optimal μ, that is, we used least-square regression to find the best possible choice of combination weights for a fixed set of p approximations by setting $s = n$.

The results of these experiments are presented in Figure 1 for the Frobenius norm and in Figure 2 for the spectral norm. These results clearly show that the Ensemble Nyström performance is significantly better than any of the individual Nyström approximations. As mentioned earlier, the rank of the ensemble approximations can be p times greater than the rank of each of the base learners. Hence, to validate the results in Figures 1 and 2, we performed a simple experiment in which we compared the performance of the best base learner to the best rank k approximation of the uniform ensemble approximation (obtained via SVD of the uniform ensemble approximation). The results of this experiment, presented in Figure 3, suggest that the performance gain of the ensemble methods is not due to this increased rank.

Furthermore, the ridge regression technique is the best of the proposed techniques and generates nearly the optimal solution in terms of the percent error in Frobenius norm. We also observed that when s is increased to approximately 5% to 10% of n, linear regression without any regularization performs about as well as ridge regression for both the Frobenius and spectral norm. Figure 4 shows this comparison between linear regression and ridge regression for varying values of s using a fixed number of experts ($p = 10$). Finally we note that the Ensemble Nyström method tends to converge very quickly, and the most significant gain in performance occurs as p increases from 2 to 10.

3.2 Large-scale experiments

We now present an empirical study of the effectiveness of the Ensemble Nyström method on the SIFT-1M dataset in Table 1 containing 1 *million* data points. As is common practice with large-scale datasets, we worked on a cluster of several machines for this dataset. We present results comparing the performance of the Ensemble Nyström method, using both uniform and ridge regression mixture weights, with

[1] Similar results (not reported here) were observed for other values of k and l as well.

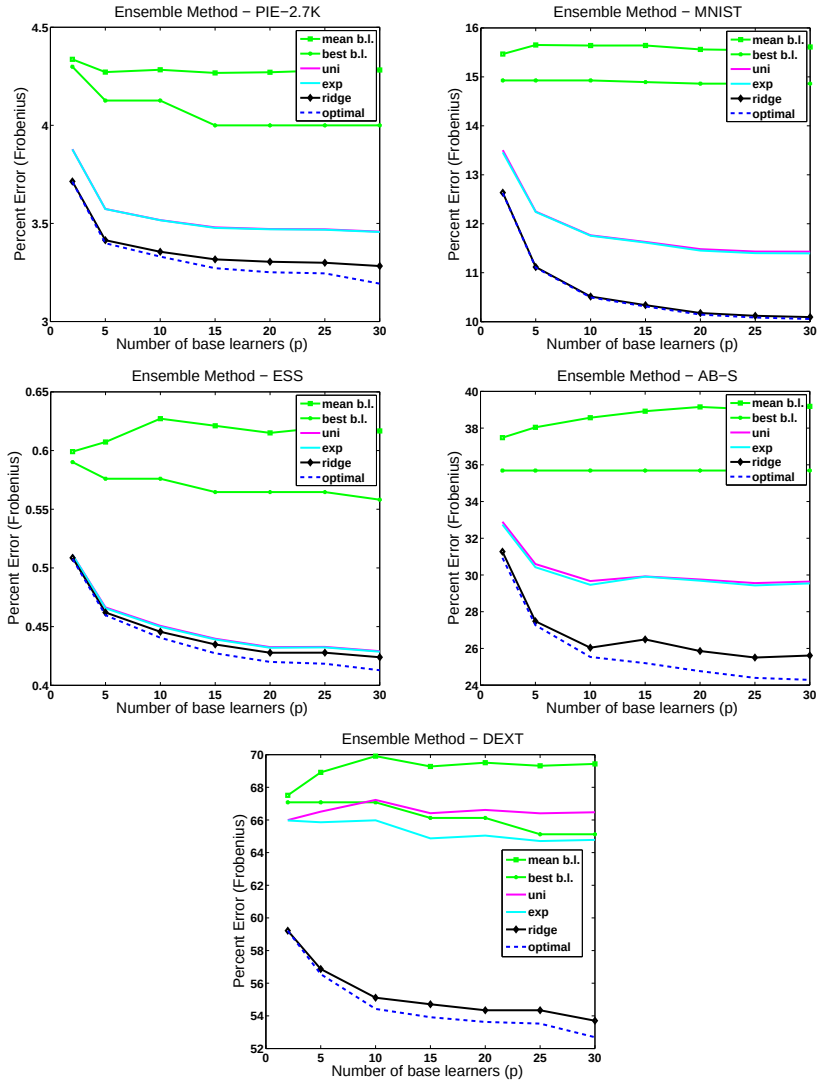

Fig. 1 Percent error in Frobenius norm for Ensemble Nyström method using uniform ('uni'), exponential ('exp'), ridge ('ridge') and optimal ('optimal') mixture weights as well as the best ('best b.l.') and mean ('mean b.l.') of the p base learners used to create the ensemble approximations.

that of the best and mean performance across the p Nyström approximations used to construct $\widetilde{\mathbf{K}}^{ens}$. We also make comparisons with the K-means adaptive sampling technique [54, 55]. Although the K-means technique is quite effective at generating informative columns by exploiting the data distribution, the cost of performing K-means becomes expensive for even moderately sized datasets, making it difficult to use in large-scale settings. Nevertheless, in this work, we include the K-means

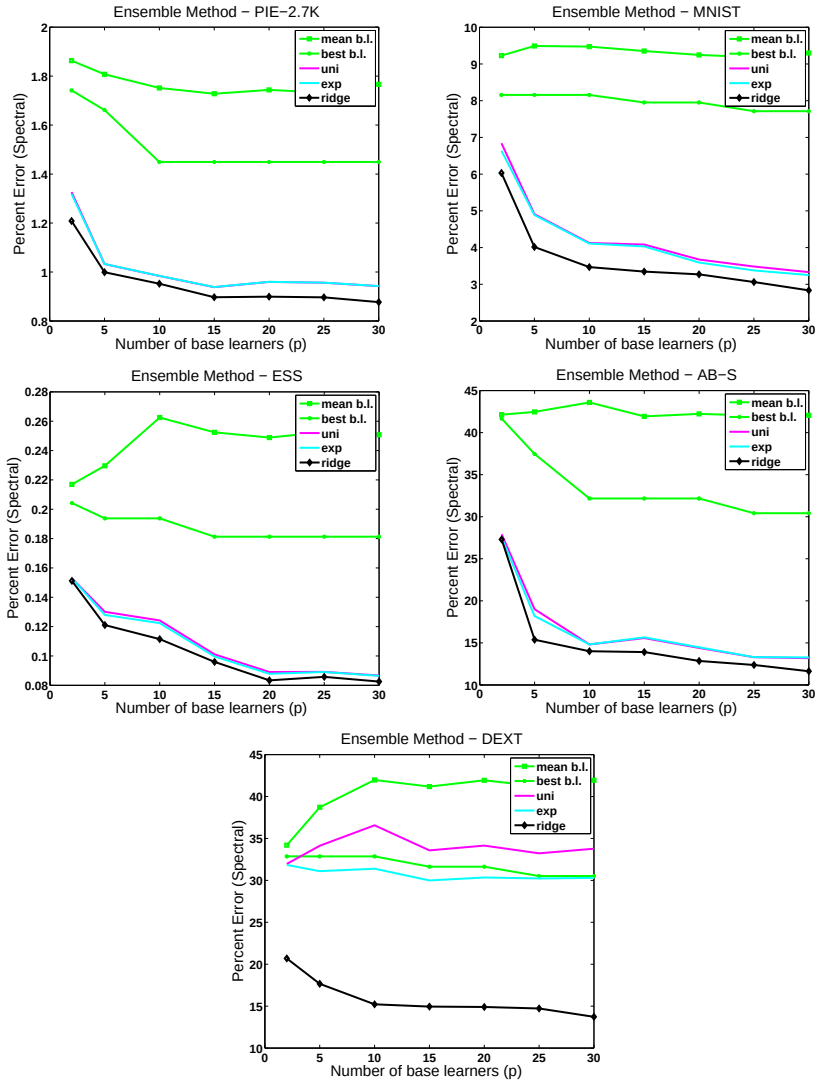

Fig. 2 Percent error in spectral norm for Ensemble Nyström method using various mixture weights and the best/mean of the p approximations. Legend entries are the same as in Figure 1.

method in our comparison, and present results for various subsamples of the SIFT-1M dataset, with n ranging from 5K to 1M.

For a fair comparison, we performed 'fixed-time' experiments. We first searched for an appropriate l such that the percent error for the Ensemble Nyström method with ridge weights was approximately 10%, and measured the time required by the cluster to construct this approximation. We then allotted an equal amount of time (within 1 second) for the other techniques, and measured the quality of the resulting

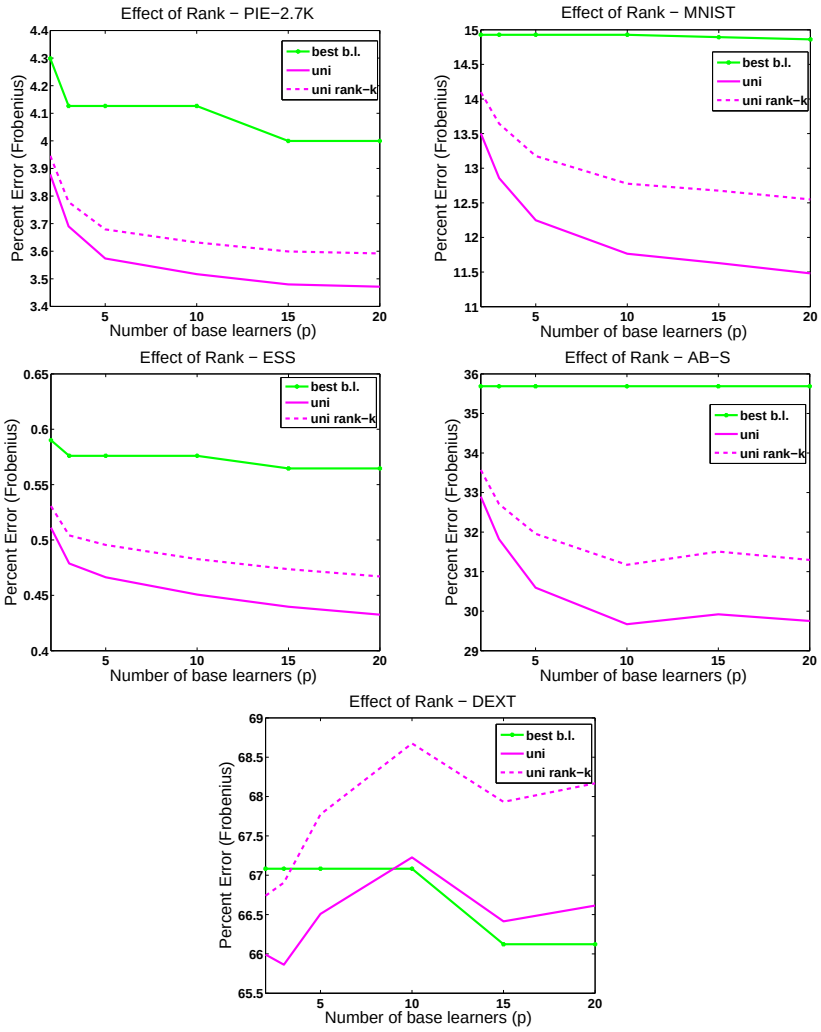

Fig. 3 Percent error in Frobenius norm for Ensemble Nyström method using uniform ('uni') mixture weights, the optimal rank-k approximation of the uniform ensemble result ('uni rank-k') as well as the best ('best b.l.') of the p base learners used to create the ensemble approximations.

approximations. For these experiments, we set $k=50$ and $p=10$, based on the results from the previous section. Furthermore, in order to speed up computation on this large dataset, we decreased the size of the validation and hold-out sets to $s=2$ and $s'=2$, respectively.

The results of this experiment, presented in Figure 5, clearly show that the Ensemble Nyström method is the most effective technique given a fixed amount of time. Furthermore, even with the small values of s and s', Ensemble Nyström with ridge-regression weighting outperforms the uniform Ensemble Nyström method.

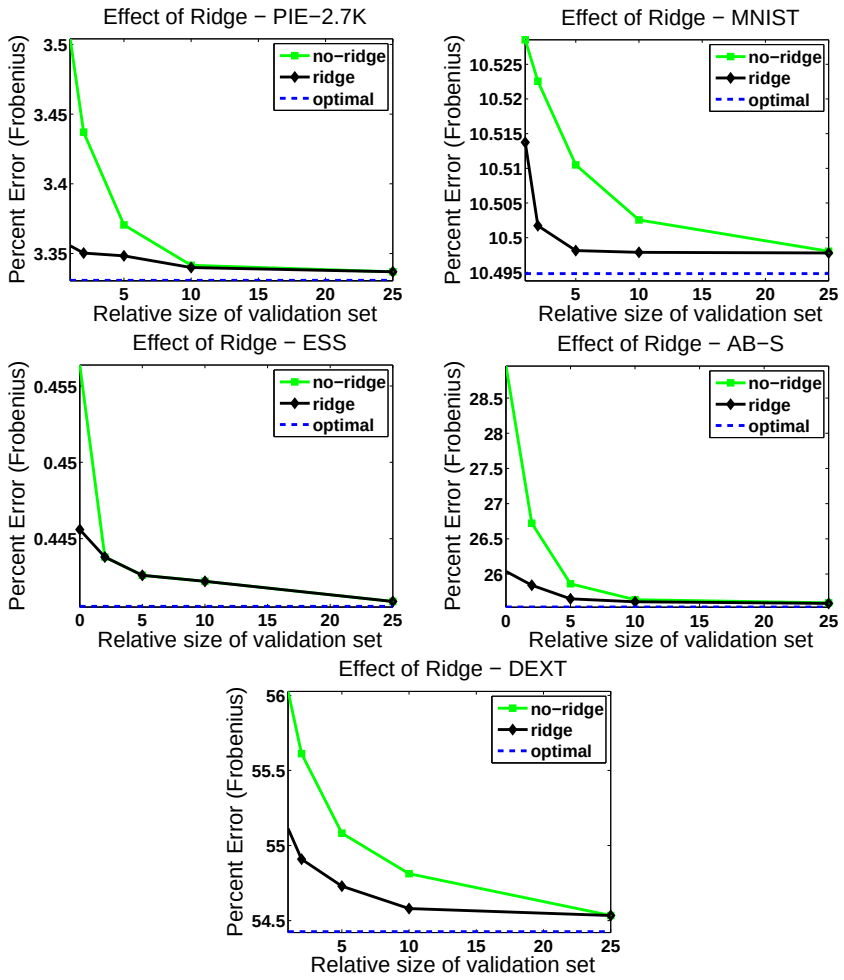

Fig. 4 Comparison of percent error in Frobenius norm for the Ensemble Nyström method with $p=10$ experts with weights derived from linear ('no-ridge') and ridge ('ridge') regression. The dotted line indicates the optimal combination. The relative size of the validation set equals $s/n\times 100$.

We also observe that due to the high computational cost of K-means for large datasets, the K-means approximation does not perform well in this 'fixed-time' experiment. It generates an approximation that is worse than the mean standard Nyström approximation and its performance increasingly deteriorates as n approaches 1M. Finally, we note that although the space requirements are 10 times greater for Ensemble Nyström in comparison to standard Nyström (since $p=10$ in this experiment), the space constraints are nonetheless quite reasonable. For instance, when working with 1M points, the Ensemble Nyström method with ridge

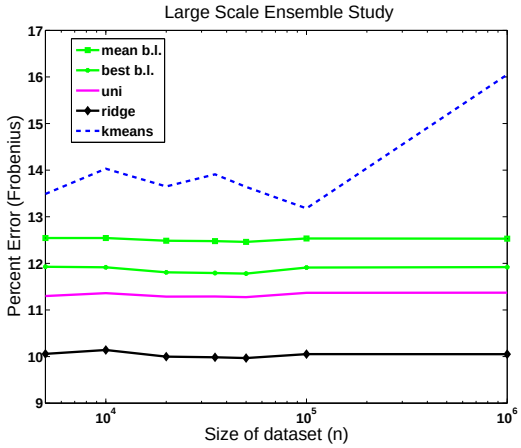

Fig. 5 Large-scale performance comparison with SIFT-1M dataset. For a fixed computational time, the Ensemble Nyström approximation with ridge weights tends to outperform other techniques.

regression weights only required approximately 1% of the columns of **K** to achieve an error of 10%.

4 Summary and Open Questions

A key element of Nyström approximation is the number of sampled columns used by it. More samples typically result in better accuracy. However, the number of samples that can be processed by a single Nyström approximation is limited due to the computational constraints, restricting its accuracy. In this work, we discussed an ensemble based meta-algorithm for combining multiple Nyström approximations. These ensemble algorithms show consistent and significant performance improvement across a number of different data sets. Moreover, they naturally fit within a distributed computing environment, thus making them quite efficient in large-scale settings. These ensemble algorithms also have better theoretical guarantees than individual Nyström approximation.

One interesting fact revealed by the experiments is that as the number of individual Nyström approximations is increased in the ensemble, the reconstruction error does not go towards zero. The error tends to saturate after a relatively small number of learners and adding more does not benefit the ensemble. Even though this counter-intuitive behavior is a good thing in practice since one does not need to use a large number of base learners, it raises intriguing theoretical questions. Why does the error from Ensemble Nyström converge? What is the value to which it is converging? Can this error be brought arbitrarily close to zero? We believe that a

better understanding of these questions may lead to even better ways of designing ensemble algorithms for matrix approximation in the future.

5 Bibliographical and Historical Remarks

There has been a wide array of work on low-rank matrix approximation within the numerical linear algebra and computer science communities. Most of it has been inspired by the celebrated result of Johnson and Lindenstrauss [31], which showed that random low-dimensional embeddings preserve Euclidean geometry. This result has led to a family of random projection algorithms, which involves projecting the original matrix onto a random low-dimensional subspace [30, 37, 42]. Alternatively, SVD can be used to generate 'optimal' low-rank matrix approximations, as mentioned earlier. However, both the random projection and the SVD algorithms involve storage and operating on the entire input matrix. SVD is more computationally expensive than random projection methods, though neither are linear in n in terms of time and space complexity. When dealing with sparse matrices, there exist less computationally intensive techniques such as Jacobi, Arnoldi, Hebbian and more recent randomized methods [23, 25, 28, 44] for generating low-rank approximations. These iterative methods require computation of matrix-vector products at each step and involve multiple passes through the data. Hence, these algorithms are not suitable for large, dense matrices. Matrix sparsification algorithms [1, 2], as the name suggests, attempt to sparsify dense matrices to speed up future storage and computational burdens, though they too require storage of the input matrix and exhibit superlinear processing time.

Alternatively, sampling-based approaches can be used to generate low-rank approximations. Research in this area dates back to classical theoretical results that show, for any arbitrary matrix, the existence of a subset of k columns for which the error in matrix projection (as defined in [33]) can be bounded relative to the optimal rank-k approximation of the matrix [46]. Deterministic algorithms such as rank-revealing QR [26] can achieve nearly optimal matrix projection errors. More recently, research in the theoretical computer science community has been aimed at deriving bounds on matrix projection error using sampling-based approximations, including additive error bounds using sampling distributions based on leverage scores, i.e., the squared L_2 norms of the columns [17, 22, 45]; relative error bounds using adaptive sampling techniques [16, 29]; and, relative error bounds based on distributions derived from the singular vectors of the input matrix, in work related to the column-subset selection problem [10, 19]. However, as discussed in [33], the task of matrix projection involves projecting the input matrix onto a low-rank subspace, which requires superlinear time and space with respect to n and is not typically feasible for large-scale matrices.

There does however, exist another class of sampling-based approximation algorithms that only store and operate on a subset of the original matrix. For arbitrary rectangular matrices, these algorithms are known as 'CUR' approximations

(the name 'CUR' corresponds to the three low-rank matrices whose product is an approximation to the original matrix). The theoretical performance of CUR approximations has been analyzed using a variety of sampling schemes, although the column-selection processes associated with these analyses often require operating on the entire input matrix [19, 24, 40, 50]. In the context of symmetric positive semidefinite matrices, the Nyström method is the most commonly used algorithm to efficiently generate low-rank approximations. The Nyström method was initially introduced as a quadrature method for numerical integration, used to approximate eigenfunction solutions [6, 41]. More recently, it was presented in [53] to speed up kernel algorithms and has been studied theoretically using a variety of sampling schemes [7, 8, 14, 18, 32–34, 49, 52, 54, 55]. It has also been used for a variety of machine learning tasks ranging from manifold learning to image segmentation [21, 43, 51]. A closely related algorithm, known as the Incomplete Cholesky Decomposition [4,5,20], can also be viewed as a specific sampling technique associated with the Nyström method [5]. As noted by [11,52], the Nyström approximation is related to the problem of matrix completion [11, 12], which attempts to complete a low-rank matrix from a random sample of its entries. However, the matrix completion setting assumes that the target matrix is low-rank and only allows for limited access to the data. In contrast, the Nyström method, and sampling-based low-rank approximation algorithms in general, deal with full-rank matrices that are amenable to low-rank approximation. Furthermore, when we have access to the underlying kernel function that generates the kernel matrix of interest, we can generate matrix entries on-the-fly as desired, providing us with more flexibility accessing the original matrix.

References

1. Dimitris Achlioptas and Frank Mcsherry. Fast computation of low-rank matrix approximations. *Journal of the ACM*, 54(2), 2007.
2. Sanjeev Arora, Elad Hazan, and Satyen Kale. A fast random sampling algorithm for sparsifying matrices. In *Approx-Random*, 2006.
3. A. Asuncion and D.J. Newman. UCI machine learning repository. http://www.ics.uci.edu/ mlearn/MLRepository.html, 2007.
4. Francis R. Bach and Michael I. Jordan. Kernel Independent Component Analysis. *Journal of Machine Learning Research*, 3:1–48, 2002.
5. Francis R. Bach and Michael I. Jordan. Predictive low-rank decomposition for kernel methods. In *International Conference on Machine Learning*, 2005.
6. Christopher T. Baker. *The numerical treatment of integral equations*. Clarendon Press, Oxford, 1977.
7. M.-A. Belabbas and P. J. Wolfe. On landmark selection and sampling in high-dimensional data analysis. arXiv:0906.4582v1 [stat.ML], 2009.
8. M. A. Belabbas and P. J. Wolfe. Spectral methods in machine learning and new strategies for very large datasets. *Proceedings of the National Academy of Sciences of the United States of America*, 106(2):369–374, January 2009.
9. Bernhard E. Boser, Isabelle Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Conference on Learning Theory*, 1992.

10. Christos Boutsidis, Michael W. Mahoney, and Petros Drineas. An improved approximation algorithm for the column subset selection problem. In *Symposium on Discrete Algorithms*, 2009.
11. Emmanuel J. Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.
12. Emmanuel J. Candès and Terence Tao. The power of convex relaxation: Near-optimal matrix completion. arXiv:0903.1476v1 [cs.IT], 2009.
13. Corinna Cortes, Mehryar Mohri, Dmitry Pechyony, and Ashish Rastogi. Stability of transductive regression algorithms. In *International Conference on Machine Learning*, 2008.
14. Corinna Cortes, Mehryar Mohri, and Ameet Talwalkar. On the impact of kernel approximation on learning accuracy. In *Conference on Artificial Intelligence and Statistics*, 2010.
15. Corinna Cortes and Vladimir N. Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.
16. Amit Deshpande, Luis Rademacher, Santosh Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. In *Symposium on Discrete Algorithms*, 2006.
17. Petros Drineas, Ravi Kannan, and Michael W. Mahoney. Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix. *SIAM Journal of Computing*, 36(1), 2006.
18. Petros Drineas and Michael W. Mahoney. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *Journal of Machine Learning Research*, 6:2153–2175, 2005.
19. Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan. Relative-error CUR matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, 2008.
20. Shai Fine and Katya Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2002.
21. Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the Nyström method. *Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.
22. Alan Frieze, Ravi Kannan, and Santosh Vempala. Fast Monte-Carlo algorithms for finding low-rank approximations. In *Foundation of Computer Science*, 1998.
23. Gene Golub and Charles Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 2nd edition, 1983.
24. S. A. Goreinov, E. E. Tyrtyshnikov, and N. L. Zamarashkin. A theory of pseudoskeleton approximations. *Linear Algebra and Its Applications*, 261:1–21, 1997.
25. G. Gorrell. Generalized Hebbian algorithm for incremental Singular Value Decomposition in natural language processing. In *European Chapter of the Association for Computational Linguistics*, 2006.
26. Ming Gu and Stanley C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal of Scientific Computing*, 17(4):848–869, 1996.
27. A. Gustafson, E. Snitkin, S. Parker, C. DeLisi, and S. Kasif. Towards the identification of essential genes using targeted genome sequencing and comparative analysis. *BMC:Genomics*, 7:265, 2006.
28. Nathan Halko, Per Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions. arXiv:0909.4061v1 [math.NA], 2009.
29. Sariel Har-peled. Low-rank matrix approximation in linear time, manuscript, 2006.
30. Piotr Indyk. Stable distributions, pseudorandom generators, embeddings, and data stream computation. *Journal of the ACM*, 53(3):307–323, 2006.
31. W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.
32. Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. Ensemble Nyström method. In *Neural Information Processing Systems*, 2009.
33. Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. On sampling-based approximate spectral decomposition. In *International Conference on Machine Learning*, 2009.

34. Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. Sampling techniques for the Nyström method. In *Conference on Artificial Intelligence and Statistics*, 2009.
35. Yann LeCun and Corinna Cortes. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/, 1998.
36. Mu Li, James T. Kwok, and Bao-Liang Lu. Making large-scale Nyström approximation possible. In *International Conference on Machine Learning*, 2010.
37. Edo Liberty. *Accelerated dense random projections*. Ph.D. thesis, computer science department, Yale University, New Haven, CT, 2009.
38. N. Littlestone and M. K. Warmuth. The Weighted Majority algorithm. *Information and Computation*, 108(2):212–261, 1994.
39. David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.
40. Michael W Mahoney and Petros Drineas. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.
41. E.J. Nyström. Über die praktische auflösung von linearen integralgleichungen mit anwendungen auf randwertaufgaben der potentialtheorie. *Commentationes Physico-Mathematicae*, 4(15):1–52, 1928.
42. Christos H. Papadimitriou, Hisao Tamaki, Prabhakar Raghavan, and Santosh Vempala. Latent Semantic Indexing: a probabilistic analysis. In *Principles of Database Systems*, 1998.
43. John C. Platt. Fast embedding of sparse similarity graphs. In *Neural Information Processing Systems*, 2004.
44. Vladimir Rokhlin, Arthur Szlam, and Mark Tygert. A randomized algorithm for Principal Component Analysis. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1100–1124, 2009.
45. Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM*, 54(4):21, 2007.
46. A. F. Ruston. Auerbachs theorem. *Mathematical Proceedings of the Cambridge Philosophical Society*, 56:476–480, 1964.
47. Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.
48. Terence Sim, Simon Baker, and Maan Bsat. The CMU pose, illumination, and expression database. In *Conference on Automatic Face and Gesture Recognition*, 2002.
49. Alex J. Smola and Bernhard Schölkopf. Sparse Greedy Matrix Approximation for machine learning. In *International Conference on Machine Learning*, 2000.
50. G. W. Stewart. Four algorithms for the efficient computation of truncated pivoted QR approximations to a sparse matrix. *Numerische Mathematik*, 83(2):313–323, 1999.
51. Ameet Talwalkar, Sanjiv Kumar, and Henry Rowley. Large-scale manifold learning. In *Conference on Vision and Pattern Recognition*, 2008.
52. Ameet Talwalkar and Afshin Rostamizadeh. Matrix coherence and the Nyström method. In *Conference on Uncertainty in Artificial Intelligence*, 2010.
53. Christopher K. I. Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Neural Information Processing Systems*, 2000.
54. Kai Zhang and James T. Kwok. Density-weighted Nyström method for computing large kernel eigensystems. *Neural Computation*, 21(1):121–146, 2009.
55. Kai Zhang, Ivor Tsang, and James Kwok. Improved Nyström low-rank approximation and error analysis. In *International Conference on Machine Learning*, 2008.