A novel greedy algorithm for Nyström approximation

Ahmed K. Farahat **Ali Ghodsi** **Mohamed S. Kamel**
University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1

Abstract

The Nyström method is an efficient technique for obtaining a low-rank approximation of a large kernel matrix based on a subset of its columns. The quality of the Nyström approximation highly depends on the subset of columns used, which are usually selected using random sampling. This paper presents a novel recursive algorithm for calculating the Nyström approximation, and an effective greedy criterion for column selection. Further, a very efficient variant is proposed for greedy sampling, which works on random partitions of data instances. Experiments on benchmark data sets show that the proposed greedy algorithms achieve significant improvements in approximating kernel matrices, with minimum overhead in run time.

1 INTRODUCTION

The Nyström method (Williams and Seeger, 2001) is an efficient technique for obtaining a low-rank approximation of a large kernel matrix, using only a subset of its columns. It can also be used to efficiently approximate the singular values and vectors of a kernel matrix (Williams and Seeger, 2001; Kumar et al., 2009c). The Nyström method has been successfully used in many large-scale applications including efficient learning of kernel-based models such as Gaussian processes (Williams and Seeger, 2001) and support vector machines (Cortes et al., 2010), fast multidimensional scaling (Platt, 2005), approximate spectral clustering (Fowlkes et al., 2004), and large-scale manifold learning (Talwalkar et al., 2008).

The quality of the Nyström approximation highly depends on the subset of selected columns. Although uniform sampling has been the most common technique for column selection (Williams and Seeger, 2001), a considerable amount of research work has been conducted to theoretically and empirically study other sampling techniques. These techniques include: non-uniform sampling, using probabilities calculated based on the kernel matrix (Drineas and Mahoney, 2005; Drineas et al., 2007; Kumar et al., 2009b); adaptive sampling, in which probabilities are updated based on intermediate approximations of the kernel matrix (Deshpande et al., 2006; Kumar et al., 2009c); and deterministic sampling, where columns are selected such that some criterion function is optimized (Smola and Schölkopf, 2000; Zhang et al., 2008).

In this paper, a greedy algorithm is proposed for simultaneously calculating the Nyström approximation and selecting representative columns. The proposed algorithm is based on a novel recursive formula for the Nyström approximation which allows a greedy selection criterion to be calculated efficiently at each iteration. First, a recursive formula is derived, in which a rank-l Nyström approximation of a kernel matrix is constructed by calculating a rank-1 Nyström approximation based on one column, and then calculating the rank-$(l-1)$ Nyström approximation of the residual matrix. Next, a novel criterion is developed for selecting a representative column at each iteration of the recursive algorithm. The selected column corresponds to the direction which best represents other data points in the high-dimensional feature space implicitly defined by the kernel. The paper also presents an efficient algorithm for greedy sampling, which partitions data points into random groups, and selects the column which best represents the centroids of these groups in the high-dimensional feature space. This approximate criterion is very efficient and it obtains highly accurate Nyström approximations.

The rest of this paper is organized as follows. Section 2 defines the notations used throughout the paper. Section 3 reviews the basic Nyström method. Section 4 proposes a recursive algorithm for calculating the Nyström approximation. Section 5 extends the algorithm by proposing the greedy selection criteria. Re-

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15.

lated work is then discussed in Section 6. Section 7 presents an empirical evaluation of the proposed algorithms. Finally, Section 8 concludes the paper.

2 NOTATIONS

Throughout the paper, scalars, vectors, sets, and matrices are shown in small, small bold italic, script, and capital letters, respectively. In addition, the following notations are used.

For a vector $\boldsymbol{x} \in \mathbb{R}^p$:

$\boldsymbol{x}_i$	i-th element of $\boldsymbol{x}$.
$\|\boldsymbol{x}\|$	the Euclidean norm (ℓ_2-norm) of $\boldsymbol{x}$.

For a matrix $A \in \mathbb{R}^{p \times q}$:

A_{ij}	(i,j)-th entry of A.
$A_{i:}$	i-th row of A.
$A_{:j}$	j-th column of A.

For a kernel matrix $K \in \mathbb{R}^{p \times p}$:

$\tilde{K}$	a low rank approximation of K.
$\tilde{K}_k$	the best rank-k approximation of K obtained using singular value decomposition.
$\tilde{K}_{\mathcal{S}}$	rank-l Nyström approximation of K based on the set $\mathcal{S}$ of columns, where $\|\mathcal{S}\| = l$.
$\tilde{K}_{\mathcal{S},k}$	rank-k Nyström approximation of K based on the set $\mathcal{S}$ of columns, where $\|\mathcal{S}\| = l$ and $k \leq l$.

3 THE NYSTRÖM METHOD

The Nyström method obtains a low-rank approximation of a kernel matrix using a subset of its columns. Let K be an $n \times n$ symmetric positive semi-definite (SPSD) kernel matrix defined over n data instances. The Nyström method starts by selecting a subset of $l \ll n$ columns of K (usually by random sampling). These columns represent the similarities between the subset of l data instances and all data instances. Let $\mathcal{S}$ be the set of the indices of selected columns, and $\mathcal{R}$ be the set of the indices of remaining columns. Without loss of generality, the columns and rows of K can be arranged as follows:

$$K = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}, \tag{1}$$

where A, B and C are sub-matrices of K whose elements are $\{K_{ij} : i,j \in \mathcal{S}\}$, $\{K_{ij} : i \in \mathcal{S}, j \in \mathcal{R}\}$, and $\{K_{ij} : i,j \in \mathcal{R}\}$ respectively, and K_{ij} denotes the element of K at row i and column j.

The Nyström method calculates a rank-l approximation of K as (Williams and Seeger, 2001):

$$\tilde{K}_{\mathcal{S}} = DA^{-1}D^T, \tag{2}$$

where $D = \begin{bmatrix} A & B \end{bmatrix}^T$ is an $n \times l$ matrix which consists of the selected columns of K.

The Nyström method can also be used to approximate the leading singular values and vectors of K using those of A (Williams and Seeger, 2001), which is sometimes referred to as the approximate spectral decomposition (Kumar et al., 2009c).The k leading singular values and vectors of K can be approximated as:

$$\tilde{\Sigma}_k = \frac{n}{l}\Lambda_k, \quad \tilde{U}_k = \sqrt{\frac{l}{n}} D V_k \Lambda_k^{-1}. \tag{3}$$

where $k \leq l \ll n$. V_k and U_k are $l \times k$ and $n \times k$ matrices whose columns are the k leading singular vectors of A and K respectively. Λ_k and Σ_k are $k \times k$ matrices whose diagonal elements are the k leading singular values of A and K respectively.

The approximate singular values and vectors of K can be used to map data points to a k-dimensional space:

$$Y = \tilde{\Sigma}_k^{1/2} \tilde{U}_k^T = \Lambda_k^{-1/2} V_k^T D^T, \tag{4}$$

where Y is a $k \times n$ matrix whose columns represent data instances in the k-dimensional space. The kernel matrix over data points in the k-dimensional space represents a rank-k approximation of K which can be calculated as:

$$\tilde{K}_{\mathcal{S},k} = Y^T Y = D V_k \Lambda_k^{-1} V_k^T D^T. \tag{5}$$

Throughout the rest of the paper, "Nyström approximation" and "rank-l Nyström approximation" are used interchangeably to refer to $\tilde{K}_{\mathcal{S}}$, while "rank-k Nyström approximation" refers to $\tilde{K}_{\mathcal{S},k}$.

The computational complexity of calculating A^{-1} is $\mathcal{O}\left(l^3\right)$, and those of calculating Y and $\tilde{K}_{\mathcal{S},k}$ are $\mathcal{O}\left(l^3 + nlk\right)$ and $\mathcal{O}\left(l^3 + nlk + n^2k\right)$, respectively. It should be noted that the approximate singular vectors, as well as the basis of the k-dimensional space are, however, non-orthonormal (Kumar et al., 2009c). In some applications, additional steps might be required to obtain orthonormal vectors. This, however, increases the computational complexity.

4 RECURSIVE NYSTRÖM METHOD

In this section, an algorithm is derived which calculates the Nyström approximation in a recursive manner. At each iteration of the algorithm, one column is selected and a rank-1 Nyström approximation is calculated based on that column. A residual matrix is then calculated and the same steps are repeated recursively on the residual matrix.

Let $q \in \mathcal{S}$ be the index of one of the selected columns, α be the q-th diagonal element of K, $\boldsymbol{\delta}$ be the q-th column of K, and $\boldsymbol{\beta}$ be a column vector of length $l-1$ whose elements are $\{K_{iq} : i \in \mathcal{S} \setminus \{q\}\}$. Without loss of generality, the rows and columns of K (and accordingly A and D) can be rearranged such that the first row and column correspond to q. The matrices A and D in Equation (2) can be written as:

$$A = \begin{bmatrix} \alpha & \boldsymbol{\beta}^T \\ \boldsymbol{\beta} & \Gamma \end{bmatrix}, \quad D = \begin{bmatrix} \boldsymbol{\delta} & \Delta^T \end{bmatrix}, \tag{6}$$

where Γ is a $(l-1) \times (l-1)$ sub-matrix of A whose elements are $\{K_{ij} : i,j \in \mathcal{S} \setminus \{q\}\}$, and Δ is a $(l-1) \times (n)$ sub-matrix of D whose elements are $\{K_{ij} : i \in \mathcal{S} \setminus \{q\}, j \in \{1,..,n\}\}$.

Let $S = \Gamma - \frac{1}{\alpha}\boldsymbol{\beta}\boldsymbol{\beta}^T$ be the Schur complement (Lütkepohl, 1996) of α in A. Use the block-wise inversion formula (Lütkepohl, 1996) of A^{-1} and substitute with D and A^{-1} in Equation (2):

$$\tilde{K}_{\mathcal{S}} = \begin{bmatrix} \boldsymbol{\delta} & \Delta^T \end{bmatrix} \begin{bmatrix} \frac{1}{\alpha} + \frac{1}{\alpha^2}\boldsymbol{\beta}^T S^{-1}\boldsymbol{\beta} & -\frac{1}{\alpha}\boldsymbol{\beta}^T S^{-1} \\ -\frac{1}{\alpha}S^{-1}\boldsymbol{\beta} & S^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}^T \\ \Delta \end{bmatrix} \tag{7}$$

The right-hand side of (7) can be simplified to:

$$\tilde{K}_{\mathcal{S}} = \frac{1}{\alpha}\boldsymbol{\delta}\boldsymbol{\delta}^T + \left(\Delta - \frac{1}{\alpha}\boldsymbol{\beta}\boldsymbol{\delta}^T\right)^T S^{-1} \left(\Delta - \frac{1}{\alpha}\boldsymbol{\beta}\boldsymbol{\delta}^T\right) \tag{8}$$

Let $\tilde{K}_{\{q\}} = \frac{1}{\alpha}\boldsymbol{\delta}\boldsymbol{\delta}^T$ be the rank-1 Nyström approximation of K obtained using the column corresponding to q [1], and $E^{(K)}_{\{q\}}$ be an $n \times n$ residual matrix which is calculated as: $E^{(K)}_{\{q\}} = K - \tilde{K}_{\{q\}}$. It can be shown that $E^{(\Gamma)}_{\{q\}} = S$ and $E^{(\Delta)}_{\{q\}} = \Delta - \frac{1}{\alpha}\boldsymbol{\beta}\boldsymbol{\delta}^T$ are the sub-matrices of $E^{(K)}_{\{q\}}$ corresponding to Γ and Δ respectively. $\tilde{K}_{\mathcal{S}}$ can be written in terms of $E^{(\Gamma)}_{\{q\}}$ and $E^{(\Delta)}_{\{q\}}$ as:

$$\tilde{K}_{\mathcal{S}} = \tilde{K}_{\{q\}} + E^{(\Delta)^T}_{\{q\}} {E^{(\Gamma)}_{\{q\}}}^{-1} E^{(\Delta)}_{\{q\}}. \tag{9}$$

The second term is the Nyström approximation of $E^{(K)}_{\{q\}}$ based on $\mathcal{S} \setminus \{q\}$. This means that rank-$l$ Nyström approximation of K can be constructed in a recursive manner by first calculating a rank-1 Nyström approximation of K based on the column corresponding to q, and then calculating the rank-$(l-1)$ Nyström approximation of the residual matrix based on the columns corresponding to the remaining elements of $\mathcal{S}$.

Based on this recursion, the rank-l Nyström approximation of K can be expressed as a summation of rank-1 approximations calculated at different iterations of the recursive formula:

$$\tilde{K}_{\mathcal{S}} = \sum_{t=1}^{l} \boldsymbol{\omega}^{(t)}\boldsymbol{\omega}^{(t)T}, \tag{10}$$

where $\boldsymbol{\omega}^{(t)} = \boldsymbol{\delta}^{(t)}/\sqrt{\alpha^{(t)}}$, $\boldsymbol{\delta}^{(t)}$ is the column sampled at iteration t, and $\alpha^{(t)}$ is the corresponding diagonal element. $\boldsymbol{\delta}^{(t)}$ and $\alpha^{(t)}$ can be efficiently calculated as:

$$\boldsymbol{\delta}^{(t)} = K_{:q} - \sum_{r=1}^{t-1} \boldsymbol{\omega}^{(r)}_q \boldsymbol{\omega}^{(r)}, \quad \alpha^{(t)} = \boldsymbol{\delta}^{(t)}_q, \tag{11}$$

where q is the index of the column selected at iteration t, $K_{:q}$ denotes the q-th column of K, and $\boldsymbol{\delta}_q$ denotes the q-th element of $\boldsymbol{\delta}$. $\tilde{K}_{\mathcal{S}}$ can also be expressed in a matrix form as: $W^T W$, where W is an $l \times n$ matrix whose t-th row is $\boldsymbol{\omega}^{(t)T}$. The columns of W can be used to represent data instances in a l-dimensional space. However, as the rows of W are non-orthogonal, additional steps are applied to obtain an orthogonal basis. The proposed algorithm calculates the k leading singular vectors of W (or equivalently, the eigenvectors of WW^T), and then uses these vectors to represent data instances in a low-dimension space as follows:

$$Y = \Omega_k^T W, \tag{12}$$

where Y is a $k \times n$ matrix whose columns represent data instances in the k-dimensional space, and Ω_k is a $k \times k$ matrix whose columns are the k leading singular vectors of W. The corresponding rank-k Nyström approximation of K is:

$$\tilde{K}_{\mathcal{S},k} = Y^T Y = W^T \Omega_k \Omega_k^T W. \tag{13}$$

Although the recursive Nyström algorithm calculates the same rank-l Nyström approximation $\tilde{K}_{\mathcal{S}}$ as the traditional Nyström formula (Equation 2), it calculates different estimates of Y and $\tilde{K}_{\mathcal{S},k}$. The advantage of the recursive algorithm is that the basis of low-dimension representation is orthogonal, and that $\tilde{K}_{\mathcal{S},k}$ is the best rank-k approximation of $\tilde{K}_{\mathcal{S}}$.

The computational complexity of calculating $\boldsymbol{\delta}^{(t)}$ (Equation 11) in terms of previous $\boldsymbol{\omega}$'s is $\mathcal{O}(nt)$, and that of W is $\mathcal{O}(nl^2)$. The computational complexity of orthogonalization steps is $\mathcal{O}(l^3 + nl^2)$. Thus, the computational complexity of calculating Y is $\mathcal{O}(l^3 + nlk + nl^2)$ and that of $\tilde{K}_{\mathcal{S},k}$ is $\mathcal{O}(l^3 + nlk + n^2k + nl^2)$. This is the same complexity as the traditional Nyström method with orthogonalization.

5 GREEDY SAMPLING CRITERION

The recursive nature of the Nyström method can be used to develop an efficient greedy algorithm for sam-

[1] This can be obtained using Equation (2) when A is a scalar and D is a column vector.

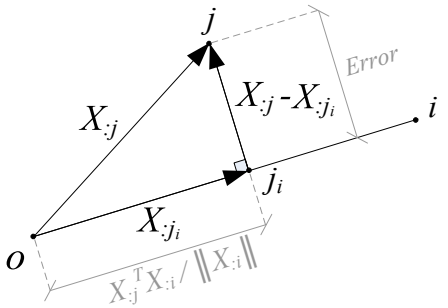

Figure 1: Projection for rank-1 Nyström approximation.

pling columns while calculating the low-rank approximation. The basic idea here is to select, at each iteration, the column that constructs the best rank-1 Nyström approximation of the current residual matrix. Thus, there is a need to define a quality measure for the rank-1 Nyström approximation obtained by the recursive Nyström method at each iteration, and then to select the column that maximizes this quality measure. In the recursive algorithm, the residual matrix is initially equal to K, and the goal is to select a column q from the n columns of K. For a candidate column i, the rank-1 Nyström approximation of matrix K based on that column is calculated as:

$$\tilde{K}_{\{i\}} = \frac{1}{K_{ii}} K_{:i} K_{:i}^T, \tag{14}$$

where $K_{:i}$ denotes the i-th column of K, and K_{ii} denotes the i-th diagonal element of K.

A key observation about Equation (14) is that the rank-1 Nyström approximation based on the i-th column implicitly projects all data points onto a line which contains data point i and the origin in the high-dimensional feature space defined by the kernel (as illustrated in Figure 1). To prove that, assume that a linear kernel is used (The same proof applies to other kernel types, as any kernel matrix implicitly maps data vectors to a high-dimensional linear space.) The kernel matrix is calculated as $K = X^T X$, where X is an $d \times n$ data matrix, and d is the number of features. Let $X_{:i}$ be the i-th column of X. $X_{:i}$ also represents a vector that connects data point i and the origin. $K_{:i}$ and K_{ii} can be written as: $K_{:i} = X^T X_{:i}$, and $K_{ii} = \|X_{:i}\|^2$, where $\|.\|$ is the ℓ_2 norm. Based on this, $\tilde{K}_{\{i\}}$ can be expressed as:

$$\tilde{K}_{\{i\}} = X^T \frac{X_{:i}}{\|X_{:i}\|} \frac{X_{:i}^T}{\|X_{:i}\|} X, \tag{15}$$

where $X^T X_{:i}/\|X_{:i}\|$ is a column vector whose j-th element is $X_{:j}^T X_{:i}/\|X_{:i}\|$. This value is the scalar projection of data point j onto $X_{:i}$. This means that $\tilde{K}_{\{i\}}$ implicitly projects all data points into a 1-dimensional subspace (i.e., a line) which contains data point i and the origin, and then calculates the inner-products between the projected points.

Based on this observation, an efficient criterion can be developed for column selection. The criterion selects the column $\boldsymbol{\delta} = K_{:q}$ which achieves the least squared error between data points in the feature space and their projections onto $X_{:q}$ (also called the reconstruction error). The intuition behind this criterion is that greedily minimizing reconstruction error in the high-dimensional feature space leads to minimizing the difference between kernel matrices in the original and reconstructed spaces. It should be noted that this is similar to principal components analysis (PCA), in which the projection onto the principal subspace minimizes the squared reconstruction error.

Let j_i be the projection of data point j onto $X_{:i}$. $X_{:j_i}$ represents a vector in the direction of $X_{:i}$ whose length is the scalar projection of data point j onto $X_{:i}$. The goal is to find a data point q such that the sum of squared errors between data points and their projections onto $X_{:q}$ is minimized. This can be expressed as the following optimization problem:

$$q = \underset{i}{arg\,min}\ \Sigma_{j=1}^n \|X_{:j} - X_{:j_i}\|^2 \tag{16}$$

Since vector $(X_{:j} - X_{:j_i})$ is orthogonal to $X_{:j_i}$, the distance between a data point j and its projection j_i is: $\|X_{:j} - X_{:j_i}\|^2 = \|X_{:j}\|^2 - \|X_{:j_i}\|^2$, and the objective function of (16) is: $\Sigma_{j=1}^n \|X_{:j} - X_{:j_i}\|^2 = \Sigma_{j=1}^n \|X_{:j}\|^2 - \Sigma_{j=1}^n \|X_{:j_i}\|^2$. The term $\Sigma_{j=1}^n \|X_{:j}\|^2$ is the sum of the lengths of all data vectors which is a constant for different values of i, and the term $\Sigma_{j=1}^n \|X_{:j_i}\|^2$ can be written as: $\Sigma_{j=1}^n (X_{:j}^T X_{:i}/\|X_{:i}\|)^2 = \|X^T X_{:i}/\|X_{:i}\|\|^2 = \|K_{:i}/\sqrt{K_{ii}}\|^2$. Accordingly, the optimization problem (16) is equivalent to:

$$q = \underset{i}{arg\,max}\ \|\frac{1}{\sqrt{K_{ii}}} K_{:i}\|^2. \tag{17}$$

This means that to obtain the best rank-1 approximation according to the squared error criterion, the proposed algorithm evaluates $\|K_{:i}/\sqrt{K_{ii}}\|^2$ for all the columns of K, and selects the column with the maximum criterion function. The same selection procedure is then applied during the next iterations of the recursive algorithm on the new residual matrices (i.e., $\|E_{:i}/\sqrt{E_{ii}}\|^2$).

The computational complexity of the selection criterion is $\mathcal{O}\left(n^2 + n\right)$ per iteration, and it requires $\mathcal{O}\left(n^2\right)$ memory to store the residual of the whole kernel matrix after each iteration. In the rest of this section, two novel techniques are proposed to reduce the memory and time requirements of the greedy selection criterion.

Algorithm 1 Greedy Nyström Approximation

Inputs: K, l, k, **Outputs:** $\mathcal{S}$, $\tilde{K}_{\mathcal{S}}$, $\tilde{K}_{\mathcal{S},k}$, Y

1. Initialize $\mathcal{S} = \{\ \}$, Generate a random partitioning P, Calculate G: $G_{ji} = \sum_{r \in \mathcal{P}_j} K_{ir}$
2. Initialize $\boldsymbol{f}_i^{(0)} = \|G_{:i}\|^2$, and $\boldsymbol{g}_i^{(0)} = K_{ii}$
3. Repeat $t = 1 \rightarrow l$:
   (a) $q = arg\ \underset{i}{max}\ \boldsymbol{f}_i^{(t)} / \boldsymbol{g}_i^{(t)}, \quad \mathcal{S} = \mathcal{S} \cup \{q\}$
   (b) $\boldsymbol{\delta}^{(t)} = K_{:q} - \sum_{r=1}^{t-1} \boldsymbol{\omega}_q^{(r)} \boldsymbol{\omega}^{(r)}$, $\alpha^{(t)} = \boldsymbol{\delta}_q^{(r)}$
   (c) $\boldsymbol{\gamma}^{(t)} = G_{:q} - \sum_{r=1}^{t-1} \boldsymbol{\omega}_q^{(r)} \boldsymbol{\upsilon}^{(r)}$
   (d) $\boldsymbol{\omega}^{(t)} = \boldsymbol{\delta}^{(t)} / \sqrt{\alpha^{(t)}}$, $\boldsymbol{\upsilon}^{(t)} = \boldsymbol{\gamma}^{(t)} / \sqrt{\alpha^{(t)}}$
   (e) Update $\boldsymbol{f}_i$'s, $\boldsymbol{g}_i$'s (Equation 19)
4. $W = \left[\ w^{(1)} \quad ... \quad w^{(l)}\ \right]^T$, $\tilde{K}_{\mathcal{S}} = W^T W$
5. $\Omega = eigvec(WW^T)$, $Y = \Omega_k^T W$, $\tilde{K}_{\mathcal{S},k} = Y^T Y$

Memory-Efficient Sampling To reduce the memory requirements of the greedy algorithm, the sampling criterion for each data instance can be calculated in a recursive manner as follows. Let $\boldsymbol{f}_i = \|E_{:i}\|^2$ and $\boldsymbol{g}_i = E_{ii}$ be the numerator and denominator of the criterion function for data point i respectively, $\boldsymbol{f} = [\boldsymbol{f}_i]_{i=1..n}$, and $\boldsymbol{g} = [\boldsymbol{g}_i]_{i=1..n}$. It can be shown that $\boldsymbol{f}$ and $\boldsymbol{g}$ can be calculated recursively as follows[2]:

$$\begin{aligned}\boldsymbol{f}^{(t)} &= \Big(\boldsymbol{f} - 2\left(\boldsymbol{\omega} \circ \left(K\boldsymbol{\omega} - \Sigma_{r=1}^{t-2}\left(\boldsymbol{\omega}^{(r)^T}\boldsymbol{\omega}\right)\boldsymbol{\omega}^{(r)}\right)\right) \\ &\quad + \|\boldsymbol{\omega}\|^2 (\boldsymbol{\omega} \circ \boldsymbol{\omega})\Big)^{(t-1)}, \\ \boldsymbol{g}^{(t)} &= \Big(\boldsymbol{g} - (\boldsymbol{\omega} \circ \boldsymbol{\omega})\Big)^{(t-1)}.\end{aligned} \tag{18}$$

where $\circ$ represents the Hadamard product operator, and $\|.\|$ is the ℓ_2 norm. This means that the greedy criterion can be memory-efficient by only maintaining two score variables for each data point, $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$, and updating them at each iteration based on their previous values and the selected columns so far.

Partition-Based Sampling In order to reduce the computational complexity, a novel partition-based criterion is proposed, which reduces the number of scalar projections to be calculated at each iteration. The criterion partitions data points into $c \ll n$ random groups, and selects the column of K which best represents the centroids of these groups in the high-dimensional feature space. Let $\mathcal{P}_j$ be the set of data points that belong to the j-th partition, $P = \{\mathcal{P}_1, \mathcal{P}_2, ...\mathcal{P}_c\}$ be a random partitioning of data points into c groups, and G be an $c \times n$ matrix whose element G_{ji} is the inner-product of the centroid of the j-th group and the i-th data point, weighted with the size of the j-th group. The use of weighted inner-products avoids any bias towards larger groups when calculating the sum of scalar projections. As the scalar projections are implicitly calculated in a high-dimensional linear space defined by the kernel matrix K, G_{ji} can be calculated in this linear space as: $G_{ji} = \sum_{r \in \mathcal{P}_j} K_{ir}$. In general, it requires $\mathcal{O}\left(n^2\right)$ to calculate G given K. However, G needs to be calculated only once during the calculation of K. In the case of a linear kernel, this complexity could be significantly reduced by calculating the centroids of each group in the feature space, and then the inner-products between each centroid and all data points. This computational complexity could be reduced further if the data matrix is very sparse. In addition, there is no need to calculate and store the whole kernel matrix in order to calculate G.

Let $H^{(t)}$ be the residual of G at iteration t, and $\boldsymbol{\gamma}^{(t)}$ be the column of H corresponding to the selected column at iteration t, which can be calculated as: $\boldsymbol{\gamma}^{(t)} = G_{:q} - \sum_{r=1}^{t-1} \boldsymbol{\omega}_q^{(r)} \boldsymbol{\upsilon}^{(r)}$. It can be shown that for partition-based sampling, $\boldsymbol{f}$ and $\boldsymbol{g}$ can be calculated as:

$$\begin{aligned}\boldsymbol{f}^{(t)} &= \Big(\boldsymbol{f} - 2\left(\boldsymbol{\omega} \circ \left(G^T\boldsymbol{\upsilon} - \Sigma_{r=1}^{t-2}\left(\boldsymbol{\upsilon}^{(r)T}\boldsymbol{\upsilon}\right)\boldsymbol{\omega}^{(r)}\right)\right) \\ &\quad + \|\boldsymbol{\upsilon}\|^2 (\boldsymbol{\omega} \circ \boldsymbol{\omega})\Big)^{(t-1)}, \\ \boldsymbol{g}^{(t)} &= \Big(\boldsymbol{g} - (\boldsymbol{\omega} \circ \boldsymbol{\omega})\Big)^{(t-1)}.\end{aligned} \tag{19}$$

where $\boldsymbol{\upsilon}^{(t)} = \boldsymbol{\gamma}^{(t)} / \sqrt{\alpha^{(t)}}$. The computational complexity of the new update formulas is $\mathcal{O}\left(ncl + nl^2\right)$ (or $\mathcal{O}\left(nc + nt\right)$ per iteration). Algorithm 1 shows the complete greedy Nyström algorithm.

6 RELATED WORK

Different sampling schemes have been used with the Nyström method. Williams and Seeger (2001), who first proposed the use of Nyström approximation for kernel methods, used uniform sampling without replacement to select columns. This has been the most commonly used sampling scheme for Nyström methods. Non-uniform sampling has also been used with Nyström methods. This includes non-uniformly sampling columns based on the corresponding diagonal elements of the kernel matrix (Drineas and Mahoney, 2005), or the norms of its columns (Drineas et al., 2007). Recently, Kumar et al. (2009b) showed that uniform sampling without replacement outperforms other random sampling techniques on real data sets.

[2]The proof is omitted due to space limitation.

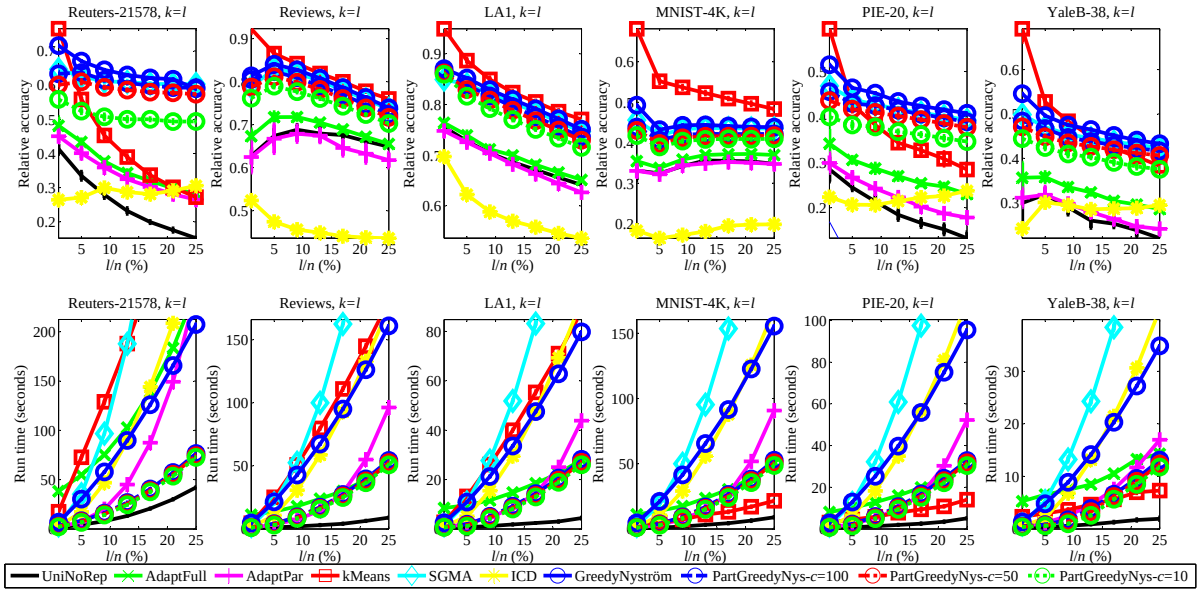

Figure 2: The relative accuracy and run time of rank-l approximations $\tilde{K}_{\mathcal{S}}$ for different methods.

Adaptive sampling has also been used with Nyström methods. These techniques update sampling probabilities based on intermediate approximations of the kernel matrix. Deshpande et al. (2006) suggested an adaptive sampling algorithm which iteratively samples subsets of columns using probabilities calculated based on the low-rank approximation error so far. This adaptive mechanism is more effective than fixed sampling. However, it is computationally more complex, as it requires the calculation of the Nyström approximation at each iteration of the algorithm. A more efficient algorithm for adaptive sampling (Kumar et al., 2009c) calculates sampling probabilities based on the approximation error of a small part of the kernel matrix. A more recent work (Kumar et al., 2009a) uses an ensemble of Nyström approximations to obtain a better low-rank approximation of the kernel matrix.

Besides random sampling, deterministic sampling has also been used with Nyström methods. Sparse greedy matrix approximation (SGMA) (Smola and Schölkopf, 2000) is a related algorithm, which selects a set of basis kernel functions, and represents other kernel functions as a linear combination of these basis functions: $\tilde{K} = K^{\mathcal{S}:}T$, where $K^{\mathcal{S}:}$ denotes the subset of selected columns, and T is a matrix of coefficients. The authors showed that low-rank approximation which optimizes the approximation error in the reproducing kernel Hilbert space (RKHS) is equal to Nyström approximation (Schölkopf and Smola, 2002, chap. 10). They also proposed a greedy algorithm to select columns of the kernel matrix and recursively update T based on the newly selected columns. The selection criterion used by SGMA is based on maximizing the improvement of the low-rank approximation error in RKHS. To reduce the complexity of this algorithm, a probabilistic speedup was suggested by the authors to evaluate the criterion function for only a random subset of columns. This makes the complexity of the selection criterion $\mathcal{O}(Nnl^2)$ (or $\mathcal{O}(Nnt)$ per iteration), where N is the size of the random subset. Ouimet and Bengio (2005) proposed another greedy sampling algorithm which recursively selects samples that are far (using some threshold) from the subspace spanned by previously selected samples. Incomplete Cholesky decomposition (Fine and Scheinberg, 2002) can also be used to greedily select columns for the Nyström method. Zhang et al. (2008) recently proposed an algorithm which first applies the k-means algorithm to cluster data points, and then uses the centroids of clusters for calculating the Nyström approximation. As the k-means algorithm scales as $\mathcal{O}(ndlt)$, where d is the number of features and t is the number of iterations, this algorithm is computationally infeasible for data sets with large number of features.

On the other hand, there is considerable research work on low-rank approximation of rectangular matrices. Most of this work is based on random sampling of columns (and/or rows) (Frieze et al., 2004; Drineas et al., 2007; Mahoney and Drineas, 2009). One deterministic approach has been proposed by Çivril and Magdon-Ismail (2008), who suggested a greedy algorithm for low-rank construction of a rectangular matrix, based on selecting columns that best fit the space spanned by the leading singular vectors of the matrix.

Table 1: Properties of data sets used to evaluate different Nyström methods. n and d are the number of instances and features respectively.

Data set	Type	n	d
Reuters-21578	Documents	5946	18933
Reviews	Documents	4069	36746
LA1	Documents	3204	29714
MNIST-4K	Digit Images	4000	784
PIE-20	Face Images	3400	1024
Yale-B-38	Face Images	2414	1024

Comparison to Related Work Like adaptive sampling, the greedy algorithm presented in this paper selects columns based on intermediate approximations of the kernel matrix. However, at each iteration, the greedy algorithm deterministically selects one column, while adaptive methods randomly sample a subset of columns. In term of computational complexity, the complexity of adaptive sampling based on the full kernel (Deshpande et al., 2006) is $\mathcal{O}(n^2v + nv^2 + v^3)$ per iteration, where v is the number of samples selected so far. The greedy selection criterion without partitioning ($\mathcal{O}(n^2l)$) is therefore less complex than the last iteration of the adaptive algorithm with the full kernel (when $v = l$). On the other hand, the greedy criterion without partitioning is more complex than adaptive sampling based on part of the kernel (Kumar et al., 2009c), which is $\mathcal{O}(nv^2 + v^3)$ per iteration.

In comparison to SGMA (Smola and Schölkopf, 2000), it can be shown that maximizing the improvement in approximation error is equivalent to minimizing the squared reconstruction error in the feature space. However, the basic selection criterion presented here ($\mathcal{O}\left(n^2l\right)$) is more efficient than that of SGMA with probabilistic speedup ($\mathcal{O}(nNl^2)$) when $l/n \geq 1/N$. In addition, the approximation of K as $K^{\mathcal{S}:}T$ does not allow SGMA to be directly applied to approximate spectral decomposition and dimension reduction. In this case, the Nyström method has to be applied to columns selected by SGMA, which requires extra computational cost. In comparison to k-means (Zhang et al., 2008), which is $\mathcal{O}(ndlt)$, the greedy selection criterion is computationally less complex for data sets with large number of features (when $dt > n$). On the other hand, the partition-based selection criterion ($\mathcal{O}(ncl+nl^2)$) is much less complex than the two adaptive sampling methods, SGMA, and sampling based on k-means centroids.

The greedy Nyström algorithm is also different from the greedy algorithm proposed by Çivril and Magdon-Ismail (2008), as the latter depends on the availability of the leading singular vectors to select columns.

7 EXPERIMENTS AND RESULTS

Experiments have been conducted on six benchmark data sets, whose properties are summarized in Table 1 The *Reuters-21578* is the training set of the Reuters-21578 collection (Lewis, 1999). The *Reviews* and *LA1* are document data sets from TREC collections[3]. The pre-processed versions of *Reviews* and *LA1* that are distributed with the CLUTO Toolkit (Karypis, 2003) were used. The *MNIST-4K* is a subset of the MNIST data set of handwritten digits[4]. The *PIE-20* and *YaleB-38* are pre-processed subsets of the CMU PIE (Sim et al., 2003) and Extended Yale Face (Lee et al., 2005) data sets respectively (He et al., 2005).

Similar to previous work (Kumar et al., 2009b), the low-rank approximations obtained by greedy Nyström algorithm are compared to those obtained by other Nyström methods relative to the best low-rank approximation obtained by singular value decomposition. In particular, the following quality measure is used:

$$\textit{Relative Accuracy} = \frac{\|K - \tilde{K}_r\|_F}{\|K - \tilde{K}_{Nys}\|_F}, \qquad (20)$$

where K is the kernel matrix, $\tilde{K}_r$ is the best rank-r approximation obtained using singular decomposition, $\tilde{K}_{Nys}$ is the rank-r Nyström approximation (i.e., $\tilde{K}_{\mathcal{S}}$, or $\tilde{K}_{\mathcal{S},k}$), and $\|.\|_F$ is the Frobenius norm. The relative accuracy is between 0 and 1 with higher values indicating a better low-rank approximation. The run times of different algorithms are also compared.

The basic greedy Nyström algorithm (**GreedyNyström**) and its partition-based variant (**PartGreedyNys**) are compared to six well-known Nyström methods: (1) **UniNoRep**: Uniform sampling without replacement, which has been shown to outperform other random sampling methods (Kumar et al., 2009b); (2) **AdaptFull**: Adaptive sampling based on the full kernel matrix (Deshpande et al., 2006), (3) **AdaptPart**: Adaptive sampling based on a part of the kernel matrix (Kumar et al., 2009c), (4) **k-means**, Nyström method based on k-means centroids (Zhang et al., 2008), (5) **SGMA**: The SGMA algorithm with probabilistic speedup (Smola and Schölkopf, 2000), and (6) **ICD**: The incomplete Cholesky decomposition with symmetric pivoting (Fine and Scheinberg, 2002).

For adaptive sampling, similar to Kumar et al. (2009c), 10 iterations were used (i.e., $l/10$ columns were sampled in each iteration). For SGMA, a random subset size of 59 was used, as suggested by Smola and Schölkopf (2000). For k-means, the k-means algorithm

[3] `http://trec.nist.gov`

[4] `http://yann.lecun.com/exdb/mnist`

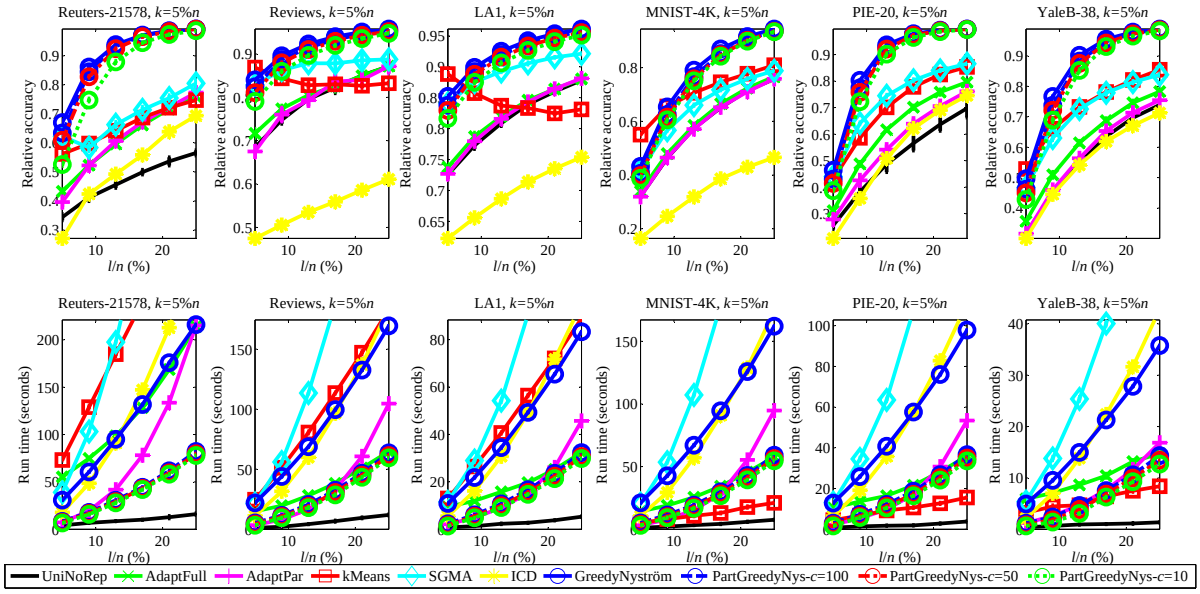

Figure 3: The relative accuracy and run time of rank-k approximations $\tilde{K}_{\mathcal{S},k}$ for different methods.

by Zhang et al. (2008) was used, with the same number of iterations. Experiments with randomness were repeated 10 times, and the average and standard deviation of measures were calculated. Linear kernels were used for document data sets, and Gaussian kernels with $\sigma = 10$ for image data sets.

Two sets of experiments were conducted to evaluate the quality of rank-l and rank-k Nyström approximations ($\tilde{K}_{\mathcal{S}}$ and $\tilde{K}_{\mathcal{S},k}$) for different sampling methods. For $\tilde{K}_{\mathcal{S},k}$, as SGMA and ICD do not directly allow the calculation rank-k approximation from $\tilde{K}$, SGMA and ICD were used for selecting columns and then a traditional Nyström formula was applied.

Figures 2 and 3 show the relative accuracies and run times for the two experiements. It can be observed from results that the greedy Nyström method (**GreedyNyström**) achieves significant improvement in estimating low-rank approximations of a kernel matrix, compared to other sampling-based methods. It also achieves better accuracy than **SGMA** and **k-means** for most data sets. Although the **k-means** achieves better accuracy for some data sets, it obtains much worse accuracy for others. This inconsistency could be due to the nature of the k-means algorithm, which might obtain a poor local minimum. It can also be seen that **GreedyNyström** is more efficient than **SGMA** and **AdaptFull**, but is computationally more complex than **UniNoRep** and **AdaptPart**. The latter two methods, however, obtain inferior accuracies. **GreedyNyström** is also computationally less complex than **k-means** for data sets with large number of features. On the other hand, the partition-based algorithm (**PartGreedyNys**) outperforms the two adaptive sampling methods in obtaining low-rank approximations, and it requires small overhead in run time compared to **UniNoRep**. **PartGreedyNys** obtains slightly lower accuracies than **GreedyNyström** and **SGMA** when calculating $\tilde{K}_{\mathcal{S}}$, but in much less time, and it outperforms all other deterministic methods when calculating $\tilde{K}_{\mathcal{S},k}$. **PartGreedyNys** is also not sensitive to the number of random partitions used. It can also be noted that **ICD** obtains inferior approximation accuracies compared to other methods[5].

8 CONCLUSIONS

This paper presents a novel recursive algorithm for calculating the Nyström approximation and an effective greedy criterion for column selection. The proposed criterion obtains, at each iteration, the rank-1 approximation which minimizes the reconstruction error in the high-dimensional space implicitly defined by the kernel. It has been empirically shown that the proposed algorithm consistently achieves a significant improvement in the accuracy of low-rank approximation compared to traditional Nyström methods, and is less computationally demanding than other deterministic methods. In addition, a partition-based algorithm for greedy sampling is provided, which achieves very good approximation accuracies and is more efficient than adaptive and deterministic sampling methods.

[5]This was also observed by Zhang et al. (2008) and Talwalkar (2010)

References

A. Çivril and M. Magdon-Ismail. Deterministic sparse column based matrix reconstruction via greedy approximation of SVD. In *the 19th International Symposium on Algorithms and Computation*, pages 414–423. Springer-Verlag, 2008.

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

A. Deshpande, L. Rademacher, S. Vempala, and G. Wang. Matrix approximation and projective clustering via volume sampling. In *the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1117–1126. ACM, 2006.

P. Drineas and M. 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.

P. Drineas, R. Kannan, and M.W. Mahoney. Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix. *SIAM Journal on Computing*, 36(1):158–183, 2007.

S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2002.

C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the Nyström method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):214–225, 2004.

A. Frieze, R. Kannan, and S. Vempala. Fast Monte-Carlo algorithms for finding low-rank approximations. *Journal of the ACM*, 51(6):1041, 2004.

X. He, S. Yan, Y. Hu, P. Niyogi, and H.J. Zhang. Face recognition using laplacianfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(3):328–340, 2005.

G. Karypis. CLUTO - a clustering toolkit. Technical Report #02-017, University of Minnesota, Department of Computer Science, 2003.

S. Kumar, M. Mohri, and A. Talwalkar. Ensemble Nyström Method. In *Advances in Neural Information Processing Systems 22*, pages 1060–1068, 2009a.

S. Kumar, M. Mohri, and A. Talwalkar. Sampling techniques for the Nyström method. In *the 12th International Conference on Artificial Intelligence and Statistics*, pages 304–311, 2009b.

S. Kumar, M. Mohri, and A. Talwalkar. On sampling-based approximate spectral decomposition. In *the 26th Annual International Conference on Machine Learning*, pages 553–560. ACM, 2009c.

K.C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(5):684–698, 2005.

D.D. Lewis. Reuters-21578 text categorization test collection distribution 1.0, 1999.

H. Lütkepohl. *Handbook of Matrices*. John Wiley & Sons Inc, 1996.

M.W. Mahoney and P. Drineas. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697, 2009.

M. Ouimet and Y. Bengio. Greedy spectral embedding. In *the 10th International Workshop on Artificial Intelligence and Statistics*, pages 253–260, 2005.

J.C. Platt. Fastmap, MetricMap, and Landmark MDS are all Nyström algorithms. In *the 10th International Workshop on Artificial Intelligence and Statistics*, pages 261–268, 2005.

B. Schölkopf and A.J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression database. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25 (12):1615–1618, 2003.

A.J. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. In *Pthe 17th Annual International Conference on Machine Learning*, pages 911–918. ACM, 2000.

A. Talwalkar. *Matrix Approximation for Large-scale Learning*. PhD thesis, New York University, 2010.

A. Talwalkar, S. Kumar, and H. Rowley. Large-scale manifold learning. In *the 21st IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, pages 682–688. MIT Press, 2001.

K. Zhang, I.W. Tsang, and J.T. Kwok. Improved Nyström low-rank approximation and error analysis. In *the 25th Annual International Conference on Machine Learning*, pages 1232–1239. ACM, 2008.