# A Brief Report of Low Rank Kernel Matrix Approximation

Yan Ren

2021103739

January 26, 2022

# Contents

**1 Introduction** **3**

1.1 Low rank Approximation . . . . . 3

1.2 Kernel Methods . . . . . 3

**2 SVD** **4**

2.1 SVD . . . . . 4

2.2 Best Rank-k Approximation . . . . . 4

2.3 SVD Computation and Drawbacks . . . . . 5

**3 Nyström Method** **6**

3.1 Nyström Method in Low Rank Approximation . . . . . 6

3.2 Error Bound . . . . . 7

3.3 Drawbacks of Nyström Method . . . . . 7

**4 BKA** **8**

4.1 Relation to K-Means Clustering . . . . . 8

4.2 Low Rank Structure . . . . . 10

**5 MEKA** **13**

5.1 Steps of MEKA . . . . . 13

5.2 Time Complexity and Storage Usage . . . . . 15

**6 Summary & Discussion** **16**

# 1 Introduction

The problem of low rank matrix approximation is raised due to the rapid increase of the data size. Although there are many methods theoretically best for specific problems, the time complexity and storage usage aften stop people from using them pratically. The problem is especially severe in the case of $n \gg d$, where $n$ means data size and $d$ indicates the dimension. For example, when applying kernel methods, kernel matrix of size $n \times n$ is computed as an important intermediate matrix. But the computation is time-consuming and the storage is also expensive for such a huge matrix. Low rank approximation of these matrices are studied as one of main solutions. Other solutions include sketch, which has been focused in class.

## 1.1 Low rank Approximation

For a given matrix $G$, low rank approximation aims to find a rank-k matrix $\hat{G}$ to minimize $\|G - \hat{G}\|$, where $dim(G) = n \times n,\ dim(\hat{G}) = k \times k,\ k < n$. Singular Value Decomposition (SVD) yields the best rank-k approximation. However, SVD is time-consuming and therefore prohibiting in reality. Other methods like Nyström Method improve the time and space efficiency with some estimation accuracy lost. Theory about SVD is reviewed in Section 2.

## 1.2 Kernel Methods

Kernel methods [5] are machine learning algorithms that first map samples from input space to a high-dimensional feature space. For example, for the tasks of clustering, kernel methods use kernel functions to project data points to higher dimension. If points of different groups are linearly separable in higher dimension but not in the observed lower dimension, kernel methods achieve higher accuracy score. Nonetheless, $n \times n$ kernel matrix $G$ is dependent on data size, and thus the storage and computation of kernel matrix becomes tricky if $n$ is large.

Let us note kernel function as $K(x_i, x_j), i, j = 1, ..., n$, and kernel matrix as $G_{n \times n}$ where $G_{ij} = K(x_i, x_j)$. The notation is used below.

In a word, low rank approximation is significant to solve the storage and computation cost of large dataset, and kernel matrix is an important and typical object. The rest of the report is outlined as follows. Section 2 reviews SVD and the best rank-k approximation. Section 3 illustrated a method often used to get rank-k approximation with respectively large $\|G - \hat{G}\|$

but more time and storage efficient compared with SVD. The standard Nyström Method is discussed in detail. Section 4 is more related to kernel matrix approximation problem. BKA is introduced and the advantages of performing k-means before approximation is proved. Last, in section 5, a state-of-art method Memory Efficient Kernel Approximation (MEKA) (first proposed in reference [6]) is introduced. Content related to $\epsilon - covering$ is used in this part. Summary and discussion are in Section 6.

# 2 SVD

Singular Value Decomposition (SVD) of a matrix is a factorization of that matrix into three matrices. In our case of rank-k approximation, the best rank-k approximation solution is gotten as well.

## 2.1 SVD

**Theorem 1.** *(Singular value decomposition). Let* $\mathbf{A}$ *be an* $n \times d$ *matrix with rank* $r$*. Then there exist matrices* $\mathbf{U}, \mathbf{D}$ *and* $\mathbf{V}$ *such that*

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$$

*where* $\mathbf{U}$ *is an* $n \times r$ *orthogonal matrix,* $\mathbf{V}$ *is an* $d \times r$ *orthogonal matrix, and* $\mathbf{D}$ *is an* $r \times r$ *matrix with* $d_{i,j} = 0$ *for* $i \neq j$*, and* $d_{1,1} \geq d_{2,2} \geq \cdots \geq d_{r,r} > 0$*.*

Theorem 1 is proved by mathematical induction, which is discussed in class, thus omitted here. The notation is used below.

## 2.2 Best Rank-k Approximation

**Theorem 2.** *(Echart-Young Theorem). Suppose* $\mathbf{A} \in \mathbb{R}^{n \times d}$*. Let* $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ *be the SVD of* $\mathbf{A}$*. Let* $\mathbf{U}_k$ *and* $\mathbf{V}_k$ *denote the first* $k$ *columns of* $\mathbf{U}$ *and* $\mathbf{V}$*, respectively. Let* $\mathbf{D}_k = \operatorname{diag}(d_{1,1}, \ldots, d_{k,k})$*. Define* $\mathbf{A}_k := \mathbf{U}_k\mathbf{D}_k\mathbf{V}_k^\top = \sum_{i=1}^{k} d_{i,i}\mathbf{u}_k\mathbf{v}_k^\top$*. Then for any* $\mathbf{B} \in \mathbb{R}^{n \times d}$ *with rank* $k$*,*

$$\|\mathbf{A} - \mathbf{A}_k\| \leq \|\mathbf{A} - \mathbf{B}\|$$

Proof is omitted to avoid unnecessary and redundant copy from the handout in class.

## 2.3 SVD Computation and Drawbacks

As we see in Therorem 2, SVD yeilds good theoretical property. However, it is not practical due to computation cost.

**Power Method** Reference [2] summarizes many SVD computation methods. By conducting eigenvalue decomposition of $AA^T$ (a positive-semidefinite matrix), SVD is conducted. Relative efficient ideas of this kind is always to find square roots of eigenvalues of $AA^T$ without actually computing it.

Power method [3] is one of this kind. Still, $dim(X) = n \times d$ means the data matrix. According to Theorem 1, $\exists$ orthogonal matrix $U$ $(dim(U) = n \times k)$, $V$ $(dim(V) = k \times d)$ and diagonal matrix $\Sigma$ $(dim(\Sigma) = k \times k)$ subject to

$$A = U\Sigma V^T = \sum_{i=1}^{r} d_{i,i} u_i v_i^T \tag{1}$$

Define $B = AA^T$

$$\begin{aligned} B = AA^T &= \left(\sum_i d_{i,i}\mathbf{u_i}\mathbf{v_i}^T\right)\left(\sum_j \sigma_j \mathbf{v_j}\mathbf{u_j}^T\right) \\ &= \sum_{i,j} d_{i,i}\sigma_j \mathbf{u_i}\mathbf{v_i}^T\mathbf{v_j}\mathbf{u_j}^T = \sum_{i,j} d_{i,i}\sigma_j \mathbf{u_i}\left(\mathbf{v}_i^T \cdot \mathbf{v_j}\right)\mathbf{u_j}^T \\ &= \sum_i d_{i,i}^2 \mathbf{u_i}\mathbf{u_i}^T \end{aligned} \tag{2}$$

Similarly, we have

$$B^k = \sum_i d_{i,i}^{2k}\mathbf{u_i}\mathbf{u_i}^T \to d_{1,1}^{2k}\mathbf{u_1}\mathbf{u_1}^T. \tag{3}$$

When $k \to \infty$, $\frac{d_{i,i}^k}{d_{1,1}^k} \to 0$, thus the last part of formula 3 exists.

In this way, $B^k \to d_{1,1}^{2k} u_1 u_1^T$ provides a way to get $u_1$ and $d_{1,1}$ by successively powering $B$. However, powering matrix $B$ will be time-consuming. A substitute way is to calculate $B^k\mathbf{x}$, where $x$ is a random unit vector. It is clear that $B^k\mathbf{x} = AA^T\left(B^{k-1}\mathbf{x}\right)$. According to formula 3, $B^k\mathbf{x} \approx d_{1,1}^{2k}\mathbf{u_1}(\mathbf{u_1}^T\mathbf{x})$. The right side of the equation is scalar operation, which is much faster than matrix powering.

The time complexity of power method in SVD is $O(kn^2)$. It is much faster than operating eigenvalue decomposition on $AA^T$, but still prohibitive in practice when $n$ is too large.

# 3 Nyström Method

When $n$ gets too large, one natural way is sampling. The Nyström Method [7] is one of the most widely used technique to approximate the kernel matrix given a sampled subset of columns. Using Standard Nyström Method, $G$ can be approximated by $\tilde{G} = CW_k^+C^T$. The proof of this in the case of low rank kernel matrix approximation is performed in subsection 3.1. The full proof can be found in reference [7].

## 3.1 Nyström Method in Low Rank Approximation

**Theorem 3.** *(Nyström Method in Low Rank Approximation)*
*$G$ is a kernel matrix for some kernel method. Consider the first $k < n$ points in the data set. Then there exists a matrix $\tilde{G}$ of rank $k$ : $\tilde{G} = C_{n\times k}W_k^+C_{k\times n}^T$, where $(C_{n\times k})_{ij} = K(x_i, x_j),\ i = 1, ..., n,\ j = 1, ..., k$.*

*Proof.* Suppose SVD of $X$ is $X_{n\times d} = U_{n\times q}\Sigma_{q\times q}V_{q\times d}^\top$

$$\begin{cases} X = U\Sigma V^\top \\ X^\top = V\Sigma U^\top \end{cases} \Rightarrow \begin{cases} XX^\top = U\Sigma^2U^\top \\ X^\top X = V\Sigma^2V^\top \end{cases} \tag{4}$$

Let

$$(X_k)_{k\times d} := (U_k)_{k\times q}(\Sigma_k)_{q\times q}\left(V_k^\top\right)_{q\times d} \tag{5}$$

Define

$$\begin{cases} W_{n\times n} & = XX^\top = U\Sigma^2U^\top \\ (W_k)_{k\times k} & = X_kX_k^\top = (U_k)_{k\times q}(\Sigma_k)_{q\times q}^2(U_k)_{q\times k}^\top \end{cases} \tag{6}$$

It is clear that $U = XV\Sigma^{-1}$.

Use $\tilde{U}$ to approximate $U$.

$$\begin{aligned} \tilde{U} :&= XV_k\Sigma_k^{-1} \\ &= XX_k^\top U_k\Sigma_k^{-1}\Sigma_k^{-1} \\ &= XX_k^\top U_k\Sigma_k^{-2} \end{aligned} \tag{7}$$

Define $\tilde{G} := \tilde{U}\Sigma_k^2\tilde{U}^\top$

$$\begin{aligned} \tilde{G} &= XX_k^\top U_k\Sigma_k^{-2}\Sigma_k^2\Sigma_k^{-2}U_k^\top X_kX^\top \\ &= \left(XX_k^\top\right)\left(U_k\Sigma_k^{-2}U_k^\top\right)\left(XX_k^\top\right)^\top \\ &= \left(XX_k^\top\right)\left(W_k^{-1}\right)\left(XX_k^\top\right)^\top \\ &= C_{n\times k}W_k^+C_{k\times n}^\top \end{aligned} \tag{8}$$

□

## 3.2 Error Bound

According to the proof of Theorem 3, the approximation occurs in equation 7. Theorem 3 suggests one possible approximation. Theorem 4 shows that the error can be bounded with probability.

**Theorem 4.** *(Error Bound of Standard Nyström Method)*
*Let* $\widetilde{\mathbf{K}}$ *denote the rank- k Nyström approximation of* $\mathbf{K}$ *based on m 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 m :*

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

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

*where* $\beta(m,n) = 1-\frac{1}{2\max\{m,n-m\}}$*,* $K_{max} = max_i K_{ii}$*, and* $d_k$ *max the distance* $max_{ij} = \sqrt{K_{ii}+K_{jj}-2K_{ij}}$*.*

The proof of Theorem 4 can be found in reference [4].

As can be seen, although Nyström Methodis not the best rank-k approximation solution, it can be bounded witih probability at least $1-\delta$, which ensures the accuracy of Nyström Methodwhen there is low rank structure originally of the matrix.

## 3.3 Drawbacks of Nyström Method

Although Nyström Method includes a series of good solution to low rank matrix approximation problem and run faster than SVD, it may not perform well for kernel matrices can be dense and approximately full rank.

However, kernel matrices are not always low-rank. Set Gaussian kernel as an example, where $K(x_i,x_j) = e^{-\gamma\|x_i-x_j\|_2^2}$. $\gamma$ is the scale parameter here. It influences the rank structure of kernel matrix $G$. If $\gamma \to 0$, then $G \to ee^T$, $e \in \mathbb{R}^n$. In this case, $G$ has a low rank structure. While $\gamma \to \infty$, $G \to$ an identity matrix with full rank. $G$ has a block-clustering structure.

Nyström Method performs well only when $G$ has a low rank structure ($\gamma$ is small). In the other case, even the SVD best rank-k cannot approximate well enough.

# 4 BKA

Block Kernel Approximation (BKA) is one of the solution to $\gamma \to \infty$, Considering the discussion of kernel matrix rank discussed in subsection 3.3. BKA split the data (size $n$) in $c$ clusters. BKA approximates the kernel matrix as:

$$G \approx \tilde{G} \equiv \begin{bmatrix} G^{(1,1)} & 0 & \dots & 0 \\ 0 & G^{(2,2)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & G^{(c,c)} \end{bmatrix} \tag{9}$$

Off-diagonal matrices are ignored. BKA only considers the diagonal blocks. BKA is a good choice when the kernel matrix has a block structure. But in other cases, $\tilde{G}$ is far from $G$ because the ignorance of off-diagonal blocks. It should be noticed that the computation and storage of all diagonal blocks are also expensive.

## 4.1 Relation to K-Means Clustering

Error of BKA can be measured by

$$\|\tilde{G} - G\|_F^2 = \sum_{i,j} K\left(x_i, x_j\right)^2 - \sum_{s=1}^{c} \sum_{i,j \in \mathcal{V}_s} K\left(x_i, x_j\right)^2 \tag{10}$$

where $\mathcal{V}_s$ is a subset of $\{1, 2, ..., n\}$ and $\mathcal{V}_1, \mathcal{V}_2, ..., \mathcal{V}_c$ is a partition of data.

To minimize the approximation error of BKA, $\|\tilde{G} - G\|_F^2$ should be minimized. Once the kernel function is specified, $\sum_{i,j} K\left(x_i, x_j\right)^2$ is a constant. So minimizing the error equals to maximizing $\sum_{s=1}^{c} \sum_{i,j \in \mathcal{V}_s} K\left(x_i, x_j\right)^2$.

If $\sum_{s=1}^{c} \sum_{i,j \in \mathcal{V}_s} K\left(x_i, x_j\right)^2$ is set a the goal to be maximized, all data will be grouped as one cluster inevitably. To balance the group size influence, the improved goal to be maximized is

$$D^{kernel}\left(\{\mathcal{V}_s\}_{s=1}^{c}\right) = \sum_{s=1}^{c} \frac{1}{|\mathcal{V}_s|} \sum_{i,j \in \mathcal{V}_s} K\left(x_i, x_j\right)^2 \tag{11}$$

Theorem 5 shows that for shift-invariant kernel matrices, after performing k-means clustering and rearranging data by k-means clusters, the maximum of $\sum_{s=1}^{c}\sum_{i,j\in\mathcal{V}_s} K\left(x_i,x_j\right)^2$ is achieved.

**Shift-Invariant Kernel Matrix** A kernel function $K(x_i,x_j)$ is shift-invariant if the kernel value depends only on $x_i-x_j$, which can be written as $K(x_i,x_j)=f(\eta(x_i-x_j))$. Note $u$ as the unit vector in the direction of $x_i-x_j$, then we have $K(x_i,x_j)=f(\eta(x_i-x_j))=g_u(\eta\|x_i-x_j\|)$. The notation is used again later.

**Theorem 5.** *For any shift-invariant kernel that satisfies that $g_u(t)$ is differentiable for all $t\neq 0$*

$$D^{kernel}\left(\{\mathcal{V}_s\}_{s=1}^{c}\right)\geq \bar{C}-\eta^2R^2D^{kmeans}\left(\{\mathcal{V}_s\}_{s=1}^{c}\right)$$

*where $\bar{C}=\frac{nf(0)^2}{2}$, $R$ is a constant depending on the kernel function, and $D^{kmeans}\equiv\sum_{s=1}^{c}\sum_{i\in\mathcal{V}_s}\|x_i-m_s\|_2^2$ is the k-means objective function, where $m_s=\left(\sum_{i\in\mathcal{V}_s}x_i\right)/|\mathcal{V}_s|$, $s=1,\cdots,c$, are the cluster centers.*

*Proof.* By the mean value theorem,

$$\begin{aligned} K\left(x_i,x_j\right) &= g_u\left(\eta\left\|x_i-x_j\right\|_2\right)\\ &= g_u(0)+\eta g_u'(s)\left\|x_i-x_j\right\|_2 \end{aligned} \tag{12}$$

where $s\in\left(0,\left\|x_i-x_j\right\|_2\right)$. Notice that $f(0)=g_u(0)$ by definition. Define $R:=\sup_{\theta\in\mathbb{R},\|v\|=1}|g_v'(\theta)|$

$$\begin{aligned} f(0) &\leqslant K\left(x_i,x_j\right)+\eta R\left\|x_i-x_j\right\|_2\\ f^2(0) &\leqslant K\left(x_i,x_j\right)^2+\eta^2R^2\left\|x_i-x_j\right\|_2^2+2\eta RK\left(x_i,x_j\right)\left\|x_i-x_j\right\|_2\\ f^2(0) &\leqslant 2K\left(x_i,x_j\right)^2+2\eta^2R^2\left\|x_i-x_j\right\|_2^2\\ K\left(x_i,x_j\right)^2 &\geqslant \frac{1}{2}f^2(0)-\eta^2R^2\left\|x_i-x_j\right\|_2^2 \end{aligned} \tag{13}$$

Plug it into $D^{kernel}\left(\{\mathcal{V}_s\}_{s=1}^c\right)$, we have

$$\begin{aligned} D^{\text{kernel}}\left(\{V_s\}_{s=1}^c\right) &= \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{i,j \in v_s} K\left(x_i, x_j\right)^2 \\ &\geqslant \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{1,j \in V_s} \left(\frac{1}{2} f^2(0) - \eta^2 R^2 \left\|x_i - x_j\right\|_2^2\right) \\ &= \frac{1}{2} f^2(0) \sum_{s=1}^{c} \frac{1}{\|V_s\|} \left(|V_s|\left(|V_s| - 1\right)\right) - \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{1,j \in V_s} \eta^2 R^2 \left\|x_i - x_j\right\|_2^2 \\ &= \frac{n}{2} f^2(0) - \eta^2 R^2 \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{1,j \in V_s} \left\|x_i - x_j\right\|_2^2 \end{aligned} \tag{14}$$

On the other hand, according to definition, we have

$$\begin{aligned} D^{\text{kmeans}}\left(\{V_s\}_{s=1}^c\right) &= \sum_{s=1}^{c} \sum_{i \in V_s} \left\|x_i - m_s\right\|_2^2 \\ &= \sum_{s=1}^{c} \sum_{i \in V_s} \left\| x_i - \frac{1}{|V_s|} \sum_{j \in V_s} x_j \right\|_2^2 \\ &= \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{i \in V_s} \left\| \sum_{j \in V_s} \left(x_i - x_j\right) \right\|_2^2 \\ &= \sum_{s=1}^{c} \frac{1}{|V_s|} \sum_{i,j \in V_s} \left\|x_i - x_j\right\|_2^2 \end{aligned} \tag{15}$$

Combine the inequation 14 and equation 15, the theorem is proven. □

Theorem 5 indicates that after clustering by k-means, $D^{\text{kernel}}\left(\{V_s\}_{s=1}^c\right)$ gets a higher lower bound. K-means is helpful to perform BKA.

## 4.2 Low Rank Structure

It has been proved that after performing k-means, $D^{\text{kmeans}}\left(\{V_s\}_{s=1}^c\right)$, thus $D^{\text{kernel}}\left(\{V_s\}_{s=1}^c\right)$ gets a higher lower bound. In fact, all blocks have low rank structure after k-means clustering, both off-diagonal blocks and diagonal blocks.

Before proving the above result, $\epsilon$-net theorem is introduced. $\epsilon$-net can also be called is $\epsilon$-covering.

Define $\mathcal{N}(S, \eta)$ as the smallest possible cardinality of an $\eta$-covering of $S$, and $\varepsilon_k(S)$ as the

minimum radius of balls to cover points in $S$.

$$\begin{aligned}&\mathcal{N}(S,\eta) = inf\{Card(K) : K \text{ is a } \epsilon - covering\}\\&\varepsilon_k(S) = \inf\{\varepsilon > 0 : \exists \text{ closed balls } D_1,\ldots,D_k \text{ with radius } \varepsilon \text{ covering } S\}\end{aligned} \tag{16}$$

Notice that $\varepsilon_k(S) \leq \eta \Longleftrightarrow \mathcal{N}(S,\eta) \leq k$

**Lemma 1.** *Define*

$$\varphi_k(S) = \sup\{\delta > 0 \mid \exists x_1,\ldots,x_{k+1} \in S \text{ s.t. for } i \neq j, d(x_i,x_j) > 2\delta\} \tag{17}$$

*then for all* $\varphi_k(S) \leq \varepsilon_k(S) \leq 2\varphi_k(S)$

It should be noticed that $\varphi_k(S)$ is actually the maximum radius of packing with $k$ balls. Then by definition of covering and packing, the lemma is clear.

**Lemma 2.** *$B_R \in \mathbb{R}^d$ is the ball with zero center and $R$ radius. Then for all $k \geq 1, k^{-\frac{1}{d}} \leq \varepsilon_k(B_1) \leq 4(k+1)^{-\frac{1}{d}}$*

*Proof.* By definition of packing, $\varphi_k(B_1) \leq 1$. Let $\rho < \varphi_k(B_1)$, there exists $x_1,\ldots,x_{k+1} \in S$ s.t. for $i \neq j, d(x_i,x_j) > 2\rho$ then $\varphi_k(B_1)$ can be understood as the supremum of $\rho$. Let $D_j = \rho B_1 + x_j,\ j = 1,2,...,k+1$, then

$$\|x\| \leq \|x - x_j\| + \|x_j\| \leq \rho + 1 < 2. \tag{18}$$

So $D_j \subseteq B_2$. Define a measure $v$ in $\mathbb{R}^d$ which satistifies $\nu(\lambda B) = \lambda^d \nu(B)$ for all measurable set $B \in \mathbb{R}^d$. Consider that $D_j \subseteq B_2$ and $D_i \cap D_j = \emptyset, i,j = 1,...,k+1$, we have

$$\begin{aligned}&\sum_{i=1}^{k+1} \nu(D_i) \leq \nu(B_2) \Rightarrow \sum_{i=1}^{k+1} \rho^N \nu(B_1) \leq 2^N \nu(B_1)\\\Rightarrow\quad &(k+1)\rho^N \leq 2^N \Rightarrow \rho \leq 2(k+1)^{-\frac{1}{N}}\end{aligned} \tag{19}$$

Any $\rho$ satistifies $\rho \leq 2(k+1)^{-\frac{1}{N}}$, so the supremum satistifies the inequation as well. Thus $\varepsilon_k(B_1) \leq 2\varphi_k(S) \leq 4(k+1)^{-\frac{1}{N}}$. Similarly, the other part of lemma is proven. More details can be found in reference [1] □

**Theorem 6.** *($\epsilon$-net Theorem)* $\ln \mathcal{N}(B_R,\eta) \leq d \ln\left(\frac{4R}{\eta}\right)$

*Proof.* Let $k = \lceil \left(\frac{4R}{\eta}\right)^N - 1 \rceil$. Then

$$k+1 \geq \left(\frac{4R}{\eta}\right)^N \Rightarrow \frac{\eta}{R} \geq 4(k+1)^{-\frac{1}{N}} \tag{20}$$

By lemma 2, $\varepsilon_k(B_1) \leq 4(k+1)^{-\frac{1}{d}}$. Combine it with inequation 20, we have

$$\begin{aligned} \varepsilon_k(B_1) \leq \frac{\eta}{R} &\Rightarrow \varepsilon_k(B_R) \leq \eta \\ &\mathcal{N}(B_R, \eta) \leq k \end{aligned} \tag{21}$$

The last transformation is done because $\varepsilon_k(S) \leq \eta \Longleftrightarrow \mathcal{N}(S, \eta) \leq k$ What is more, we have $k \leq (\frac{4R}{\eta})^d$. Combine it with inequation 21, $\epsilon$-net theorem is proven. □

**Theorem 7.** *Given data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$, and a partition $\mathcal{V}_1, \ldots, \mathcal{V}_c$, and assume $f$ is Lipschitz continuous, then for any $s, t (s = t \text{ or } s \neq t)$*

$$\left\| G^{(s,t)} - G_k^{(s,t)} \right\|_F \leq 4Ck^{-1/d} \sqrt{|\mathcal{V}_s|\,|\mathcal{V}_t|} \min(r_s, r_t),$$

*where $G_k^{(s,t)}$ is the best rank-k approximation to $G^{(s,t)}$; $C$ is the Lipschitz constant of the shift-invariant function $f$; $r_s$ is the radius of the s-th cluster.*

*Proof.* In s-th cluster: $x_1, ..., x_{n_s}$, t-th cluster: $y_1, ..., y_{n_y}$

Suppose the radius of t-th cluster is $r_t$. Apply $\epsilon$-net theorem 6 for t-th cluster, we can find $k$ balls with radius $\hat{r} = k^{-1/d} 4 r_t$ to cover $\{y_j\}_{j=1}^{n_t}$.

Assume centers of the balls for t-th cluster are $m_1, m_2, ..., m_k$. Define

$$\bar{G}^{(s,t)} = \bar{U}\bar{V}^T \tag{22}$$

$$\begin{cases} \bar{U}_{i,q} = K(x_i, m_q) \\ \bar{V}_{j,q} = I\{y_j \in Ball(m_q)\} \end{cases} \tag{23}$$

$\bar{G}^{(s,t)}$ has a low rank structure for $rank(\bar{G}^{(s,t)}) \leq rank(\bar{U}) \leq k$.

Note $\left(G^{(s,t)}\right)^*$ as the best rank-k approximation, then

$$\begin{aligned} \left\| G^{(s,t)} - \left(G^{(s,t)}\right)^* \right\|_F &\leqslant \left\| G^{(s,t)} - \bar{G}^{(s,t)} \right\|_F \\ &\leqslant \sqrt{\sum_{i,j} \left( G_{ij}^{(s,t)} - \bar{G}_{i,j}^{(s,t)} \right)^2} \\ &\leqslant \sqrt{n_s n_t c^2 \left( k^{-\frac{1}{d}} 4 r_t \right)^2} \\ &= C k^{-\frac{1}{d}} 4 r_t \sqrt{n_s n_t} \end{aligned} \tag{24}$$

Similarly, apply $\epsilon$-net theorem 6 for s-th cluster, we have both

$$\begin{cases} \left\| G^{(s,t)} - \left(G^{(s,t)}\right)^* \right\|_F \leqslant C \cdot R^{-\frac{1}{d}} 4 r_t \sqrt{n_s n_t} \\ \left\| G^{(s,t)} - \left(G^{(s,t)}\right)^* \right\|_F \leqslant C \cdot k^{-\frac{1}{d}} 4 r_s \sqrt{n_s n_t} \end{cases} \tag{25}$$

$$\Rightarrow \left\| G^{(s,t)} - \left(G^{(s,t)}\right)^* \right\|_F \leqslant C \cdot k^{-\frac{1}{d}} 4 \sqrt{n_s n_t} min(r_s, r_t). \tag{26}$$

The theorem has been proven. □

The difference between best rank-k approximation matrix and the original matrix can be bounded by $C \cdot k^{-\frac{1}{d}} 4 \sqrt{n_s n_t} min(r_s, r_t)$, where $s = t$ or $s \neq t$. The conditions of the theorem are well satisfied for most of kernel functions are Lipschitz continuous.

Theorem 7 suggests that each block(diagonal or off-diagonal block) of the kernel matrix will be low rank if we find the partition by k-means and the radius of the cluster is small. Inspired by this theorem, MEKA in Section 5 is raised.

# 5 MEKA

In the previous pages, several common method of low rank kernel matrix approximation have been discussed. SVD yeilds the best rank-k approximation with heavy storage and computation burden. Nyström Methodapproximate the result of SVD with less computation but only works with matrix with original low rank structure. The condition is not always satisfied in when it comes to kernel matrix. BKA performs well for block structured matrix, which is not always satisfied as well. But after k-means clustering, diagonal block error decrease and all blocks have low rank structure.

Memory Efficient Kernel Approximation (MEKA) is proposed in reference [6] to take advantages of Nyström Methodand BKA with more storage and computation efficiency.

## 5.1 Steps of MEKA

MEKA first performs BKA, with k-means clustering for less diagonal error and low rank structure for all blocks. For each diagonal blocks, Nyström Methodis applied. Different from BKA, off-diagonal blocks are not ignored. Those off-diagonal blocks do not use Nyström Methodseperately for efficiency. Instead, a link matrix $L^{(s,t)}$ is computated to link $G^{(s,t)}$ with

$G^{(s,s)}$ and $G^{(t,t)}$. Low rank structure of all blocks guarantees the accuracy of Nyström Method, the scale of each problem is also much smaller. Reference [6] also show that the principal angles between the dominant singular subspace of a diagonal block $G^{(s,s)}$ and that of an off-diagonal block $G^{(s,t)}$ for different ranks k are similar. There is substantial overlap between the dominant singular subspaces of the diagonal and off-diagonal block. So the idea of link matrix makes sense.

**Algorithm 1:** Memory Efficient Kernel Approximation (MEKA)

**Input :** Data points $(xi)_{i=1}^{n}$, scaling parameter $\gamma$, rank $k$, and no. of clusters $c$.

**Output:** The rank-ck approximation $G = WLW^T$. Generate the partition $V_1, ..., V_c$ by k-means;

1 **for** $s = 1, ..., c$ **do**

2 Perform the rank-k approximation $G^{(s,s)} \approx W^{(s)} L^{(s,s)} \left(W^{(s)}\right)^T$ by standard Nyström Method

3 **end for**

4 **foreach** $(s,t), s \neq t$ **do**

5 Sample a submatrix $\bar{G}^{(s,t)}$ from $G^{(s,t)}$ with row index set $v_s$ and column index set $v_t$;

6 Form $W_{v_s}^{(s)}$ by selecting the rows in $W^{(s)}$ according to index set $v_s$;

7 Form $W_{v_t}^{(t)}$ by selecting the rows in $W^{(t)}$ according to index set $v_t$;

8 Solve the least squares problem: $\bar{G}^{(s,t)} \approx W_{v_s}^{(s)} L^{(s,t)} \left(W_{v_t}^{(t)}\right)^T$ to get $L^{(s,t)}$.

9 **end foreach**

**Compute $W^{(s)}$** Since we aim to deal with dense kernel matrices of huge size, we use the standard Nyström Methodapproximation to compute low-rank basis for each diagonal block. By theorem 3, we have $G^{(s,s)} \approx W^{(s)} L^{(s,s)} \left(W^{(s)}\right)^T$.

**Compute $L^{(s,t)}$** We approximate $G^{(s,t)}$ by $W^{(s)} L^{(s,t)} \left(W^{(t)}\right)^T$. Intuitively, the goal should be minimizing $\left\| G^{(s,t)} - W^{(s)} L^{(s,t)} \left(W^{(t)}\right)^T \right\|_F$. However, the scale of the problem is too large, the idea of sampling is applied again here to reduce problem size.

**Choose $k_s$ for Each Cluster** The partition of data to $c$ clusters are done by k-means. $k_s$ means we do rank-$k_s$ approximation for diagonal block $G^{(s,s)}$. For simplicity, we set $k_s = k, \forall s = 1, ..., c$ in practice. However, the method based on eigenvalue shows good theoretical property. Suppose that data has been standarized, then the steps are as follows.

1. Do reduced eigenvalue decomposition for each diagonal blocks. Get top-2k eigenvalue of each blocks.

2. Rank all $2ck$ eigenvalues, select top-ck ones. The number of selected eigenvalues are $k_s$

Here, only top-2k eigenvalues are computed instead of all eigenvalue for efficiency consideration.

**Theorem 8.** *Let $\Delta$ denote a matrix consisting of all off-diagonal blocks of $G$, so $\Delta^{(s,t)} = G^{(s,t)}$ for $s \neq t$ and all zeros when $s = t$. We sample cm points from the dataset uniformly at random without replacement and split them according to the partition from k-means, such that each cluster has $m_s$ benchmark points and $\sum_{s=1}^{c} m_s = \text{cm}$. Let $G_{ck}$ be the best rank-ck approximation of $G$, and $\tilde{G}$ be the rank-ck approximation from MEKA. Suppose we choose the rank $k_s$ for each diagonal block using the eigenvalue based approach as mentioned in Section 5.1, then with probability at least $1 - \delta$, the following inequalities hold for any sample of size* cm *:*

$$\begin{aligned} \|G - \tilde{G}\|_2 &\leq \|G - G_{ck}\|_2 + \frac{1}{\sqrt{c}} \frac{2n}{\sqrt{m}} G_{\max}(1 + \theta) + 2\|\Delta\|_2 \\ \|G - \tilde{G}\|_F &\leq \|G - G_{ck}\|_F + \left(\frac{64k}{m}\right)^{\frac{1}{4}} n G_{\max}(1 + \theta)^{\frac{1}{2}} + 2\|\Delta\|_F \end{aligned} \tag{27}$$

*where $\theta = \sqrt{\frac{n-m}{n-0.5} \frac{1}{\beta(m,n)} \log \frac{1}{\delta}} d^G_{\max} / G^{\frac{1}{2}}_{\max}$; $\beta(m,n) = 1 - \frac{1}{2\max\{m, n-m\}}$; $G_{\max} = \max_i G_{ii}$; and $d^G_{\max}$ represents the distance $\max_{ij} \sqrt{G_{ii} + G_{jj} - 2G_{ij}}$.*

The bound is similar to Nyström Method. The theorem ensure the accuracy of MEKA.

## 5.2 Time Complexity and Storage Usage

MEKA is not only accurate, but also computation and storage efficient, which is an important advantage. The used storage and time complexity is listed here as Table 1 for completeness of the report. Details of the complexity can also be found in reference [6]

Table 1: Time Complexity of Low Rank Approximation Method

| Method | Storage | Rank | Time Complexity |
|---|---|---|---|
| Nyström Method | $O(cnk)$ | $ck$ | $O\left(cnm(ck+d)+(cm)^3\right)$ |
| SVD | $O(cnk)$ | $ck$ | $O\left(n^3+n^2d\right)$ |
| MEKA | $O\left(nk+(ck)^2\right)$ | $ck$ | $O\left(nm(k+d)+cm^3+T_L+T_C\right)$ |

# 6 Summary & Discussion

In this report, some common low rank kernel matrix approximation methods are studied. Kernel matrix is the object of approximation for its importance and universality. SVD is the theoretical best rank-k approximation with low efficiency. Nyström Method is a widely used method to make up for SVD's cost with sacrifice of accuracy in the case of kernel matrix without low rank structure. BKA aims to settle problems with block structured kernel matrix. K-means clustering plays an important part in BKA to ensure low approximation error. Finally, MEKA is briefly introduced as a combination of Nyström Method and BKA. MEKA considers both original low rank structure and block structure. The time and storage efficiency is another superiority, but not emphasized in this report.

There are many methods to do low rank approximation. One key idea is sampling. Special structures of matrix should also be considered. Both Nyström Methodand BKA suffer a loss from only considering low rank structure or block structure. MEKA outperforms because of detailed consideration of structures. Another inspiring idea is preprocessing data to form better property, such as k-means clustering here.

The report is the extension of the topics: *low rank approximation* and *covering and packing* discussed in class.

# References


[1] Felipe Cucker and Steve Smale. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39:1–49, 2002.

[2] L. (Ed.) Hogben. *Handbook of Linear Algebra (2nd ed.)*. Chapman and Hall/CRC., 2013.

[3] Hopcroft and Kannan. Lecture notes (forthcoming book). [EB/OL], 2012. https://www.cs.princeton.edu/courses/archive/spring12/cos598C/svdchapter.pdf.

[4] Sanjiv Kumar, Mehryar Mohri, and Ameet Talwalkar. Ensemble nystrom method. In *Advances in Neural Information Processing Systems 22 - Proceedings of the 2009 Conference*, Advances in Neural Information Processing Systems 22 - Proceedings of the 2009 Conference, pages 1060–1068, 2009. 23rd Annual Conference on Neural Information Processing Systems, NIPS 2009 ; Conference date: 07-12-2009 Through 10-12-2009.

[5] Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[6] Si Si, Cho-Jui Hsieh, and Inderjit S. Dhillon. Memory efficient kernel approximation. In *International Conference on Machine Learning (ICML)*, jun 2014.

[7] Christopher K. I. Williams and Matthias W. Seeger. Using the nyström method to speed up kernel machines. In *NIPS*, 2000.