# 7. Appendix

## 7.1. Proof of lower approximation error

**Theorem 1.** *Using Algorithm 1 to generate $m$ landmark points, we can guarantee that the approximation quality will become better than the traditional Nyström approximation with initial $s$ landmark points:*

$$\|G - \bar{G}\| \leq \|G - \tilde{G}\|, \tag{15}$$

*where $\tilde{G}$ and $\bar{G}$ are the approximation of $G$ from standard Nyström and Algorithm 1 respectively.*

*Proof.* Let us first compare our method with standard Nyström. The generalization to other sampling strategies based Nyström is straight forward. Let $G$ denote the kernel matrix form on the $n$ data points, and suppose $s$ landmark points $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_s$ are selected uniformly at random from the data. Let us define the sampling matrix $S \in R^{n\times s}$ to be a zero-one matrix where $S_{ij} = 1$ if $i$-th sample in the dataset is selected as landmark point. $C$ is a $n \times s$ matrix consisting of the corresponding $s$ columns selected from $G$ and $W$ consists of the kernel matrix formed by these $s$ landmark points. So by standard Nyström, $\tilde{G} = CW^+G^T$, $C = GS$ and $W = S^T GS$.

Using $\boldsymbol{m}_1, \cdots, \boldsymbol{m}_s$ as initial landmark points in Algorithm 1, after fast transforms, we totally have $m = sd$ landmark points $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_m$, of which the last $s$ points are the original landmark points and the rest $m - s$ are new landmark points. Assume the new kernel matrix $G_H$ is the kernel matrix on the union of the original $n$ data points and $m - s$ new added landmark points. So $G$ is a block in $G_H$. Similarly we define $S_H$, $C_H$, and $W_H$ as sampling matrix, $m$ sampled columns in $G_H$ and kernel matrix formed by $m$ landmark points respectively. So $C_H = G_H S_H$ and $W_H = S_H^T G_H S_H$. Let the decomposition of $G_H$ be $G_H = L_H^T L_H$. So

$$G_H = L_H^T L_H = [\begin{array}{c} \bar{L}^T \\ L^T \end{array}][\begin{array}{cc} \bar{L} & L \end{array}] = [\begin{array}{cc} \bar{L}^T\bar{L} & \bar{L}^T L \\ L^T\bar{L} & L^T L \end{array}]. \tag{16}$$

Since $G$ is a block in $G_H$, the decomposition of $G$ is $L^T L$.

Since $C_H = G_H S_H = L_H^T L_H S_H$ and let the singular value decomposition of $L_H S_H$ be $U_H \Sigma_H V_H^T$, $C_H = L_H^T U_H \Sigma_H V_H^T$. Also we have

$$W_H = S_H^T G_H S_H \quad = S_H^T L_H^T L_H S_H = V_H \Sigma_H^2 V_H^T. \tag{17}$$

The Nyström approximation on $G_H$ is written as

$$\begin{aligned} G_H &= C_H W_H^+ C_H^T \\ &= L_H^T U_H \Sigma_H V_H^T V_H \Sigma_H^{-2} V_H^T V_H \Sigma_H U_H^T L_H \\ &= L_H^T U_H U_H^T L_H. \end{aligned} \tag{18}$$

So we have

$$\begin{aligned} G_H - C_H W_H^+ C_H^T &= L_H^T L_H - L_H^T U_H U_H^T L_H \\ &= (L_H - U_H U_H^T L_H)^T (L_H - U_H U_H^T L_H). \end{aligned} \tag{19}$$

The Nyström approximation error on the original $n$ data points or $G$ part is

$$\begin{aligned} (G_H - C_H W_H^+ C_H^T)_G &= L^T L - L^T U_H U_H^T L \\ &= (L - U_H U_H^T L)^T (L - U_H U_H^T L). \end{aligned} \tag{20}$$

According to Lemma 1 in (Drineas & Mahoney, 2005), we have the standard Nyström approximation on $G$ as

$$\begin{aligned} G - CW^+C^T &= L^T L - L^T U U^T L \\ &= (L - UU^T L)^T (L - UU^T L). \end{aligned} \tag{21}$$

where $LS$'s SVD is $U\Sigma V^T$.

Since $U$ is the basis for the range space of $LS$ and $U_H$ is the basis for the range space of $L_H S_H$, so $range(U) \subseteq range(U_H)$. According to the proposition 8.5 in (Halko et al., 2011), we have

$$\|L - U_H U_H^T L\|_2 \leq \|L - UU^T L\|_2, \tag{22}$$

so

$$\|(G_H - C_H W_H^+ C_H^T)_G\| \leq \|G - CW^+C^T\|, \tag{23}$$

or

$$\|G - \bar{G}\| \leq \|G - \tilde{G}\|. \tag{24}$$

□

## 7.2. Lemma 1

**Lemma 1.** *If the kernel function can be written as* (3)*, assume the maximum distance between the samples and the original point is a bounded number $R$, and $f, g$ are differentiable, then*

$$K(\boldsymbol{a}, \boldsymbol{b})^2 - K(\boldsymbol{c}, \boldsymbol{d})^2 \leq \eta(\|\boldsymbol{a} - \boldsymbol{c}\|^2 + \|\boldsymbol{b} - \boldsymbol{d}\|^2) \tag{25}$$

*for any $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d} \in \mathbb{R}^d$, where*

$$\eta = 4M_f^4 L_g^2 R^2 + 4M_f^2 M_g^2 L_f^2,$$

*where $M_f = \max_{\|\boldsymbol{x}\| \leq R} |f(\boldsymbol{x})|$, $M_g = \max_{\|\boldsymbol{u}\| \leq R} |g(\boldsymbol{u})|$, $L_f = \max_{\|\boldsymbol{x}\| \leq R} |f'(\boldsymbol{x})|$, $L_g = \max_{\|\boldsymbol{u}\| \leq R} |g'(\boldsymbol{u})|$.*

*Proof.* For any $\boldsymbol{a},\boldsymbol{b},\boldsymbol{c},\boldsymbol{d}\in\mathbb{R}^d$, we have

$$\begin{aligned}
&(K(\boldsymbol{a},\boldsymbol{b})-K(\boldsymbol{c},\boldsymbol{d}))^2\\
=&\Big(f(\boldsymbol{a})f(\boldsymbol{b})g(\boldsymbol{a}^T\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})g(\boldsymbol{c}^T\boldsymbol{d})\Big)^2\\
=&\Big((f(\boldsymbol{a})f(\boldsymbol{b})g(\boldsymbol{a}^T\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})g(\boldsymbol{a}^T\boldsymbol{b}))\\
&+(f(\boldsymbol{c})f(\boldsymbol{d})g(\boldsymbol{a}^T\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})g(\boldsymbol{c}^T\boldsymbol{d}))\Big)^2\\
\le&2\Big(g(\boldsymbol{a}^T\boldsymbol{b})(f(\boldsymbol{a})f(\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d}))\Big)^2\\
&+2\Big(f(\boldsymbol{c})f(\boldsymbol{d})(g(\boldsymbol{a}^T\boldsymbol{b})-g(\boldsymbol{c}^T\boldsymbol{d}))\Big)^2\\
\le&2M_g^2\Big(f(\boldsymbol{a})f(\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})\Big)^2\\
&+2M_f^4\Big(g(\boldsymbol{a}^T\boldsymbol{b})-g(\boldsymbol{c}^T\boldsymbol{d})\Big)^2.
\end{aligned}$$

We can then bound each term by

$$\begin{aligned}
&\Big(f(\boldsymbol{a})f(\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})\Big)^2\\
\le&\Big(f(\boldsymbol{a})f(\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{b})+f(\boldsymbol{c})f(\boldsymbol{b})-f(\boldsymbol{c})f(\boldsymbol{d})\Big)^2\\
\le&2(f(\boldsymbol{a})-f(\boldsymbol{c}))^2f(\boldsymbol{b})^2+2(f(\boldsymbol{b})-f(\boldsymbol{d}))^2f(\boldsymbol{c})^2\\
\le&2M_f^2\Big((f(\boldsymbol{a})-f(\boldsymbol{c}))^2+(f(\boldsymbol{b})-f(\boldsymbol{d}))^2\Big)\\
=&2M_f^2\Big(f'(\xi_1)^2\|\boldsymbol{a}-\boldsymbol{c}\|^2+f'(\xi_2)^2\|\boldsymbol{b}-\boldsymbol{d}\|^2\Big)\\
\le&2M_f^2L_f^2(\|\boldsymbol{a}-\boldsymbol{c}\|^2+\|\boldsymbol{b}-\boldsymbol{d}\|^2)
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
&(g(\boldsymbol{a}^T\boldsymbol{b})-g(\boldsymbol{c}^T\boldsymbol{d}))^2\\
=&(g'(\xi)(\boldsymbol{a}^T\boldsymbol{b}-\boldsymbol{c}^T\boldsymbol{d}))^2\\
\le&L_g^2(\boldsymbol{a}^T\boldsymbol{b}-\boldsymbol{c}^T\boldsymbol{b}+\boldsymbol{c}^T\boldsymbol{b}-\boldsymbol{c}^T\boldsymbol{d})^2\\
=&L_g^2((\boldsymbol{a}-\boldsymbol{c})^T\boldsymbol{b}+(\boldsymbol{b}-\boldsymbol{d})^T\boldsymbol{c})^2\\
\le&2L_g^2(\|(\boldsymbol{a}-\boldsymbol{c})^T\boldsymbol{b}\|^2+2\|(\boldsymbol{b}-\boldsymbol{d})^T\boldsymbol{c}\|^2)\\
\le&2L_g^2R^2(\|\boldsymbol{a}-\boldsymbol{c}\|^2+\|\boldsymbol{b}-\boldsymbol{d}\|^2)
\end{aligned}$$

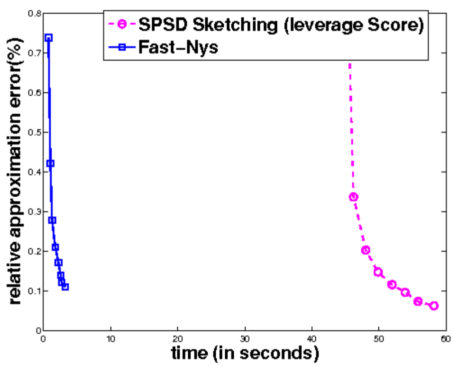
This proves (25). □

### 7.3. Parameters for the experimental results

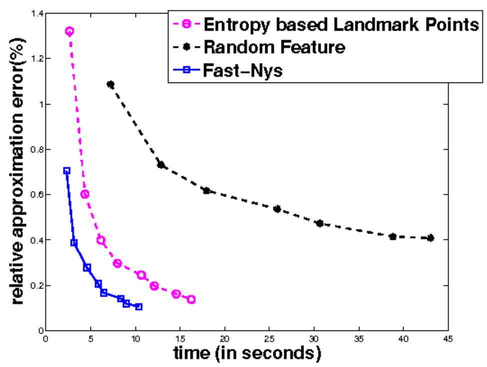
- All the experiments were conducted on a machine with an Intel Xeon X5440 2.83GHz CPU and 32G RAM. We tried to have the best implementation for each algorithm. Fast-Nys, DC-Pred++, Nys, KNys, RKS, Fastfood are all implemented in C sharing the same modules. LDKL is the highly optimized C++ implementation published along with the original paper (Jose et al., 2013).
- The degree for the polynomial kernel and homogeneous kernel is set to be 3.
- We do data normalization with mean to be 0 and variance to be 1 before running our algorithms.
- When working on fast prediction experiments, we first form the low-rank approximation for the kernel matrix and apply liblinear to perform the classification.
- For fast prediction parameters($\gamma$ is the width parameters for Gaussian kernel and $C$ is the regularization term in Liblinear SVM):
  - cifar: $\gamma=2^{-10}$,$C=64$;
  - mnist: $\gamma=2^{-10}$,$C=128$;
  - a9a: $C=32$;
- For kernel approximation:
  - magic: $\gamma=0.01$
  - ijcnn: $\gamma=0.01$
  - webspam: $\gamma=1$
- When working on prediction, we use random samples as the initial landmarks for Fast-Nys. The number of initial landmarks ranges from 2 to 10.
- When using kmeans Nyström, we randomly sample 10000 data samples to perform clustering.
- For LDKL, for a fair comparison, we disable the SSD operation.
- We use an alternating minimization algorithm to find the seeds in our algorithm. The algorithm usually converges to a reasonably good solution in 10 iterations, so we fix the number of iterations to be 10 for all the experiments. For example, on MNIST dataset with k=10, the initial objective function value (using random samples) is 1750260, after 10 iterations it drops to 90041, and the converged solution has objective function value 89872.

### 7.4. Comparison with other kernel approximation methods

We show the comparison between fast-Nys with leverage score (Gittens & Mahoney, 2013b) and entropy based landmark points (Brabanter et al., 2010) in Nyström approximation and random feature (Rahimi & Recht, 2007).




(a) MNIST, Gaussian




(b) CIFAR, Gaussian

*Figure 5.* Low-rank kernel approximation results. $x$-axis is the time and $y$ axis shows the relative kernel approximation error. Methods with approximation error above the top of $y$-axis are not shown. (a) compares Fast-Nys with sampling landmark points based on leverage score (Gittens & Mahoney, 2013a). Since this method needs to compute the entire kernel, it is much slower than our method. (b) compares Fast-Nys with entropy based landmark points based Nyström approximation (Brabanter et al., 2010) and random feature (Rahimi & Recht, 2008). We can also observe Fast-Nys achieves much lower approximation error than these two methods.