Fast DPP Sampling for Nyström with Application to Kernel Methods

Fast DPP Sampling for Nyström with Application to Kernel Methods

Chengtao Li ctli@mit.edu
Stefanie Jegelka stefje@csail.mit.edu
Suvrit Sra suvrit@mit.edu
Massachusetts Institute of Technology

Abstract

The Nyström method has long been popular for scaling up kernel methods. Its theoretical guarantees and empirical performance rely critically on the quality of the *landmarks* selected. We study landmark selection for Nyström using Determinantal Point Processes (DPPs), discrete probability models that allow tractable generation of *diverse* samples. We prove that landmarks selected via DPPs guarantee bounds on approximation errors; subsequently, we analyze implications for kernel ridge regression. Contrary to prior reservations due to cubic complexity of DPP sampling, we show that (under certain conditions) Markov chain DPP sampling requires only *linear* time in the size of the data. We present several empirical results that support our theoretical analysis, and demonstrate the superior performance of DPP-based landmark selection compared with existing approaches.

1 Introduction

Low-rank matrix approximation is an important ingredient of modern machine learning methods. Numerous learning tasks rely on multiplication and inversion of matrices, operations that scale cubically in the number of data points N, and therefore quickly become a bottleneck for large data. In such cases, low-rank matrix approximations promise speedups with a tolerable loss in accuracy.

A notable instance is the *Nyström method* [32, 43], which takes a positive semidefinite matrix $K \in \mathbb{R}^{N\times N}$ as input, selects from it a small subset C of columns $K_{\cdot,C}$, and constructs the approximation $\tilde{K} = K_{\cdot,C}K^{\dagger}_{C,C}K_{C,\cdot}$. The matrix $\tilde{K}$ is then used in place of K, which can decrease runtimes from $\mathcal{O}(N^3)$ to $\mathcal{O}(N|C|^3)$, a huge savings (since typically $|C| \ll N$).

Since its introduction into machine learning, the Nyström method has been applied to a wide spectrum of problems, including kernel ICA [6, 36], kernel and spectral methods in computer vision [8, 18], manifold learning [39, 40], regularization [35], and efficient approximate sampling [1]. Recent work [2, 5, 12] shows risk bounds for Nyström applied to various kernel methods.

The most important step of the Nyström method is the selection of the subset C, the so-called *landmarks*. This choice governs the approximation error and subsequent performance of the approximated learning methods [12]. The most basic strategy is to sample landmarks uniformly at random [43]. More sophisticated non-uniform selection strategies include deterministic greedy schemes [37], incomplete Cholesky decomposition [7, 17], sampling with probabilities proportional to diagonal values [14] or to column norms [15], sampling based on leverage scores [19], via K-means [44], or using submatrix determinants [9].

We study landmark selection using *Determinantal Point Processes (*DPP*)*, discrete probability models that allow tractable sampling of diverse non-independent subsets [26, 31]. Our work generalizes the determinant based scheme of Belabbas and Wolfe [9].[1] We refer to our scheme as DPP-Nyström, and analyze it from several perspectives.

[1] The authors do not make any connection to DPPs.

A key quantity in our analysis is the error of the Nyström approximation. Suppose k is the target rank; then for selecting $c \geq k$ landmarks, Nyström's error is typically measured using the Frobenius or spectral norm relative to the best achievable error via rank-k SVD K_k; i.e., we measure

$$\frac{\|K - K_{\cdot,C} K_{C,C}^{\dagger} K_{C,\cdot}\|_F}{\|K - K_k\|_F} \quad \text{or} \quad \frac{\|K - K_{\cdot,C} K_{C,C}^{\dagger} K_{C,\cdot}\|_2}{\|K - K_k\|_2}.$$

Several authors also use additive instead of relative bounds. However, such bounds are very sensitive to scaling, and become loose even if a single entry of the matrix is large. Thus, we focus on the above relative error bounds.

First, we analyze this approximation error. Previous analyses [9] fix a cardinality $c = k$; we allow the general case of selecting $c \geq k$ columns. Our relative error bounds rely on the properties of characteristic polynomials. Empirically, DPP-Nyström obtains approximations competitive to state-of-the-art methods.

Second, we consider its impact on kernel methods. Specifically, we address the impact of Nyström-based kernel approximations on kernel ridge regression. This task has been noted as the main application in [2, 5]. We show risk bounds of DPP-Nyström that hold in expectation. Empirically, it achieves the best performance among competing methods.

Third, we consider the efficiency of DPP-Nyström; specifically, its tradeoff between error and running time. Since its proposal, determinantal sampling has so far not been used widely in practice due to valid concerns about its scalability. We consider a Gibbs sampler for k-DPP, and analyze its mixing time using a *path coupling* [11] argument. We prove that under certain conditions the chain is fast mixing, which implies a *linear* running time for DPP sampling of landmarks. Empirical results indicate that the chain yields favorable results within a small number of iterations, and the best efficiency-accuracy traedoffs compared to state-of-art methods (Figure 6).

2 Background and Notation

Throughout, we are approximating a given positive semidifinite (PSD) matrix $K \in \mathbb{R}^{N \times N}$ with eigendecomposition $K = U \Lambda U^{\top}$ and eigenvalues $\lambda_1 \geq \ldots \geq \lambda_N$. We use $K_{i,\cdot}$ for the i-th row and $K_{\cdot,j}$ for the j-th column, and, likewise, $K_{C,\cdot}$ for the rows of K and $K_{\cdot,C}$ for the columns of K indexed by $C \subseteq [N]$. Finally, $K_{C,C}$ is the submatrix of K with rows and columns indexed by C. In this notation, $K_k = U_{\cdot,[k]} \Lambda_{[k],[k]} U_{\cdot,[k]}^{\top}$ is the best rank-k approximation to K in both Frobenius and spectral norm. We write $r(\cdot)$ for the rank and $(\cdot)^{\dagger}$ for the pseudoinverse, and denote a decomposition of K by $B^{\top} B$, where $B \in \mathbb{R}^{r(K) \times N}$.

The Nyström Method. The *standard Nyström* method selects a subset $C \subseteq [N]$ of $c = |C|$ *landmarks*, and approximates K with $K_{\cdot,C} K_{C,C}^{\dagger} K_{C,\cdot}$. The actual set of landmarks affects the approximation quality, and is hence the subject of a substantial body of research [7, 9, 12, 14, 15, 17, 19, 37, 44]. Besides various landmark selection methods, there exist variations of the standard Nyström method. The *ensemble Nyström method* [27], for instance, uses a weighted combination of approximations. The *modified Nyström method* constructs an approximation $K_{\cdot,C} K_{\cdot,C}^{\dagger} K K_{C,\cdot}^{\dagger} K_{C,\cdot}$ [38]. In this paper, we focus on the standard Nyström method.

Determinantal Point Processes. A *determinantal point process* $\mathrm{DPP}(K)$ is a distribution over all subsets of a ground set $\mathcal{Y}$ of cardinality N that is determined by a PSD kernel $K \in \mathbb{R}^{N \times N}$. The probability of observing a subset $C \subseteq [N]$ is proportional to $\det(K_{C,C})$, that is,

$$\Pr(C) = \det(K_{C,C}) / \det(K + I). \tag{2.1}$$

When conditioning on a fixed cardinality, one obtains a k-DPP [25]. To avoid confusion with the target rank k, and since we use cardinality $c = |C|$, we will refer to this distribution as c-DPP[2], and

[2]Note that we refer to DPP-Nyström as `kDPP` in experimental parts.

note that

$$\Pr(C \mid |C| = c) = \det(K_{C,C}) e_c(K)^{-1} [\![|C| = c]\!],$$

where $e_c(K)$ is the c-th coefficient of the characteristic polynomial $\det(\lambda I - K) = \sum_{j=0}^{N} (-1)^j e_j(K) \lambda^{N-j}$.

Sampling from a (c-)DPP can be done in polynomial time, but requires a full eigendecomposition of K [23], which is prohibitive for large N. A number of approaches have been proposed for more efficient sampling [1, 29, 42]. We follow an alternative approach based on Gibbs sampling and show that it can offer fast polynomial-time DPP sampling and Nyström approximations.

3 Dpp for the Nyström Method

Next, we consider sampling c landmarks $C \subseteq [N]$ from c-DPP(K), and use the approximation $\tilde{K} = K_{\cdot,C} K_{C,C}^{\dagger} K_{C,\cdot}$. We call this approach DPP-Nyström. It was essentially introduced in [9], but without making the explicit connection to DPPs. Our analysis builds on this connection and subsumes existing results that only apply to c being the rank k of the target approximation.

We begin with error bounds for matrix approximations:

Theorem 1 (Relative Error). *If $C \sim c$-DPP(K), then* DPP*-Nyström satisfies the relative error bounds*

$$\mathbb{E}_C \left[\frac{\|K - K_{\cdot C}(K_{C,C})^{\dagger} K_{C\cdot}\|_F}{\|K - K_k\|_F} \right] \leq \left(\frac{c+1}{c+1-k} \right) \sqrt{N-k},$$

$$\mathbb{E}_C \left[\frac{\|K - K_{\cdot C}(K_{C,C})^{\dagger} K_{C\cdot}\|_2}{\|K - K_k\|_2} \right] \leq \left(\frac{c+1}{c+1-k} \right) (N-k).$$

These bounds hold in expectation. An additional argument based on [33] yields high probability bounds, too (Appendix A).

To show Theorem 1, we exploit a property of characteristic polynomials observed in [22]. But first recall that the coefficients of characteristic polynomials satisfy $e_c(K) = \sum_{|S|=c} \det(B_{\cdot,S}^{\top} B_{\cdot,S}) = e_c(\Lambda)$.

Lemma 2 (Guruswami and Sinop [22]). *For any $c \geq k > 0$, it holds that*

$$\frac{e_{c+1}(K)}{e_c(K)} \leq \frac{1}{c+1-k} \sum_{i>k} \lambda_i.$$

With Lemma 2 in hand, we are ready to prove Theorem 1.

Proof (Thm. 1). We begin with the Frobenius norm error, and then show the spectral norm result. Using the decomposition $K = B^{\top} B$, it holds that

$$\begin{aligned}
\mathbb{E}_C \left[\|K - K_{\cdot C} K_{C,C}^{\dagger} K_{C\cdot}\|_F \right] &= \mathbb{E}_C \left[\|B^{\top} B - B^{\top} B_{\cdot,C} (B_{\cdot,C}^{\top} B_{\cdot,C})^{\dagger} B_{\cdot,C}^{\top} B\|_F \right] \\
&= \mathbb{E}_C \left[\|B^{\top} (I - B_{\cdot,C} (B_{\cdot,C}^{\top} B_{\cdot,C})^{\dagger} B_{\cdot,C}^{\top}) B\|_F \right] = \mathbb{E}_C \left[\|B^{\top} (I - U^C (U^C)^{\top}) B\|_F \right],
\end{aligned}$$

where $U^C \Sigma^C (V^C)^{\top}$ is the SVD of $B_{\cdot,C}$. Next, we extend $U^C \in \mathbb{R}^{r(K) \times c}$ to an orthogonal basis $[U^C \ (U^C)^{\perp}] \in \mathbb{R}^{r(K) \times r(K)}$ of $\mathbb{R}^N$. Using that $I - U^C (U^C)^{\top} = (U^C)^{\perp} ((U^C)^{\perp})^{\top}$ and applying

Cauchy-Schwartz yields

$$\begin{aligned}
&\mathbb{E}_C\left[\|B^\top(I - U^C(U^C)^\top)B\|_F\right] = \mathbb{E}_C\left[\|B^\top(U^C)^\perp((U^C)^\perp)^\top B\|_F\right] \\
&= \mathbb{E}_C\left[\sqrt{\textstyle\sum_{i,j}(b_i^\top(U^C)^\perp((U^C)^\perp)^\top b_j)^2}\right] \leq \mathbb{E}_C\left[\sqrt{(\textstyle\sum_{i,j}\|b_i^\top(U^C)^\perp\|_2^2\|b_j^\top(U^C)^\perp\|_2^2)}\right] \\
&= \mathbb{E}_C\left[\textstyle\sum_i \|b_i^\top(U^C)^\perp\|_2^2\right] = \frac{1}{e_c(K)}\sum_{|C|=c}\sum_i \det(B_{\cdot,C}^\top B_{\cdot,C})\|b_i^\top(U^C)^\perp\|_2^2 \\
&\overset{(a)}{=} \frac{1}{e_c(K)}\sum_{|C|=c}\sum_{i\notin C}\det(B_{\cdot,C\cup\{i\}}B_{\cdot,C\cup\{i\}}^\top) \\
&\overset{(b)}{=} (c+1)\frac{e_{c+1}(K)}{e_c(K)}.
\end{aligned}$$

In (a), we use that $(U^C)^\perp$ projects vectors onto the null (column) space of B, and (b) uses the definition of e_c. With Lemma 2, it follows that

$$\begin{aligned}
(c+1)\tfrac{e_{c+1}(K)}{e_c(K)} &\leq \tfrac{c+1}{c+1-k}\textstyle\sum_{i>k}\lambda_i \\
&\leq \tfrac{c+1}{c+1-k}\sqrt{N-k}\sqrt{\textstyle\sum_{i>k}\lambda_i^2} = \tfrac{c+1}{c+1-k}\sqrt{N-k}\|K-K_k\|_F.
\end{aligned}$$

The bound on the Frobenius norm immediately implies the bound on the spectral norm:

$$\begin{aligned}
\mathbb{E}_C\left[\|K - K_{\cdot C}(K_{C,C})^\dagger K_{C\cdot}\|_2\right] &\leq \mathbb{E}_C\left[\|K - K_{\cdot C}K_{C,C}^\dagger K_{C\cdot}\|_F\right] \\
\leq \frac{c+1}{c+1-k}\sqrt{N-k}\|K-K_k\|_F &\leq \frac{c+1}{c+1-k}(N-k)\|K-K_k\|_2
\end{aligned}$$

□

Remarks. Compared to previous bounds (e.g., [19] on uniform and leverage score sampling), our bounds seem somewhat weaker asymptotically (since as $c \to N$ they do not converge to 1). This suggests that there is an opportunity for further tightening our bounds, which may be worthwhile, given than in Section Sec. 6.1 our extensive experiments on various datasets with DPP-Nyström show that it attains superior accuracies compared with various state-of-art methods.

4 Low-rank Kernel Ridge Regression

Our theoretical (Section 3) and empirical (Section 6.1) results suggest that DPP-Nyström is well-suited for scaling kernel methods. In this section, we analyze its implications on kernel ridge regression. The experiments in Section 6 confirm our results empirically.

We have N training samples $\{(x_i, y_i)\}_{i=1}^N$, where $y_i = z_i + \epsilon_i$ are the observed labels under zero-mean noise with finite covariance. We minimize a regularized empirical loss

$$\min_{f\in\mathcal{F}} \frac{1}{N}\sum_{i=1}^N \ell(y_i, f(x_i)) + \frac{\gamma}{2}\|f\|^2$$

over an RKHS $\mathcal{F}$. Equivalently, we solve the problem

$$\min_{\alpha\in\mathbb{R}^N} \frac{1}{N}\sum_{i=1}^N \ell(y_i, (K\alpha)_i) + \frac{\gamma}{2}\alpha^\top K\alpha,$$

for the corresponding kernel matrix K. With the squared loss $\ell(y, f(x)) = \frac{1}{2}(y - f(x))^2$, the resulting estimator is

$$\hat{f}(x) = \sum_{i=1}^N \hat{\alpha}_i k(x, x_i), \quad \hat{\alpha} = (K + n\gamma I)^{-1}y, \tag{4.1}$$

and the prediction for $\{x_i\}_{i=1}^N$ is given by $\hat{z} = K(K+N\gamma I)^{-1}y \in \mathbb{R}^N$. Denoting the noise covariance by F, we obtain the risk

$$\begin{aligned}\mathcal{R}(\hat{z}) &= \tfrac{1}{N}\mathbb{E}_\varepsilon\|\hat{z}-z\|^2 \\ &= N\gamma^2 z^\top (K+N\gamma I)^{-2}z + \tfrac{1}{N}\operatorname{tr}(FK^2(K+N\gamma I)^{-2}) \\ &= \operatorname{bias}(K) + \operatorname{var}(K). \end{aligned} \tag{4.2}$$

Observe that the bias term is matrix-decreasing (in K) while the variance term is matrix-increasing. Since the estimator (4.1) requires expensive matrix inversions, it is common to replace K in (4.1) by an approximation $\tilde{K}$. If $\tilde{K}$ is constructed via Nyström we have $\tilde{K} \preceq K$, and it directly follows that the variance shrinks with this substitution, while the bias increases. Denoting the predictions from $\tilde{K}$ by $\hat{z}_{\tilde{K}}$, Theorem 3 completes the picture of how using $\tilde{K}$ affects the risk.

Theorem 3. *If $\tilde{K}$ is constructed via* DPP*-Nyström, then*

$$\mathbb{E}_C\left[\sqrt{\frac{\mathcal{R}(\hat{z}_{\tilde{K}})}{\mathcal{R}(\hat{z})}}\right] \le 1 + \frac{(c+1)}{N\gamma}\frac{e_{c+1}(K)}{e_c(K)}.$$

Again, using [33], we obtain bounds that hold with high probability (Appendix A).

Proof. We build on [2, 5]. Knowing that $\operatorname{Var}(\tilde{K}) \le \operatorname{Var}(K)$ as $\tilde{K} \preceq K$, it remains to bound the bias. Using $K = B^\top B$ and $\tilde{K} = B^\top B_{\cdot,C}(B_{\cdot,C}^\top B_{\cdot,C})^\dagger B_{\cdot,C}^\top B$, we obtain

$$\begin{aligned} K - \tilde{K} &= B^\top(I - B_{\cdot,C}(B_{\cdot,C}^\top B_{\cdot,C})^\dagger B_{\cdot,C}^\top)B \\ &= B^\top (U^C)^\perp((U^C)^\perp)^\top B \preceq \|B^\top(U^C)^\perp((U^C)^\perp)^\top B\|_F I \\ &= \sqrt{\textstyle\sum_{i,j}(b_i^\top(U^C)^\perp((U^C)^\perp)^\top b_j)^2}\, I \\ &\preceq \sqrt{(\textstyle\sum_{i,j}\|b_i^\top(U^C)^\perp\|_2^2\|b_j^\top(U^C)^\perp\|_2^2)}\, I \\ &= \textstyle\sum_i \|b_i^\top(U^C)^\perp\|_2^2 I = \nu_C I, \end{aligned}$$

where $\nu_C = \sum_i \|b_i^\top(U^C)^\perp\|_2^2 \le \sum_i \|b_i^\top\|_2^2 = \operatorname{tr}(K)$. Since $(K-\tilde{K})$ and $\nu_C I$ commute, we have

$$\begin{aligned} &\|(\tilde{K}+N\gamma I)^{-1}(K-\tilde{K})\|_2^2 \\ &\quad = \|(\tilde{K}+N\gamma I)^{-1}(K-\tilde{K})^2(\tilde{K}+N\gamma I)^{-1}\|_2 \\ &\quad \le \nu_C^2\|(\tilde{K}+N\gamma I)^{-2}\|_2 \le \left(\frac{\nu_C}{N\gamma}\right)^2. \end{aligned}$$

It follows that

$$\begin{aligned} &\|(\tilde{K}+N\gamma I)^{-1}z - (K+N\gamma I)^{-1}z\|_2 \\ &\quad = \|(\tilde{K}+N\gamma I)^{-1}(K-\tilde{K})(K+N\gamma I)^{-1}z\|_2 \\ &\quad \le \|(\tilde{K}+N\gamma I)^{-1}(K-\tilde{K})\|_2\|(K+N\gamma I)^{-1}z\|_2 \\ &\quad \le \frac{\nu_C}{N\gamma}\|(K+N\gamma I)^{-1}z\|_2. \end{aligned}$$

Hence,

$$\begin{aligned} &\sqrt{z^\top(\tilde{K}+N\gamma I)^{-2}z} = \|(\tilde{K}+N\gamma I)^{-1}z\|_2 \\ &\le \|(K+N\gamma I)^{-1}z\|_2 + \|(\tilde{K}+N\gamma I)^{-1}z - (K+N\gamma I)^{-1}z\|_2 \\ &\le (1+\frac{\nu_C}{N\gamma})\|(K+N\gamma I)^{-1}z\|_2 \\ &= (1+\frac{\nu_C}{N\gamma})\sqrt{z^\top(K+N\gamma I)^{-2}z}. \end{aligned}$$

Finally, this inequality implies that

$$\sqrt{\frac{\text{bias}(\tilde{K})}{\text{bias}(K)}} \leq (1 + \frac{\nu_C}{N\gamma}).$$

Taking the expectation over $C \sim c\text{-DPP}(K)$ yields

$$\mathbb{E}_C\left[\sqrt{\frac{\text{bias}(\tilde{K})}{\text{bias}(K)}}\right] \leq 1 + \mathbb{E}_C\left[\frac{\nu_C}{N\gamma}\right] = 1 + \frac{(c+1)}{N\gamma}\frac{e_{c+1}(K)}{e_c(K)}.$$

Together with the fact that $\text{var}(\tilde{K}) \leq \text{var}(K)$, we obtain

$$\begin{aligned}\mathbb{E}_C\left[\sqrt{\frac{\mathcal{R}(\hat{z}_{\tilde{K}})}{\mathcal{R}(\hat{z})}}\right] &= \mathbb{E}_C\left[\sqrt{\frac{\text{bias}(\tilde{K}) + \text{var}(\tilde{K})}{\text{bias}(K) + \text{var}(K)}}\right] \\ &\leq 1 + \frac{(c+1)}{N\gamma}\frac{e_{c+1}(K)}{e_c(K)} \qquad (4.3)\end{aligned}$$

for any $k \leq c$. □

Remarks. Theorem 3 quantifies how the learning results depend on the decay of the spectrum of K. In particular, the ratio $e_{c+1}(K)/e_c(K)$ closely relates to the effective rank of K: if $\lambda_c > a$ and $\lambda_{c+1} \ll a$, this ratio is almost zero, resulting in near-perfect approximations and no loss in learning.

There exist works that consider Nyström methods in this scenario [2, 5]. Our theoretical bounds could also be tightened in this setting, possibly by a tighter bound on the elementary symmetric polynomial ratio. This theoretical exercise may be worthwhile given our extensive experiments comparing DPP-Nyström against other state-of-art methods in Sec. 6.2 that reveal the superior performance of DPP-Nyström.

5 Fast Mixing Markov Chain Dpp

Despite its excellent empirical performance and strong theoretical results, determinantal sampling for Nyström has rarely been used in applications due to the computational cost of $\mathcal{O}(N^3)$ for directly sampling from a DPP, which involves an eigendecomposition. Instead, we follow a different route: an MCMC sampler, which offers a promising alternative if the chain mixes fast enough. Recent empirical results provide initial evidence [24], but without a theoretical analysis[3]; other recent works [21, 34] do not apply to our cardinality-constrained setting. We offer a theoretical analysis that confirms fast mixing (i.e., polynomial or even *linear*-time sampling) under certain conditions, and connect it to our empirical results. The empirical results in Section 6 illustrate the favorable performance of DPP-Nyström in trading off time and error. Concurrently with this paper, Anari et al. [4] derived a different, general analysis of fast mixing that also confirms our observations.

Algorithm 1 shows a Gibbs sampler for k-DPP. Starting with a uniformly random set Y_0, at iteration t, we try to swap an element $y^{\text{in}} \in Y_t$ with an element $y^{\text{out}} \notin Y_t$, according to $\Pr(Y_t)$ and $\Pr(Y_t \cup \{y^{\text{out}}\} \setminus \{y^{\text{in}}\})$. The stationary distribution of this chain is exactly the desired k-DPP(K).

The *mixing time* $\tau(\varepsilon)$ of the chain is the number of iterations until the distribution over the states (subsets) is close to the desired one, as measured by total variation: $\tau(\varepsilon) = \min\{t | \max_{Y_0} \text{TV}(Y_t, \pi) \leq \epsilon\}$. We bound $\tau(\varepsilon)$ via coupling techniques. Given a Markov chain (Y_t) on a state space Ω with

[3] The analysis in [24] is not correct.

Algorithm 1 Gibbs sampler for c-DPP

Input: K the kernel matrix, $\mathcal{Y} = [N]$ the ground set
Output: Y sampled from exact c-DPP(K)
Randomly Initialize $Y \subseteq \mathcal{Y}$, $|Y| = c$
while not mixed **do**
  Sample b from uniform Bernoulli distribution
  if $b = 1$ **then**
    Pick $y^{\text{in}} \in Y$ and $y^{\text{out}} \in \mathcal{Y} \backslash Y$ uniformly randomly
    $q(y^{\text{in}}, y^{\text{out}}, Y) \leftarrow \frac{\det(K_{Y \cup \{y^{\text{out}}\} \backslash \{y^{\text{in}}\}})}{\det K_{Y \cup \{y^{\text{out}}\} \backslash \{y^{\text{in}}\}} + \det(K_Y)}$
    $Y \leftarrow Y \cup \{y^{\text{out}}\} \backslash \{y^{\text{in}}\}$ with prob. $q(y^{\text{in}}, y^{\text{out}}, Y)$
  end if
end while

transition matrix P, a *coupling* is a new chain (Y_t, Z_t) on $\Omega \times \Omega$ such that both (Y_t) and (Z_t), if considered marginally, are Markov chains with the same transition matrix P. The key point of coupling is to construct such a new chain to encourage Y_t and Z_t to *coalesce* quickly. If, in the new chain, $\Pr(Y_t \neq Z_t) \leq \varepsilon$ for some fixed t regardless of the starting state (Y_0, Z_0), then $\tau(\varepsilon) \leq t$ [3].

Such coalescing chains can be difficult to construct. *Path coupling* [11] relieves this burden by reducing the coupling to adjacent states in an appropriately constructed state graph. The coupling of arbitrary states follows by aggregation over a path between the states. Path coupling is formalized in the following lemma.

Lemma 4. *[11, 16] Let δ be an integer-valued metric on $\Omega \times \Omega$ where $\delta(\cdot, \cdot) \leq D$. Let E be a subset of $\Omega \times \Omega$ such that for all $(Y_t, Z_t) \in \Omega \times \Omega$ there exists a path $Y_t = X_0, \ldots, X_r = Z_t$ between Y_t and Z_t where $(X_i, X_{i+1}) \in E$ for $i \in [r-1]$ and $\sum_i \delta(X_i, X_{i+1}) = \delta(Y_t, Z_t)$. Suppose a coupling $(R, T) \to (R', T')$ of the Markov chain is defined on all pairs in E such that there exists an $\alpha < 1$ such that $\mathbb{E}[\delta(R', T')] \leq \alpha \delta(R, T)$ for all $(R, T) \in E$, then we have*

$$\tau(\varepsilon) \leq \frac{\log(D\varepsilon^{-1})}{(1-\alpha)}.$$

The lemma says that if we have a contraction of the two chains in expectation ($\alpha < 1$), then the chain mixes fast. With the path coupling lemma, we obtain a bound on the mixing time that can be *linear* in the data set size N.

The actual mixing time depends on three quantities that relate to how sensitive the transition probabilities are to swapping a single element in a set of size c. Consider an arbitrary set S of columns, $|S| = c - 1$, and complete it to two c-sets $R = S \cup \{r\}$ and $T = S \cup \{t\}$ that differ in exactly one element. Our quantities are, for $u \notin R \cup T$, and $v \in S$:

$$\begin{aligned} p_1(S, r, t, u) &= \min\{q(r, u, R), q(t, u, T)\} \\ p_2(S, r, t, u) &= \min\{q(v, t, R), q(v, u, T)\} \\ p_3(S, r, t, v, u) &= |q(v, u, R) - q(v, u, T)|. \end{aligned}$$

Theorem 5. *Let the contraction coefficient α be given by*

$$\alpha = \max_{|S|=c-1, r, t \in [n] \backslash S, r \neq t} \sum_{u_3 \in S, u_4 \notin S \cup \{r,t\}} p_3(S, r, t, u_3, u_4) - \sum_{u_1 \notin S \cup \{r,t\}} p_1(S, r, t, u_1) - \sum_{u_2 \in S} p_2(S, r, t, u_2).$$

When $\alpha < 1$, the mixing time for the Gibbs sampler in Algorithm 1 is bounded as

$$\tau(\varepsilon) \leq \frac{2c(N-c)\log(c\varepsilon^{-1})}{(1-\alpha)}.$$

Proof. We bound the mixing time via path coupling. Let $\delta(R,T) = |R \oplus T|/2$ be half the Hamming distance on the state space, and define E to consist of all state pairs (R,T) in $\Omega \times \Omega$ such that $\delta(R,T) = 1$. We intend to show that for all states $(R,T) \in E$ and next states $(R',T') \in E$, we have $\mathbb{E}[\delta(R',T')] \leq \alpha\delta(R,T)$ for an appropriate α.

Since $\delta(R,T) = 1$, the sets R and T differ in only two entries. Let $S = R \cap T$, so $|S| = c-1$ and $R = S \cup \{r\}$ and $T = S \cup \{t\}$. For a state transition, we sample an element $r^{\text{in}} \in R$ and $r^{\text{out}} \in [n]\backslash R$ as switching candidates for R, and elements $t^{\text{in}} \in T$ and $t^{\text{out}} \in [n]\backslash T$ as switching candidates for T. Let b_R and b_T be the Bernoulli random variables indicating whether we try to make a transition. In our coupling we always set $b_R = b_T$. Hence, if $b_R = 0$ then both chains will not transition and the distance of states remains. For $b_R = b_T = 1$, we distinguish four cases:

Case C1 If $r^{\text{in}} = r$ and $r^{\text{out}} = t$, we let $t^{\text{in}} = t$ and $t^{\text{out}} = r$. As a result, $\delta(R',T') = 0$.

Case C2 If $r^{\text{in}} = r$ and $r^{\text{out}} = u_1 \notin S \cup \{r,t\}$, we let $t^{\text{in}} = t$ and $t^{\text{out}} = u_1$. In this case, if both chains transition, then the resulting distance is zero, otherwise it remains one. With probability $p_1(S,r,t,u_1) = \min\{q(r,u_1,R), q(t,u_1,T)\}$ both chains transition.

Case C3 If $r^{\text{in}} = u_2 \in S$ and $r^{\text{out}} = t$, we let $t^{\text{in}} = u_2$ and $t^{\text{out}} = r$. Again, if both chains transition, then the resulting distance is $\delta(R',T') = 0$, otherwise it remains one. With probability $p_2(S,r,t,u_2) = \min\{q(u_2,t,R), q(u_2,u_1,T)\}$ both chains transition.

Case C4 If $r^{\text{in}} = u_3 \in S$ and $r^{\text{out}} = u_4 \notin S \cup \{r,t\}$, we let $t^{\text{in}} = u_3$ and $t^{\text{out}} = u_4$. If both chains make the same transition (both move or do not move), the resulting distance is one, otherwise it increases to 2. The distance increases with probability $p_3(S,r,t,u_3,u_4) = |q(u_3,u_4,R) - q(u_3,u_4,T)|$.

With those four cases, we can now bound $\mathbb{E}[\delta(R',T')]$. For all $(R,T) \in E$, i.e., $\delta(R,T) = 1$:

$$\begin{aligned}
\frac{\mathbb{E}[\delta(R',T')]}{\mathbb{E}[\delta(R,T)]} &= \frac{1}{2} + \Pr(C2)\mathbb{E}[\delta(R',T')|C2] + \Pr(C3)\mathbb{E}[\delta(R',T')|C3] + \Pr(C4)\mathbb{E}[\delta(R',T')|C4] \\
&= \frac{1}{2} + \frac{1}{2c(n-c)}\Big(\sum_{u_1 \notin S\cup\{r,t\}} (1 - p_1(u_1)) + \sum_{u_2 \in S} (1 - p_2(u_2)) + \sum_{\substack{u_3 \in S, \\ u_4 \notin S\cup\{r,t\}}} (1 + p_3(u_3,u_4))\Big) \\
&= \frac{1}{2c(n-c)}\Big(2c(n-1) + \sum_{\substack{u_3 \in S, \\ u_4 \notin S\cup\{r,t\}}} p_3(u_3,u_4) - \sum_{u_1 \notin S\cup\{r,t\}} p_1(u_1) - \sum_{u_2 \in S} p_2(u_2) - 1\Big),
\end{aligned}$$

where we did not explicitly write the arguments S, r, t to p_1, p_2, p_3. For

$$\alpha = \max_{\substack{|S|=c-1, \\ r,t \in [n]\backslash S, \\ r \neq t}} \sum_{\substack{u_3 \in S, \\ u_4 \notin S\cup\{r,t\}}} p_3(u_3,u_4) - \sum_{u_1 \notin S\cup\{r,t\}} p_1(u_1) - \sum_{u_2 \in S} p_2(u_2)$$

and $\alpha < 1$ the Path Coupling Lemma 4 implies that

$$\tau(\varepsilon) \leq \frac{2c(N-c)\log(c\varepsilon^{-1})}{(1-\alpha)}.$$

□

Remarks. If $\alpha < 1$ is fixed, then the mixing time (running time) depends only linearly on N. The coefficient α itself depends on our three quantities. In particular, fast mixing requires p_3 (the difference between transition probabilities) to be very small compared to p_1, p_2, at least on average. The difference p_3 measures how exchangeable two points r and t are. This notion of symmetry is

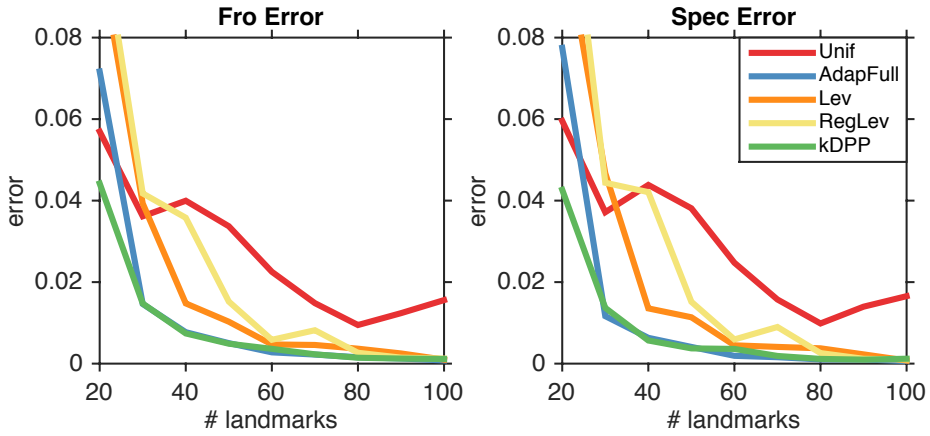


Figure 1: Relative Frobenius/spectral norm errors from different kernel approximations (Ailerons data).

closely related to a symmetry that determines the complexity of submodular maximization [41] (indeed, $F(S) = \log\det K_S$ is a submodular function). This symmetry only needs to hold for most pairs r, t, and most swapping points u, v. It holds for kernels with sufficiently fast-decaying similarities, similar to the conditions in [34] for unconstrained sampling.

One iteration of the sampler can be implemented efficiently in $\mathcal{O}(c^2)$ time using block inversion [20]. Additional speedups via quadrature are also possible [30]. Together with the analysis of mixing time, this leads to fast sampling methods for k-DPPs.

6 Experiments

In our experiments, we evaluate the performance of DPP-Nyström on both kernel approximation and kernel learning tasks, in terms of running time and accuracy.

We use 8 datasets: Abalone, Ailerons, Elevators, CompAct, CompAct(s), Bank32NH, Bank8FM and California Housing[4]. We subsample 4,000 points from each dataset (3,000 training and 1,000 test). Throughout our experiments, we use an RBF kernel and choose the bandwidth σ and regularization parameter λ for each dataset by 10-fold cross-validation. We initialize the Gibbs sampler via Kmeans++ and run for 3,000 iterations. Results are averaged over 3 random subsets of data.

6.1 Kernel Approximation

We first explore DPP-Nyström (`kDPP` in the figures) for approximating kernel matrices. We compare to uniform sampling (`Unif`) and leverage score sampling (`Lev`) [19] as baseline landmark selection methods. We also include AdapFull (`AdapFull`) [13] that performs quite well in practice but scales poorly, as $\mathcal{O}(N^2)$, with the size of dataset. Although sampling with regularized leverage scores (`RegLev`) [2] is not originally designed for kernel approximations, we include its results to see how regularization affects leverage score sampling.

Figure 1 shows example results on the Ailerons data; further results may be found in the appendix. DPP-Nyström performs well, achieving the lowest error as measured in both spectral

[4] http://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html

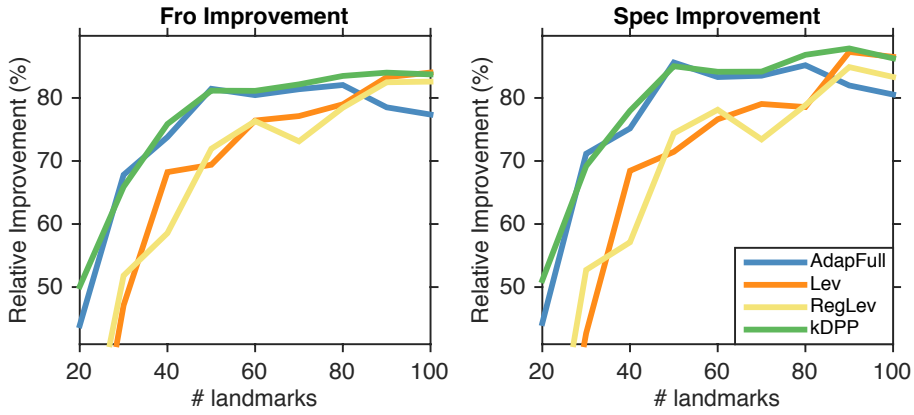


Figure 2: Improvement in relative Frobenius/spectral norm errors (%) over `Unif` (with corresponding landmark sizes) for kernel approximation, averaged over all datasets.

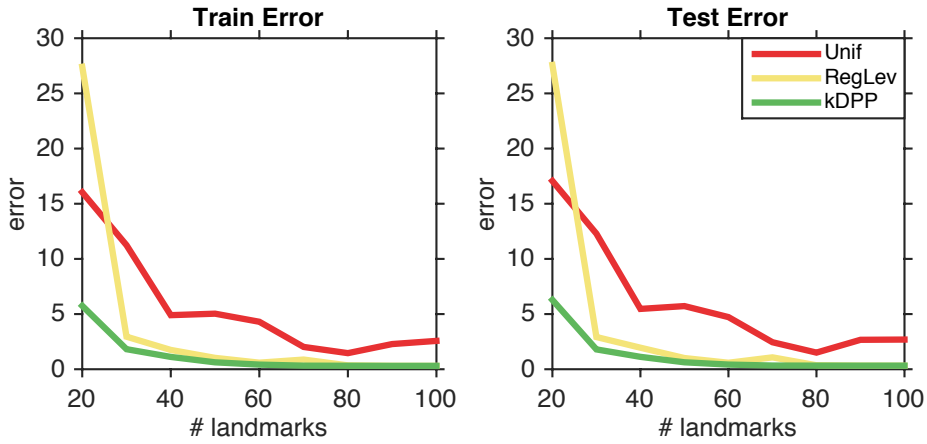


Figure 3: Training and test errors for kernel ridge regression with different Nyström approximations (Ailerons data).

and Frobenius norm. The only method that is on par in terms of accuracy is `AdapFull`, which has a much higher running time.

For a different perspective, Figure 2 shows the improvement in error over `Unif`. Relative improvements are averaged over all data sets. Again, the performance of DPP-Nyström almost always dominate those of other methods, and achieves an up to 80% reduction in error.

6.2 Kernel Ridge Regression

Next, we apply DPP-Nyström to kernel ridge regression, comparing against uniform sampling (`Unif`) [5] and regularized leverage score sampling (`RegLev`) [2] which have theoretical guarantees for this task. Figure 3 illustrates an example result: non-uniform sampling greatly improves accuracy, with `kDPP` improving over regularized leverage scores in particular for a small number of landmarks, where a single column has a larger effect.

Figure 4 displays the average improvement over `Unif`, averaged over 8 data sets. Again, the

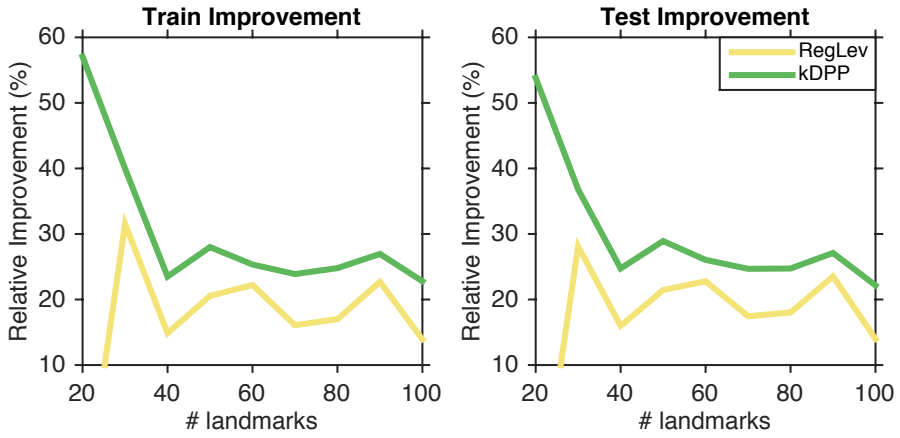


Figure 4: Improvements in training/test errors (%) over uniform sampling (with same number of landmarks) in kernel ridge regression, averaged over all datasets.

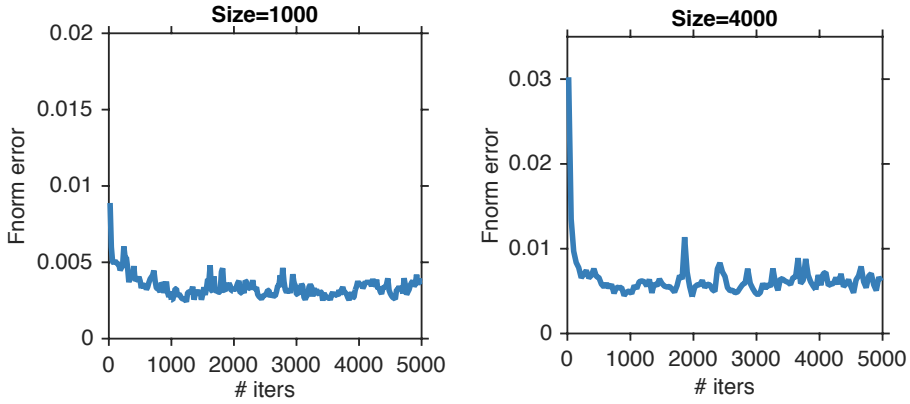


Figure 5: Relative Frobenius norm error of DPP-Nyström with 50 landmarks as changing across iterations of the Markov Chain (Ailerons data).

performance of `kDPP` dominates `RegLev` and `Unif`, and leads to gains in accuracy. On average `kDPP` consistently achieves more than 20% improvement over `Unif`.

6.3 Mixing of the Gibbs Markov Chain

In the next experiment, we empirically study the mixing of the Gibbs chain with respect to matrix approximation errors, the ultimate measure that is of interest in our application of the sampler. We use $c = 50$ and choose N as 1,000 and 4,000. To exclude impacts of the initialization, we pick the initial state Y_0 uniformly at random. We run the chain for 5,000 iterations, monitoring how the error changes with the number of iterations. Example results on the Ailerons data are shown in Figure 5. Empirically, the error drops very quickly and afterwards fluctuates only little, indicating a fast convergence of the approximation error. Other error measures and larger c, included in the appendix, confirm this trend.

Notably, our empirical results suggest that the mixing time does not increase much as N increases greatly, suggesting that the Gibbs sampler remains fast even for large N.

In Theorem 5, the mixing time depends on the quantity α. By subsampling 1,000 random sets S and column indices r, t, we approximately computed α on our data sets. We find that, as expected, $\alpha < 1$ in particular for kernels with a smaller bandwidth, and in general α increases with k. In accordance with the theory, we found that the mixing time (in terms of error) too increases with k. In practice, we observe a fast drop in error even for cases where $\alpha > 1$, indicating that Theorem 5 is conservative and that the iterative MCMC approach is even more widely applicable.

6.4 Time-Error Tradeoffs

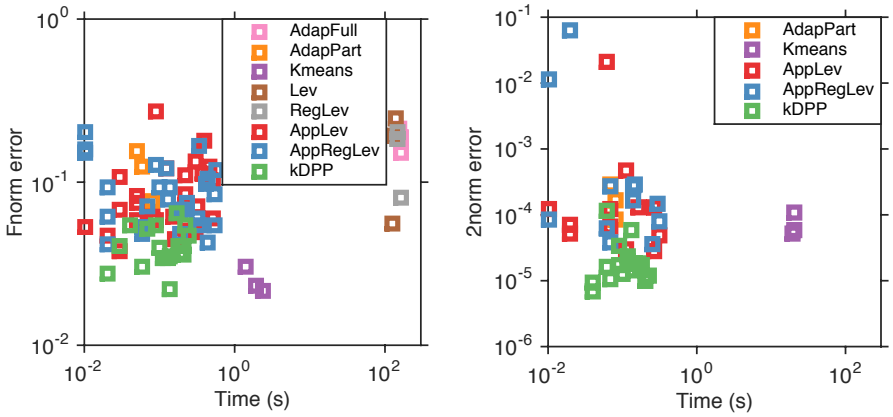


Figure 6: Time-Error tradeoffs with 20 landmarks on Ailerons (size 4,000) and California Housing (size 12,000). Time and Errors are shown on a log scale. Bottom left is the best (low error, low running time), top right is the worst. We did not include `AdapFull`, `Lev` and `RegLev` on California Housing due to their long running times.

Iterative methods like the Gibbs sampler offer tradeoffs between time and error. The longer the Markov Chain runs, the closer the sampling distribution is to the desired DPP, and the higher the accuracy obtained by Nyström. We hence explicitly show the time and accuracy trade-off of the sampler on Ailerons (of size 4,000) for up to 200 and California Housing (of size 12,000) for up to 100 iterations.

A similar tradeoff occurs with leverage scores. For the experiments in the other sections, we computed the (regularized) leverage scores for `Lev` and `RegLev` exactly. This requires a full, computationally expensive eigendecomposition. For a fast, rougher approximation, we here compare to an approximation mentioned in [2]. Concretely, we sample p elements with probability proportional to the diagonal entries of kernel matrices K_{ii}, and then use a Nyström-like method to construct an approximate low-rank decomposition of K, and compute scores based on this approximation. We vary p from 20 to 340 on Ailerons and 20 to 140 on California Housing to show the tradeoff for approximate leverage score sampling (`AppLev`) and regularized leverage score sampling (`AppRegLev`). We also include AdapPartial (`AdapPart`) [28] that approximates `AdapFull` and is much more efficient, and Kmeans Nyström (`Kmeans`) [44] that empirically perform very well in kernel approximation.

Figure 6 summarizes and compares the tradeoffs offered by these different methods on the Ailerons and California Housing datasets. The x axis indicates time, the y axis error, so the lower left is the preferred corner. We see that `AdapFull`, `Lev` and `RegLev` are expensive and perform worse than `kDPP`. The approximate variants `AdapPart`, `AppLev` and `AppRegLev` have comparable efficiency but higher error. On the smaller data, `Kmeans` is accurate but needs more time than `kDPP`,

while on the larger data it is dominated in both accuracy and time by kDPP. Overall, on the larger data, DPP-Nyström offers the best tradeoff of accuracy and efficiency.

7 Conclusion

In this paper, we revisited the use of k-Determinantal Point Processes for sampling good landmarks for the Nyström method. We theoretically and empirically observe its competitive performance, for both matrix approximation and ridge regression, compared to state-of-the-art methods.

To make this accurate method scalable to large matrices, we consider an iterative approach, and analyze it theoretically as well as empirically. Our results indicate that the iterative approach, a Gibbs sampler, achieves good landmark samples quickly; under certain conditions even in a number of iteratons linear in N, for an N by N matrix. Finally, our empirical results demonstrate that among state-of-the-art methods, the iterative sampler yields the best tradeoff between efficiency and accuracy.

Acknowledgements

This research was partially supported by an NSF CAREER award 1553284, NSF grant IIS-1409802, and a Google Research Award. We also thank Xixian Chen for discussions.

References

[1] R. H. Affandi, A. Kulesza, E. Fox, and B. Taskar. Nyström approximation for large-scale determinantal processes. In *AISTATS*, pages 85–98, 2013.

[2] A. E. Alaoui and M. W. Mahoney. Fast randomized kernel methods with statistical guarantees. *NIPS*, 2015.

[3] D. J. Aldous. Some inequalities for reversible Markov chains. *Journal of the London Mathematical Society*, pages 564–576, 1982.

[4] N. Anari, S. O. Gharan, and A. Rezaei. Monte Carlo Markov chain algorithms for sampling strongly Rayleigh distributions and determinantal point processes. In *COLT*, 2016.

[5] F. R. Bach. Sharp analysis of low-rank kernel matrix approximations. *COLT*, 2013.

[6] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *JMLR*, pages 1–48, 2003.

[7] F. R. Bach and M. I. Jordan. Predictive low-rank decomposition for kernel methods. In *ICML*, pages 33–40, 2005.

[8] M.-A. Belabbas and P. J. Wolfe. On landmark selection and sampling in high-dimensional data analysis. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, pages 4295–4312, 2009.

[9] 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*, pages 369–374, 2009.

[10] J. Borcea, P. Brändén, and T. Liggett. Negative dependence and the geometry of polynomials. *Journal of the American Mathematical Society*, pages 521–567, 2009.

[11] R. Bubley and M. Dyer. Path coupling: A technique for proving rapid mixing in Markov chains. In *FOCS*, pages 223–231, 1997.

[12] C. Cortes, M. Mohri, and A. Talwalkar. On the impact of kernel approximation on learning accuracy. In *AISTATS*, pages 113–120, 2010.

[13] A. Deshpande, L. Rademacher, S. Vempala, and G. Wang. Matrix approximation and projective clustering via volume sampling. In *SODA*, pages 1117–1126, 2006.

[14] P. Drineas and M. W. Mahoney. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *JMLR*, pages 2153–2175, 2005.
[15] 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*, pages 158–183, 2006.
[16] M. Dyer and C. Greenhill. A more rapidly mixing Markov chain for graph colorings. *Random Structures and Algorithms*, pages 285–317, 1998.
[17] S. Fine and K. Scheinberg. Efficient SVM training using low-rank kernel representations. *JMLR*, pages 243–264, 2002.
[18] C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the Nyström method. *TPAMI*, pages 214–225, 2004.
[19] A. Gittens and M. W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. *ICML*, 2013.
[20] G. H. Golub and C. F. Van Loan. *Matrix computations*. JHU Press, 2012.
[21] A. Gotovos, H. Hassani, and A. Krause. Sampling from probabilistic submodular models. In *NIPS*, pages 1936–1944, 2015.
[22] V. Guruswami and A. K. Sinop. Optimal column-based low-rank matrix reconstruction. In *SODA*, pages 1207–1214, 2012.
[23] J. B. Hough, M. Krishnapur, Y. Peres, and B. Virág. Determinantal processes and independence. *Probability Surveys*, pages 206–229, 2006.
[24] B. Kang. Fast determinantal point process sampling with application to clustering. In *NIPS*, pages 2319–2327, 2013.
[25] A. Kulesza and B. Taskar. k-DPPs: Fixed-size determinantal point processes. In *ICML*, pages 1193–1200, 2011.
[26] A. Kulesza and B. Taskar. Determinantal point processes for machine learning. *arXiv preprint arXiv:1207.6083*, 2012.
[27] S. Kumar, M. Mohri, and A. Talwalkar. Ensemble Nyström method. In *NIPS*, pages 1060–1068, 2009.
[28] S. Kumar, M. Mohri, and A. Talwalkar. Sampling methods for the nyström method. *The Journal of Machine Learning Research*, pages 981–1006, 2012.
[29] C. Li, S. Jegelka, and S. Sra. Efficient sampling for k-determinantal point processes. *AISTATS*, 2016.
[30] C. Li, S. Sra, and S. Jegelka. Gaussian quadrature for matrix inverse forms with applications. In *ICML*, 2016.
[31] O. Macchi. The coincidence approach to stochastic point processes. *Advances in Applied Probability*, pages 83–122, 1975.
[32] E. J. Nyström. Über die praktische Auflösung von Integralgleichungen mit Anwendungen auf Randwertaufgaben. *Acta Mathematica*, pages 185–204, 1930.
[33] R. Pemantle and Y. Peres. Concentration of Lipschitz functionals of determinantal and other strong Rayleigh measures. *Combinatorics, Probability and Computing*, pages 140–160, 2014.
[34] P. Rebeschini and A. Karbasi. Fast mixing for discrete point processes. *COLT*, 2015.
[35] A. Rudi, R. Camoriano, and L. Rosasco. Less is more: Nyström computational regularization. *NIPS*, 2015.
[36] H. Shen, S. Jegelka, and A. Gretton. Fast kernel-based independent component analysis. *IEEE Transactions on Signal Processing*, pages 3498–3511, 2009.
[37] A. J. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. *ICML*, 2000.
[38] S. Sun, J. Zhao, and J. Zhu. A review of Nyström methods for large-scale machine learning. *Information Fusion*, pages 36–48, 2015.
[39] A. Talwalkar, S. Kumar, and H. Rowley. Large-scale manifold learning. In *CVPR*, 2008.

[40] A. Talwalkar, S. Kumar, M. Mohri, and H. Rowley. Large-scale SVD and manifold learning. *JMLR*, pages 3129–3152, 2013.

[41] J. Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM Journal on Computing*, 42(1):265–304, 2013.

[42] S. Wang, C. Zhang, H. Qian, and Z. Zhang. Using the matrix ridge approximation to speedup determinantal point processes sampling algorithms. In *AAAI*, 2014.

[43] C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *NIPS*, pages 682–688, 2001.

[44] K. Zhang, I. W. Tsang, and J. T. Kwok. Improved Nyström low-rank approximation and error analysis. In *ICML*, pages 1232–1239, 2008.

A Bounds that hold with High Probability

To show high probability bounds we employ concentration results on homogeneous strongly Rayleigh measures. Specifically, we use the following theorem.

Theorem 6 (Pemantle and Peres [33]). *Let $\mathbb{P}$ be a k-homogeneous strongly Rayleigh probability measure on $\{0,1\}^N$ and f an ℓ-Lipschitz function on $\{0,1\}^N$, then*

$$\mathbb{P}(f - \mathbb{E}[f] \geq a\ell) \leq \exp\{-a^2/8k\}.$$

It is known that a k-DPP is a homogeneous strongly Rayleigh measure on $\{0,1\}^N$ [4, 10], thus Theorem 6 applies to results obtained with k-DPP. Concretely, for the bound in Theorem 1 that holds in expectation, we have the following bound that holds with high probability:

Corollary 7. *When sampling $C \sim k$-DPP(K), for any $\delta \in (0,1)$, with probability at least $1-\delta$ we have*

$$\frac{\|K - K_{\cdot C}(K_{C,C})^{\dagger}K_{C\cdot}\|_F}{\|K - K_k\|_F} \leq \left(\frac{c+1}{c+1-k}\right)\sqrt{N-k} + \sqrt{8c\log(1/\delta)}\sqrt{\frac{\sum_{i=1}^N \lambda_i^2}{\sum_{i=k+1}^N \lambda_i^2}},$$
$$\frac{\|K - K_{\cdot C}(K_{C,C})^{\dagger}K_{C\cdot}\|_2}{\|K - K_k\|_2} \leq \left(\frac{c+1}{c+1-k}\right)(N-k) + \sqrt{8c\log(1/\delta)}\frac{\lambda_1}{\lambda_{k+1}},$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$ are the eigenvalues of K.

Proof. The Lipschitz constants of the relative errors are upper bounded by $\sqrt{\frac{\sum_{i=1}^N \lambda_i^2}{\sum_{i=k+1}^N \lambda_i^2}}$ and $\frac{\lambda_1}{\lambda_{k+1}}$, respectively. Applying Theorem 6 yields the results. □

For the bound in Theorem 3 that holds in expectation, we have the following bound that holds with high probability:

Corollary 8. *If $\tilde{K}$ is constructed via* DPP*-Nyström, then with probability at least $1-\delta$, $\sqrt{\frac{\text{bias}(\tilde{K})}{\text{bias}(K)}}$ is upper-bounded by*

$$1 + \frac{1}{N\gamma}\left(\frac{(c+1)e_{c+1}(K)}{e_c(K)} + \sqrt{8c\log(1/\delta)}tr(K)\right).$$

Proof. Consider the function $f_C(K) = \nu_C = \sum_i \|b_i^\top (U^C)^\perp\|_2^2 \leq \sum_i \|b_i^\top\|_2^2 = \text{tr}(K)$. Since $0 \leq f_C(K) \leq \text{tr}(K)$, it follows that the Lipschitz constant for f_C is at most $\text{tr}(K)$. Thus when $C \sim k$-DPP and $\delta \in (0,1)$, by applying Theorem 6 we see that the inequality $\nu_C \leq \mathbb{E}[\nu_C] + \sqrt{8c\log(1/\delta)}\text{tr}(K)$ holds with probability at least $1-\delta$. Hence

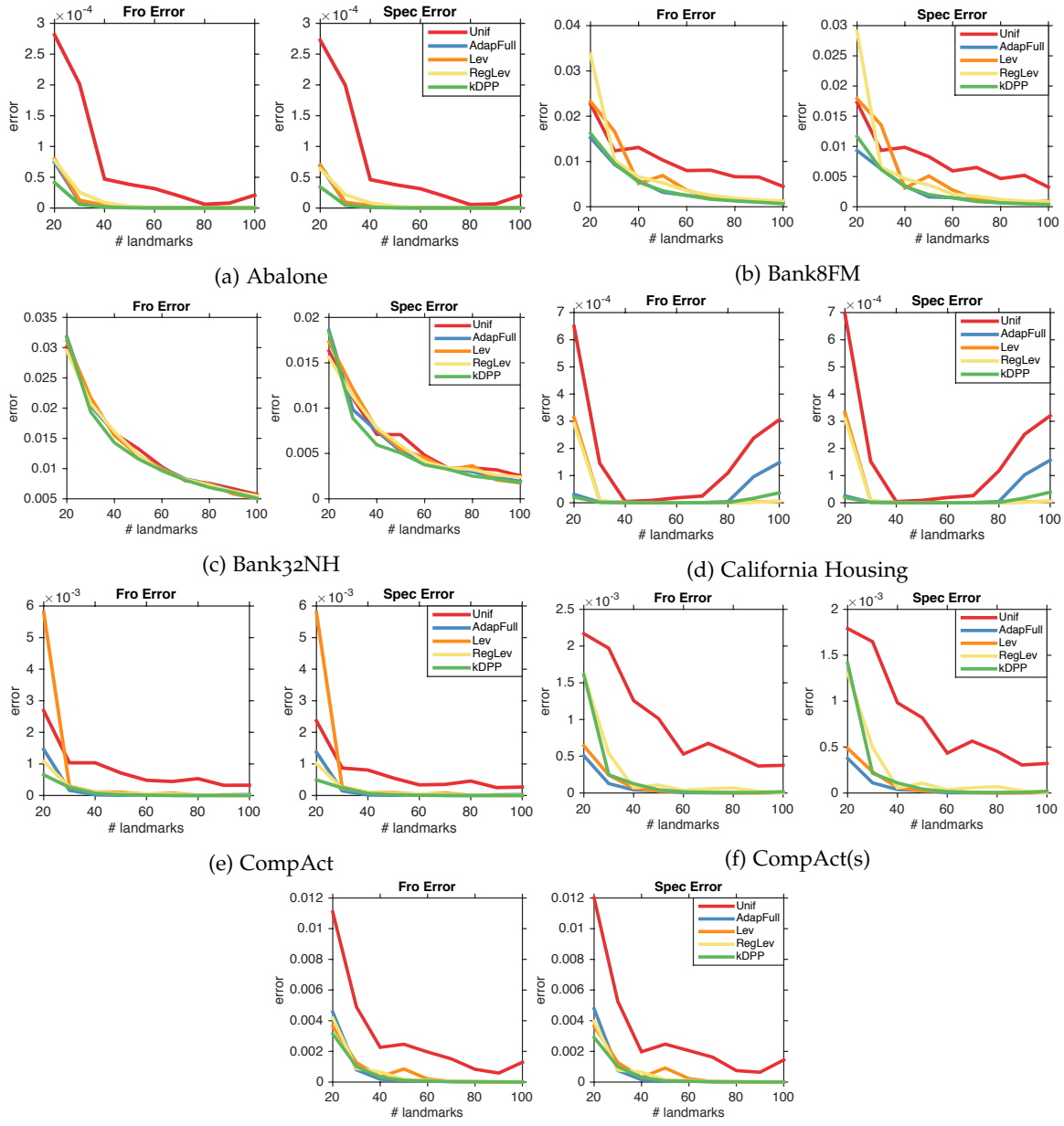
$$\begin{aligned}\mathbb{E}_C\left[\sqrt{\frac{\text{bias}(\tilde{K})}{\text{bias}(K)}}\right] &\leq 1 + \mathbb{E}\left[\frac{\nu_C}{N\gamma}\right] + \sqrt{8c\log(1/\delta)}\frac{\text{tr}(K)}{N\gamma} \\ &= 1 + \frac{1}{N\gamma}\left(\frac{(c+1)e_{c+1}(K)}{e_c(K)} + \sqrt{8c\log(1/\delta)}\text{tr}(K)\right)\end{aligned}$$

holds with probability at least $1-\delta$. □

B Supplementary Experiments

B.1 Kernel Approximation

Fig. 7 shows the matrix norm relative error of various methods in kernel approximation on the remaining 7 datasets mentioned in the main text.



(a) Abalone

(b) Bank8FM

(c) Bank32NH

(d) California Housing

(e) CompAct

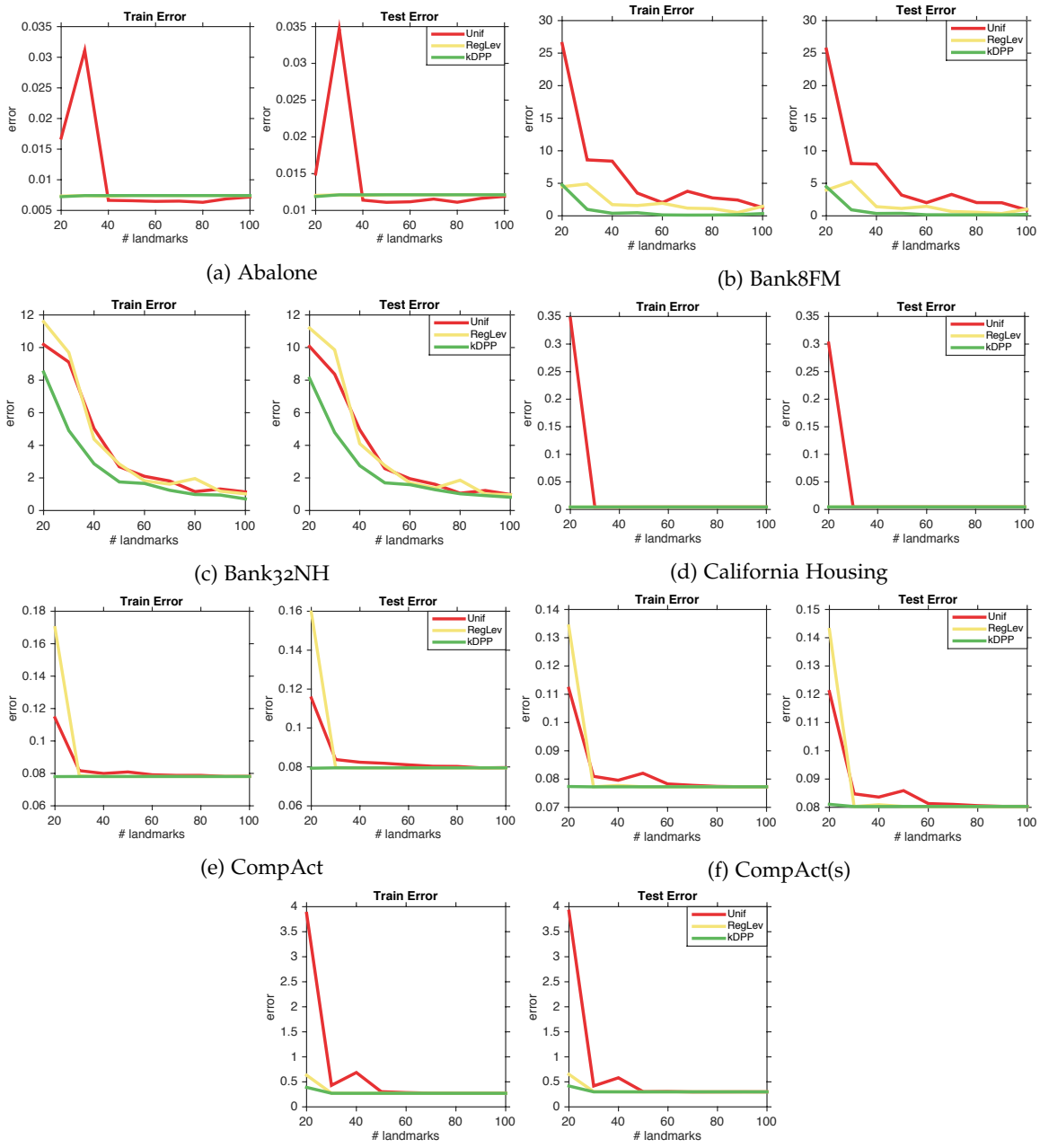
(f) CompAct(s)

(g) Elevators

Figure 7: Relative Frobenius norm and spectral norm error achieved by different kernel approximation algorithms on the remaining 7 data sets.

B.2 Approximated Kernel Ridge Regression

Fig. 8 shows the training and test error of various methods for kernel ridge regression on the remaining 7 datasets.



(a) Abalone

(b) Bank8FM

(c) Bank32NH

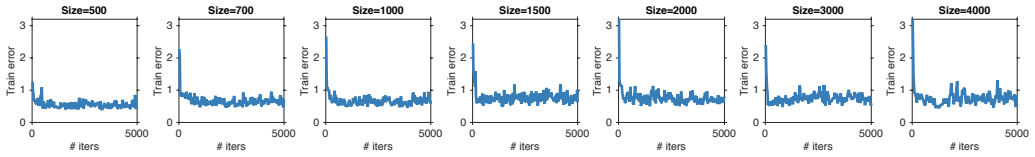
(d) California Housing

(e) CompAct

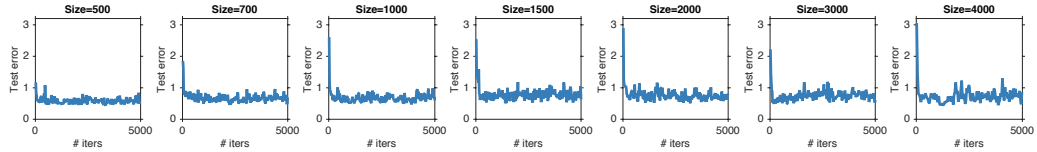
(f) CompAct(s)

(g) Elevators

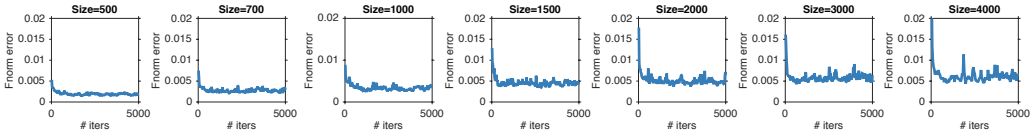
Figure 8: Training and test error achieved by different Nyström kernel ridge regression algorithms on the remaining 7 regression datasets.



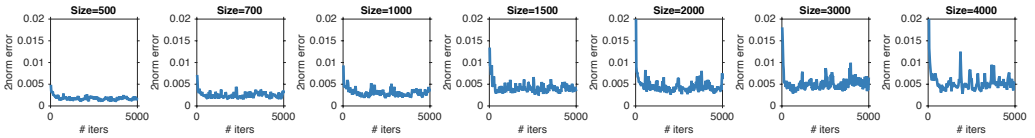
(a) Training error



(b) Test error



(c) Relative Frobenius norm error



(d) Relative Spectral norm error

Figure 9: Performance of Markov chain DPP-Nyström with 50 landmarks on Ailerons. Runs for 5,000 iterations.

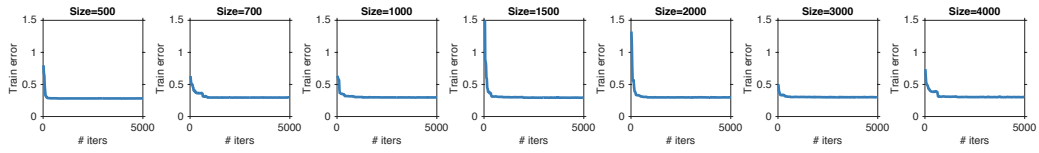
B.3 Mixing of Markov Chain k-Dpp

We first show the mixing of the Gibbs DPP-Nyström with 50 landmarks with different performance measures: relative spectral norm error, training error and test error of kernel ridge regression in Fig. 9.

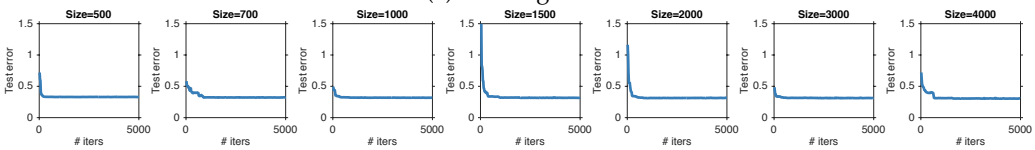
We also show corresponding results with respect to 100 and 200 landmarks in Fig. 10 and Fig. 11, so as to illustrate that for varying number of landmarks the chain is indeed fast mixing and will give reasonably good result within a small number of iterations.

B.4 Running Time Analysis

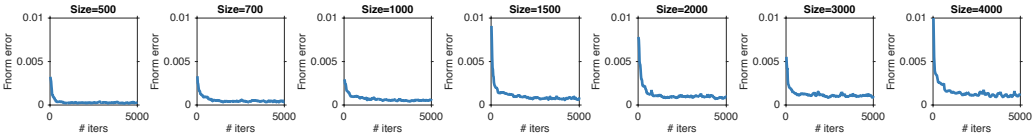
We next show time-error trade-offs for various sampling methods on small and larger datasets with respect to Fnorm and 2norm errors. We sample 20 landmarks from Ailerons dataset of size 4,000 and California Housing of size 12,000. The result is shown in Figure 12 and Figure 13 and similar trends as the example results in the main text could be spotted: on small scale dataset (size 4,000) `kDPP` get very good time-error trade-off. It is more efficient than `Kmeans`, though the error is a bit larger. While on larger dataset (size 12,000) the efficiency is further enhanced while the error is even lower than `Kmeans`. It also have lower variances in both cases compared to `AppLev` and `AppRegLev`. Overall, on larger dataset we obtain the best time-error trade-off with `kDPP`.



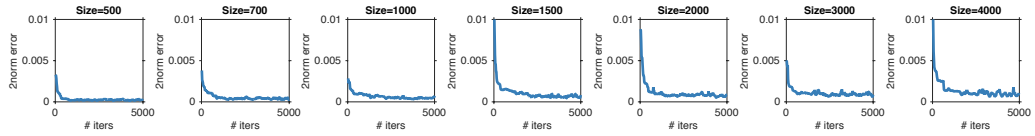
(a) Training error



(b) Test error

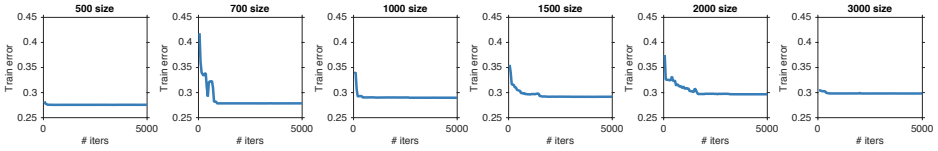


(c) Relative Frobenius norm error

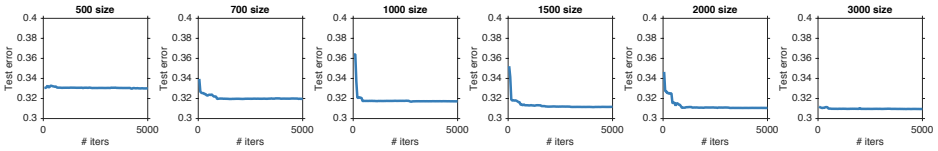


(d) Relative Spectral norm error

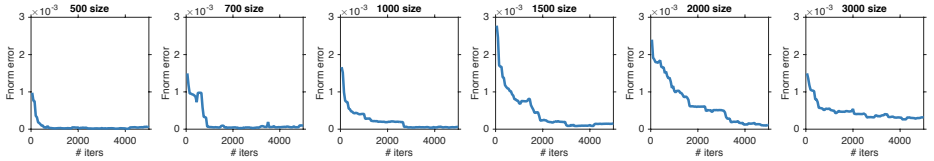
Figure 10: Performance of Markov chain DPP-Nyström with 100 landmarks on Ailerons. Runs for 5,000 iterations.



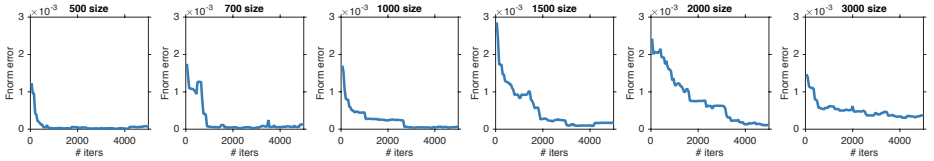
(a) Training error



(b) Test error

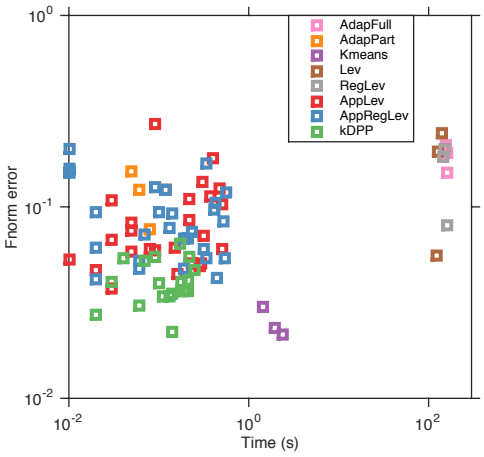


(c) Relative Frobenius norm error

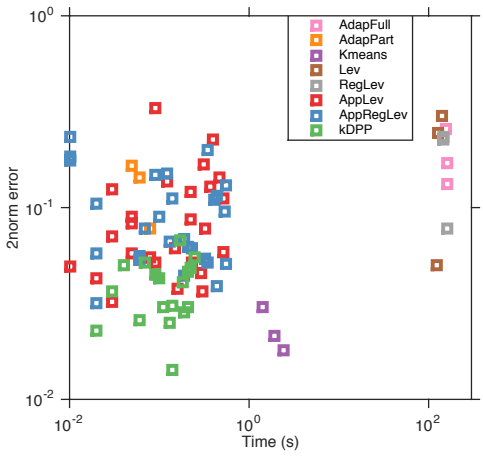


(d) Relative Spectral norm error

Figure 11: Performance of Markov chain DPP-Nyström with 200 landmarks on Ailerons. Runs for 5,000 iterations.

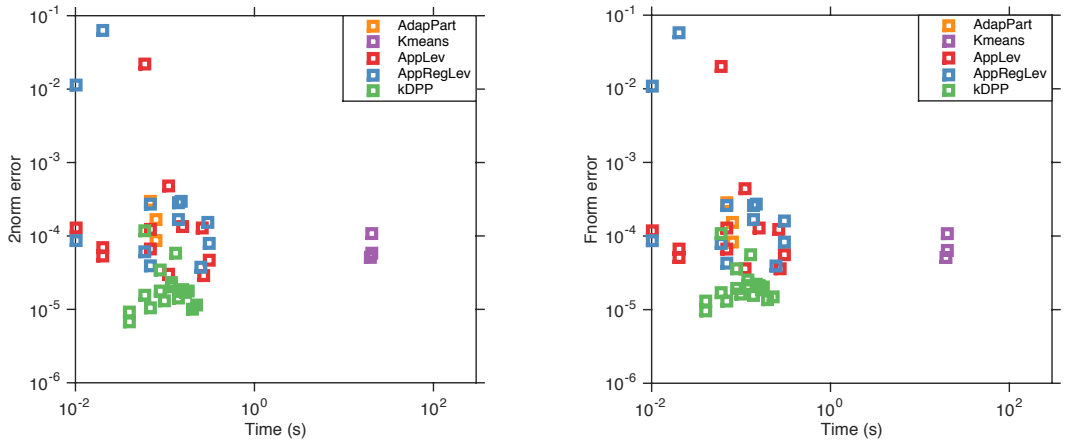


(a) Fnorm Error vs. Time



(b) 2norm Error vs. Time

Figure 12: Time-Error tradeoff with 20 landmarks on Ailerons of size 4,000. Time and Errors shown in log-scale.



(a) 2norm Error vs. Time

(b) Training Error vs. Time

Figure 13: Time-Error tradeoff with 20 landmarks on California Housing of size 12,000. Time and Errors shown in log-scale. We didn't include `AdapFull`, `Lev` and `RegLev` due to their inefficiency on larger datasets.