Proceedings of Machine Learning Research vol 99:1–21, 2019 32nd Annual Conference on Learning Theory

Fast determinantal point processes via distortion-free intermediate sampling

Michał Dereziński MDEREZIN@BERKELEY.EDU
Department of Statistics, UC Berkeley

Editors: Alina Beygelzimer and Daniel Hsu

Abstract

Given a fixed $n \times d$ matrix $\mathbf{X}$, where $n \gg d$, we study the complexity of sampling from a distribution over all subsets of rows where the probability of a subset is proportional to the squared volume of the parallelepiped spanned by the rows (a.k.a. a determinantal point process). In this task, it is important to minimize the preprocessing cost of the procedure (performed once) as well as the sampling cost (performed repeatedly). To that end, we propose a new determinantal point process algorithm which has the following two properties, both of which are novel: (1) a preprocessing step which runs in time $O\big(\text{number-of-non-zeros}(\mathbf{X}) \cdot \log n\big) + \text{poly}(d)$, and (2) a sampling step which runs in $\text{poly}(d)$ time, independent of the number of rows n. We achieve this by introducing a new *regularized* determinantal point process (R-DPP), which serves as an intermediate distribution in the sampling procedure by reducing the number of rows from n to $\text{poly}(d)$. Crucially, this intermediate distribution does not distort the probabilities of the target sample. Our key novelty in defining the R-DPP is the use of a Poisson random variable for controlling the probabilities of different subset sizes, leading to new determinantal formulas such as the normalization constant for this distribution. Our algorithm has applications in many diverse areas where determinantal point processes have been used, such as machine learning, stochastic optimization, data summarization and low-rank matrix reconstruction.

Keywords: determinantal point processes, subset selection, low-rank approximation

1. Introduction

Determinantal point processes (DPP) form a family of distributions sampling diverse subsets of points from a given domain, where the diversity is measured by the squared volume of the parallelepiped spanned by the points in some predefined space. DPPs have found applications in a variety of fields such as physics (Macchi, 1975), statistics (Bardenet et al., 2017), machine learning (Kulesza and Taskar, 2012), computational geometry (Deshpande et al., 2006), graph theory (Guenoche, 1983) and others. The applications include:

1. *Data summarization and diverse recommendation* (e.g., Gillenwater et al., 2012; Gong et al., 2014; Lin and Bilmes, 2011): selecting a representative sample of items, e.g. documents in a corpus, frames in a video or products in an online store.
2. *Row-based low-rank matrix reconstruction* (e.g., Deshpande et al., 2006; Guruswami and Sinop, 2012): determinantal sampling is the optimal way of selecting few rows of a matrix that preserve its low-rank approximation.
3. *Stochastic optimization and Monte Carlo sampling* (e.g., Zhang et al., 2017; Bardenet and Hardy, 2016): DPP sampling has been used to reduce the variance inherent in i.i.d. sampling and improve convergence.

Let $\mathbf{X} \in \mathbb{R}^{n\times d}$ be a tall and thin matrix, i.e. $n \gg d$. Determinantal point process $\mathrm{DPP}(\mathbf{X})$ is defined as a distribution over all 2^n subsets $S \subseteq \{1..n\}$ such that

$$\Pr(S) \;=\; \frac{\det(\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)}, \tag{1}$$

where $\mathbf{X}_S \in \mathbb{R}^{|S|\times d}$ denotes the submatrix of $\mathbf{X}$ containing rows indexed by S. DPP algorithms are typically divided into a preprocessing step, which needs to be performed once per given matrix $\mathbf{X}$, and a sampling step, which happens each time we wish to sample set $S \sim \mathrm{DPP}(\mathbf{X})$. In this paper, we improve on the best known time complexity of both steps by demonstrating:

1. the first DPP algorithm with *input sparsity time* preprocessing: $\mathrm{nnz}(\mathbf{X})\log n + \mathrm{poly}(d)$;
2. the first exact DPP algorithm s.t. sampling takes $\mathrm{poly}(d)$ time, independent of n.

Here, $\mathrm{nnz}(\mathbf{X})$ denotes the number of non-zero entries. Before this work, the best known DPP algorithms (see Section 2 and references therein) had preprocessing times $\Theta(nd^2)$, or $\Omega(\mathrm{nnz}(\mathbf{X})k^2)$ if we allow conditioning on a fixed subset size $|S| = k$ (which can be as large as d), and sampling times at least $\Omega(n\,|S|)$. Our DPP algorithm is based on the following general recipe for sampling from some "target" joint distribution: $i)$ generate a larger sample of $\mathrm{poly}(d)$ rows $\sigma = (\sigma_1,\dots,\sigma_k) \in \{1..n\}^k$ from an "intermediate" distribution; $ii)$ downsample to a smaller subset using the "target" distribution. Crucially, the intermediate sampling is not allowed to distort the target distribution:

$$\textbf{Goal:}\quad \sigma \overset{i)}{\sim} \text{intermediate}\overbrace{(\mathbf{X})}^{n\times d} \text{ and } S \overset{ii)}{\sim} \text{target}\overbrace{(\mathbf{X}_\sigma)}^{\mathrm{poly}(d)\times d} \quad\Longrightarrow\quad \overbrace{\{\sigma_i\}_{i\in S}}^{\sigma_S} \sim \overbrace{\text{target}(\mathbf{X})}^{\mathrm{DPP}(\mathbf{X})}.$$

A simplified version of this approach was recently suggested by Dereziński et al. (2018). They sample from a different but related family of determinantal distributions called *size $k \geq d$ volume sampling*, which has the unique property that it can play the role of both the "intermediate" and the "target" distribution. DPPs on the other hand do not have that property and no candidate for an intermediate distribution was previously known. To that end, we develop a new family of joint distributions: *regularized* determinantal point processes (R-DPP). For a p.s.d. matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$ and a Poisson mean parameter $r > 0$ an $\text{R-DPP}^r(\mathbf{X},\mathbf{A})$ samples over all sequences $\sigma \in \bigcup_{k=0}^\infty \{1..n\}^k$ so that

$$\Pr(\sigma) \;\propto\; \det\big(\mathbf{A}+\mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big)\,\frac{r^k\mathrm{e}^{-r}}{k!}, \qquad \text{where } k \text{ is the length of } \sigma.$$

Rescaling with Poisson probabilities $\frac{r^k\mathrm{e}^{-r}}{k!}$ is *essential* for the normalization constant of this distribution to have a closed form (a key part of our analysis). We obtain it via the following new determinantal formula: if $\sigma = (\sigma_1,\dots,\sigma_K)$ is a sequence of K i.i.d. samples from $\{1..n\}$, then

$$\text{for}\quad K \sim \text{Poisson}(r), \qquad \mathbb{E}\Big[\det\big(\mathbf{A}+\mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big)\Big] = \det\Big(\mathbf{A}+\mathbb{E}\big[\mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big]\Big),$$

where the expectations are over the random variable K as well as the i.i.d. samples. Similar formulas have been known for fixed length K but only in the unregularized case where $\mathbf{A} = \mathbf{0}$ (see Dereziński et al., 2019). Our result shows that regularization can be introduced by randomizing the sequence

length with Poisson distribution. In Section 4 we use this formula to show a number of connections between R-DPPs and DPPs. For example, the latter can be obtained from the former when the Poisson mean parameter r converges to 0 along with regularization $\mathbf{A}$ set to $\frac{r}{n}\mathbf{I}$:

$$\text{R-DPP}^r\Big(\mathbf{X},\ \frac{r}{n}\mathbf{I}\Big) \xrightarrow{r\to 0} \text{DPP}(\mathbf{X}).$$

This means that the family of R-DPPs contains DPPs in its closure and can thus be viewed as an extension. We also show that DPPs are preserved under subsampling with an R-DPP:

$$\text{if} \quad \sigma \sim \text{R-DPP}^r(\mathbf{X}, \mathbf{I}) \quad \text{and} \quad S \sim \text{DPP}(\mathbf{X}_\sigma), \quad \text{then} \quad \sigma_S \sim \text{DPP}\Big(\sqrt{\tfrac{r}{n}}\,\mathbf{X}\Big).$$

This suggests that R-DPP is a good candidate for an "intermediate" distribution when sampling a DPP. To make sampling from R-DPPs efficient, we further generalize them so that the marginal row probabilities can be reweighted with an arbitrary i.i.d. distribution. On a very high level, our strategy is to show that when the length of sequence σ is sufficiently large in expectation (but still independent of n) and the R-DPP is reweighted by ridge leverage scores, then it becomes very close to i.i.d. sampling, so we can use that as a proposal distribution for a rejection sampling scheme.

In the following section we give some background and related work on DPP algorithms, then in Section 3 we present our main algorithm (Algorithm 2) and the associated result (Theorem 2), along with an example application in low-rank matrix reconstruction. Section 4 introduces R-DPPs along with their basic properties and the remaining Sections 5 and 6 are devoted to proving Theorem 2.

2. Background and related work

Several settings have been used in the literature for studying the complexity of DPP algorithms. Below we review those which are slightly different than the one we presented in Section 1, but are still relevant to our discussion (see Kulesza and Taskar, 2012, for details).

L-ensemble We defined a DPP in terms of a matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, where each element $i \in \{1..n\}$ is described by a row vector $\mathbf{x}_i^\top \in \mathbb{R}^d$ (suggested by Deshpande et al., 2006; Kulesza and Taskar, 2010). An equivalent parameterization can be defined in terms of the so-called *ensemble* matrix $\mathbf{L} = \mathbf{X}\mathbf{X}^\top \in \mathbb{R}^{n\times n}$, where the (i,j)th entry represents the dot product $\mathbf{x}_i^\top \mathbf{x}_j$. In this case, the probability (1) of a subset $S \subseteq \{1..n\}$ can be written as $\Pr(S) = \frac{\det(\mathbf{L}_{S,S})}{\det(\mathbf{I}+\mathbf{L})}$, where $\mathbf{L}_{S,S} \in \mathbb{R}^{|S|\times|S|}$ denotes the submatrix of $\mathbf{L}$ with both row and column entries indexed by S. Naturally, one representation can be converted to the other in preprocessing, so our results are still useful in the $\mathbf{L}$-ensemble case when $\text{rank}(\mathbf{L}) \ll n$. However, since matrix $\mathbf{L}$ is much larger than $\mathbf{X}$, the preprocessing cost may increase.

k-DPP In some practical applications, when a subset of particular size is desired, a DPP can be restricted only to subsets S such that $|S| = k$ for some $k \in \{1..d\}$, and referred to as a k-DPP (Kulesza and Taskar, 2011). This is equivalent to sampling a set S from a standard DPP, but accepting the sample only if $|S| = k$. This distribution is also sometimes called *size $k \leq d$ volume sampling* (Deshpande et al., 2006), not to be confused with size $k \geq d$ volume sampling mentioned in Section 1. The special case of $k = d$ will be further discussed in Section 2.2. An alternative way of controlling the subset size is by rescaling matrix $\mathbf{X}$ with some α so that the expected subset size for $S \sim \text{DPP}(\alpha\mathbf{X})$, i.e. $\mathbb{E}\big[|S|\big]$ matches a desired value (see Section 3 for more details). A

restriction to k-DPP can lead to faster sampling algorithms (when k is small), as discussed in the following sections.

A classical DPP algorithm introduced by Hough et al. (2006) uses the singular value decomposition (SVD) of either $\mathbf{L}$ or $\mathbf{X}$ to produce an exact sample S from the DPP in time $O(n\,|S|^2)$. However, this runtime does not include the cost of SVD, considered as a preprocessing step, which takes $O(n^3)$ and $O(nd^2)$ for $\mathbf{L}$ and $\mathbf{X}$, respectively. Since then, a number of methods were proposed to improve on this basic approach. We survey these techniques in the following two sections, and present a runtime comparison in Tables 1 and 2 (we omit the big-O notation in the tables).

	preprocessing
SVD	nd^2
sketching	$\mathrm{nnz}(\mathbf{X})k^2 + nk^4$
MCMC	$nd \cdot \mathrm{poly}(k)$
this paper	$\mathrm{nnz}(\mathbf{X}) + d^3k^2$

Table 1: Preprocessing costs for approximate k-DPPs from $\mathbf{X}$ (omitting log terms), compared to our DPP algorithm.

	sampling
bottom-up	nk^2
i.i.d.+top-down	$nk + k^4$
MCMC	$n \cdot \mathrm{poly}(k)$
this paper	d^3k

Table 2: Sampling cost (after preprocessing) for DPP/k-DPP methods, compared to our DPP algorithm ($k = |S| \leq d \leq n$).

2.1. Previous approximate preprocessing techniques

Approximate preprocessing methods were studied primarily for k-DPPs (i.e. $|S| = k$) rather than for standard DPPs, because bounding the subset size makes volume approximations easier to control. Deshpande and Rademacher (2010) and Gillenwater et al. (2012) suggested to use volume-preserving sketching of Magen and Zouzias (2008) to reduce the dimension d of matrix $\mathbf{X}$ to $r = \widetilde{O}(k^2)$, which allows approximate sampling from a k-DPP. Here, the cost of sketching is $\widetilde{O}(\mathrm{nnz}(\mathbf{X})k^2)$, and it is followed by computing SVD of an $n \times r$ matrix in time $\widetilde{O}(nk^4)$. Also, Anari et al. (2016) proposed to use a fast-mixing MCMC algorithm whose stationary distribution is a k-DPP, where preprocessing cost is $O(nd \cdot \mathrm{poly}(k))$ for matrix $\mathbf{X}$, and $O(n \cdot \mathrm{poly}(k))$ for matrix $\mathbf{L}$ (sampling time is similar). Other approximation techniques such as Nystrom (Affandi et al., 2013) and coresets (Li et al., 2016) yield approximate DPP distributions however their accuracy is data-dependent. Table 1 compares the preprocessing costs for the approximate k-DPP methods offering data-independent accuracy guarantees with that of our DPP algorithm. Note that our approach is specifically designed for sampling from a full DPP, not a k-DPP. Also, unlike these approximate methods our algorithm samples exactly from $\mathrm{DPP}(\rho\mathbf{X})$ for some $\rho \approx 1$, which is a much more precise guarantee.

2.2. Sampling a DPP as a mixture of volume samples

Any DPP is a mixture of so-called *elementary DPPs* (see Hough et al., 2006; Kulesza and Taskar, 2012). These elementary DPPs have been independently studied in the context of *volume sampling* (Avron and Boutsidis, 2013). For $\mathbf{X} \in \mathbb{R}^{n\times d}$, volume sampling is given by

$$S \sim \mathrm{VS}(\mathbf{X}): \quad \Pr(S) = \frac{\det(\mathbf{X}_S)^2}{\det(\mathbf{X}^\top\mathbf{X})} \quad \text{for } S \text{ s.t. } |S| = d.$$

The mixture decomposition shown by Hough et al. (2006) implies that DPP sampling can be divided into two steps: first sample one element from the mixture, then generate a sample from that elementary DPP. Specifically, consider the eigendecompositions $\mathbf{L} = \sum_{i=1}^{d} \lambda_i \mathbf{v}_i \mathbf{v}_i^\top$ and $\mathbf{X}^\top\mathbf{X} = \sum_{i=1}^{d} \lambda_i \widehat{\mathbf{v}}_i \widehat{\mathbf{v}}_i^\top$. For convenience, let us put the eigenvectors of $\mathbf{L}$ into a matrix $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_d] \in \mathbb{R}^{n \times d}$. Then $S \sim \mathrm{DPP}(\mathbf{X})$ can be produced by sampling a subset T of eigenvector indices (step 1) and performing volume sampling w.r.t. the $n \times |T|$ matrix constructed from those vectors (step 2):

$$S \sim \mathrm{VS}\big(\mathbf{V}_{*,T}\big), \qquad \text{where for each } i \in \{1..d\}, \text{ independently,} \qquad \Pr(i \in T) = \frac{\lambda_i}{1+\lambda_i}.$$

Here, $\mathbf{V}_{*,T}$ denotes the submatrix of $\mathbf{V}$ consisting of the columns indexed by T. We used vectors from the decomposition of $\mathbf{L}$, but this can be easily obtained from the decomposition of $\mathbf{X}^\top\mathbf{X}$ during preprocessing, since $\mathbf{v}_i = \frac{1}{\sqrt{\lambda_i}}\mathbf{X}\widehat{\mathbf{v}}_i$. So given the eigendecomposition we can sample the set T easily, and it remains to perform the volume sampling step.

In Table 2 we review the running times for different approaches of sampling $S \sim \mathrm{VS}\big(\mathbf{V}_{*,T}\big)$, compared to a different MCMC-based DPP sampler (Anari et al., 2016) and our algorithm. The classical approach from DPP literature (Hough et al., 2006) samples "bottom-up", adding one point at a time and at each step projecting the remaining points onto the subspace orthogonal to that point (see Algorithm 1). Curiously, a diametrically opposed "top-down" approach of Dereziński and Warmuth (2018), which eliminates points one at a time instead of adding them, achieves the same asymptotic runtime with high probability. The volume sampling algorithm of Dereziński et al. (2018) further improves on this downsampling strategy by introducing an intermediate i.i.d. oversampling step, which is what inspired our approach. Note that we are the first to apply the "top-down" algorithms to DPP sampling (they were previously known only in the context of volume sampling). Unfortunately, in all of these methods sampling time is linear[1] in n (they have to read the matrix $\mathbf{V}_{*,T}$ for each sampled set) which may not be acceptable when $n \gg d$ and we need to perform the sampling repeatedly. Our algorithm, which uses the mixture decomposition only as a subroutine, samples in time independent of n.

Algorithm 1 Bottom-up volume sampling

input: $\mathbf{V} \in \mathbb{R}^{n \times k}$, s.t. $\mathbf{V}^\top\mathbf{V} = \mathbf{I}$
output: $S \sim \mathrm{VS}(\mathbf{V})$
for $i = 1..k$
  Sample $\sigma_i \sim \big(\|\mathbf{V}_{1,*}\|^2, \ldots, \|\mathbf{V}_{n,*}\|^2\big)$
  $\mathbf{V} \leftarrow \mathbf{V}\Big(\mathbf{I} - \frac{(\mathbf{V}_{\sigma_i,*})^\top \mathbf{V}_{\sigma_i,*}}{\|\mathbf{V}_{\sigma_i,*}\|^2}\Big)$
end for
return $S = \{\sigma_1, \ldots, \sigma_k\}$

3. Main result

As discussed in Section 1, the high level strategy of our algorithm is to design an "intermediate" sampling distribution, which reduces the size of the matrix $\mathbf{X}$ from $n \times d$ to $\mathrm{poly}(d) \times d$ while preserving the "target" distribution (here, a DPP). Another key property of the intermediate distribution is that it has be close to i.i.d., so that we can implement it efficiently. Thus, our algorithm will use an i.i.d. sampling distribution defined by a vector $l = (l_1, \ldots, l_n)$ of importance weights assigned to each row of $\mathbf{X}$, and use it as a proposal for rejection sampling from the intermediate distribution. Overall, the three main components of our method are:

1. A different strategy was proposed by Li et al. (2016), which uses coreset construction to approximately sample from a DPP in time independent of n, however the sampling accuracy is data-dependent.

$$\overbrace{\underbrace{(1)\text{ i.i.d. sampling}}_{\sigma\sim\text{intermediate}(\mathbf{X})}\quad\underbrace{(2)\text{ rejection sampling}}}\qquad\overbrace{(3)\text{ downsampling}}^{S\sim\text{target}(\mathbf{X}_\sigma)}.$$

In Section 4 we define a regularized determinantal point process (R-DPP) and use it as the intermediate distribution. For the i.i.d. sampling weights we use a particular variant of so-called ridge leverage scores (Alaoui and Mahoney, 2015). Let $\mathbf{x}_i^\top$ denote the ith row of $\mathbf{X}$ and w.l.o.g. assume that all rows are non-zero.

Algorithm 2 Fast determinantal point process sampling

1: **input:** $\mathbf{X}\in\mathbb{R}^{n\times d}$, $\mathbf{A}\in\mathbb{R}^{d\times d}$, sampling oracle for $i\sim\tilde{l}=(\tilde{l}_1,\ldots,\tilde{l}_n)$
2: **repeat**
3: sample $K\sim\text{Poisson}(q)$, for $q=\lceil 2d\tilde{s}\rceil$, $\tilde{s}=\text{tr}\big(\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1}\big)$
4: sample $\sigma_1,\ldots,\sigma_K\overset{\text{i.i.d.}}{\sim}(l_1,\ldots,l_n)$, for $l_i=\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i$, rejection sampling via $\tilde{l}$
5: sample $Acc\sim\text{Bernoulli}\Big(\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{C_K\det(\mathbf{I}+\mathbf{A})}\Big)$, for $\widetilde{\mathbf{X}}=\Big[\sqrt{\frac{\tilde{s}}{l_i(q-\tilde{s})}}\,\mathbf{x}_i^\top\Big]$, $C_K=(\frac{q}{q-\tilde{s}})^{K+d}\mathrm{e}^{-\tilde{s}}$
6: **until** $Acc=\text{true}$ (if $Acc=\text{true}$, then $\sigma_1,\ldots,\sigma_K$ is distributed as an R-DPP)
7: **return** $\sigma_{\widetilde{S}}$, where $\widetilde{S}\sim\text{DPP}\big(\widetilde{\mathbf{X}}_\sigma\big)$

Other than matrix $\mathbf{X}$, the algorithm takes additional inputs: matrix $\mathbf{A}\approx\mathbf{X}^\top\mathbf{X}$ and a sampling oracle for $\tilde{l}$ which approximates l. The inputs $\mathbf{A}$ and $\tilde{l}$ are computed in the preprocessing step, which can be easily performed in time $O(nd^2)$, but standard sketching techniques can be used to achieve input sparsity time preprocessing. In line 7 of the algorithm we invoke a different DPP sampling procedure for a matrix of reduced size. This can be for example the classical algorithm discussed in Section 2.2 with the volume sampling part implemented by Algorithm 1 and SVD performed exactly. Our main result shows that Algorithm 2 runs in time independent of n and samples from a determinantal point process. The only trade-off coming from approximate preprocessing is that it samples from $\text{DPP}(\rho\mathbf{X})$ instead of $\text{DPP}(\mathbf{X})$, for some $\rho\approx 1$. This rescaling only affects the distribution of sample size $|\sigma_{\widetilde{S}}|$ (and not the conditional probability given the size), and it also implies a weaker (but more standard) approximation guarantee based on *total variation* distance.

Definition 1 (total variation) *A distribution on a finite domain Ω with probability mass function q is an ϵ-approximation of a distribution on Ω with probability mass function p if*

$$\frac{1}{2}\sum_{x\in\Omega}\big|q(x)-p(x)\big|\leq\epsilon.$$

Theorem 2 *Suppose that $\epsilon\in[0,1]$, $C\geq 0$ and $\mathbf{X}\in\mathbb{R}^{n\times d}$, $\mathbf{A}\in\mathbb{R}^{d\times d}$, $\tilde{l}=(\tilde{l}_1,\ldots,\tilde{l}_n)$ satisfy:*

$$(1-\eta)\mathbf{X}^\top\mathbf{X}\preceq\mathbf{A}\preceq(1+\eta)\mathbf{X}^\top\mathbf{X},\qquad\text{for }\ \eta=\frac{\epsilon}{4\bar{s}+C\ln 9/\epsilon},\tag{2}$$

$$\frac{1}{2}\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i\leq\tilde{l}_i\leq\frac{3}{2}\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i,\qquad\text{for all }\ i\in\{1..n\},\tag{3}$$

where $\bar{s}=\max\{1,\mathbb{E}[|S|]\}\leq d$ for $S\sim\text{DPP}(\mathbf{X})$. The following are true for Algorithm 2:

1. *It returns $\sigma_{\widetilde{S}}\sim\text{DPP}(\rho\mathbf{X})$, where $|\rho-1|\leq\eta$ and $\big|\mathbb{E}[|\sigma_{\widetilde{S}}|]-\mathbb{E}[|S|]\big|\leq\epsilon$;*

2. *There is* $C = O(1)$ *for which this distribution is an* ϵ*-approximation of* $\mathrm{DPP}(\mathbf{X})$*;*

3. *The algorithm has time complexity* $O(d^3 \bar{s} \log^2 1/\delta)$ *w.p. at least* $1 - \delta$*.*

Note that both $\mathbf{X}^\top\mathbf{X}$ and the distribution l can be computed in $O(nd^2)$ time, thereby satisfying conditions (2) and (3) with $\eta = \epsilon = 0$. In this case, Algorithm 2 is an exact DPP sampler that runs in time independent of n. However, if preprocessing cost of $O(nd^2)$ is not acceptable, then the following result offers input sparsity time preprocessing for $\epsilon > 0$.

Proposition 3 *Matrix* $\mathbf{A}$ *and distribution* $\tilde{l}$ *satisfying conditions* (2) *and* (3) *can be obtained from matrix* $\mathbf{X}$ *in time* $O(\mathrm{nnz}(\mathbf{X}) \log n + d^3 \log d \cdot (\bar{s} + \log 1/\epsilon)^2/\epsilon^2)$.

The proof of Theorem 2 is spread out across Sections 5, 6 and the appendices. In Section 4 and Appendix A we define regularized determinantal point processes and show how they can be used for sampling DPPs. Then, in Section 5 and Appendix B we show that Algorithm 2 indeed samples from $\mathrm{DPP}(\rho\mathbf{X})$ (ϵ-approximation is proven in Appendix C). Finally, in Section 6 we present the key steps in proving the time bounds, with the details (including the proof of Proposition 3) given in Appendix D.

Application: row-based low-rank matrix reconstruction

One application of DPPs aims to find a small subset $S \subseteq \{1..n\}$ of rows of matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ such that the subspace spanned by those rows captures the full matrix nearly as well as the best rank-k approximation in terms of the Frobenius norm $\|\cdot\|_F$. Let $\mathbf{P}_S = (\mathbf{X}_S)^+\mathbf{X}_S$ be the projection matrix onto the span of vectors $\{\mathbf{x}_i\}_{i\in S}$. Guruswami and Sinop (2012) showed that if $S \sim \mathrm{DPP}(\mathbf{X})$ then for any $k \leq s \leq d$,

$$\mathbb{E}\Big[\big\|\mathbf{X} - \mathbf{X}\mathbf{P}_S\big\|_F^2 \;\Big|\; |S| = s\Big] \leq \frac{s+1}{s+1-k}\,\|\mathbf{X} - \mathbf{X}_{(k)}\|_F^2 \quad \text{where} \quad \mathbf{X}_{(k)} = \operatorname*{argmin}_{\mathbf{Y}\colon \mathrm{rank}(\mathbf{Y})=k} \|\mathbf{X} - \mathbf{Y}\|_F,$$

and that for any $s = o(n)$ the bound is tight up to lower order terms.[2] In particular, for the ratio $\frac{s+1}{s+1-k}$ to become $1+\epsilon$, we need $s_{k,\epsilon} = k/\epsilon + k - 1$ rows sampled from a DPP. Even though it is most natural to use fixed-size DPPs in this context, the sample size $|S|$ for a standard DPP is sufficiently concentrated around its mean to offer near-optimal guarantees for this task. In fact, as shown by Pemantle and Peres (2014) (see Lemma 14 in Appendix C), for $S \sim \mathrm{DPP}(\mathbf{X})$, w.p. $\geq \frac{1}{2}$ we have $\big||S| - \bar{s}\big| \leq c\sqrt{\bar{s}}$ for some absolute constant c, where $\bar{s} = \max\{1, \mathbb{E}[|S|]\}$. The expected sample size is derived as:

$$\mathbb{E}\big[|S|\big] = \sum_{i=1}^{d} \frac{\lambda_i}{1+\lambda_i}, \quad \text{where} \quad \lambda_1, \ldots, \lambda_d \text{ are the eigenvalues of } \mathbf{X}^\top\mathbf{X}. \tag{4}$$

Since $\bar{s}$ is monotonic w.r.t. the eigenvalues we can use a simple binary search to find a rescaling $\alpha\mathbf{X}$ for which $\bar{s} - c\sqrt{\bar{s}} \approx s_{k,\epsilon}$. Thus, if $S \sim \mathrm{DPP}(\alpha\mathbf{X})$, then for $\Delta_{k,\epsilon} = [s_{k,\epsilon}, s_{k,\epsilon} + \delta_{k,\epsilon}]$, where $\delta_{k,\epsilon} = 2c\sqrt{\bar{s}} = O\big(\sqrt{s_{k,\epsilon}}\big)$, we have:

2. Other methods are known for this and related tasks which achieve near-optimal bounds (e.g., see Deshpande et al., 2006; Boutsidis et al., 2011; Boutsidis and Woodruff, 2017).

$$\mathbb{E}\Big[\big\|\mathbf{X}-\mathbf{X}\mathbf{P}_S\big\|_F^2 \;\big|\; |S|\in\Delta_{k,\epsilon}\Big] \le (1+\epsilon)\,\|\mathbf{X}-\mathbf{X}_{(k)}\|_F^2 \quad \text{and} \quad \Pr\big(|S|\in\Delta_{k,\epsilon}\big)\ge\frac{1}{2}.$$

We can obtain the same guarantee if we replace the eigenvalues of $\mathbf{X}^\top\mathbf{X}$ in (4) with those of matrix $\mathbf{A}$ satisfying condition (2) with $\eta=\frac{1}{4s_{k,\epsilon}}$, so by Theorem 2 the total cost of obtaining such a sample would be[3] $O(\mathrm{nnz}(\mathbf{X})\log n + d^3\log d\cdot s_{k,\epsilon}^2)$. Note that we needed the fact that Algorithm 2 returns $\mathrm{DPP}(\rho\mathbf{X})$ rather than just an approximation of $\mathrm{DPP}(\mathbf{X})$. This raises the following natural question: can our techniques be extended to sampling from fixed-size DPPs, so that the optimal sample size $s_{k,\epsilon}$ can be achieved exactly and in time $O(\mathrm{nnz}(\mathbf{X})\log n+\mathrm{poly}(d))$? We leave this as a new direction for future work.

4. Regularized determinantal point processes (R-DPP)

We propose a new family of determinantal sampling distributions which will be used in Sections 5 and 6 to prove Theorem 2. The crucial property of this family is that the determinantal sampling probabilities can be regularized by adding an arbitrary fixed positive semi-definite (p.s.d.) matrix inside of the determinant, while maintaining many of the natural properties of a DPP, such as a simple normalization constant. This is achieved by controlling the size of the sample with a Poisson random variable. In the proofs that follow we will use the shorthand $[n]\overset{def}{=}\{1..n\}$.

Definition 4 *Given matrix $\mathbf{X}\in\mathbb{R}^{n\times d}$, distribution $p=(p_1,\dots,p_n)$, p.s.d. matrix $\mathbf{A}\in\mathbb{R}^{d\times d}$ and $r>0$, we define $\text{R-DPP}_p^r(\mathbf{X},\mathbf{A})$ as a distribution over all index sequences $\tilde\sigma\in\bigcup_{k=0}^\infty\{1..n\}^k$, s.t.*

$$\Pr(\tilde\sigma)=\frac{\det(\mathbf{A}+\mathbf{X}_{\tilde\sigma}^\top\mathbf{X}_{\tilde\sigma})}{\det\big(\mathbf{A}+r\,\mathbb{E}_{j\sim p}[\mathbf{x}_j\mathbf{x}_j^\top]\big)}\,\frac{r^k e^{-r}}{k!}\prod_{i=1}^k p_{\tilde\sigma_i},\qquad \textit{for}\quad \tilde\sigma\in\{1..n\}^k. \tag{5}$$

Whenever p is uniform, we will write $\text{R-DPP}^r(\mathbf{X},\mathbf{A})$. Of course, we need to establish that this is in fact a valid distribution, i.e. that it sums to one. We achieve this by showing a new variant of the classical Cauchy-Binet formula (the classical formula is stated in (6) below).

Lemma 5 *Given $\mathbf{X}\in\mathbb{R}^{n\times d}$ and p.s.d. $\mathbf{A}\in\mathbb{R}^{d\times d}$, if $\sigma=(\sigma_1,\dots,\sigma_K)\overset{\text{i.i.d.}}{\sim}p=(p_1,\dots,p_n)$, then*

$$\textit{for}\quad K\sim\text{Poisson}(r),\qquad \mathbb{E}\Big[\det\big(\mathbf{A}+\mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big)\Big]=\det\Big(\mathbf{A}+\mathbb{E}\big[\mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big]\Big).$$

Remark 6 *The classical Cauchy-Binet formula states that for a matrix $\mathbf{X}\in\mathbb{R}^{n\times d}$, we have*

$$\sum_{S\subseteq[n]:\,|S|=d}\det(\mathbf{X}_S)^2=\det(\mathbf{X}^\top\mathbf{X}). \tag{6}$$

Probabilistic extensions of the formula previously appeared in the context of volume sampling (Dereziński et al., 2019), and also much earlier in a different context (van der Vaart, 1965). In these cases $\mathbf{A}=\mathbf{0}$ and K is fixed.

3. If we forgo exact DPP sampling, then *projection-cost preserving sketches* (Cohen et al., 2015) may offer further speed-ups.

Proof Previously shown identities for fixed size K do not generalize naturally to the regularized setting unless randomness in K is introduced. We start the proof by applying the Cauchy-Binet formula (6) to the term under the expectation. Let $\mathbf{A} = \mathbf{B}^\top\mathbf{B}$ be any decomposition of $\mathbf{A}$ s.t. $\mathbf{B} \in \mathbb{R}^{b\times d}$ where $b = \operatorname{rank}(\mathbf{A})$. To apply the Cauchy-Binet formula, we sum over all d-element subsets of the union of rows of matrices $\mathbf{B}$ and $\mathbf{X}_\sigma$:

$$\det\big(\mathbf{A} + \mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big) = \sum_{\substack{S\subseteq[b]\\ |S|\geq d-K}} \sum_{\substack{T\subseteq[K]:\\ |T|=d-|S|}} \det\left(\begin{bmatrix}\mathbf{B}_S\\ \mathbf{X}_{\sigma_T}\end{bmatrix}\right)^2.$$

Applying the law of total expectation w.r.t. the Poisson variable K, we obtain

$$\begin{aligned}
\mathbb{E}\Big[\det\big(\mathbf{A} + \mathbf{X}_\sigma^\top\mathbf{X}_\sigma\big)\Big] &= \sum_{k=0}^{\infty} \frac{r^k\mathrm{e}^{-r}}{k!} \sum_{\substack{S\subseteq[b]\\ |S|\geq d-k}} \sum_{\substack{T\subseteq[k]:\\ |T|=d-|S|}} \mathbb{E}\left[\det\left(\begin{bmatrix}\mathbf{B}_S\\ \mathbf{X}_{\sigma_T}\end{bmatrix}\right)^2 \mid K=k\right]\\
&\overset{(a)}{=} \sum_{S\subseteq[b]} \sum_{k=d-|S|}^{\infty} \frac{r^k\mathrm{e}^{-r}}{k!}\binom{k}{d-|S|}\mathbb{E}\left[\det\left(\begin{bmatrix}\mathbf{B}_S\\ \mathbf{X}_\sigma\end{bmatrix}\right)^2 \mid K=d-|S|\right]\\
&\overset{(b)}{=} \sum_{S\subseteq[b]} (d-|S|)! \sum_{\substack{T\subseteq[n]:\\ |T|=d-|S|}} \det\left(\begin{bmatrix}\mathbf{B}_S\\ \mathbf{X}_T\end{bmatrix}\right)^2 \left(\prod_{i\in T} p_i\right) \sum_{k=d-|S|}^{\infty} \frac{r^k\mathrm{e}^{-r}}{k!}\binom{k}{d-|S|}\\
&\overset{(c)}{=} \sum_{S\subseteq[b]} \sum_{\substack{T\subseteq[n]:\\ |T|=d-|S|}} \det\left(\begin{bmatrix}\mathbf{B}_S\\ \big[\sqrt{rp_i}\,\mathbf{x}_i^\top\big]_{i\in T}\end{bmatrix}\right)^2 \underbrace{\sum_{k=d-|S|}^{\infty} \frac{r^{k-d+|S|}\mathrm{e}^{-r}}{(k-d+|S|)!}}_{1}\\
&\overset{(d)}{=} \det\Big(\mathbf{B}^\top\mathbf{B} + r\sum_{i=1}^{n} p_i\mathbf{x}_i\mathbf{x}_i^\top\Big),
\end{aligned}$$

where (a) follows from the exchangeability of sequence σ (so that the value of the expectation is the same for any subset T), in (b) we expand the expectation and note that only unique sequences σ will have a non-zero determinant (hence the switch to subsets and the term $(d-|S|)!$) and in (c) we absorb the factors $r^{d-|S|}$ and p_i into the determinant by treating the product of p_i's as the determinant of a $d\times d$ diagonal matrix with 1's at diagonal entries corresponding to the rows of $\mathbf{B}_S$ and $\sqrt{rp_i}$ at the entries corresponding to $\mathbf{X}_T$, and then using that $\det(\mathbf{V}\mathbf{W}) = \det(\mathbf{V})\det(\mathbf{W})$. Finally, (d) is the classical Cauchy-Binet. It was crucial that we were able to absorb the subset size $|S|$ into the Poisson series, which allowed the formula to collapse to a single determinant. This completes the proof because $\mathbb{E}[\mathbf{X}_\sigma^\top\mathbf{X}_\sigma] = \mathbb{E}[K]\,\mathbb{E}[\mathbf{x}_{\sigma_1}\mathbf{x}_{\sigma_1}^\top] = r\sum_i p_i\mathbf{x}_i\mathbf{x}_i^\top$. ■

In what sense is R-DPP a natural extension of a determinantal point process? Naively, we might say that setting the matrix $\mathbf{A}$ to an all-zeros matrix would recover the classical distribution, however this is not the case because when $\mathbf{A}=\mathbf{0}$ only samples of size d or larger will have non-zero probability. Instead, we can demonstrate a connection to both DPP and volume sampling distributions in a different way: we show that they can be obtained as the limiting distributions of R-DPP when the regularization and sample size parameter r converge to zero (in two separate ways), which is remarkable as the two distributions in most cases produce vastly different samples.

Theorem 7 *For $\mathbf{X} \in \mathbb{R}^{n\times d}$, ignoring the ordering in the sequences sampled with* R-DPP*, we have*

$$\text{R-DPP}^r\Big(\mathbf{X},\ \frac{r}{n}\mathbf{I}\Big) \overset{r\to 0}{\longrightarrow} \text{DPP}(\mathbf{X}) \quad \text{(pointwise)},$$

$$\textit{whereas} \qquad \text{R-DPP}^r(\mathbf{X}, \mathbf{0}) \overset{r\to 0}{\longrightarrow} \text{VS}(\mathbf{X}) \quad \text{(pointwise)}.$$

Theorem 7 suggests that R-DPPs are likely to be of independent interest as an extension of DPPs. However, it does not say how to use them algorithmically. To that end, we make a second observation, which essentially states that DPPs are preserved under subsampling with R-DPPs. See Appendix A for proofs of Theorems 7 and 8.

Theorem 8 *For any $\mathbf{X} \in \mathbb{R}^{n\times d}$, $\alpha > 0$ and distribution p over $\{1..n\}$ s.t. $p_i > 0$, let $\widetilde{\mathbf{X}}$ denote matrix $\mathbf{X}$ with ith row rescaled by $\frac{1}{\sqrt{\alpha p_i}}$ for every $i \in \{1..n\}$. It follows that for any $r > 0$,*

$$\textit{if} \quad \widetilde{\sigma} \sim \text{R-DPP}^r_p\big(\widetilde{\mathbf{X}}, \mathbf{I}\big) \ \textit{ and } \ S \sim \text{DPP}\big(\widetilde{\mathbf{X}}_{\widetilde{\sigma}}\big), \quad \textit{then} \quad \widetilde{\sigma}_S \sim \text{DPP}\Big(\sqrt{\tfrac{r}{\alpha}}\,\mathbf{X}\Big).$$

5. Correctness of Algorithm 2

We present the first part of the proof of Theorem 2 by establishing that Algorithm 2 produces a sample from a determinantal point process. In fact, we will prove the following more precise claim:

Lemma 9 *Given $\mathbf{X} \in \mathbb{R}^{n\times d}$ and a non-zero p.s.d. matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$, Algorithm 2 returns $S \sim \text{DPP}(\rho\mathbf{X})$, with $\rho^2 = \frac{\tilde{s}}{\hat{s}}$ where $\tilde{s} = \text{tr}(\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1})$ and $\hat{s} = \sum_{i=1}^n \mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i$.*

Proof The general idea of the proof is to show that the main repeat loop is implementing an R-DPP, so that we can invoke Theorem 8. Central to this fact is the choice of numerical factor appearing in the denominator of Bernoulli sampling probability in line 5, which we denote here as $C_K = (\frac{q}{q-\tilde{s}})^{K+d}\mathrm{e}^{-\tilde{s}}$. This factor has to depend on K because otherwise the determinantal term will always dominate it for large enough K. This means that C_K needed to be carefully chosen so that:

1. the acceptance probability of line 5 is always bounded by 1,
2. we have control over how the presence of C_K changes the distribution of K,
3. and C_K is not too large so that we may have a good chance of accepting the sample.

We start by showing that the Bernoulli sampling probability in line 5 is in fact bounded by 1. We will use the following simple inequality (proven in Appendix B):

Lemma 10 *For any $d \geq 1$, $\epsilon \in [0,1]$, and non-negative integers k, q s.t. $q \geq \epsilon d$ we have*

$$\Big((1-\epsilon) + \frac{\epsilon\, k}{q}\Big)^d \leq \Big(\frac{q}{q-\epsilon d}\Big)^k e^{-\epsilon d}.$$

Let $\widetilde{\mathbf{X}} \in \mathbb{R}^{n\times d}$ be the matrix $\mathbf{X}$ where for each $i \in \{1..n\}$ the ith row is rescaled by $\sqrt{\frac{\tilde{s}}{l_i(q-\tilde{s})}}$, with $l_i = \mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i$ (same as in line 5 of the algorithm), and let $\sigma = (\sigma_1, \ldots, \sigma_K)$. We use

arithmetic-geometric mean inequality for the eigenvalues of matrix $(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)(\mathbf{I}+\mathbf{A})^{-1}$ and the fact that $\mathrm{tr}\big((\mathbf{I}+\mathbf{A})^{-1}\big) = d-\tilde{s}$, obtaining:

$$\begin{aligned}
\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{\det(\mathbf{I}+\mathbf{A})} &= \det\big((\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)(\mathbf{I}+\mathbf{A})^{-1}\big) \\
&\leq \left(\frac{1}{d}\mathrm{tr}\big((\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)(\mathbf{I}+\mathbf{A})^{-1}\big)\right)^d = \left(\frac{d-\tilde{s}}{d} + \frac{1}{d}\mathrm{tr}\big(\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma(\mathbf{I}+\mathbf{A})^{-1}\big)\right)^d \\
&= \left(1-\frac{\tilde{s}}{d} + \frac{\tilde{s}}{d(q-\tilde{s})}\sum_{i=1}^K \overbrace{\frac{1}{l_{\sigma_i}}\mathbf{x}_{\sigma_i}^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_{\sigma_i}}^{1}\right)^d \\
&= \left(1-\frac{\tilde{s}}{d} + \frac{\tilde{s}}{d}\frac{K}{q-\tilde{s}}\right)^d \leq \left(1-\frac{\tilde{s}}{d} + \frac{\tilde{s}}{d}\frac{K}{q}\right)^d\left(\frac{q}{q-\tilde{s}}\right)^d \\
&\overset{(*)}{\leq} \left(\frac{q}{q-\tilde{s}}\right)^{K+d}\mathrm{e}^{-\tilde{s}} = C_K,
\end{aligned}$$

where $(*)$ follows from Lemma 10 invoked with $\epsilon = \frac{\tilde{s}}{d}$ and $k = K$. Having established the validity of the rejection sampling in Algorithm 2, we now compute the distribution of sample σ at the point of exiting the **repeat** loop. Denoting $r = q-\tilde{s}$ as the desired Poisson mean parameter and $\hat{s} = \sum_i l_i = \mathrm{tr}(\mathbf{X}^\top\mathbf{X}(\mathbf{I}+\mathbf{A})^{-1})$ as the normalization for the sampling probabilities in line 4,

$$\Pr(\sigma\,|\,Acc) \;\propto\; \overbrace{\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{(\frac{q}{r})^{K+d}\mathrm{e}^{-\tilde{s}}\det(\mathbf{I}+\mathbf{A})}}^{\Pr(Acc\,|\,\sigma)} \overbrace{\frac{q^K\mathrm{e}^{-q}}{K!}}^{\Pr(K)} \overbrace{\prod_i \frac{l_{\sigma_i}}{\hat{s}}}^{\Pr(\sigma\,|\,K)} \;\propto\; \det\Big(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma\Big)\frac{r^K\mathrm{e}^{-r}}{K!}\prod_{i=1}^K\frac{l_{\sigma_i}}{\hat{s}},$$

where in the above we omitted the normalization for the sake of clarity. Comparing the obtained unnormalized probability to the one given in (5), we conclude that the sample is distributed according to $\text{R-DPP}_l^r\big(\widetilde{\mathbf{X}},\,\mathbf{I}\big)$. Note how the factor C_K interplays with $\Pr(K)$ to "transform" the variable from being Poisson(q) to Poisson(r). Invoking Theorem 8 for the matrix $\mathbf{X}$, $\alpha = \frac{\hat{s}}{\tilde{s}}r$ and distribution $\big(\frac{l_1}{\hat{s}},\ldots,\frac{l_n}{\hat{s}}\big)$ we conclude that Algorithm 2 returns $\sigma_{\widetilde{S}} \sim \mathrm{DPP}(\rho\mathbf{X})$ where $\rho^2 = \tilde{s}/\hat{s}$. ■

To bound the rescaling factor ρ we use condition (2) of Theorem 2 which ensures that $\mathbf{A} = (1\pm\eta)\mathbf{X}^\top\mathbf{X}$, implying that

$$\tilde{s} = \mathrm{tr}\big(\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1}\big) \leq (1+\eta)\,\mathrm{tr}\big(\mathbf{X}^\top\mathbf{X}(\mathbf{I}+\mathbf{A})^{-1}\big) = (1+\eta)\,\hat{s},$$

and similarly $\tilde{s} \geq (1-\eta)\hat{s}$. We obtain $\rho^2 = \tilde{s}/\hat{s} \in [1-\eta, 1+\eta]$ implying that $|\rho-1| \leq \eta$, as claimed in Theorem 2. Having established this, we use formula (4) to similarly show that:

$$\mathbb{E}\big[|\sigma_{\widetilde{S}}|\big] = \mathrm{tr}\big(\rho^2\mathbf{X}^\top\mathbf{X}(\mathbf{I}+\rho^2\mathbf{X}^\top\mathbf{X})^{-1}\big) \leq \left(1+\frac{\epsilon}{4\bar{s}}\right)\cdot\mathbb{E}\big[|S|\big] \leq \mathbb{E}\big[|S|\big] + \epsilon/4, \tag{7}$$

(lower bound follows identically). The total variation bound (restated below) is proven in Appendix C.

Lemma 11 *There is $C > 0$ s.t. for any matrix $\mathbf{X}$ and $\epsilon \leq 1$, if $|\rho^2-1| \leq \frac{\epsilon}{4\bar{s}+C\ln 9/\epsilon}$, where $\bar{s} = \max\{1, \mathbb{E}[|S|]\}$ for $S \sim \mathrm{DPP}(\mathbf{X})$, then $\mathrm{DPP}(\rho\mathbf{X})$ is an ϵ-approximation of $\mathrm{DPP}(\mathbf{X})$.*

6. Efficiency of rejection sampling

We complete the proof of Theorem 2 by bounding the time complexity of Algorithm 2 and that of preprocessing. The key step is to lower bound the probability of exiting the **repeat** loop in lines 2-6. In the following lemma we show that if matrix $\mathbf{A}$ is a sufficiently good approximation of $\mathbf{X}^\top\mathbf{X}$, then the acceptance probability $\Pr(\mathit{Acc} = \text{true})$ is lower bounded by a constant, thus ensuring that the algorithm will leave the loop after only a few iterations.

Lemma 12 *If matrix* $\mathbf{A}$ *satisfies* $\big(1-\frac{1}{4\bar{s}}\big)\mathbf{X}^\top\mathbf{X} \preceq \mathbf{A} \preceq \big(1+\frac{1}{4\bar{s}}\big)\mathbf{X}^\top\mathbf{X}$*, then at the end of each iteration of the* **repeat** *loop in Algorithm 2, we have* $\Pr(\mathit{Acc}=\text{true}) \geq \frac{1}{6}$.

Proof As in lines 3 and 4 of Algorithm 2, let $\sigma = (\sigma_1, \dots, \sigma_K) \overset{\text{i.i.d.}}{\sim} l$ and $K \sim \text{Poisson}(q)$, where $q = \lceil 2d\tilde{s}\rceil$. Recall that at the end of the proof of Lemma 9 we noted how the presence of constant $C_K = (\frac{q}{r})^{K+d}\mathrm{e}^{-\tilde{s}}$, where $r = q - \tilde{s}$, appears to transform the distribution of the sample size K into $\widetilde{K} \sim \text{Poisson}(r)$. This can be seen even more clearly as we compute the acceptance probability after one iteration of the loop:

$$\begin{aligned}
\mathbb{E}\left[\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{(\frac{q}{r})^{K+d}\mathrm{e}^{-\tilde{s}}\det(\mathbf{I}+\mathbf{A})}\right] &= \sum_{k=0}^{\infty}\frac{q^k\mathrm{e}^{-q}}{k!}\mathbb{E}\left[\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{(\frac{q}{r})^{k+d}\mathrm{e}^{-\tilde{s}}\det(\mathbf{I}+\mathbf{A})}\,\Big|\,K=k\right] \\
&= \frac{1}{(\frac{q}{r})^d}\sum_{k=0}^{\infty}\frac{r^k\mathrm{e}^{-r}}{k!}\mathbb{E}\left[\frac{\det(\mathbf{I}+\widetilde{\mathbf{X}}_\sigma^\top\widetilde{\mathbf{X}}_\sigma)}{\det(\mathbf{I}+\mathbf{A})}\,\Big|\,K=k\right] \\
&= \left(\frac{q-\tilde{s}}{q}\right)^d\frac{\mathbb{E}\Big[\det\big(\mathbf{I}+\sum_{i=1}^{\widetilde{K}}\widetilde{\mathbf{x}}_{\sigma_i}\widetilde{\mathbf{x}}_{\sigma_i}^\top\big)\Big]}{\det(\mathbf{I}+\mathbf{A})} \\
&\overset{(*)}{=} \left(1-\frac{\tilde{s}}{q}\right)^d\frac{\det\big(\mathbf{I}+r\,\mathbb{E}\big[\widetilde{\mathbf{x}}_{\sigma_1}\widetilde{\mathbf{x}}_{\sigma_1}^\top\big]\big)}{\det(\mathbf{I}+\mathbf{A})},
\end{aligned}$$

where $(*)$ follows from Lemma 5 applied to $\widetilde{\mathbf{X}}$, distribution l and sample size $\widetilde{K}$. Bernoulli's inequality shows that $(1-\frac{\tilde{s}}{q})^d \geq 1-\frac{d\tilde{s}}{q} \geq \frac{1}{2}$. Furthermore, it is easy to verify that $r\,\mathbb{E}[\widetilde{\mathbf{x}}_{\sigma_1}\widetilde{\mathbf{x}}_{\sigma_1}^\top] = \tilde{s}\,\mathbb{E}[\frac{1}{l_{\sigma_1}}\mathbf{x}_{\sigma_1}\mathbf{x}_{\sigma_1}^\top] = \rho^2\mathbf{X}^\top\mathbf{X}$, where $\rho^2 = \tilde{s}/\hat{s}$. To lower bound the ratio of determinants we use the following lemma shown in Appendix C.

Lemma 13 *For p.s.d. matrices* $\mathbf{B}$, $\mathbf{C}$ *such that* $(1-\gamma)\mathbf{C} \preceq \mathbf{B} \preceq (1+\gamma)\mathbf{C}$ *with* $\gamma \in (0,1)$,

$$e^{-\frac{\gamma}{1-\gamma}s}\det(\mathbf{I}+\mathbf{C}) \leq \det(\mathbf{I}+\mathbf{B}) \leq e^{\gamma s}\det(\mathbf{I}+\mathbf{C}), \quad \textit{where} \quad s = \operatorname{tr}\big(\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}\big).$$

Setting $\mathbf{B} = \mathbf{A}$ and $\mathbf{C} = \rho^2\mathbf{X}^\top\mathbf{X}$, we have $\mathbf{B} \preceq (1+\eta)\mathbf{X}^\top\mathbf{X} \preceq \frac{1+\eta}{1-\eta}\mathbf{C}$ and similar lower bound follows, so applying Lemma 13 with $\gamma = \frac{2\eta}{1-\eta}$ and $s = \operatorname{tr}(\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}) \leq \frac{5}{4}\bar{s}$ (see (7)):

$$\frac{\det(\mathbf{I}+\rho^2\mathbf{X}^\top\mathbf{X})}{\det(\mathbf{I}+\mathbf{A})} \geq \mathrm{e}^{-\frac{2\eta}{1-\eta}\frac{5}{4}\bar{s}} \geq \mathrm{e}^{-\frac{1}{2\bar{s}}\frac{4}{3}\frac{5}{4}\bar{s}} = \mathrm{e}^{-\frac{5}{6}} \geq \frac{1}{3},$$

where we used the fact that $\eta \leq \frac{1}{4\bar{s}}$. Thus the acceptance probability is at least $\frac{1}{2}\cdot\frac{1}{3} = \frac{1}{6}$. ■

The remaining steps in proving the time complexity bound for Algorithm 2 are standard, and so they were relegated to Appendix D along with the proof of Proposition 3.

Acknowledgments

We thank the NSF for funding via the NSF TRIPODS program. Part of this work was done while MD was visiting the Simons Institute for the Theory of Computing.

References

Raja Hafiz Affandi, Alex Kulesza, Emily Fox, and Ben Taskar. Nystrom approximation for large-scale determinantal processes. In Carlos M. Carvalho and Pradeep Ravikumar, editors, *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, pages 85–98, Scottsdale, Arizona, USA, 29 Apr–01 May 2013. PMLR. URL http://proceedings.mlr.press/v31/affandi13a.html.

Ahmed El Alaoui and Michael W. Mahoney. Fast randomized kernel ridge regression with statistical guarantees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 775–783, Montreal, Canada, December 2015.

Nima Anari, Shayan Oveis Gharan, and Alireza Rezaei. Monte carlo markov chain algorithms for sampling strongly rayleigh distributions and determinantal point processes. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 103–115, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR. URL http://proceedings.mlr.press/v49/anari16.html.

Haim Avron and Christos Boutsidis. Faster subset selection for matrices and applications. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1464–1499, 2013.

Rémi Bardenet and Adrien Hardy. Monte Carlo with Determinantal Point Processes. working paper or preprint, May 2016. URL https://hal.archives-ouvertes.fr/hal-01311263.

Rémi Bardenet, Frédéric Lavancier, Xavier Mary, and Aurélien Vasseur. On a few statistical applications of determinantal point processes. *ESAIM: Procs*, 60:180–202, 2017. doi: 10.1051/proc/201760180. URL https://doi.org/10.1051/proc/201760180.

Christos Boutsidis and David P. Woodruff. Optimal cur matrix decompositions. *SIAM Journal on Computing*, 46(2):543–589, 2017. doi: 10.1137/140977898. URL https://doi.org/10.1137/140977898.

Christos Boutsidis, Petros Drineas, and Malik Magdon-Ismail. Near optimal column-based matrix reconstruction. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 305–314, Oct 2011. doi: 10.1109/FOCS.2011.21.

Clément Canonne. A short note on poisson tail bounds. Technical report, Columbia University, 2017.

Kenneth L. Clarkson and David P. Woodruff. Low-rank approximation and regression in input sparsity time. *J. ACM*, 63(6):54:1–54:45, January 2017. ISSN 0004-5411. doi: 10.1145/3019134. URL http://doi.acm.org/10.1145/3019134.

Michael B. Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 163–172, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3536-2. doi: 10.1145/2746539.2746569. URL http://doi.acm.org/10.1145/2746539.2746569.

Michał Dereziński and Manfred K. Warmuth. Reverse iterative volume sampling for linear regression. *Journal of Machine Learning Research*, 19(23):1–39, 2018. URL http://jmlr.org/papers/v19/17-781.html.

Michał Dereziński, Manfred K. Warmuth, and Daniel Hsu. Leveraged volume sampling for linear regression. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 2510–2519. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7517-leveraged-volume-sampling-for-linear-regression.pdf.

Michał Dereziński, Manfred K. Warmuth, and Daniel Hsu. Correcting the bias in least squares regression with volume-rescaled sampling. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 2019.

Amit Deshpande and Luis Rademacher. Efficient volume sampling for row/column subset selection. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 329–338, Las Vegas, USA, October 2010.

Amit Deshpande, Luis Rademacher, Santosh Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, pages 1117–1126, Miami, FL, USA, January 2006.

Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, and David P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *J. Mach. Learn. Res.*, 13(1):3475–3506, December 2012.

Jennifer Gillenwater, Alex Kulesza, and Ben Taskar. Discovering diverse and salient threads in document collections. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 710–720, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=2390948.2391026.

Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. Diverse sequential subset selection for supervised video summarization. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2069–2077. Curran Associates, Inc., 2014.

A Guenoche. Random spanning tree. *Journal of Algorithms*, 4(3):214 – 220, 1983. ISSN 0196-6774. doi: https://doi.org/10.1016/0196-6774(83)90022-6. URL http://www.sciencedirect.com/science/article/pii/0196677483900226.

Venkatesan Guruswami and Ali K. Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1207–1214, Kyoto, Japan, January 2012.

J Ben Hough, Manjunath Krishnapur, Yuval Peres, Bálint Virág, et al. Determinantal processes and independence. *Probability surveys*, 3:206–229, 2006.

Alex Kulesza and Ben Taskar. Structured determinantal point processes. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1171–1179. Curran Associates, Inc., 2010.

Alex Kulesza and Ben Taskar. k-DPPs: Fixed-Size Determinantal Point Processes. In *Proceedings of the 28th International Conference on Machine Learning*, pages 1193–1200, Bellevue, WA, USA, June 2011.

Alex Kulesza and Ben Taskar. *Determinantal Point Processes for Machine Learning*. Now Publishers Inc., Hanover, MA, USA, 2012.

Chengtao Li, Stefanie Jegelka, and Suvrit Sra. Efficient sampling for k-determinantal point processes. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1328–1337, Cadiz, Spain, 09–11 May 2016. PMLR. URL http://proceedings.mlr.press/v51/li16f.html.

Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 510–520, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL http://dl.acm.org/citation.cfm?id=2002472.2002537.

Odile Macchi. The coincidence approach to stochastic point processes. *Advances in Applied Probability*, 7(1):83–122, 1975. ISSN 00018678. URL http://www.jstor.org/stable/1425855.

Avner Magen and Anastasios Zouzias. Near optimal dimensionality reductions that preserve volumes. In Ashish Goel, Klaus Jansen, José D. P. Rolim, and Ronitt Rubinfeld, editors, *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 523–534, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-85363-3.

Xiangrui Meng and Michael W. Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 91–100, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2029-0. doi: 10.1145/2488608.2488621. URL http://doi.acm.org/10.1145/2488608.2488621.

Jelani Nelson and Huy L. Nguyên. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, pages 117–126, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-5135-7. doi: 10.1109/FOCS.2013.21. URL http://dx.doi.org/10.1109/FOCS.2013.21.

Robin Pemantle and Yuval Peres. Concentration of lipschitz functionals of determinantal and other strong rayleigh measures. *Combinatorics, Probability and Computing*, 23(1):140–160, 2014. doi: 10.1017/S0963548313000345.

Joel A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, August 2012.

H. Robert van der Vaart. A note on wilks' internal scatter. *Ann. Math. Statist.*, 36(4):1308–1312, 08 1965. doi: 10.1214/aoms/1177700006. URL https://doi.org/10.1214/aoms/1177700006.

Cheng Zhang, Hedvig Kjellström, and Stephan Mandt. Determinantal point processes for mini-batch diversification. In *33rd Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, 11 August 2017 through 15 August 2017*. AUAI Press Corvallis, 2017.

Appendix A. Properties of regularized determinantal point processes

We give the proofs omitted from Section 4.

Theorem 7 *For $\mathbf{X} \in \mathbb{R}^{n\times d}$, ignoring the ordering in the sequences sampled with* R-DPP*, we have*

$$\text{R-DPP}^r\Big(\mathbf{X},\, \frac{r}{n}\mathbf{I}\Big) \overset{r\to 0}{\longrightarrow} \text{DPP}(\mathbf{X}) \quad \text{(pointwise)}, \tag{8}$$

$$\textit{whereas} \qquad \text{R-DPP}^r(\mathbf{X}, \mathbf{0}) \overset{r\to 0}{\longrightarrow} \text{VS}(\mathbf{X}) \quad \text{(pointwise)}. \tag{9}$$

Proof To show (8) we use a fact which is a simple consequence of the Sylvester's theorem, namely that $\det(\frac{r}{n}\mathbf{I} + \mathbf{X}_S^\top\mathbf{X}_S) = (\frac{r}{n})^{d-k}\det(\frac{r}{n}\mathbf{I} + \mathbf{X}_S\mathbf{X}_S^\top)$ for a set S of size k. It follows that

$$\begin{aligned}\Pr(\widetilde{\sigma}=S) &= k!\,\frac{\det(\frac{r}{n}\mathbf{I}+\mathbf{X}_S^\top\mathbf{X}_S)}{\det(\frac{r}{n}\mathbf{I}+\frac{r}{n}\mathbf{X}^\top\mathbf{X})}\,\frac{r^k\mathrm{e}^{-r}}{k!\,n^k} = \frac{(\frac{r}{n})^{d-k}\det(\frac{r}{n}\mathbf{I}+\mathbf{X}_S\mathbf{X}_S^\top)}{(\frac{r}{n})^d\det(\mathbf{I}+\mathbf{X}^\top\mathbf{X})}\Big(\frac{r}{n}\Big)^k\mathrm{e}^{-r}\\ &= \frac{\det(\frac{r}{n}\mathbf{I}+\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)}\mathrm{e}^{-r} \overset{r\to 0}{\longrightarrow} \frac{\det(\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)},\end{aligned}$$

where $\widetilde{\sigma} = S$ should be interpreted as if $\widetilde{\sigma}$ was an unordered multiset. Next, we prove (9):

$$\Pr(\widetilde{\sigma}=S) = k!\,\frac{\det(\mathbf{X}_S^\top\mathbf{X}_S)}{\det(\frac{r}{n}\mathbf{X}^\top\mathbf{X})}\,\frac{r^k\mathrm{e}^{-r}}{k!\,n^k} = \frac{\det(\mathbf{X}_S^\top\mathbf{X}_S)}{\det(\mathbf{X}^\top\mathbf{X})}\Big(\frac{r}{n}\Big)^{k-d}\mathrm{e}^{-r} \overset{r\to 0}{\longrightarrow} \mathbf{1}_{[k=d]}\frac{\det(\mathbf{X}_S^\top\mathbf{X}_S)}{\det(\mathbf{X}^\top\mathbf{X})},$$

because $\det(\mathbf{X}_S^\top\mathbf{X}_S) = 0$ whenever $k < d$. ■

Theorem 8 *For any $\mathbf{X} \in \mathbb{R}^{n\times d}$, $\alpha > 0$ and distribution p over $\{1..n\}$ s.t. $p_i > 0$, let $\widetilde{\mathbf{X}}$ denote matrix $\mathbf{X}$ with ith row rescaled by $\frac{1}{\sqrt{\alpha p_i}}$ for every $i \in \{1..n\}$. It follows that for any $r > 0$,*

$$\textit{if} \quad \widetilde{\sigma} \sim \text{R-DPP}_p^r\big(\widetilde{\mathbf{X}}, \mathbf{I}\big) \;\; \textit{and} \;\; S \sim \text{DPP}\big(\widetilde{\mathbf{X}}_{\widetilde{\sigma}}\big), \quad \textit{then} \quad \widetilde{\sigma}_S \sim \text{DPP}\Big(\sqrt{\tfrac{r}{\alpha}}\,\mathbf{X}\Big).$$

Proof Using the law of total probability we compute the probability of sampling set T of size t:

$$\begin{aligned}\Pr(\widetilde{\sigma}_S=T) &= \sum_{\widetilde{\sigma}}\Pr(\widetilde{\sigma}_S=T\,|\,\widetilde{\sigma})\Pr(\widetilde{\sigma})\\ &\overset{(a)}{=} \sum_{k=t}^{\infty}\sum_{S\subseteq[k]}\sum_{\widetilde{\sigma}:\,|\widetilde{\sigma}|=k}\mathbf{1}_{[\widetilde{\sigma}_S=T]}\overbrace{\frac{\det(\widetilde{\mathbf{X}}_{\widetilde{\sigma}_S}\widetilde{\mathbf{X}}_{\widetilde{\sigma}_S}^\top)}{\cancel{\det(\mathbf{I}+\widetilde{\mathbf{X}}_{\widetilde{\sigma}}\widetilde{\mathbf{X}}_{\widetilde{\sigma}}^\top)}}}^{\Pr(S\,|\,\widetilde{\sigma})}\overbrace{\frac{\cancel{\det(\mathbf{I}+\widetilde{\mathbf{X}}_{\widetilde{\sigma}}^\top\widetilde{\mathbf{X}}_{\widetilde{\sigma}})}}{\det\big(\mathbf{I}+r\sum_i\frac{p_i}{\alpha p_i}\mathbf{x}_i\mathbf{x}_i^\top\big)}\frac{r^k\mathrm{e}^{-r}}{k!}\prod_{i=1}^k p_{\widetilde{\sigma}_i}}^{\Pr(\widetilde{\sigma})}\\ &\overset{(b)}{=} \sum_{k=t}^{\infty}\binom{k}{t}t!\frac{\det(\widetilde{\mathbf{X}}_T\widetilde{\mathbf{X}}_T^\top)}{\det(\mathbf{I}+\frac{r}{\alpha}\mathbf{X}^\top\mathbf{X})}\frac{r^k\mathrm{e}^{-r}}{k!}\prod_{i\in T}p_i\\ &= \frac{\det(\widetilde{\mathbf{X}}_T\widetilde{\mathbf{X}}_T^\top)}{\det(\mathbf{I}+\frac{r}{\alpha}\mathbf{X}^\top\mathbf{X})}\Big(\prod_{i\in T}p_i\Big)\sum_{k=t}^{\infty}\frac{k!}{(k-t)!}\frac{r^k\mathrm{e}^{-r}}{k!}\\ &= \frac{\det(\frac{r}{\alpha}\mathbf{X}_T\mathbf{X}_T^\top)\big(\prod_{i\in T}\frac{1}{p_i}\big)}{\det(\mathbf{I}+\frac{r}{\alpha}\mathbf{X}^\top\mathbf{X})}\Big(\prod_{i\in T}p_i\Big)\sum_{k=t}^{\infty}\frac{r^{k-t}\mathrm{e}^{-r}}{(k-t)!} = \frac{\det(\frac{r}{\alpha}\mathbf{X}_T\mathbf{X}_T^\top)}{\det(\mathbf{I}+\frac{r}{\alpha}\mathbf{X}\mathbf{X}^\top)},\end{aligned}$$

where the cancellation in (a) follows from Sylvester's Theorem, and in (b) we use the exchangeability of sequence $\widetilde{\sigma}$ to observe that for any subset S of size t the value of the proceeding sum is the same (the factor $\binom{k}{t}$ counts the number of such subsets S and $t!$ counts the number of sequences of length t that correspond to set T). ■

Appendix B. Proof of Lemma 10

We present the omitted proof of the inequality from Lemma 10, stated here again.

Lemma 10 *For any $d \geq 1$, $\epsilon \in [0,1]$, and non-negative integers k, q s.t. $q \geq \epsilon d$ we have*

$$\Big(\big(1-\epsilon\big) + \frac{\epsilon\, k}{q}\Big)^d \leq \Big(\frac{q}{q-\epsilon d}\Big)^k e^{-\epsilon d}. \tag{10}$$

Proof We start with the case of $d = 1$. Denote the left hand side of (10) as L_k and the right hand side as R_k. If $k = q$ then

$$R_q = \left(\frac{q}{q-\epsilon}\right)^q \mathrm{e}^{-\epsilon} = \frac{\mathrm{e}^{-\epsilon}}{\big(1-\frac{\epsilon}{q}\big)^q} \geq 1 = 1-\epsilon+\epsilon\frac{q}{q} = L_q.$$

Let us now consider the multiplicative change in L_k as we increase or decrease k by one:

$$\frac{L_k}{L_{k+1}} = \frac{L_{k+1}-\frac{\epsilon}{q}}{L_{k+1}} = 1 - \frac{\epsilon}{q\,L_{k+1}} \begin{cases} \leq 1-\frac{\epsilon}{q} & \text{for } k \leq q-1 \text{ because } L_{k+1} \leq 1, \\ \geq 1-\frac{\epsilon}{q} & \text{for } k \geq q \text{ because } L_{k+1} \geq 1. \end{cases}$$

Observe that $\frac{R_k}{R_{k+1}} = 1-\frac{\epsilon}{q}$ for any k, so by induction over decreasing $k \leq q-1$,

$$L_k = L_{k+1}\frac{L_k}{L_{k+1}} \leq R_{k+1}\Big(1-\frac{\epsilon}{q}\Big) = R_{k+1}\frac{R_k}{R_{k+1}} = R_{k+1},$$

and for increasing $k \geq q$ similar induction shows that $L_{k+1} = L_k\frac{L_{k+1}}{L_k} \leq R_k\frac{R_{k+1}}{R_k} = R_{k+1}$. Finally, we use the case $d = 1$ to show the inequality for arbitrary $d \geq 1$:

$$\begin{aligned}\Big(1-\epsilon+\epsilon\frac{k}{q}\Big)^d &\overset{(a)}{\leq} \left(\Big(1-\frac{\epsilon}{q}\Big)^{-k}\mathrm{e}^{-\epsilon}\right)^d = \left(\frac{1}{(1-\frac{\epsilon}{q})^d}\right)^k \mathrm{e}^{-\epsilon d} \\ &\overset{(b)}{\leq} \left(\frac{1}{1-\frac{\epsilon d}{q}}\right)^k \mathrm{e}^{-\epsilon d} = \Big(\frac{q}{q-\epsilon d}\Big)^k \mathrm{e}^{-\epsilon d},\end{aligned}$$

where (a) follows from (10) applied for $d = 1$ and (b) is Bernoulli's inequality. ■

Appendix C. Total variation bound for Algorithm 2

We start by showing an approximation lemma about determinants which was earlier given in Section 5 (we restate it here).

Lemma 13 *For p.s.d. matrices $\mathbf{B}$, $\mathbf{C}$ such that $(1-\gamma)\mathbf{C} \preceq \mathbf{B} \preceq (1+\gamma)\mathbf{C}$ with $\gamma \in (0,1)$,*

$$\mathrm{e}^{-\frac{\gamma}{1-\gamma}s}\det(\mathbf{I}+\mathbf{C}) \leq \det(\mathbf{I}+\mathbf{B}) \leq \mathrm{e}^{\gamma s}\det(\mathbf{I}+\mathbf{C}), \quad \textit{where} \quad s = \mathrm{tr}\big(\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}\big).$$

Proof Let $\lambda_1, \dots, \lambda_d$ denote the eigenvalues of $\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}$. Then,

$$\begin{aligned}\frac{\det(\mathbf{I}+\mathbf{B})}{\det(\mathbf{I}+\mathbf{C})} &= \det\big(\mathbf{I}+(\mathbf{B}-\mathbf{C})(\mathbf{I}+\mathbf{C})^{-1}\big)\\ &\leq \det\big(\mathbf{I}+\gamma\,\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}\big)\\ &= \prod_{i=1}^{d}(1+\gamma\lambda_i) \leq \prod_{i=1}^{d}\mathrm{e}^{\gamma\lambda_i} = \mathrm{e}^{\gamma\,\mathrm{tr}(\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1})},\end{aligned}$$

which gives the upper bound. Similarly, we have

$$\begin{aligned}\frac{\det(\mathbf{I}+\mathbf{C})}{\det(\mathbf{I}+\mathbf{B})} &= \det\big(\mathbf{I}+(\mathbf{C}-\mathbf{B})(\mathbf{I}+\mathbf{B})^{-1}\big)\\ &\leq \det\big(\mathbf{I}+\gamma\,\mathbf{C}(\mathbf{I}+(1-\gamma)\mathbf{C})^{-1}\big)\\ &\leq \det\Big(\mathbf{I}+\frac{\gamma}{1-\gamma}\,\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1}\Big) \leq \mathrm{e}^{\frac{\gamma}{1-\gamma}\,\mathrm{tr}(\mathbf{C}(\mathbf{I}+\mathbf{C})^{-1})},\end{aligned}$$

so by inverting both sides we obtain the lower bound. ■

We are ready to prove that $\mathrm{DPP}(\rho\mathbf{X})$ is an approximation of $\mathrm{DPP}(\mathbf{X})$ in terms of total variation distance (restated here).

Lemma 11 *There is $C > 0$ s.t. for any matrix $\mathbf{X}$ and $\epsilon \leq 1$, if $|\rho^2 - 1| \leq \frac{\epsilon}{4\bar{s}+C\ln 9/\epsilon}$, where $\bar{s} = \max\{1, \mathbb{E}[|S|]\}$ for $S \sim \mathrm{DPP}(\mathbf{X})$, then $\mathrm{DPP}(\rho\mathbf{X})$ is an ϵ-approximation of $\mathrm{DPP}(\mathbf{X})$.*

Proof The larger the size of subset S the harder it is to control its approximate probability because it is defined via the determinant of a larger matrix. To overcome this we use the following standard concentration bound for determinantal point processes which shows that the probability of sampling a large subset is negligibly small. For simplicity, we state only a special case of the cited result.

Lemma 14 (based on Pemantle and Peres, 2014, Theorem 3.5) *Given any $\mathbf{X}$, if $S \sim \mathrm{DPP}(\mathbf{X})$, then for any $a > 0$ we have:*

$$\Pr\big(|S| - \mathbb{E}[|S|] \geq a\big) \leq 3\exp\bigg(-\frac{a^2}{16(a+2\,\mathbb{E}[|S|])}\bigg).$$

In (7) we showed that the expected subset size $|S|$ for both $\mathrm{DPP}(\mathbf{X})$ and $\mathrm{DPP}(\rho\mathbf{X})$ is bounded by $\bar{s}+\epsilon/4 \leq (1+\frac{1}{4})\bar{s}$, so setting $a = (3/4)\bar{s} + 80\ln 9/\epsilon$ in Lemma 14 (applied to either distribution),

$$\begin{aligned}\Pr\Big(|S| \geq 2\bar{s} + 80\ln 9/\epsilon\Big) &\leq \Pr\Big(|S| - \mathbb{E}[|S|] \geq a\Big)\\ &\leq 3\exp\bigg(-\frac{((3/4)\bar{s}+80\ln 9/\epsilon)^2}{16((13/4)\bar{s}+80\ln 9/\epsilon)}\bigg)\\ &\leq 3\exp\bigg(-\frac{(3/4)\bar{s}+80\ln 9/\epsilon}{5\cdot 16}\bigg) \leq \frac{\epsilon}{3}.\end{aligned}$$

So letting $k = 2\bar{s} + 80\ln 9/\epsilon$ and $C = 160$ we have $|\rho^2 - 1| \leq \frac{\epsilon}{2k}$. We can now bound the total variation distance between the distributions $p(S) = \frac{\det(\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)}$ and $q(S) = \frac{\det(\rho^2\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\rho^2\mathbf{X}\mathbf{X}^\top)}$.

$$\begin{aligned}
\frac{1}{2}\sum_{S\subseteq\{1..n\}} |p(S) - q(S)| &\overset{(a)}{\leq} \frac{\epsilon}{3} + \frac{1}{2}\sum_{S:|S|\leq k}\left|\frac{\det(\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)} - \frac{\rho^{2|S|}\det(\mathbf{X}_S\mathbf{X}_S^\top)}{\det(\mathbf{I}+\rho^2\mathbf{X}\mathbf{X}^\top)}\right| \\
&= \frac{\epsilon}{3} + \frac{1}{2}\sum_{S:|S|\leq k} p(S)\cdot\left|1 - \rho^{2|S|}\frac{\det(\mathbf{I}+\mathbf{X}\mathbf{X}^\top)}{\det(\mathbf{I}+\rho^2\mathbf{X}\mathbf{X}^\top)}\right| \\
&\overset{(b)}{\leq} \frac{\epsilon}{3} + \frac{(1+\frac{\epsilon}{2k})^k\,\mathrm{e}^{\frac{\epsilon}{4\bar{s}}\cdot\bar{s}} - 1}{2}\sum_{S:|S|\leq k} p(S) \\
&\leq \frac{\epsilon}{3} + \frac{\mathrm{e}^{(3/4)\epsilon} - 1}{2} \leq \epsilon,
\end{aligned}$$

where (a) uses Lemma 14 and in (b) we used Lemma 13 and the fact that $|\rho^2 - 1| \leq \frac{\epsilon}{2k}$. ■

Appendix D. Time complexity proofs

In this Section we show the runtime bounds for both Algorithm 2 and the preprocessing.

D.1. Sampling cost (proof of Theorem 2, part 3)

Lemma 12 implies that with probability at least $1-\delta$ Algorithm 2 will perform $\ln(\frac{1}{\delta})/\ln(\frac{6}{5})$ iterations of the loop. We next analyze the cost of one such iteration. Note that in line 4 we are supposed to sample exactly from the distribution $l = (l_1, \ldots, l_n)$ even though we are only given its approximation $\tilde{l}$ with condition (3) stating that $\frac{1}{2}l_i \leq \tilde{l}_i \leq \frac{3}{2}l_i$ for all $i \in \{1..n\}$. We can do this via simple rejection sampling performed for each $t \in \{1..K\}$:

$$\text{Sample } i \sim \tilde{l}, \quad a \sim \text{Bernoulli}\big(\tfrac{l_i}{2\tilde{l}_i}\big), \quad \text{if } a = \text{true, then } \sigma_t = i, \text{ else repeat.}$$

From the approximation guarantee it follows that the Bernoulli probability is bounded by 1 and never less than $\frac{1}{3}$. One such probability requires computing $l_i = \mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i$ which takes $O(d^2)$ if the matrix inverse $(\mathbf{I}+\mathbf{A})^{-1}$ was precomputed (in time $O(d^3)$). How many times will we need to compute l_i? Let m denote the total number of i.i.d. samples from l needed throughout the algorithm, i.e. the sum of all of the Poisson variables K. Conditioned on m, with probability at least $1-\delta$ the total cost of sampling from line 4 over the course of the algorithm is $O(md^2\ln(\frac{1}{\delta}))$. The total cost of computing the determinants from line 5 as well as the cost of sampling from $\mathrm{DPP}(\widetilde{\mathbf{X}}_\sigma)$ is $O(md^2)$ so they do not add to the asymptotic runtime. Since the number of iterations of the **repeat** loop is w.p. $\geq 1-\delta$ bounded by $c = \lceil\ln(\frac{1}{\delta})/\ln(\frac{6}{5})\rceil$, let variable $\widehat{m}$ be the sum of c independent copies of K. Then $\widehat{m} \sim \mathrm{Poisson}(c\,q)$ and a Poisson tail bound (Canonne, 2017) for any $\alpha > 0$ yields

$$\Pr\big(\widehat{m} \geq c\,(q+\alpha)\big) \leq \mathrm{e}^{-\frac{(c\,\alpha)^2}{c\,\alpha + c\,q}} \leq \mathrm{e}^{2\ln(\delta)\frac{\alpha^2}{\alpha+q}} = \delta^{\frac{2\alpha^2}{\alpha+q}},$$

which is less than δ for $\alpha = q$. Thus, with probability at least $1-2\delta$ we have $m \leq \widehat{m} \leq 2cq = O(d(\tilde{s}+1)\log\frac{1}{\delta})$, and the overall time complexity of Algorithm 2 becomes $O(d^3(\tilde{s}+1)\log^2\frac{1}{\delta})$. Since $\tilde{s} = \mathrm{tr}(\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1}) \leq \frac{1+1/(4\bar{s})}{1-1/(4\bar{s})}\cdot\bar{s} \leq \bar{s}+1$, we obtain the bound in Theorem 2.

D.2. Preprocessing cost (proof of Proposition 3)

Matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$ and estimate distribution $\tilde{l}$ satisfying approximation guarantees (2) and (3) can be computed efficiently using standard sketching and/or sampling techniques. Here, we outline the basic steps needed to obtain this, and discuss the time complexity achievable for each step:

1. Compute $\frac{1}{2}$-approximate leverage score distribution $p = (p_1, \ldots, p_n)$, i.e. such that the probabilities p_i satify (here "$(\cdot)^+$" is the Moore-Penrose pseudo-inverse):

$$p_i \geq \frac{\mathbf{x}_i^\top (\mathbf{X}^\top \mathbf{X})^+ \mathbf{x}_i}{2\,\mathrm{rank}(\mathbf{X})}.$$

2. Sample $r(\eta)$ row indices $\sigma_1, \ldots, \sigma_{r(\eta)} \overset{\text{i.i.d.}}{\sim} p$, so that with high probability

$$(1-\eta)\mathbf{X}^\top\mathbf{X} \preceq \overbrace{\frac{1}{r(\eta)}\sum_i \frac{1}{p_{\sigma_i}}\mathbf{x}_{\sigma_i}\mathbf{x}_{\sigma_i}^\top}^{\mathbf{A}} \preceq (1+\eta)\mathbf{X}^\top\mathbf{X}. \tag{11}$$

3. Having found matrix $\mathbf{A}$ we compute the approximate distribution $\tilde{l}$ satisfying

$$\frac{1}{2}\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i \leq \tilde{l}_i \leq \frac{3}{2}\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i,$$

   which is similar to the leverage scores, except with matrix $\mathbf{X}^\top\mathbf{X}$ replaced by $\mathbf{I}+\mathbf{A}$.

Step 1 can be performed in time $O(\mathrm{nnz}(\mathbf{X})\log n + d^3\log^2 d + d^2\log n)$ by employing the sparse subspace embedding technique developed by Clarkson and Woodruff (2017). Similar running times are offered by embeddings proposed by Nelson and Nguyên (2013); Meng and Mahoney (2013). Step 2 is an application of the standard matrix concentration bounds due to Tropp (2012), which show that it suffices to sample $r(\eta) = O(d\eta^{-2}\log d)$ rows from distribution p to satisfy (11) with high probability. The computation of matrix $\mathbf{A}$ then takes $O(r(\eta)d^2) = O(d^3\eta^{-2}\log d)$. Finally, step 3 is very similar to step 1, except we are estimating ridge leverage score type values. A standard approach of doing this is to observe that $\mathbf{x}_i^\top(\mathbf{I}+\mathbf{A})^{-1}\mathbf{x}_i$ is the squared norm of the ith row in matrix $\mathbf{X}(\mathbf{I}+\mathbf{A})^{-\frac{1}{2}}$. All of the row norms of this matrix can be estimated in time $O(\mathrm{nnz}(\mathbf{X})\log n + d^3 + d^2\log n)$ using the Johnson-Lindenstraus tranform as described by Drineas et al. (2012). Thus, the overall time complexity is $O(\mathrm{nnz}(\mathbf{X})\log n + d^3\eta^{-2}\log d + d^2\log n)$, where recall that $\eta^{-1} = O((\bar{s}+\log 1/\epsilon)/\epsilon)$. Note that the term $d^2\log n$ can be omitted from the time complexity, because if $\log n = \Omega(d)$, then $\mathrm{nnz}(\mathbf{X})\log n = \Omega(\mathrm{nnz}(\mathbf{X})\,d)$ and we can compute both $\mathbf{X}^\top\mathbf{X}$ and l exactly in time $O(\mathrm{nnz}(\mathbf{X})\,d + d^3)$. Also, our procedure requires a constant factor estimate of $\bar{s}$ to compute η. In fact, we can first use $\eta_0 = \frac{1}{2}$ and let $\max\{1, \mathrm{tr}(\mathbf{A}(\mathbf{I}+\mathbf{A})^{-1})\}$ be the estimate for $\bar{s}$ and then perform a second more accurate estimation.