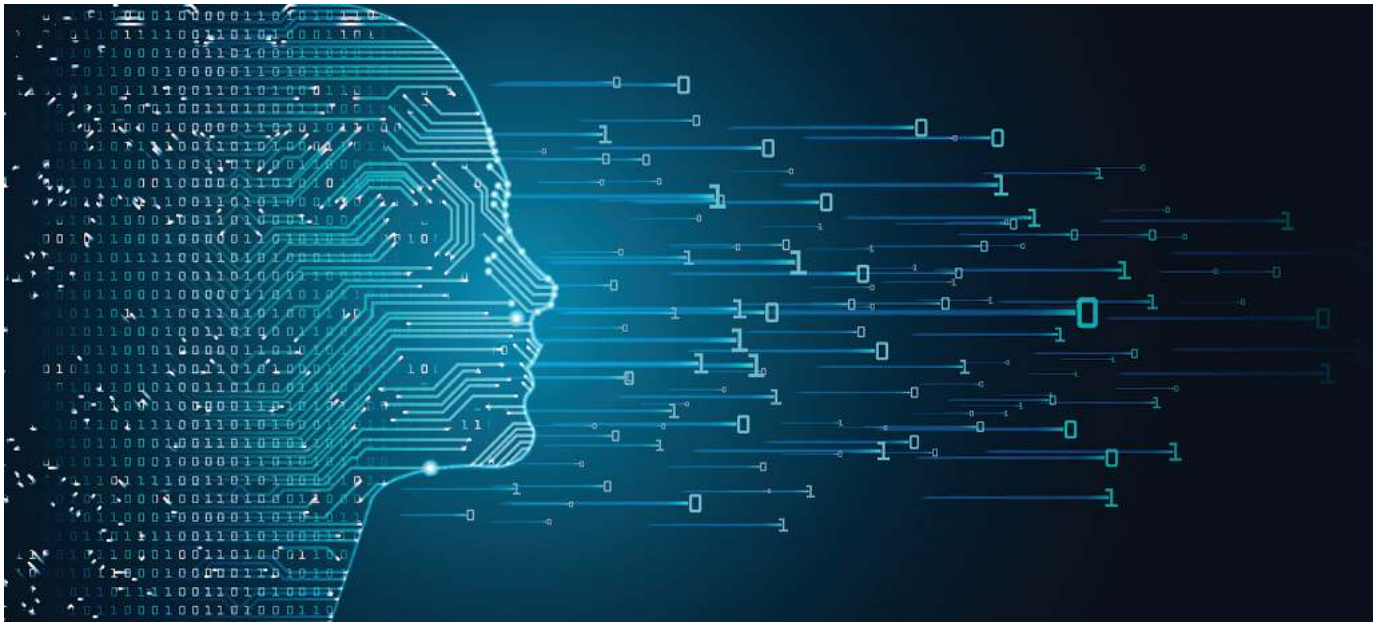
Determinantal Point Processes in Randomized Numerical Linear Algebra



Michał Dereziński and Michael W. Mahoney

1. Introduction

Randomized Numerical Linear Algebra (RandNLA) is an area which uses randomness, most notably random sampling and random projection methods, to develop improved algorithms for ubiquitous matrix problems. It began as a niche area in theoretical computer science about fifteen years ago [11], and since then the area has exploded. Matrix problems are central to much of applied mathematics, from traditional scientific computing and partial differential equations to statistics, machine learning, and artificial intelligence. Generalizations and variants of matrix problems are central to many other areas of mathematics, via more general transformations and algebraic structures, nonlinear optimization, infinite-dimensional operators, etc. Much of the work in RandNLA has been propelled by recent developments in machine learning, artificial intelligence, and large-scale data science, and RandNLA both draws upon and contributes back to both pure and applied mathematics.

A seemingly different topic, but one which has a long history in pure and applied mathematics, is that of Determinantal Point Processes (DPPs). A DPP is a stochastic point process, the probability distribution of which is characterized by subdeterminants of some matrix. Such processes were first introduced to model the distribution of fermions at thermal equilibrium [19]. In the context of random matrix theory, DPPs emerged as the eigenvalue

Michał Dereziński is a postdoctoral research scientist in the department of statistics at the University of California at Berkeley. His email address is mderezin@berkeley.edu.

Michael W. Mahoney is an associate professor in the department of statistics at the University of California at Berkeley and a research scientist at the International Computer Science Institute. His email address is mmahoney@stat.berkeley.edu.

A complete list of references is available in the technical report version of this paper: https://arxiv.org/pdf/2005.03185.pdf.

Communicated by Notices *Associate Editor Reza Malek-Madani.*

DOI: https://doi.org/10.1090/noti2202

distribution for standard random matrix ensembles, and they are of interest in other areas of mathematics such as graph theory, combinatorics, and quantum mechanics [17]. More recently, DPPs have also attracted significant attention within machine learning and statistics as a tractable probabilistic model that is able to capture a balance between quality and diversity within data sets and that admits efficient algorithms for sampling, marginalization, conditioning, etc. [18]. This resulted in practical application of DPPs in experimental design, recommendation systems, stochastic optimization, and more.

Until very recently, DPPs have had little if any presence within RandNLA. However, recent work has uncovered deep connections between these two topics. The purpose of this article is to provide an overview of RandNLA, with an emphasis on discussing and highlighting these connections with DPPs. In particular, we will show how random sampling with a DPP leads to new kinds of unbiased estimators for the classical RandNLA task of least squares regression, enabling a more refined statistical and inferential understanding of RandNLA algorithms. We will also demonstrate that a DPP is, in some sense, an optimal randomized method for low-rank approximation, another ubiquitous matrix problem. Finally, we also discuss how a standard RandNLA technique, called leverage score sampling, can be derived as the marginal distribution of a DPP, as well as the algorithmic consequences this has for efficient DPP sampling.

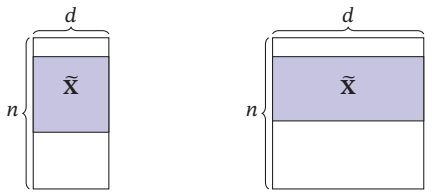
We start (in Section 2) with a brief review of a prototypical RandNLA algorithm, focusing on the ubiquitous least squares problem and highlighting key aspects that will put in context the recent work we will review. In particular, we discuss the trade-offs between standard sampling methods from RandNLA, including uniform sampling, norm-squared sampling, and leverage score sampling. Next (in Section 3), we introduce the family of DPPs, highlighting some important subclasses and the basic properties that make them appealing for RandNLA. Then (in Section 4), we describe the fundamental connections between certain classes of DPPs and the classical RandNLA tasks of least squares regression and low-rank approximation, as well as the relationship between DPPs and the RandNLA method of leverage score sampling. Finally (in Section 5), we discuss the algorithmic aspects of both leverage scores and DPPs. We conclude (in Section 6) by briefly mentioning several other connections between DPPs and RandNLA, as well as a recently introduced class of random matrices, called determinant preserving, which has proven useful in this line of research.

2. RandNLA: Randomized Numerical Linear Algebra

Much of the early work in numerical linear algebra (NLA) focused on deterministic algorithms. However, Monte Carlo sampling approaches demonstrated that randomness inside the algorithm is a powerful computational resource which can lead to both greater efficiency and robustness to worst-case data [12, 20]. The success of RandNLA methods has been proven in many domains, e.g., when the randomized least squares solvers such as Blendenpik or LSRN have outperformed the established high performance computing software LAPACK or other methods in parallel/distributed environments, respectively, or when RandNLA methods have been used in conjunction with traditional scientific computing solvers for low-rank approximation problems.

In a typical RandNLA setting, we are given a large dataset in the form of a real-valued matrix, say $\mathbf{X} \in \mathbb{R}^{n\times d}$, and our goal is to compute quantities of interest quickly. To do so, we efficiently downsize the matrix using a randomized algorithm, while approximately preserving its inherent structure, as measured by some objective. In doing so, we obtain a new matrix $\widetilde{\mathbf{X}}$ (often called a *sketch* of $\mathbf{X}$) which is either smaller or sparser than the original matrix. This procedure can often be represented by matrix multiplication, i.e., $\widetilde{\mathbf{X}} = \mathbf{S}\mathbf{X}$, where $\mathbf{S}$ is called the *sketching matrix*. Many applications of RandNLA follow the *sketch-and-solve* paradigm: Instead of performing a costly operation on $\mathbf{X}$, we first construct $\widetilde{\mathbf{X}}$ (the *sketch*); we then perform the expensive operation (more cheaply) on the smaller $\widetilde{\mathbf{X}}$ (the *solve*); and we use the solution from $\widetilde{\mathbf{X}}$ as a proxy for the solution we would have obtained from $\mathbf{X}$. Here, cost often means computational time, but it can also be communication or storage space or even human work.

Many different approaches have been established for randomly downsizing data matrices $\mathbf{X}$ (see [12, 20] for a detailed survey). While some methods randomly zero out most of the entries of the matrix, most randomly keep only a small random subset of rows and/or columns. In either case, however, the choice of randomness is crucial in preserving the structure of the data. For example, if the data matrix $\mathbf{X}$ contains a few dominant entries/rows/columns (e.g., as measured by their absolute value or norm or some other "importance" score), then we should make sure that our sketch is likely to retain the information they carry. This leads to data-dependent sampling distributions that will be the focus of our discussion. However, data-oblivious sketching techniques, where the random sketching matrix $\mathbf{S}$ is independent of $\mathbf{X}$ (typically called a "random rotation" or a "random projection," even if it is not precisely a rotation or projection in the linear algebraic sense), have also proven very successful [16, 24]. Among the most common



Rank-preserving sketch (e.g., Projection DPP) *Low-rank approximation* (e.g., L-ensemble DPP)

Figure 1. Two RandNLA settings which are differentiated by whether the sketch (matrix $\widetilde{\mathbf{X}}$) aims to preserve the rank of $\mathbf{X}$ (left) or obtain a low-rank approximation (right). In Section 4 we associate each setting with a different DPP.

examples of such random transformations are i.i.d. Gaussian matrices, fast Johnson-Lindenstrauss transforms and count sketches, all of which provide different trade-offs between efficiency and accuracy. These "data-oblivious random projections" can be interpreted either in terms of the Johnson-Lindenstrauss lemma or as a preconditioner for the "data aware random sampling" methods we discuss.

Most RandNLA techniques can be divided into one of two settings, depending on the dimensionality or aspect ratio of $\mathbf{X}$, and on the desired size of the sketch (see Figure 1):

1. Rank-preserving sketch. When $\mathbf{X}$ is a tall full-rank matrix (i.e., $n \gg d$), then we can reduce the larger dimension while preserving the rank.
2. Low-rank approximation. When $\mathbf{X}$ has comparably large dimensions (i.e., $n \sim d$), then the sketch typically has a much lower rank than $\mathbf{X}$.

The classical application of rank-preserving sketches is least squares regression, where, given matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ and vector $\mathbf{y} \in \mathbb{R}^n$, we wish to find:

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \mathcal{L}(\mathbf{w}) \text{ for } \mathcal{L}(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2.$$

The least squares solution can be computed exactly, using the Moore-Penrose inverse, $\mathbf{w}^* = \mathbf{X}^\dagger\mathbf{y}$. In a traditional RandNLA setup, in order to avoid solving the full problem, our goal is to use the sketch-and-solve paradigm to obtain an (ϵ, δ)-approximation of $\mathbf{w}^*$, i.e., $\widehat{\mathbf{w}}$ such that:

$$\mathcal{L}(\widehat{\mathbf{w}}) \le (1+\epsilon)\mathcal{L}(\mathbf{w}^*) \text{ with probability } 1-\delta. \qquad (1)$$

Imposing statistical modeling assumptions on the vector $\mathbf{y}$ leads to different objectives, such as the mean squared error (MSE):

$$\mathrm{MSE}\big[\widehat{\mathbf{w}}\big] = \mathbb{E}\,\|\widehat{\mathbf{w}} - \boldsymbol{\beta}\|^2, \text{ given } \mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\xi},$$

where $\boldsymbol{\xi}$ is a noise vector with a known distribution, and $\boldsymbol{\beta}$ is fixed but unknown. There has been work on statistical aspects of RandNLA methods, and these statistical objectives pose different challenges than the standard RandNLA guarantees (some of which can be addressed by DPPs; see Section 4).

To illustrate the types of guarantees achieved by RandNLA methods on the least squares task, we will focus on row sampling, i.e., sketches consisting of a small random subset of the rows of $\mathbf{X}$, in the case that $n \gg d$. Concretely, the considered meta-strategy is to draw random i.i.d. row indices $j_1, ..., j_k$ from $\{1, ..., n\}$, with each index distributed according to $(p_1, ..., p_n)$, and then solve the subproblem formed from those indices:

$$\widehat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}} \|\widetilde{\mathbf{X}}\mathbf{w} - \tilde{\mathbf{y}}\|^2 = \widetilde{\mathbf{X}}^\dagger\tilde{\mathbf{y}}, \qquad (2)$$

where $\tilde{\mathbf{x}}_i^\top = \sqrt{\frac{1}{kp_{j_i}}}\,\mathbf{x}_{j_i}^\top$ and $\tilde{y}_i = \sqrt{\frac{1}{kp_{j_i}}}\,y_{j_i}$ for $i = 1, ..., k$ denote the ith row of $\widetilde{\mathbf{X}}$ and entry of $\tilde{\mathbf{y}}$, respectively. The rescaling is introduced to account for the biases caused by nonuniform sampling. We consider the following standard sampling distributions:

1. Uniform: $p_i = 1/n$ for all i.
2. Squared norms: $p_i = \|\mathbf{x}_i\|^2/\|\mathbf{X}\|_F^2$, where $\|\cdot\|_F$ is the Frobenius (Hilbert-Schmidt) norm.
3. Leverage scores: $p_i = l_i/d$ for $l_i = \|\mathbf{x}_i\|^2_{(\mathbf{X}^\top\mathbf{X})^{-1}}$ (ith leverage score of $\mathbf{X}$) and $\|\mathbf{v}\|_{\mathbf{A}} = \sqrt{\mathbf{v}^\top\mathbf{A}\mathbf{v}}$.

Both squared norm and leverage score sampling are standard RandNLA techniques used in a variety of applications [11, 14]. The following theorem (which, for convenience, we state with a failure probability of $\delta = 0.1$) puts together the results that allow us to compare each row sampling distribution in the context of least squares. Below, we use C to denote an absolute positive constant.

Theorem 1. *Estimator $\widehat{\mathbf{w}}$ constructed as in (2) is an $(\epsilon, 0.1)$-approximation, as in (1), if:*

1. $k \ge C(\mu d\log d + \mu d/\epsilon)$ *for Uniform, where* $\mu = \max_i \frac{n}{d} l_i \ge 1$ *is the matrix coherence of* $\mathbf{X}$.
2. $k \ge C(\kappa d\log d + \kappa d/\epsilon)$ *for Squared norms, where* $\kappa \ge 1$ *is the condition number of* $\mathbf{X}^\top\mathbf{X}$.
3. $k \ge C(d\log d + d/\epsilon)$ *for Leverage scores.*

Recall that (when considering least squares) we typically assume that $n \gg d$, so any of the three sample sizes k may be much smaller than n. Thus, each sampling method offers a potentially useful guarantee for the number of rows needed to achieve a $(1+\epsilon)$-approximation. However, in the case of both uniform and squared norm sampling, the sample size depends not only on the dimension d, but also on other data-dependent quantities. For uniform sampling, that quantity is *matrix coherence* μ, which measures a degree of nonuniformity among the data points, with respect to the canonical axes. For squared norm sampling, that quantity is the *condition number* κ (ratio between the largest and the smallest eigenvalue) of the $d\times d$ data covariance $\mathbf{X}^\top\mathbf{X}$. Leverage score sampling avoids both of these dependencies.

We now briefly discuss a key structural property, called the *subspace embedding*, which is needed to show the guarantees of Theorem 1. This important property—first introduced into RandNLA for data-aware random sampling by [14] and then for data-oblivious random projection by [23]—is ubiquitous in the analysis of many RandNLA techniques. Remarkably, most DPP results do *not* rely on subspace embedding techniques, which is an important differentiating factor for this class of sampling distributions.

Definition 1. A sketching matrix $\mathbf{S}$ is a $(1 \pm \epsilon)$ subspace embedding for the column space of $\mathbf{X}$ if:

$$(1-\epsilon)\|\mathbf{X}\mathbf{v}\|^2 \leq \|\mathbf{S}\mathbf{X}\mathbf{v}\|^2 \leq (1+\epsilon)\|\mathbf{X}\mathbf{v}\|^2 \quad \forall \mathbf{v} \in \mathbb{R}^d.$$

The matrix $\widetilde{\mathbf{X}}$ used in (2) for constructing $\widehat{\mathbf{w}}$ can be written as $\widetilde{\mathbf{X}} = \mathbf{S}\mathbf{X}$, by letting the ith row of $\mathbf{S}$ be the scaled standard basis vector $\sqrt{\frac{1}{kp_{j_i}}}\,\mathbf{e}_{j_i}$. Relying on established measure concentration results for random matrices, we can show that $\mathbf{S}$ is a subspace embedding (up to some failure probability) for each of the i.i.d. sampling methods from Theorem 1. However, only leverage score sampling (or random projections, which precondition the input to have approximately uniform leverage scores) achieves this for $O(d \log d)$ samples, independent of any input-specific quantities such as the coherence or condition number.

The least squares task formulated in (1), as well as the subspace embedding condition, require using rank-preserving sketches. However, these ideas can be naturally extended to the task of low-rank approximation. In particular, as discussed in more detail in Section 4.3, low-rank approximation can be reformulated as a set of least squares problems. Similarly, leverage score sampling has been extended to adapt to the low-rank setting. In Section 4.4, we discuss one of these extensions, called *ridge leverage scores*, and its connections to DPPs.

In the next sections, we show how non-i.i.d. sampling via DPPs goes beyond the standard RandNLA analysis. Among other things, this will allow us to obtain approximation guarantees with fewer than $d \log d$ samples and without a failure probability.

3. DPPs: Determinantal Point Processes

In this section, we define DPPs and related families of distributions (see Figure 2 for a diagram), including some basic properties and intuitions. A detailed introduction to DPPs can be found in [18]. We focus on sampling over a discrete domain $[n] = \{1, \ldots, n\}$ (continuous domains are discussed in [17]).

Definition 2 (Determinantal Point Process)**.** Let $\mathbf{K}$ be an $n \times n$ positive semidefinite (p.s.d.) matrix with operator norm $\|\mathbf{K}\| \leq 1$. Point process $S \subseteq [n]$ is drawn according to DPP($\mathbf{K}$), denoted as $S \sim \text{DPP}(\mathbf{K})$, if for any $T \subseteq [n]$,

$$\Pr\{T \subseteq S\} = \det(\mathbf{K}_{T,T}).$$

Here, $\mathbf{K}_{T,T}$ denotes the $|T| \times |T|$ submatrix indexed by the set T. Matrix $\mathbf{K}$ is called the *marginal kernel* of S (it can be shown that any $\mathbf{K}$ as in Definition 2 defines a DPP). If $\mathbf{K}$ is diagonal, then DPP($\mathbf{K}$) corresponds to a series of n independent biased coin flips deciding whether to include each index i into the set S. A more interesting distribution is obtained for a general $\mathbf{K}$, in which case the inclusion events are no longer independent. Some of the key properties that make DPPs useful as a mathematical framework are:

1. Negative correlation: if $i \neq j$ and $\mathbf{K}_{ij} \neq 0$, then
$$\Pr(i \in S \mid j \in S) < \Pr(i \in S).$$
2. Cardinality: while the size $|S|$ is in general random, its expectation equals $\text{tr}(\mathbf{K})$ and the variance also has a simple expression.
3. Restriction: for $R \subseteq [n]$, the set $\tilde{S} = S \cap R$ is distributed as $\text{DPP}(\mathbf{K}_{R,R})$ (after relabeling).
4. Complement: the complement set $\tilde{S} = [n] \backslash S$ is distributed as $\text{DPP}(\mathbf{I} - \mathbf{K})$.

In the context of linear algebra, a slightly more restrictive definition of DPPs has proven useful.

Definition 3 (L-ensemble)**.** Let $\mathbf{L}$ be an $n \times n$ p.s.d. matrix. Point process $S \subseteq [n]$ is drawn according to $\text{DPP}_{\text{L}}(\mathbf{L})$ and called an L-ensemble if

$$\Pr\{S\} = \frac{\det(\mathbf{L}_{S,S})}{\det(\mathbf{I} + \mathbf{L})}.$$

It can be shown that any L-ensemble is a DPP by setting $\mathbf{K} = \mathbf{L}(\mathbf{I} + \mathbf{L})^{-1}$, but not vice versa. (However, there are extensions of the L-ensemble parameterization which cover all DPPs.) Unlike Definition 2, this definition explicitly gives the probabilities of individual sets. These probabilities sum to one as a consequence of a determinantal identity (see Theorem 2.1 in [18] which is a special case of the classical formula for the Fredholm determinant). Furthermore, the L-ensemble formulation provides a natural geometric interpretation in the context of row sampling for RandNLA. Suppose that we let $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$ for some $n \times d$ matrix $\mathbf{X}$. Then, the probability of sampling subset S according to $\text{DPP}_{\text{L}}(\mathbf{L})$ satisfies:

$$\Pr\{S\} \propto \text{Vol}^2_{|S|}\big(\{\mathbf{x}_i : i \in S\}\big).$$

Namely, this sampling probability is proportional to the squared $|S|$-dimensional volume of the parallelepiped spanned by the rows of $\mathbf{X}$ indexed by S. This immediately implies that the size of S will never exceed the rank of $\mathbf{X}$ (which is bounded by d). Furthermore, such a distribution ensures that the set of sampled rows will be nondegenerate: no row can be obtained as a linear combination of the

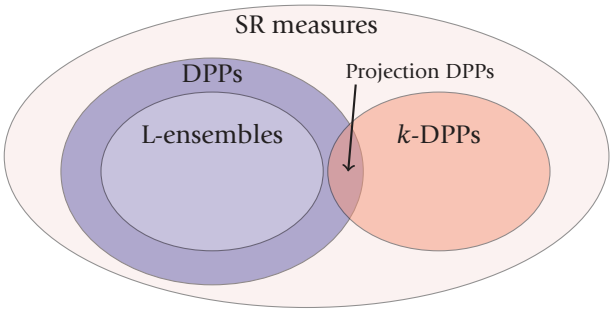


Figure 2. A diagram illustrating different classes of determinantal distributions within a broader class of Strongly Rayleigh (SR) measures: DPPs (Definition 2), L-ensembles (Definition 3), k-DPPs (Definition 4), and Projection DPPs (Remark 1).

others. Intuitively, this property is desirable for RandNLA sampling as it avoids redundancies. Also, all else being equal, rows with larger norms are generally preferred as they contribute more to the volume.

While the subset size of a DPP is in most cases a random variable, it is easy to constrain the cardinality to some fixed value k. The resulting distribution is not a DPP, in the sense of Definition 2, but it retains many useful properties of proper DPPs.

Definition 4 (Cardinality constrained DPP). We will use $k\text{-DPP}_{\text{L}}(\mathbf{L})$ to denote a distribution obtained by constraining $\text{DPP}_{\text{L}}(\mathbf{L})$ to only subsets of size $|S| = k$.

While a k-DPP is not, in general, a DPP in the sense of Definition 2, both families belong to a broader class of negatively correlated distributions called Strongly Rayleigh (SR) measures. See Figure 2.

At the intersection of DPPs and k-DPPs lies a family of distributions called Projection DPPs. This family is of particular importance to RandNLA.

Remark 1 (Projection DPP). Point process $S \sim k\text{-DPP}_{\text{L}}(\mathbf{L})$ satisfies Definition 2 iff $k = \text{rank}(\mathbf{L})$, in which case we call it a Projection DPP since its marginal kernel $\mathbf{K} = \mathbf{L}\mathbf{L}^{\dagger}$ is an orthogonal projection (recall that $(\cdot)^{\dagger}$ denotes the Moore-Penrose inverse).

We chose to introduce Projection DPPs via the connection to L-ensembles to highlight once again the geometric interpretation. In this viewpoint, letting $\mathbf{L} = \mathbf{X}\mathbf{X}^{\top}$ for an $n \times d$ matrix $\mathbf{X}$ with full column rank, a Projection DPP associated with the L-ensemble $\mathbf{L}$ has marginal kernel $\mathbf{K} = \mathbf{X}\mathbf{X}^{\dagger}$, which is a d-dimensional projection onto the span of the columns of $\mathbf{X}$. Furthermore, the probability of a row subset $\mathbf{X}_S$ under $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^{\top})$ is proportional to the squared d-dimensional volume spanned by it, i.e., $\det(\mathbf{X}_S)^2$. Here, the normalization constant is obtained via the classical Cauchy-Binet formula:

$$\sum_{S:|S|=d} \det(\mathbf{X}_S)^2 = \det(\mathbf{X}^{\top}\mathbf{X}).$$

The form of the probability implies that the rows $\{\mathbf{x}_i : i \in S\}$ sampled from a Projection DPP will with probability 1 capture all directions of the ambient space that are present in the matrix $\mathbf{X}$, which is crucial for rank-preserving RandNLA sketches.

4. DPPs in RandNLA

In this section, we demonstrate the fundamental connections between DPPs and standard RandNLA tasks, as well as the new kinds of RandNLA guarantees that can be achieved via these connections. Our discussion focuses on two types of DPP-based sketches (that were illustrated in Figure 1):

1. Projection DPPs as a *rank-preserving sketch*;
2. L-ensemble DPPs as a *low-rank approximation*.

These sketches can be efficiently constructed using DPP sampling algorithms which we discuss later (in Section 5). We also discuss the close relationship between DPPs and the RandNLA method of leverage score sampling, shedding light on why these two different randomized techniques have proven effective in RandNLA. For the summary, see Table 1.

4.1. Unbiased estimators.

We now define the least squares estimators that naturally arise from row sampling with Projection DPPs and L-ensembles. The definitions are motivated by the fact that the estimators are unbiased, relative to the solutions of the full least squares problems. Importantly, this property is *not* shared by i.i.d. row sampling methods used in RandNLA.

We start with the rank-preserving setting, i.e., given a tall full-rank $n \times d$ matrix $\mathbf{X}$ and a vector $\mathbf{y} \in \mathbb{R}^n$, where $n \gg d$, we wish to approximate the least squares solution $\mathbf{w}^* = \text{argmin}_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2$. To capture all of the directions present in the data and to obtain a meaningful estimate of $\mathbf{w}^*$, we must sample at least d rows from $\mathbf{X}$. We achieve this by sampling from a Projection DPP defined as $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^{\top})$ (see Remark 1). The linear system $(\mathbf{X}_S, \mathbf{y}_S)$, corresponding to the rows of $\mathbf{X}$ indexed by S, has exactly one solution because sets selected by a Projection DPP are always rank-preserving: $\hat{\mathbf{w}} = \mathbf{X}_S^{-1}\mathbf{y}_S$. Moreover, the obtained random vector is an unbiased estimator of the least squares solution $\mathbf{w}^*$ [9].

Theorem 2. *If* $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^{\top})$, *then*

$$\mathbb{E}\,\mathbf{X}_S^{-1}\mathbf{y}_S \;=\; \operatorname*{argmin}_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 = \mathbf{w}^*.$$

This seemingly simple identity relies on the negative correlations between the samples in a DPP, and thus *cannot* hold for any i.i.d. row sampling method. It is perhaps no coincidence that the marginal kernel of this distribution, i.e., $\mathbf{K} = \mathbf{X}\mathbf{X}^{\dagger} = \mathbf{X}(\mathbf{X}^{\top}\mathbf{X})^{-1}\mathbf{X}^{\top}$, coincides with the *hat matrix* (as it is known in statistics) of the ordinary least squares estimator.

		Rank-preserving sketch		*Low-rank approximation*	
		Projection DPP	$S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^\top)$	L-ensemble	$S \sim \text{DPP}_{\text{L}}(\frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$
subset size	$\mathbb{E}\,\lvert S\rvert \;=$	dimension	d	effective dim.	$\operatorname{tr}(\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^\top)$
marginal	$\Pr\{i \in S\} =$	leverage score	$\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}_i$	ridge lev. score	$\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{x}_i$
expectation	$\mathbb{E}\,\mathbf{X}_S^\dagger\mathbf{y}_S \;=$	least squares	$\operatorname{argmin}_{\mathbf{w}} \lVert\mathbf{X}\mathbf{w}-\mathbf{y}\rVert^2$	ridge regression	$\operatorname{argmin}_{\mathbf{w}} \lVert\mathbf{X}\mathbf{w}-\mathbf{y}\rVert^2+\lambda\lVert\mathbf{w}\rVert^2$

Table 1. Key properties of the DPPs discussed in Section 4, as they relate to: RandNLA tasks of least squares and ridge regression; RandNLA methods of leverage score sampling and ridge leverage score sampling.

Theorem 2 has an analogue in the context of low-rank approximation, where both the dimensions of $\mathbf{X}$ are comparably large (i.e., $n \sim d$), and so the desired sample size is typically much smaller than d. When the selected subproblem $(\mathbf{X}_S, \mathbf{y}_S)$ has fewer than d rows (i.e., it is under-determined) then it has multiple exact solutions. A standard way to address this is picking the solution with smallest Euclidean norm, defined via the Moore-Penrose inverse: $\widehat{\mathbf{w}} = \mathbf{X}_S^\dagger\mathbf{y}_S$. To sample the under-determined subproblem, we use a scaled L-ensemble DPP with the expected sample size controlled by a parameter $\lambda > 0$ [8].

Theorem 3. *If* $S \sim \text{DPP}_{\text{L}}(\frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$, *then*

$$\mathbb{E}\,\mathbf{X}_S^\dagger\mathbf{y}_S \;=\; \operatorname*{argmin}_{\mathbf{w}} \lVert\mathbf{X}\mathbf{w}-\mathbf{y}\rVert^2 + \lambda\lVert\mathbf{w}\rVert^2.$$

Thus, the minimum norm solution of the under-determined subproblem is an unbiased estimator of the Tikhonov-regularized least squares problem, i.e., ridge regression. Ridge regression is a natural extension of the standard least squares task, particularly useful when $n \sim d$ or $n \ll d$.

Theorem 3 illustrates the *implicit regularization* effect that occurs when choosing one out of many exact solutions to a subsampled least squares task (see also Section 6). Increasing the regularization $\lambda\lVert\mathbf{w}\rVert^2$ in ridge regression is interpreted as using fewer degrees of freedom, which aligns with the effect that λ has on the distribution $S \sim \text{DPP}_{\text{L}}(\frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$: larger λ means that smaller subsets S are more likely. In fact, since the marginal kernel of the L-ensemble coincides with the hat matrix of the ridge estimator, the expected subset size (trace of the marginal kernel) captures the notion of *effective dimension* (a.k.a. effective degrees of freedom) in the same way as it is commonly done for ridge regression in statistics [2]:

$$d_\lambda := \operatorname{tr}\big(\mathbf{X}(\mathbf{X}^\top\mathbf{X}+\lambda\mathbf{I})^{-1}\mathbf{X}^\top\big) = \mathbb{E}\,\lvert S\rvert. \tag{3}$$

4.2. **Exact error analysis.** The error analysis for DPP sampling differs significantly from the standard RandNLA techniques discussed in Section 2. In particular, approximation guarantees are formulated in terms of the expected error, without relying on measure concentration results. This means that we avoid failure probabilities such as the one present in Theorem 1, and the analysis is often much more precise, sometimes even exact. Furthermore, because of the non-i.i.d. nature of DPPs, these guarantees can be achieved with smaller sample sizes than for RandNLA sampling methods. Of course, this comes with computational trade-offs, which we discuss in Section 5.

We illustrate these differences in the context of *rank-preserving sketches* for least squares regression. Consider the estimator $\widehat{\mathbf{w}} = \mathbf{X}_S^{-1}\mathbf{y}_S$ from Theorem 2, where the subset S is sampled via the Projection DPP, i.e., $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^\top)$. Recall that the sample size here is only d, which is less than $d\log d$ needed by i.i.d. sampling methods such as leverage scores (or random projection methods). Nevertheless, this estimator achieves an approximation guarantee in terms of the loss $\mathcal{L}(\mathbf{w}) = \lVert\mathbf{X}\mathbf{w}-\mathbf{y}\rVert^2$. Moreover, under minimal assumptions, the expected loss is given by a closed form expression [9].

Theorem 4. *Assume that the rows of* $\mathbf{X}$ *are in general position, i.e., every set of* d *rows is nondegenerate. If* $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^\top)$, *then*

$$\mathbb{E}\,\mathcal{L}(\mathbf{X}_S^{-1}\mathbf{y}_S) = (d+1)\,\mathcal{L}(\mathbf{w}^*),$$

and the factor $d+1$ *is worst-case optimal.*

This exact error analysis is particularly useful in statistical modeling, where under additional assumptions about the vector $\mathbf{y}$, we wish to estimate accurately the generalization error of our estimator. Specifically, consider the following noisy linear model of the vector $\mathbf{y}$:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\xi}, \quad \text{where} \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}). \tag{4}$$

Here, $\boldsymbol{\beta}$ is a fixed vector that we wish to recover, and the mean squared error in this context is defined as $\mathbb{E}\,\lVert\widehat{\mathbf{w}}-\boldsymbol{\beta}\rVert^2$, where the expectation is taken over both the sampling and the noise. The full least squares solution $\mathbf{w}^*$ is an unbiased estimator of $\boldsymbol{\beta}$, with error given by $\mathbb{E}\,\lVert\mathbf{w}^*-\boldsymbol{\beta}\rVert^2 = \sigma^2\operatorname{tr}\big((\mathbf{X}^\top\mathbf{X})^{-1}\big)$. The Projection DPP estimator, which observes only d noisy measurements from $\mathbf{y}$, is also unbiased in this model, and its error scales linearly with that of $\mathbf{w}^*$ [9].

Theorem 5. *Assume that the rows of* $\mathbf{X}$ *are in general position, i.e., every set of* d *rows is nondegenerate, and consider* $\mathbf{y}$ *as in* (4). *If* $S \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^\top)$, *then*

$$\mathbb{E}\,\lVert\mathbf{X}_S^{-1}\mathbf{y}_S-\boldsymbol{\beta}\rVert^2 = (n-d+1)\,\mathbb{E}\,\lVert\mathbf{w}^*-\boldsymbol{\beta}\rVert^2.$$

A number of extensions to Theorems 4 and 5 have been proposed, covering larger sample sizes [10] as well as different statistical models [8, 9].

4.3. **Optimal approximation guarantees.** As we have seen above, the non-i.i.d. nature of DPP sampling can lead to improved approximation guarantees, compared to standard RandNLA methods, when we wish to minimize the size of the downsampled problem. We next discuss this in the *low-rank approximation* setting, i.e., when $n \sim d$. Here, cardinality constrained L-ensembles are known to achieve *optimal* $(1+\epsilon)$-approximation guarantees.

In Section 4.1, we used low-rank sketches to construct unbiased estimators for regularized least squares, given matrix $\mathbf{X}$ and a vector $\mathbf{y}$. However, even without introducing $\mathbf{y}$, a natural low-rank approximation objective for sketching $\mathbf{X}$ can be defined via a reduction to least squares. Namely, we can measure the error in reconstructing the ith row of $\mathbf{X}$ by finding the best fit among all linear combinations of the rows of the sketch $\mathbf{X}_S$. Repeating this over all rows of $\mathbf{X}$, we get:

$$\mathrm{Er}(S) := \sum_{i=1}^{n} \overbrace{\min_{\mathbf{w}} \|\mathbf{X}_S^\top \mathbf{w} - \mathbf{x}_i\|^2}^{\text{least squares}} = \left\|\mathbf{X}\mathbf{X}_S^\dagger \mathbf{X}_S - \mathbf{X}\right\|_F^2.$$

Note that $\mathbf{X}_S^\dagger \mathbf{X}_S$ is the projection onto the span of $\{\mathbf{x}_i : i \in S\}$. If the size of S is equal to some target rank r, then $\mathrm{Er}(S)$ is at least as large as the error of the best rank r approximation of $\mathbf{X}$, denoted $\mathbf{X}_{(r)}$ (obtained by projecting onto the top r right-singular vectors of $\mathbf{X}$). However, [15] showed that using a cardinality constrained L-ensemble with $k = r + r/\epsilon - 1$ rows suffices to get within a $1+\epsilon$ factor of the best rank r approximation error.

Theorem 6. *If $S \sim k$-$\mathrm{DPP}_\mathrm{L}(\mathbf{X}\mathbf{X}^\top)$, where the sample size satisfies $k \geq r + r/\epsilon - 1$, then*

$$\mathbb{E}\,\mathrm{Er}(S) \leq (1+\epsilon)\,\|\mathbf{X}_{(r)} - \mathbf{X}\|_F^2,$$

and the size $r + r/\epsilon - 1$ is worst-case optimal.

The task of finding the subset S that minimizes $\mathrm{Er}(S)$ is sometimes known as the Column Subset Selection Problem (with $\mathbf{X}$ replaced by $\mathbf{X}^\top$). Similar $1+\epsilon$ guarantees are achievable with RandNLA sampling techniques such as leverage scores. However, those require sample sizes k of at least $r \log r$, they contain a failure probability, and they suffer from additional constant factors due to less exact analysis.

Nyström method. The task of low-rank approximation is often formulated in the context of symmetric positive semidefinite (p.s.d.) matrices. Let $\mathbf{L}$ be an $n \times n$ p.s.d. matrix. We briefly discuss how DPPs can be applied in this setting via the Nyström method, which constructs a rank k approximation of $\mathbf{L}$ by using the eigendecomposition of a small $k \times k$ submatrix $\mathbf{L}_{S,S}$ for some index subset S.

Definition 5. We define the Nyström approximation of $\mathbf{L}$ based on a subset S as the $n \times n$ matrix $\widetilde{\mathbf{L}}(S) = \mathbf{L}_{[n],S}\mathbf{L}_{S,S}^\dagger \mathbf{L}_{S,[n]}$.

Originally developed in the context of obtaining numerical solutions to integral equations, this method has found applications in a number of areas such as machine learning, Gaussian Process regression, and Independent Component Analysis. Theorem 6 can be adapted to the setting of Nyström approximation, providing the optimal sample size with respect to the nuclear norm approximation error. We use $\|\mathbf{A}\|_* = \mathrm{tr}\big((\mathbf{A}^\top\mathbf{A})^{\frac{1}{2}}\big)$ to denote the nuclear norm and $\mathbf{L}_{(r)}$ as the best rank r approximation.

Theorem 7. *If $S \sim k$-$\mathrm{DPP}_\mathrm{L}(\mathbf{L})$, where the sample size satisfies $k \geq r + r/\epsilon - 1$, then*

$$\mathbb{E}\,\|\mathbf{L} - \widetilde{\mathbf{L}}(S)\|_* \leq (1+\epsilon)\,\|\mathbf{L} - \mathbf{L}_{(r)}\|_*,$$

and the size $r + r/\epsilon - 1$ is worst-case optimal.

4.4. **Connections to RandNLA methods.** The natural applicability of DPPs in the RandNLA tasks of least squares regression and low-rank approximation discussed above raises the question of how DPPs relate to traditional RandNLA sampling methods used for this task. As discussed in Section 2, one of the main RandNLA techniques for constructing rank-preserving sketches (i.e., relative to a tall matrix $\mathbf{X}$ with $n \gg d$) is i.i.d. leverage score sampling. (From this perspective, random projections can be seen as preprocessing or preconditioning the input so that leverage scores are approximately uniform, thereby enabling uniform sampling—in the randomly-transformed space—to be successfully used.) Even though leverage score sampling was developed independently of DPPs, this method can be viewed as an i.i.d. counterpart of the Projection DPP from Theorem 2.

Theorem 8. *Let $\mathbf{X}$ be $n \times d$ and rank d. For $S \sim d$-$\mathrm{DPP}_\mathrm{L}(\mathbf{X}\mathbf{X}^\top)$ and any index i, the marginal probability of $i \in S$ is the ith leverage score of $\mathbf{X}$:*

$$\Pr\{i \in S\} = \mathbf{x}_i^\top (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}_i.$$

Here, the term *marginal* refers to the marginal distribution of any one out of n binary variables $b_1, ..., b_n$ which can be used to represent the random set S via $S = \{i : b_i = 1\}$. Recall that the marginal kernel of the Projection DPP is the projection matrix $\mathbf{K} = \mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$. The marginal probabilities of this distribution lie on the diagonal of the marginal kernel, which also contains the leverage scores of $\mathbf{X}$.

Thus, leverage score sampling can be obtained as a distribution constructed from the marginals of the Projection DPP. Naturally, when going from non-i.i.d. to i.i.d. sampling, we lose all the negative correlations between the points in a DPP sample, and therefore the expectation formulas and inequalities from the preceding sections no longer hold for leverage score sampling. Furthermore, recall that to achieve a rank-preserving sketch (e.g., for least squares) with leverage score sampling for a full rank matrix

$\mathbf{X}$ we require at least $d \log d$ rows (Theorem 1), whereas a Projection DPP generates only d samples and also provides a rank-preserving sketch (Theorem 4). This shows that losing the negative correlations costs us a factor of $\log d$ in the sample size.

Another connection between leverage scores and Projection DPPs emerges in the reverse direction, i.e., going from i.i.d. to non-i.i.d. samples. Namely, a leverage score sample of size at least $2d \log d$ contains a Projection DPP with probability at least 1/2 [7].

Theorem 9. *Let $j_1, j_2, \ldots$ be a sequence of i.i.d. leverage score samples from matrix $\mathbf{X}$. There is a random set $T \subseteq \{1, 2, \ldots\}$ of size d s.t. $\max\{i \in T\} \leq 2d \log d$ with probability at least 1/2, and:*

$$\{j_i : i \in T\} \sim d\text{-DPP}_{\text{L}}(\mathbf{X}\mathbf{X}^\top).$$

Many extensions of leverage scores have been proposed for use in the low-rank approximation setting (i.e., when $n \sim d$). Arguably the most popular one is called *ridge leverage scores*. [2]. Ridge leverage scores can be recovered as the marginals of an L-ensemble.

Theorem 10. *For $S \sim \text{DPP}_{\text{L}}(\frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$ and any index i, the marginal probability of $i \in S$ is the λ-ridge leverage score of $\mathbf{X}$:*

$$\Pr\{i \in S\} = \mathbf{x}_i^\top (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{x}_i.$$

The typical sample size required for low-rank approximation with ridge leverage scores is at least $d_\lambda \log d_\lambda$, where d_λ is the ridge effective dimension (3) and also the expected size of the L-ensemble. Once again, the logarithmic factor appears as a trade-off coming from i.i.d. sampling.

A reverse connection analogous to Theorem 9, i.e., going from i.i.d. to non-i.i.d. sampling, can also be obtained for ridge leverage scores [5], although only a weaker version, with $O(d_\lambda^2)$ instead of $O(d_\lambda \log d_\lambda)$, is currently known in this setting.

Theorem 11. *Let $j_1, j_2, \ldots$ be a sequence of i.i.d. λ-ridge leverage score samples from matrix $\mathbf{X}$. There is a random set $T \subseteq \{1, 2, \ldots\}$ such that $\max\{i \in T\} \leq 2d_\lambda^2$ with probability at least 1/2, and:*

$$\{j_i : i \in T\} \sim \text{DPP}_{\text{L}}(\tfrac{1}{\lambda}\mathbf{X}\mathbf{X}^\top).$$

5. Sampling Algorithms

One of the key considerations in RandNLA is computational efficiency of constructing random sketches. For example, the i.i.d. leverage score sampling sketch defined in Section 2 requires precomputing all of the leverage scores. If done naïvely, this costs as much as performing the singular value decomposition (SVD) of the data. However, by employing fast RandNLA projection methods, one obtains efficient near-linear time complexity approximation algorithms for leverage score sampling [13].

In the case of DPPs, the challenge may seem even more daunting, since the naïve algorithm has exponential time complexity relative to the data size. However, the connections between leverage scores and DPPs (summarized in Table 1) have recently played a crucial role in the algorithmic improvements for DPP sampling. In particular, recent advances in DPP sampling have resulted in several algorithmic techniques which are faster than SVD, and in some regimes even approach the time complexity of fast leverage score sampling algorithms. See Table 2 for an overview.

5.1. Leverage scores: Approximation.

We start by discussing fast sketching methods for approximating leverage scores more rapidly than by naïvely computing them via the SVD or a QR decomposition. This is a good illustration of RandNLA algorithmic techniques, and it is also relevant in our later discussion of DPP sampling.

For simplicity, we focus on constructing leverage scores for rank-preserving sketches (i.e., for a tall $n \times d$ matrix $\mathbf{X}$), but similar ideas apply to the low-rank approximation setup [13]. Recall that the ith leverage score of $\mathbf{X}$ is given by $l_i = \mathbf{x}_i^\top (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{x}_i$, which can be expressed as the squared norm of the ith row of the matrix $\mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-\frac{1}{2}}$. Assuming that $n \gg d$, the primary computational cost of obtaining this matrix involves two expensive matrix multiplications: first, computing $\mathbf{R} = \mathbf{X}^\top \mathbf{X}$ (or a similarly expensive operation such as a QR decomposition or the SVD); and second, computing $\mathbf{X}\mathbf{R}^{-\frac{1}{2}}$. While each of these steps costs $O(nd^2)$ arithmetic operations, [13] showed that both of them can be approximated using efficient randomized sketching techniques, such as the Subsampled Randomized Hadamard Transform (SRHT) sketch [1]. The SRHT is a random sketching matrix $\mathbf{S}$ that, with high probability, satisfies the subspace embedding property (Definition 1), and that admits fast matrix-vector multiplication by exploiting recursive structure of the Hadamard matrix. The resulting overall procedure returns leverage score approximations in time $O(nd \log n + d^3 \log d)$, i.e., much faster than $O(nd^2)$ for the naïve algorithm.

A number of refinements have been proposed for approximating leverage scores, and similar approaches have also been developed for ridge leverage scores [2], which are used for low-rank approximation. In this case, we often consider the setting where instead of an $n \times d$ matrix $\mathbf{X}$, we are given an $n \times n$ matrix $\mathbf{K} = \mathbf{X}\mathbf{X}^\top$, i.e., the Gram matrix of the rows of $\mathbf{X}$ (this is particularly relevant in the context of DPPs). Here, λ-ridge leverage scores can be defined as the diagonal entries of the matrix $\mathbf{K}(\lambda \mathbf{I} + \mathbf{K})^{-1}$ and can be approximately computed in time $O(nd_\lambda^2 \,\text{poly}(\log n))$, with d_λ as in (3). When $d_\lambda \ll n$, this is less than the naïve cost of $O(n^3)$.

5.2. DPPs: Eigendecomposition.

In this and subsequent sections, we discuss several algorithmic techniques for

		Rank-preserving sketch		*Low-rank approximation*	
		Input: $n \times d$ data matrix $\mathbf{X}$, $n \gg d$ *Output:* Sample of size $k = O(d)$		*Input:* $n \times n$ kernel matrix $\mathbf{L}$ *Output:* Sample of size $k \ll n$	
		First sample	Subsequent samples	First sample	Subsequent samples
Lev. scores:	Exact	nd^2	d	n^3	k
	Approximate	$nd + d^3$	d	nk^2	k
DPPs:	Eigendecomposition	nd^2	d^3	n^3	$nk + k^3$
	Intermediate sampling	$nd + d^4$	d^4	$n \cdot \text{poly}(k)$	k^6
	Monte Carlo sampling	$n \cdot \text{poly}(d)$	$n \cdot \text{poly}(d)$	$n \cdot \text{poly}(k)$	$n \cdot \text{poly}(k)$

Table 2. Comparison of sampling cost for DPP algorithms, alongside the cost of exact and approximate leverage score sampling, given either a tall data matrix $\mathbf{X}$ or a square p.s.d. kernel $\mathbf{L}$. Most methods can also be extended to the wide data matrix $\mathbf{X}$ setting. We assume that an L-ensemble kernel $\mathbf{L}$ is used for the DPPs (if given a data matrix $\mathbf{X}$, we use $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$). We allow either a k-DPP or an L-ensemble with expected size k, however, in some cases, there are minor differences in the time complexities (in which case we give the better of the two). For simplicity, we omit the log terms in these expressions.

sampling from DPPs and k-DPPs. We focus on the general parameterization of a DPP via an $n \times n$ kernel matrix (either the marginal kernel $\mathbf{K}$ or the L-ensemble kernel $\mathbf{L}$), but we also discuss how these techniques can be applied to sampling from DPPs defined on a tall $n \times d$ matrix $\mathbf{X}$, which we used in Section 4.

We start with an important result of [17], which shows that any DPP can be decomposed into a mixture of Projection DPPs.

Theorem 12. *Consider the eigendecomposition* $\mathbf{K} = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$, *where* $\lambda_i \in [0, 1]$ *and* $\|\mathbf{u}_i\| = 1$. *For each* i, *let* s_i *be a random variable which is* 1 *with probability* λ_i *and* 0 *otherwise. Then the mixture distribution* $\text{DPP}(\sum_i s_i \mathbf{u}_i \mathbf{u}_i^\top)$ *is identical to* $\text{DPP}(\mathbf{K})$.

Note that $\sum_i s_i \mathbf{u}_i \mathbf{u}_i^\top$ randomly selects one of 2^n projection matrices (all their eigenvalues are 0 or 1), which defines a corresponding Projection DPP. In addition to the mixture decomposition, [17] also gave a simple $O(nk^2)$ time procedure for sampling from this Projection DPP, where k denotes the sample size $\sum_i s_i$. This procedure was recently accelerated to $O(nk + k^3 \log k)$ by [7]. Thus, combining the mixture decomposition and the Projection DPP algorithm, it became possible to sample from any determinantal point process in low-degree polynomial time, which significantly broadened the popularity of DPPs in the computer science community.

The overall sampling procedure proposed by [17] is very efficient, if we are given the eigendecomposition of kernel $\mathbf{K}$ or of the L-ensemble kernel $\mathbf{L}$. It can also be easily adapted to sampling cardinality constrained DPPs. However, obtaining the eigendecomposition itself can be a significant bottleneck: it costs $O(n^3)$ time for a general $n \times n$ kernel. If we are given a tall $n \times d$ matrix $\mathbf{X}$ such that $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$, as was the case in Section 4, then the sampling cost can be reduced to $O(nd^2)$. There have been a number of attempts at avoiding the eigendecomposition in this procedure, leading to several approximate algorithms. Finally, approaches using other factorizations of the kernel matrix have been proposed, and these offer computational advantages in certain settings.

5.3. DPPs: Intermediate sampling.

The DPP sampling algorithms from Section 5.2 can be accelerated with a recently introduced technique [5, 6], which uses leverage score sampling to reduce the size of the $n \times n$ kernel matrix, without distorting the underlying DPP distribution. Recall from Section 4.4 that i.i.d. leverage score sampling can be viewed as an approximation of a DPP in which we ignore the negative correlations between the sampled points. Naturally, in most cases such a sample as a whole will be a very poor approximation of a DPP. However, with high probability, it contains a DPP of a smaller size (Theorems 9 and 11).

This motivates a strategy called *distortion-free intermediate sampling*. To explain how this strategy can be implemented, we will consider the special case of a Projection DPP of size k, where this approach was first introduced by [10]. They showed that an i.i.d. sample of indices $i_1, ..., i_t$ of size $t = O(k^2)$ drawn proportionally to the leverage scores contains with high probability a subset S distributed according to the desired Projection DPP. Moreover, this subset can be found by downsampling with a DPP restricted to the smaller sample. This procedure essentially reduces the task of sampling from a DPP over a large domain $\{1, ..., n\}$ into sampling from a potentially much smaller domain of size $O(k^2)$. Surprisingly, this can be performed without any loss in accuracy, so that the final sample is drawn exactly from the target distribution. Similar intermediate sampling methods were later developed by [5, 6] for L-ensembles (where ridge leverage scores are used instead of the standard leverage scores) and k-DPPs, resulting in a linear in n preprocessing cost (instead of cubic for the eigendecomposition), and sampling cost independent of n (see Table 2).

Theorem 13. *Let* S_1, S_2 *be i.i.d. random sets from* $\text{DPP}_\text{L}(\mathbf{L})$, *with* $k = \mathbb{E}[|S|]$ *or from any* k-$\text{DPP}(\mathbf{L})$. *Then, given access to* $\mathbf{L}$, *we can return*

1. *first,* S_1 *in:* $n \cdot \text{poly}(k \log n)$ *time,*
2. *then,* S_2 *in:* $\text{poly}(k)$ *time.*

Analogous time complexity statements can be provided when $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$ and we are given $\mathbf{X}$. In this case, the first sample can be obtained in $O(nd \log n + \text{poly}(d))$ time, and each subsequent sample takes $\text{poly}(d)$ time [5]. Also, extensions of intermediate sampling exist for classes of distributions beyond DPPs, including all Strongly Rayleigh measures [21].

5.4. **DPPs: Monte Carlo sampling.** A completely different approach of (approximately) sampling from a DPP was proposed by [3], who showed that a simple fast-mixing Monte Carlo Markov chain (MCMC) algorithm has a cardinality constrained L-ensemble k-$\text{DPP}_\text{L}(\mathbf{L})$ as its stationary distribution. The state space of this chain consists of subsets $S \subseteq [n]$ of some fixed cardinality k. At each step, we choose an index $i \in S$ and $j \notin S$ uniformly at random. Letting $T = S \cup \{j\} \backslash \{i\}$, we transition from S to T with probability

$$\frac{1}{2} \min\left\{1, \frac{\det(\mathbf{L}_{T,T})}{\det(\mathbf{L}_{S,S})}\right\},$$

and otherwise, stay in S. It is easy to see that the stationary distribution of the above Markov chain is k-$\text{DPP}_\text{L}(\mathbf{L})$. Moreover, [3] showed that the mixing time can be bounded as follows.

Theorem 14. *The number of steps required to get to within ϵ total variation distance from k-$\text{DPP}_\text{L}(\mathbf{L})$ is at most* $\text{poly}(k)\, O\big(n \log(n/\epsilon)\big)$.

The advantages of these sampling procedures over the algorithm of [17] are that we are not required to perform the eigendecomposition and that the computational cost of the MCMC algorithm scales linearly with n. The disadvantages are that the sampling is approximate and that we have to run the entire chain every time we wish to produce a new sample S.

6. Looking Forward

We have briefly surveyed two established research areas which exhibit deep connections that have only recently began to emerge:

1. Randomized Numerical Linear Algebra; and
2. Determinantal Point Processes.

In particular, we discussed recent developments in applying DPPs to classical tasks in RandNLA, such as least squares regression and low-rank approximation; and we surveyed recent results on sampling algorithms for DPPs, comparing and contrasting several different approaches.

We expect that these connections will be fruitful more generally. As an example of this, we briefly mention a recently proposed mathematical framework for studying determinants, which played a key role in obtaining some of these results.

Determinant-preserving random matrices. A square random matrix $\mathbf{A}$ is determinant preserving (d.p.) if all of its subdeterminants commute with taking the expectation, i.e., if:

$$\mathbb{E}\, \det(\mathbf{A}_{S,T}) = \det(\mathbb{E}\, \mathbf{A}_{S,T})$$

for all index subsets S, T of the same size. Not all random matrices satisfy this property, however there are many non-trivial examples. For instance, consider $\mathbf{A} = X\mathbf{C}$, where X is a scalar random variable with positive variance and $\mathbf{C}$ is a nonzero deterministic square matrix. Then, $\mathbf{A}$ is d.p. if and only if $\mathbf{C}$ is rank 1. More elaborate positive examples, such as matrices with independent random entries, can be constructed by taking advantage of the algebraic structure of the d.p. class: if $\mathbf{A}$ and $\mathbf{B}$ are independent and d.p., then both $\mathbf{A} + \mathbf{B}$ and $\mathbf{AB}$ are also determinant preserving. The first examples of d.p. matrices where given by [5] (used in the analysis of the fast DPP sampling algorithm from Theorem 13). Further discussion can be found in [8].

Of course, our survey of the applications of DPPs necessarily excluded many areas where this family of distributions appears. Here, we briefly discuss some other applications of DPPs which are relevant in the context of NLA and RandNLA but did not fit in the scope of this work.

Implicit regularization. In many optimization tasks (e.g., in machine learning), the true minimizer of a desired objective is not unique or not computable exactly, so that the choice of the optimization procedure affects the output. Implicit regularization occurs when these algorithmic choices provide an effect similar to explicitly introducing a regularization penalty into the objective. This has been observed for approximate solutions returned by stochastic and combinatorial optimization algorithms, but a precise characterization of this phenomenon for RandNLA sampling methods has proven challenging. Recently, DPPs have been used to derive exact expressions for implicit regularization in RandNLA algorithms [8], connecting it to a phase transition called the double descent curve.

Optimal design of experiments. In statistics, the task of selecting a subset of data points for a downstream regression task is referred to as optimal design. In this context, it is often assumed that the coefficients y_i (or responses) are random variables obtained as a linear transformation of the vector $\mathbf{x}_i$ distorted by some mean zero noise, as in (4). A number of optimality criteria (such as A-optimality, which uses mean squared error of the least squares estimator) have been considered for selecting the subsets. DPP subset selection has been shown to provide useful guarantees for some of the most popular criteria (including A-, C-, D-, and V-optimality), leading to new approximation algorithms [7, 22].

Stochastic optimization. Randomized selection of small batches of data or subsets of parameters has been very successful in speeding up many iterative optimization algorithms. Here, nonuniform sampling can be used to

reduce the bias and/or variance in the iteration steps. In particular, [25] showed that using a DPP for sampling mini-batches in stochastic gradient descent improves the convergence rate of the optimizer.

Monte Carlo integration. DPPs have been shown to achieve theoretically improved guarantees for numerical integration, i.e., using a weighted sum of function evaluations to approximate an integral. In particular, [4] constructed a DPP for which the root mean squared errors of Monte Carlo integration decrease as $n^{-(1+1/d)/2}$, where n is the number of function evaluations and d is the dimension. This is faster than the typical $n^{-1/2}$ rate.

In conclusion, despite having been studied for at least forty-five years, DPPs are enjoying an explosion of renewed interest, with novel applications emerging on a regular basis. Their rich connections to RandNLA, which we have only briefly summarized and which offer a nice example of how deep mathematics informs practical problems and vice versa, provide a particularly fertile ground for future work.

ACKNOWLEDGMENTS. We would like to acknowledge DARPA, NSF (via the TRIPODS program), and ONR (via the BRC on RandNLA) for providing partial support for this work.

References

[1] Nir Ailon and Bernard Chazelle, *The fast Johnson-Lindenstrauss transform and approximate nearest neighbors*, SIAM J. Comput. **39** (2009), no. 1, 302–322, DOI 10.1137/060673096. MR2506527

[2] Ahmed El Alaoui and Michael W. Mahoney, *Fast randomized kernel ridge regression with statistical guarantees*, Proceedings of the 28th International Conference on Neural Information Processing Systems, 2015, pp. 775–783.

[3] Nima Anari, Shayan Oveis Gharan, and Alireza Rezaei, *Monte Carlo Markov chain algorithms for sampling strongly Rayleigh distributions and determinantal point processes*, 29th Annual Conference on Learning Theory, 2016, pp. 103–115.

[4] Rémi Bardenet and Adrien Hardy, *Monte Carlo with determinantal point processes*, Ann. Appl. Probab. **30** (2020), no. 1, 368–417, DOI 10.1214/19-AAP1504. MR4068314

[5] Michał Dereziński, *Fast determinantal point processes via distortion-free intermediate sampling*, Proceedings of the 32nd Conference on Learning Theory, 2019, pp. 1029–1049.

[6] Michał Dereziński, Daniele Calandriello, and Michal Valko, *Exact sampling of determinantal point processes with sublinear time preprocessing*, Advances in neural information processing systems, 2019, pp. 11542–11554.

[7] Michał Dereziński, Kenneth L. Clarkson, Michael W. Mahoney, and Manfred K. Warmuth, *Minimax experimental design: Bridging the gap between statistical and worst-case approaches to least squares regression*, Proceedings of the Thirty-Second Conference on Learning Theory, 2019, pp. 1050–1069.

[8] Michał Dereziński, Feynman Liang, and Michael W. Mahoney, *Exact expressions for double descent and implicit regularization via surrogate random design*, Advances in neural information processing systems, 2020.

[9] Michał Dereziński and Manfred K. Warmuth, *Reverse iterative volume sampling for linear regression*, J. Mach. Learn. Res. **19** (2018), Paper No. 23, 39. MR3862430

[10] Michał Dereziński, Manfred K Warmuth, and Daniel Hsu, *Unbiased estimators for random design regression*, arXiv preprint, arXiv:1907.03411 (2019).

[11] Petros Drineas, Ravi Kannan, and Michael W. Mahoney, *Fast Monte Carlo algorithms for matrices. I. Approximating matrix multiplication*, SIAM J. Comput. **36** (2006), no. 1, 132–157, DOI 10.1137/S0097539704442684. MR2231643

[12] P. Drineas and M. W. Mahoney, *RandNLA: Randomized numerical linear algebra*, Communications of the ACM **59** (2016), 80–90.

[13] 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** (2012), 3475–3506. MR3033372

[14] Petros Drineas, Michael W. Mahoney, and S. Muthukrishnan, *Sampling algorithms for l_2 regression and applications*, Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, 2006, pp. 1127–1136, DOI 10.1145/1109557.1109682. MR2373840

[15] Venkatesan Guruswami and Ali Kemal Sinop, *Optimal column-based low-rank matrix reconstruction*, Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, 2012, pp. 1207–1214. MR3205285

[16] N. Halko, P. G. Martinsson, and J. A. Tropp, *Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev. **53** (2011), no. 2, 217–288, DOI 10.1137/090771806. MR2806637

[17] J. Ben Hough, Manjunath Krishnapur, Yuval Peres, and Bálint Virág, *Determinantal processes and independence*, Probab. Surv. **3** (2006), 206–229, DOI 10.1214/154957806000000078. MR2216966

[18] Alex Kulesza and Ben Taskar, *Determinantal point processes for machine learning*, Now Publishers Inc., Hanover, MA, USA, 2012.

[19] Odile Macchi, *The coincidence approach to stochastic point processes*, Advances in Applied Probability **7** (1975), no. 1, 83–122.

[20] M. W. Mahoney, *Randomized algorithms for matrices and data*, Foundations and Trends in Machine Learning, NOW Publishers, Boston, 2011.

[21] N. Anari and M. Dereziński, *Isotropy and Log-Concave Polynomials: Accelerated Sampling and High-Precision Counting of Matroid Bases*, Proceedings of the 61st Annual Symposium on Foundations of Computer Science (2020).

[22] Aleksandar Nikolov, Mohit Singh, and Uthaipon Tao Tantipongpipat, *Proportional volume sampling and approximation algorithms for A-optimal design*, Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete

Algorithms, SIAM, Philadelphia, PA, 2019, pp. 1369–1386, DOI 10.1137/1.9781611975482.84. MR3909553
[23] Tamas Sarlos, *Improved approximation algorithms for large matrices via random projections*, Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, 2006, pp. 143–152.
[24] David P. Woodruff, *Sketching as a tool for numerical linear algebra*, Found. Trends Theor. Comput. Sci. **10** (2014), no. 1-2, iv+157, DOI 10.1561/0400000060. MR3285427
[25] Cheng Zhang, Hedvig Kjellström, and Stephan Mandt, *Determinantal point processes for mini-batch diversification*, 33rd Conference on Uncertainty in Artificial Intelligence, UAI 2017, 2017.


Michał Dereziński


Michael W. Mahoney

Credits

Opening graphic is courtesy of Ryzhi via Getty.
Figures 1 and 2 are courtesy of Michał Dereziński.
Photo of Michał Dereziński is courtesy of Julie Lucas.
Photo of Michael W. Mahoney is courtesy of Madeleine Fitzgerald.