Determinantal Point Processes and Randomized Numerical Linear Algebra

Michael W. Mahoney
ICSI and Department of Statistics, UC Berkeley

April 2020

(Joint work with Michał Dereziński.)

Outline

Introduction

Determinantal point processes

DPPs in Randomized Linear Algebra

Key technique: Determinant preserving random matrices

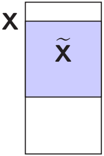
Sampling algorithms

Conclusions

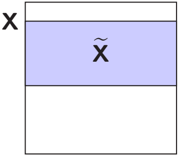
Randomized Linear Algebra

Given: data matrix $\mathbf{X}$

Goal: efficiently construct a small sketch $\widetilde{\mathbf{X}}$



Rank-preserving sketch



Low-rank approximation

Determinants

$$\det(\mathbf{A}) = \prod_i \lambda_i(\mathbf{A})$$

Some popular wisdom about determinants:

- Expensive to compute
- Numerically unstable
- Exponentially large... or exponentially small

Down With Determinants!

Sheldon Axler

1. INTRODUCTION. Ask anyone why a square matrix of complex numbers has an eigenvalue, and you'll probably get the wrong answer, which goes something

And yet... Determinantal Point Processes (DPPs)

A family of non-i.i.d. sampling distributions

1. Applications in Randomized Linear Algebra
   - Least squares regression [DW17, DWH18]
   - Low-rank approximation [DRVW06, GS12, DKM20]
   - Randomized Newton's method [DM19, MDK19]
2. Connections to i.i.d. sampling methods
   - Row norm scores
   - Leverage scores
   - Ridge leverage scores
3. Fast DPP sampling algorithms
   - Exact sampling via eigendecomposition [HKP$^+$06, KT11]
   - Intermediate sampling via leverage scores [Der19, DCV19]
   - Markov chain Monte Carlo sampling [AGR16]

Outline

L-ensemble DPPs and k-DPPs

Given a psd $n \times n$ matrix $\mathbf{L}$, sample subset $S \subseteq \{1..n\}$:

$$\text{(L-ensemble)} \quad \mathrm{DPP}(\mathbf{L}): \quad \Pr(S) = \frac{\det(\mathbf{L}_{S,S})}{\det(\mathbf{I}+\mathbf{L})} \quad \text{over all subsets.}$$

closed form normalization!

$$\text{(k-DPP)} \quad k\text{-}\mathrm{DPP}(\mathbf{L}): \quad \mathrm{DPP}(\mathbf{L}) \text{ conditioned on } |S| = k.$$

DPPs appear everywhere!

- Physics (fermions)
- Random matrix theory (eigenvalue distribution)
- Graph theory (random spanning trees)
- Optimization (variance reduction)
- Machine learning (diverse sets)

Volume (determinant) as a measure of diversity

Let $\mathbf{L} = \left[\mathbf{x}_i^\top \mathbf{x}_j\right]_{ij}$ for $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$.
Then, $\det(\mathbf{L}_{S,S}) = \mathrm{Vol}^2\left(\{\mathbf{x}_i : i \in S\}\right)$

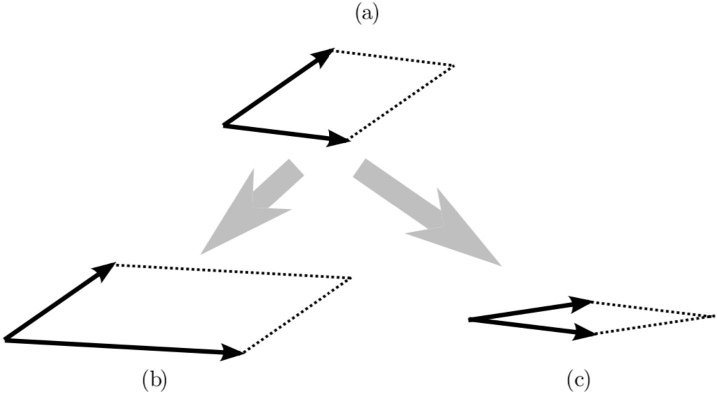
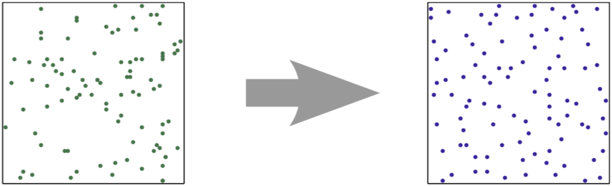


Image from [KT12]

Example: DPP vs i.i.d.

Negative correlation: $\Pr(i \in S \mid j \in S) < \Pr(i \in S)$



i.i.d. (left) versus DPP (right)

Image from [KT12]

Projection DPPs

If $\mathbf{L}$ has rank d, then $S \sim d\text{-}\mathrm{DPP}(\mathbf{L})$ is a Projection DPP

Let $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$ for a full rank $n \times d$ matrix $\mathbf{X}$

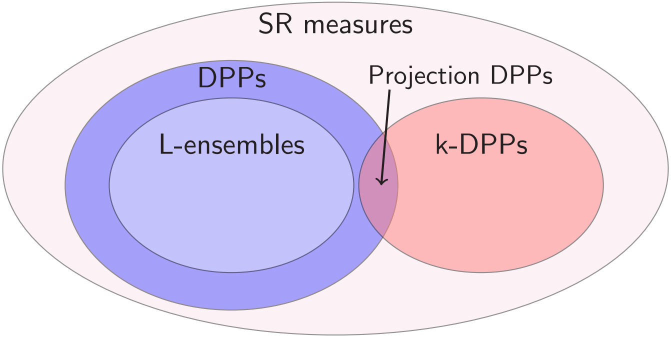
$$\text{if} \quad S \sim d\text{-}\mathrm{DPP}(\mathbf{L}) \quad \text{then} \quad \Pr(S) = \frac{\det(\mathbf{X}_S)^2}{\det(\mathbf{X}^\top \mathbf{X})}.$$

Closed form normalization (Cauchy-Binet formula).

Remark. If $k < \mathrm{rank}(\mathbf{L})$ then $k\text{-}\mathrm{DPP}(\mathbf{L})$ is not a projection DPP.
(and also does not have such a simple normalization constant)

Hierarchy of DPPs

Broader class of negatively-correlated point processes:
Strongly Rayleigh (SR) measures



Random vs fixed subset size

Let $d = \mathrm{rank}(\mathbf{L})$, and $\lambda_1, ..., \lambda_d$ be the non-zero eigenvalues of $\mathbf{L}$
If $S \sim \mathrm{DPP}(\mathbf{L})$ then:

$$|S| \sim \text{Poisson-Binomial}\left(\tfrac{\lambda_1}{\lambda_1+1}, ..., \tfrac{\lambda_d}{\lambda_d+1}\right)$$

$$\mathbb{E}\big[|S|\big] = \sum_{i=1}^{d} \frac{\lambda_i}{\lambda_i + 1} = \mathrm{tr}\big(\mathbf{L}(\mathbf{L}+\mathbf{I})^{-1}\big) < d$$

Rescaling trick: Sample $S \sim \mathrm{DPP}(\frac{1}{\lambda}\mathbf{L})$ to control $\mathbb{E}[|S|]$

$$\mathrm{Pr}(S) \propto \det(\tfrac{1}{\lambda}\mathbf{L}_{S,S}) = \lambda^{-|S|}\det(\mathbf{L}_{S,S})$$

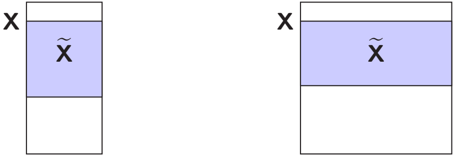
$$\underbrace{\mathrm{DPP}\big(\tfrac{1}{\lambda}\mathbf{L}\big)}_{\text{L-ensemble}} \xrightarrow{\lambda \to 0} \underbrace{d\text{-}\mathrm{DPP}\big(\mathbf{L}\big)}_{\text{Projection DPP}}$$

Outline

DPPs in Randomized Linear Algebra

Given: data matrix $\mathbf{X}$

Goal (row sampling): construct $\widetilde{\mathbf{X}}$ from few rows of $\mathbf{X}$



	Rank-preserving sketch	*Low-rank approximation*
i.i.d. sampling:	*Leverage scores*	*Ridge leverage scores*
DPP sampling:	*Projection DPPs*	*L-ensembles*

Connections to i.i.d. sampling

Given: full rank $n \times d$ matrix $\mathbf{X}$

Methods based on i.i.d. row sampling:

1. Row norm scores: $p_i = \frac{\|\mathbf{x}_i\|^2}{\|\mathbf{X}\|_F^2}$

$$\frac{\|\mathbf{x}_i\|^2}{\|\mathbf{X}\|_F^2} = \Pr\big(i \in S\big) \quad \text{for} \quad S \sim 1\text{-DPP}(\mathbf{X}\mathbf{X}^\top)$$

2. Leverage scores: $p_i = \frac{1}{d}\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}_i$

$$\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{x}_i = \Pr\big(i \in S\big) \quad \text{for} \quad S \sim d\text{-DPP}(\mathbf{X}\mathbf{X}^\top)$$

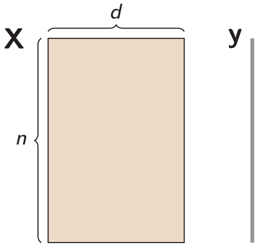
3. Ridge leverage scores: $p_i = \frac{1}{d_\lambda}\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{x}_i$

$$\mathbf{x}_i^\top(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{x}_i = \Pr\big(i \in S\big) \quad \text{for} \quad S \sim \text{DPP}(\tfrac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$$

Subsampled least squares

Given: n points $\mathbf{x}_i \in \mathbb{R}^d$ with labels $y_i \in \mathbb{R}$
Goal: Minimize loss $L(\mathbf{w}) = \sum_i (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$ over all n points

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}}\, L(\mathbf{w}) = \mathbf{X}^\dagger \mathbf{y}$$



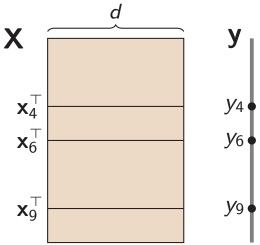
Subsampled least squares

Given: n points $\mathbf{x}_i \in \mathbb{R}^d$ with labels $y_i \in \mathbb{R}$
Goal: Minimize loss $L(\mathbf{w}) = \sum_i (\mathbf{x}_i^\top \mathbf{w} - y_i)^2$ over all n points

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} L(\mathbf{w}) = \mathbf{X}^\dagger \mathbf{y}$$

Sample $S = \{4, 6, 9\}$

Solve subproblem $(\mathbf{X}_S, \mathbf{y}_S)$



Unbiased estimators

Theorem (Rank-preserving sketch, [DW17])

If $S \sim d\text{-}\mathrm{DPP}(\mathbf{X}\mathbf{X}^\top)$*, then:*

$$\mathbb{E}[\mathbf{X}_S^{-1}\mathbf{y}_S] = \overbrace{\underset{\mathbf{w}}{\operatorname{argmin}}\, L(\mathbf{w}) = \mathbf{w}^*}^{\textit{least squares}}.$$

Theorem (Low-rank sketch, [DLM19])

If $S \sim \mathrm{DPP}(\frac{1}{\lambda}\mathbf{X}\mathbf{X}^\top)$*, then:*

$$\mathbb{E}[\mathbf{X}_S^{\dagger}\mathbf{y}_S] = \overbrace{\underset{\mathbf{w}}{\operatorname{argmin}}\, L(\mathbf{w}) + \lambda\|\mathbf{w}\|^2}^{\textit{ridge regression}}$$

Not achievable with any i.i.d. row sampling!

Merits of unbiased estimators

Simple Strategy:

1. Compute independent estimators $\mathbf{w}(S_j)$ for $j = 1, .., k$,
2. Predict with the average estimator $\frac{1}{k}\sum_{j=1}^{k} \mathbf{w}(S_j)$

If we have

$$\mathbb{E}[L(\mathbf{w}(S))] \leq (1+c)L(\mathbf{w}^*) \quad \text{and} \quad \mathbb{E}[\mathbf{w}(S)] = \mathbf{w}^*,$$

then for k independent samples $S_1, \ldots, S_k$,

$$\mathbb{E}\left[L\Big(\frac{1}{k}\sum_{j=1}^{k} \mathbf{w}(S_j)\Big)\right] \leq \Big(1 + \frac{c}{k}\Big)\, L(\mathbf{w}^*)$$

Motivation:

- Ensemble methods
- Distributed optimization
- Privacy

Connections to Gaussian sketches

Gaussian sketch

(Also gives *unbiased* estimators for least squares)

Let $\mathbf{S}$ be $\frac{1}{k}\times$ i.i.d. Gaussian with $\mathbb{E}[\mathbf{S}^\top\mathbf{S}] = \mathbf{I}$. For $k > d+1$:

$$\mathbb{E}\big[(\mathbf{X}^\top\mathbf{S}^\top\mathbf{S}\mathbf{X})^{-1}\big] = (\mathbf{X}^\top\mathbf{X})^{-1}\frac{k}{k-d-1}$$

DPP plus uniform

Let $S \sim d\text{-}\mathrm{DPP}(\mathbf{X}\mathbf{X}^\top)$, $T \sim \mathrm{Bin}(n, \frac{k-d}{n-d})$ and $\bar{\mathbf{S}} = \big[\sqrt{\frac{n}{k}}\,\mathbf{e}_i\big]^\top_{i\in S\cup T}$.
Note: $\mathbb{E}[|S|] = k$. For $k \geq d$, we have:

$$\mathbb{E}\big[(\mathbf{X}^\top\bar{\mathbf{S}}^\top\bar{\mathbf{S}}\mathbf{X})^{-1}\big] = (\mathbf{X}^\top\mathbf{X})^{-1}\frac{k}{k-d}\cdot\big(1-o_n(1)\big)$$

DPPs have a “Gaussianizing” effect on row sampling.

Outline

Determinant preserving random matrices

Definition ([DLM19])

A random $d \times d$ matrix $\mathbf{A}$ is determinant preserving (d.p.) if

$$\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}})\big] = \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big) \quad \text{for all } \mathcal{I},\mathcal{J} \subseteq [d] \text{ s.t. } |\mathcal{I}| = |\mathcal{J}|.$$

Basic examples:

- Every *deterministic* matrix
- Every *scalar* random variable
- Random matrix with i.i.d. Gaussian entries

More examples

Let $\mathbf{A} = s\,\mathbf{Z}$, where:

- $\mathbf{Z}$ is deterministic with $\mathrm{rank}(\mathbf{Z}) = r$,
- s is a scalar random variable with positive variance.

$$\mathbb{E}\big[\det(s\,\mathbf{Z}_{\mathcal{I},\mathcal{J}})\big] = \mathbb{E}[s^r]\det(\mathbf{Z}_{\mathcal{I},\mathcal{J}}) = \det\Big(\big(\mathbb{E}[s^r]\big)^{\frac{1}{r}}\,\mathbf{Z}_{\mathcal{I},\mathcal{J}}\Big),$$

Two cases:

1. If $r = 1$ then $\mathbf{A}$ is determinant preserving,
2. If $r > 1$ then $\mathbf{A}$ is not determinant preserving.

Basic properties

Lemma (Closure)

If $\mathbf{A}$ *and* $\mathbf{B}$ *are independent and determinant preserving, then:*

- $\mathbf{A} + \mathbf{B}$ *is determinant preserving,*
- $\mathbf{AB}$ *is determinant preserving.*

Lemma (Adjugate)

If $\mathbf{A}$ *is determinant preserving, then* $\mathbb{E}[\mathrm{adj}(\mathbf{A})] = \mathrm{adj}(\mathbb{E}[\mathbf{A}])$.

When $\mathbf{A}$ is invertible then $\mathrm{adj}(\mathbf{A}) = \det(\mathbf{A})\mathbf{A}^{-1}$

Note: The (i,j)th entry of $\mathrm{adj}(\mathbf{A})$ is $(-1)^{i+j}\det(\mathbf{A}_{[n]\setminus\{j\},[n]\setminus\{i\}})$.

Proof of closure under addition

First show that $\mathbf{A} + \mathbf{u}\mathbf{v}^\top$ is d.p. for fixed $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$:

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{u}_{\mathcal{I}}\mathbf{v}_{\mathcal{J}}^\top)\big] &= \mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}}) + \mathbf{v}_{\mathcal{J}}^\top \operatorname{adj}(\mathbf{A}_{\mathcal{I},\mathcal{J}})\mathbf{u}_{\mathcal{I}}\big] \\
&= \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big) + \mathbf{v}_{\mathcal{J}}^\top \operatorname{adj}\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}]\big)\mathbf{u}_{\mathcal{I}} \\
&= \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{u}_{\mathcal{I}}\mathbf{v}_{\mathcal{J}}^\top]\big).
\end{aligned}$$

Iterating this, we get $\mathbf{A} + \mathbf{Z}$ is d.p. for any fixed $\mathbf{Z}$

$$\begin{aligned}
\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}})\big] &= \mathbb{E}\Big[\mathbb{E}\big[\det(\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}}) \mid \mathbf{B}\big]\Big] \\
&= \mathbb{E}\Big[\det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}}] + \mathbf{B}_{\mathcal{I},\mathcal{J}}\big)\Big] \\
&= \det\big(\mathbb{E}[\mathbf{A}_{\mathcal{I},\mathcal{J}} + \mathbf{B}_{\mathcal{I},\mathcal{J}}]\big)
\end{aligned}$$

$\square$

Application: Expected inverse

Theorem

Let $\Pr(S) \propto \det(\mathbf{X}_S^\top \mathbf{X}_S) p^{|S|}(1-p)^{n-|S|}$ *over all* $S \subseteq [n]$*. Then:*

$$\mathbb{E}\big[(\mathbf{X}_S^\top \mathbf{X}_S)^{-1}\big] \preceq \frac{1}{p}\,(\mathbf{X}^\top \mathbf{X})^{-1}.$$

Proof Let $b_1, ..., b_n \sim \mathrm{Bernoulli}(p)$, and define $\bar{S} = \{i : b_i = 1\}$
For each $i \in [n]$, matrix $b_i \mathbf{x}_i \mathbf{x}_i^\top$ is determinant preserving
Therefore, $\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}} = \sum_{i=1}^n b_i \mathbf{x}_i \mathbf{x}_i^\top$ is determinant preserving

$$\mathbb{E}\big[(\mathbf{X}_S^\top \mathbf{X}_S)^{-1}\big] = \frac{\mathbb{E}[\det(\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}})(\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}})^\dagger]}{\mathbb{E}[\det(\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}})]} \preceq \frac{\mathbb{E}[\mathrm{adj}(\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}})]}{\mathbb{E}[\det(\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}})]}$$

$$= \frac{\mathrm{adj}(\mathbb{E}[\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}}])}{\det(\mathbb{E}[\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}}])} = \big(\mathbb{E}[\mathbf{X}_{\bar{S}}^\top \mathbf{X}_{\bar{S}}]\big)^{-1} = (p\mathbf{X}^\top \mathbf{X})^{-1}$$

□

Outline

Algorithmic challenges with sampling from DPPs

Task:

(variant 1) *Given* $\mathbf{L}$, *sample* $S \sim \mathrm{DPP}(\mathbf{L})$

(variant 2) *Given* $\mathbf{L}$ *and* k, *sample* $S \sim k\text{-}\mathrm{DPP}(\mathbf{L})$

(Task B: we are given $n \times d$ matrix $\mathbf{X} \in \mathbb{R}^d$ instead of $\mathbf{L} = \mathbf{X}\mathbf{X}^\top$)

Challenges:

1. Expensive preprocessing
   typically involves eigendecomposition of $\mathbf{L}$ in $O(n^3)$ time
2. Sampling time scales with n rather than with $|S| \ll n$
   undesirable when we need many samples $S_1, S_2, \dots \sim \mathrm{DPP}(\mathbf{L})$
3. Trade-offs between accuracy and runtime
   - exact algorithms - often too expensive
   - approximate algorithms - difficult to evaluate accuracy

Exact DPP sampling

Key result: any DPP is a mixture of Projection DPPs [HKP+06]

- Eigendecomposition $O(n^3)$
  needed only once for a given kernel
- Reduction to a projection DPP $O(n\,|S|^2)$
  needed for every sample

- Cost of first sample $S_1 \sim \mathrm{DPP}(\mathbf{L})$: $O(n^3)$
- Cost of next sample $S_2 \sim \mathrm{DPP}(\mathbf{L})$: $O(nk^2)$ $(k = \mathbb{E}[|S|])$

Extends to a k-DPP sampler [KT11]

Approximate k-DPP sampler using MCMC

1. Start from some state $S \subseteq [n]$ of size k
2. Uniformly sample $i \in S$ and $j \notin S$
3. Move to state $S - i + j$ with probability $\frac{1}{2} \min \left\{1, \frac{\det(\mathbf{L}_{S-i+j})}{\det(\mathbf{L}_S)}\right\}$
4. ...

Converges in $O(nk \log \frac{1}{\epsilon})$ steps to within ϵ total variation [AGR16]

- Cost of first sample $S_1 \sim k\text{-DPP}(\mathbf{L})$: $\quad O(n \cdot \mathrm{poly}(k))$
- Cost of next sample $S_2 \sim k\text{-DPP}(\mathbf{L})$: $\quad O(n \cdot \mathrm{poly}(k))$

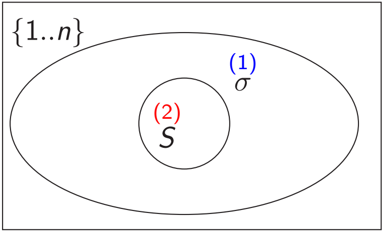
Extends to an $O(n^2 \cdot \mathrm{poly}(k))$ sampler for $\mathrm{DPP}(\mathbf{L})$ [LJS16]

Distortion-free intermediate sampling

1. Draw an intermediate sample:

$$\sigma = (\sigma_1, \ldots, \sigma_t)$$

2. Downsample: $S \subseteq [t]$
3. Return: $\{\sigma_i : i \in S\}$



What is the right intermediate sampling distribution for σ?

- Leverage scores, when S is a Projection DPP
- Ridge leverage scores, when S is an L-ensemble

Distortion-free intermediate sampling for L-ensembles

Theorem ([DCV19])

There is an algorithm which, given access to $\mathbf{L}$, *returns*

1. *first sample* $S_1 \sim \mathrm{DPP}(\mathbf{L})$ *in:* $n \cdot \mathrm{poly}(k)\,\mathrm{polylog}(n)$ *time,*
2. *next sample* $S_2 \sim \mathrm{DPP}(\mathbf{L})$ *in:* $\mathrm{poly}(k)$ *time.*

- Exact sampling
- Cost of first sample is sublinear in the size of $\mathbf{L}$
- Cost of next sample is independent of the size of $\mathbf{L}$

Outline

Conclusions

1. New fundamental connections between:
   1.1 Determinantal Point Processes
   1.2 Randomized Linear Algebra

2. New unbiased estimators and expectation formulas

3. Efficient sampling algorithms

4. *Determinant preserving random matrices*

DPP-related topics we did not cover:

- Column Subset Selection Problem
- Nyström method
- Monte Carlo integration
- Distributed/Stochastic optimization
- ...

References I

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.

Michał Dereziński, Daniele Calandriello, and Michal Valko.
Exact sampling of determinantal point processes with sublinear time preprocessing.
In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems 32, pages 11542–11554. Curran Associates, Inc., 2019.

Michał Dereziński.
Fast determinantal point processes via distortion-free intermediate sampling.
In Alina Beygelzimer and Daniel Hsu, editors, Proceedings of the Thirty-Second Conference on Learning Theory, volume 99 of Proceedings of Machine Learning Research, pages 1029–1049, Phoenix, USA, 25–28 Jun 2019.

Michał Dereziński, Rajiv Khanna, and Michael W Mahoney.
Improved guarantees and a multiple-descent curve for the column subset selection problem and the nyström method.
arXiv preprint arXiv:2002.09073, 2020.

References II

Michał Dereziński, Feynman Liang, and Michael W. Mahoney.
Exact expressions for double descent and implicit regularization via surrogate random design.
arXiv e-prints, page arXiv:1912.04533, Dec 2019.

Michał Dereziński and Michael W Mahoney.
Distributed estimation of the inverse hessian by determinantal averaging.
In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems 32, pages 11401–11411. Curran Associates, Inc., 2019.

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.

Michał Dereziński and Manfred K. Warmuth.
Unbiased estimates for linear regression via volume sampling.
In Advances in Neural Information Processing Systems 30, pages 3087–3096, Long Beach, CA, USA, 2017.

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.

References III

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.
k-DPPs: Fixed-Size Determinantal Point Processes.
In Proceedings of the 28th International Conference on Machine Learning, pages 1193–1200, 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.
Fast mixing markov chains for strongly Rayleigh measures, DPPs, and constrained sampling.
In Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16, pages 4195–4203, 2016.

Mojmír Mutný, Michał Dereziński, and Andreas Krause.
Convergence analysis of the randomized newton method with determinantal sampling.
arXiv e-prints, page arXiv:1910.11561, Oct 2019.

Thank you!