Faster Greedy MAP Inference for Determinantal Point Processes

Insu Han[1], Prabhanjan Kambadur[2], KyoungSoo Park[1] and Jinwoo Shin[1]

[1]Korea Advanced Institute of Science and Technology (KAIST) and [2]Bloomberg LP

ICML | Sydney

Thirty-fourth International Conference on Machine Learning

Determinantal Point Processes (DPPs)

Given a ground set $\mathcal{Y} = \{1, \dots, d\}$ and positive definite matrix $L \in \mathbb{R}^{d \times d}$,

$$\Pr(X) \propto \det(L_X) \quad \text{for} \quad X \subseteq \mathcal{Y},$$

where L_X is a submatrix of L indexed by items of X.

- DPPs are probabilistic models capturing both diversity and item quality of subsets.
- Most inference tasks (including normalization, marginalization, conditioning and sampling) can be done in $O(d^3)$.
- However, MAP inference is known as **NP-hard** problem, that is,

$$\arg\max_{X \subseteq \mathcal{Y}} \det L_X.$$

- The MAP inference of DPP has been used for many machine learning applications, e.g., text/video summarization, change-point detection, and informative image search.

Our Contribution: Faster MAP Inference of DPP

Since $\log\det$ is a submodular function, greedy algorithms for approximating MAP of DPP have been of typical choice.

- A naïve greedy algorithm requires $O(d^5)$ operations.

algorithm	complexity	remarks
[Minoux, 1978]	$O(d^5)$	accelerated version of a naïve greedy algorithm
[Buchbinder et al., 2015]	$O(d^4)$	symmetric greedy algorithm
[Gillenwater et al., 2012]	$O(d^4)$	multilinear softmax extension

We propose faster greedy algorithms requiring $O(d^3)$ operations.

First Ideas: Talyor Expansion

Greedy algorithms require computing the following marginal gains:

$$\log\det L_{X \cup \{i\}} - \log\det L_X$$

For their efficient computations, our key ideas are:

1. **First-order Taylor expansion for Log-determinant**

$$\log\det L_{X \cup \{i\}} - \log\det \overline{L}_X \;\approx\; \left\langle \overline{L}_X^{-1}, L_{X \cup \{i\}} - \overline{L}_X \right\rangle.$$

   - $\overline{L}_X$ is the average of $L_{X \cup \{i\}}$ for $i \in \mathcal{Y} \setminus X$.
   - $L_{X \cup \{i\}}$ and $\overline{L}_X$ differ only single column and row.
   - Single column of $\overline{L}_X^{-1}$ is computed by a linear solver, e.g., conjugate gradient descent.

2. **Partitioning**

   - For much tighter approximation, we divide $\mathcal{Y} \setminus X$ into p partitions so that

$$\|L_{X \cup \{i\}} - \overline{L}_X\|_F \gg \|L_{X \cup \{i\}} - \overline{L}_X^{(j)}\|_F,$$

   where i is in the partition $j \in \{1, \dots, p\}$.
   - To compute the marginal gains, we need to calculate extra term $(*)$:

$$\log\det L_{X \cup \{i\}} - \log\det L_X$$
$$\approx \underbrace{\left\langle (\overline{L}_X^{(j)})^{-1}, L_{X \cup \{i\}} - \overline{L}_X^{(j)} \right\rangle}_{\text{can compute by a linear solver}} + \underbrace{\left(\log\det \overline{L}_X^{(j)} - \log\det L_X \right)}_{(*)}.$$

   - $(*)$ is also computable by a linear solver under Schur complement.

The overall complexity becomes $O(d^3)$ because we choose $p = O(1)$ and

- In each greedy step, a linear solver can be used to compute both Taylor approximation and $(*)$, thus $O(p \times d^2)$ operations are required.
- The total number of greedy steps is at most d.

Second Ideas: Batch Strategy

We consider adding k-batch subset (instead of single element)

$$X \leftarrow X \cup I \quad \text{for some } |I| = k > 1$$

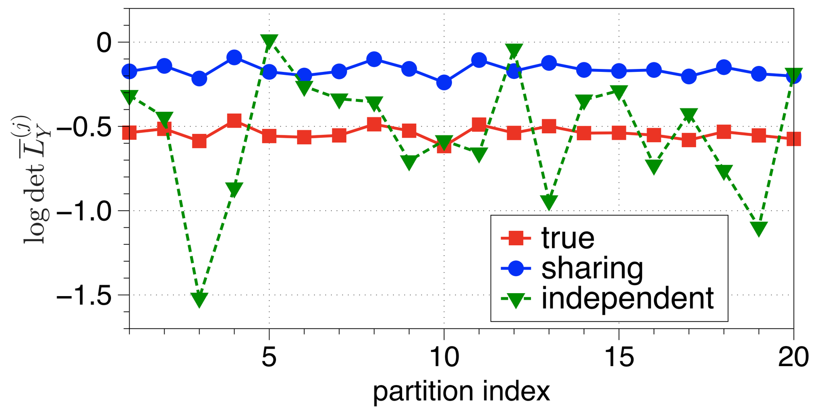
so that the number of greedy steps can be reduced at most k times.

1. **Sampling random batches**
   - For the optimal k-batch, one has to investigate $\approx \binom{d}{k}$ subsets.
   - This is expensive. Instead, we randomly sample batches and add the best of them to the current set.

2. **Log-determinant approximation under sharing randomness**
   - For k-batch strategy, one can compute the extra term $(*)$, i.e., $\log\det \overline{L}_X^{(j)} - \log\det L_X$, by running a linear solver k times.
   - Alternatively, we suggest estimating all log-determinants $\log\det \overline{L}_X^{(j)}$ by running a log-determinant approximation scheme (LDAS) [Han et al., 2015], but only once.

method	complexity	number of calls	objective
linear solver	$O(d^2)$	k	$\log\det \overline{L}_X^{(j)} - \log\det L_X$
LDAS	$O(d^2)$	1	$\log\det \overline{L}_X^{(j)}$

   - LDAS approximates $\log\det \overline{L}_X^{(j)}$ using independent random vectors. We suggest to run LDAS using the same random vectors for estimating all $\log\det \overline{L}_X^{(j)}$.



Observe that running LDAS's under sharing random vectors is better for comparing $\log\det \overline{L}_X^{(j)}$, i.e., marginal gains.

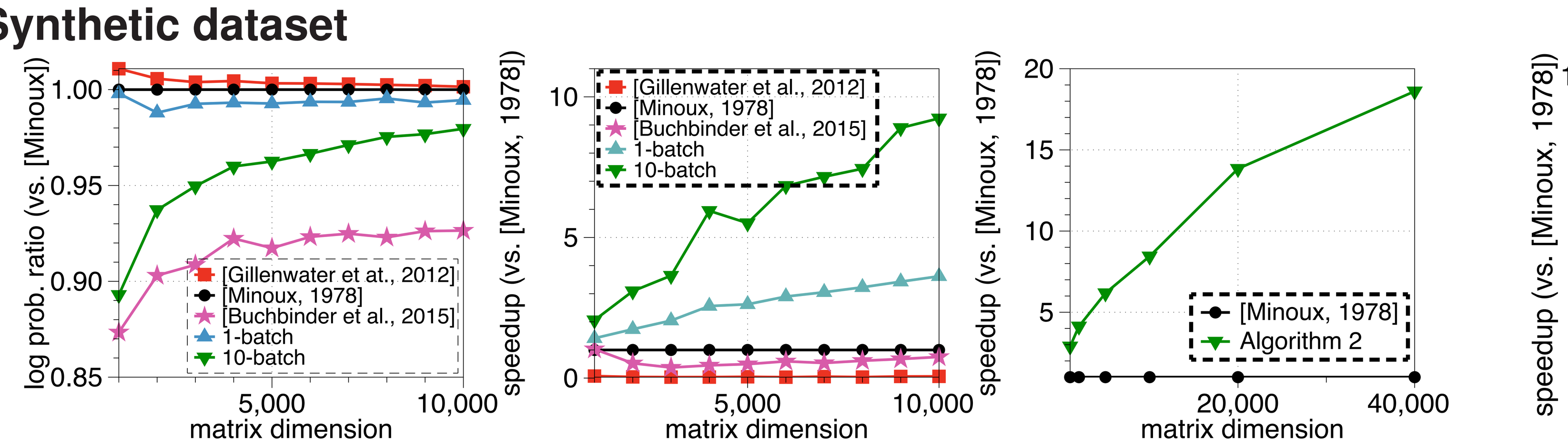
   - We provide the following error bound of LDAS under sharing random vectors, where $A = \overline{L}_X^{(j)}$ and $B = \overline{L}_X^{(j')}$.

> **Theorem** (Han, Prabhanjan, Park and Shin, 2017). *Suppose A, B are positive definite matrices whose eigenvalues are in $[\delta, 1-\delta]$ for $\delta > 0$. Let Γ_A, Γ_B be the estimations of $\log\det A$, $\log\det B$ by LDAS using the same m random vectors for both. Then, it holds that*
>
> $$\mathbf{Var}\left[\Gamma_A - \Gamma_B\right] \le \frac{32 M^2 \rho^2 (\rho+1)^2}{m (\rho-1)^6 (1-2\delta)^2} \|A - B\|_F^2$$
>
> *where $M = 5\log(2/\delta)$ and $\rho = 1 + \frac{2}{\sqrt{2/\delta - 1} - 1}$.*

On the other hand, the variance of LDAS under independent random vectors depends on $\|A\|_F^2 + \|B\|_F^2$ which is significantly larger than $\|A - B\|_F^2$ in our case.

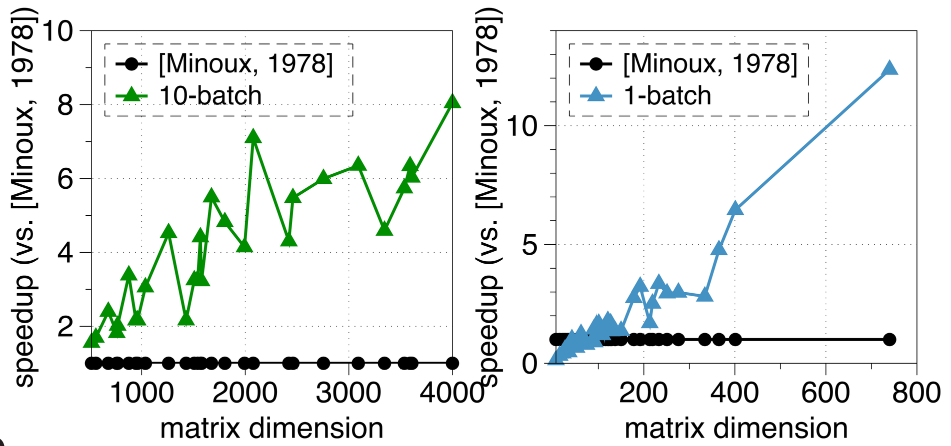
Experiments

Synthetic dataset



- Accuracy is measured by log-probability ratio of a respective algorithm to the standard (but accelerated) greedy algorithm [Minoux, 1978].
- Two versions of our algorithms: 1-batch and 10-batch with 50 batch samples.
- 1-batch achieves 0.03% and 10-batch achieves 0.3% loss on accuracy.
- 10-batch runs up-to 18 times faster than [Minoux, 1978].

Real dataset: Text and video summarizations



- The number of selected items in video summarization is small. In this case, 1-batch shows better performance than 10-batch.
- For both text and video summarization task, our algorithms run $8 \sim 10$ times faster than [Minoux, 1978] for large instances.
- Our video summaries often have higher F-score than [Minoux, 1978].

References

[Avron and Toledo, 2011] Avron, H. and Toledo, S. (2011). Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)*, 58(2):8.

[Bird, 2006] Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

[Boutsidis et al., 2015] Boutsidis, C., Drineas, P., Kambadur, P., and Zouzias, A. (2015). A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *arXiv preprint arXiv:1503.00374*.

[Buchbinder et al., 2015] Buchbinder, N., Feldman, M., Seffi, J., and Schwartz, R. (2015). A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing*, 44(5):1384–1402.

[Daley and Vere-Jones, 2007] Daley, D. J. and Vere-Jones, D. (2007). *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media.

[De Avila et al., 2011] De Avila, S. E. F., Lopes, A. P. B., da Luz, A., and de Albuquerque Araújo, A. (2011). Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, 32(1):56–68.

[Feige et al., 2011] Feige, U., Mirrokni, V. S., and Vondrak, J. (2011). Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, 40(4):1133–1153.

[Fletcher and Reeves, 1964] Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradients. *The computer journal*, 7(2):149–154.

[Gillenwater et al., 2012] Gillenwater, J., Kulesza, A., and Taskar, B. (2012). Near-optimal map inference for determinantal point processes. In *Advances in Neural Information Processing Systems*, pages 2735–2743.

[Gong et al., 2014] Gong, B., Chao, W.-L., Grauman, K., and Sha, F. (2014). Diverse sequential subset selection for supervised video summarization. In *Advances in Neural Information Processing Systems*, pages 2069–2077.

[Greenbaum, 1997] Greenbaum, A. (1997). *Iterative methods for solving linear systems*. SIAM.

[Han et al., 2015] Han, I., Malioutov, D., and Shin, J. (2015). Large-scale log-determinant computation through stochastic chebyshev expansions. In *ICML*, pages 908–917.

[Hausmann et al., 1980] Hausmann, D., Korte, B., and Jenkyns, T. (1980). Worst case analysis of greedy type algorithms for independence systems. In *Combinatorial Optimization*, pages 120–131. Springer.

[Hutchinson, 1990] Hutchinson, M. F. (1990). A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 19(2):433–450.

[Johansson, 2006] Johansson, K. (2006). Course 1 random matrices and determinantal processes. *Les Houches*, 83:1–56.

[Jordan, 1998] Jordan, M. I. (1998). *Learning in graphical models*, volume 89. Springer Science & Business Media.

[Kang, 2013] Kang, B. (2013). Fast determinantal point process sampling with application to clustering. In *Advances in Neural Information Processing Systems*, pages 2319–2327.

[Kathuria and Deshpande, 2016] Kathuria, T. and Deshpande, A. (2016). On sampling and greedy map inference of constrained determinantal point processes. *arXiv preprint arXiv:1607.01551*.

[Krause et al., 2008] Krause, A., Singh, A., and Guestrin, C. (2008). Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(Feb):235–284.

[Kulesza and Taskar, 2011] Kulesza, A. and Taskar, B. (2011). Learning determinantal point processes. In *In Proceedings of UAI*. Citeseer.

[Kulesza et al., 2012] Kulesza, A., Taskar, B., et al. (2012). Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286.

[Kumar et al., 2015] Kumar, R., Moseley, B., Vassilvitskii, S., and Vattani, A. (2015). Fast greedy algorithms in mapreduce and streaming. *ACM Transactions on Parallel Computing*, 2(3):14.

[Lee et al., 2016] Lee, D., Cha, G., Yang, M.-H., and Oh, S. (2016). Individualness and determinantal point processes for pedestrian detection. In *European Conference on Computer Vision*, pages 330–346. Springer.

[Li et al., 2016a] Li, C., Jegelka, S., and Sra, S. (2016a). Efficient sampling for k-determinantal point processes. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1328–1337.

[Li et al., 2016b] Li, C., Sra, S., and Jegelka, S. (2016b). Gaussian quadrature for matrix inverse forms with applications. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1766–1775.

[Liu et al., 2016] Liu, Y., Zhang, Z., Chong, E. K., and Pezeshki, A. (2016). Performance bounds for the k-batch greedy strategy in optimization problems with curvature. In *American Control Conference (ACC), 2016*, pages 7177–7182. IEEE.

[Macchi, 1975] Macchi, O. (1975). The coincidence approach to stochastic point processes. *Advances in Applied Probability*, 7(01):83–122.

[Mason and Handscomb, 2002] Mason, J. C. and Handscomb, D. C. (2002). *Chebyshev polynomials*. CRC Press.

[Minoux, 1978] Minoux, M. (1978). Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques*, pages 234–243. Springer.

[Mirzasoleiman et al., 2015] Mirzasoleiman, B., Badanidiyuru, A., Karbasi, A., Vondrák, J., and Krause, A. (2015). Lazier than lazy greedy. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[Nemhauser et al., 1978] Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294.

[Ouellette, 1981] Ouellette, D. V. (1981). Schur complements and statistics. *Linear Algebra and its Applications*, 36:187–295.

[Pan et al., 2014] Pan, X., Jegelka, S., Gonzalez, J. E., Bradley, J. K., and Jordan, M. I. (2014). Parallel double greedy submodular maximization. In *Advances in Neural Information Processing Systems*, pages 118–126.

[Peng and Wang, 2015] Peng, W. and Wang, H. (2015). Large-scale log-determinant computation via weighted l_2 polynomial approximation with prior distribution of eigenvalues. In *International Conference on High Performance Computing and Applications*, pages 120–125. Springer.

[Saad, 2003] Saad, Y. (2003). *Iterative methods for sparse linear systems*. SIAM.

[Sharma et al., 2015] Sharma, D., Kapoor, A., and Deshpande, A. (2015). On greedy maximization of entropy. In *ICML*, pages 1330–1338.

[Streeter and Golovin, 2009] Streeter, M. and Golovin, D. (2009). An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems*, pages 1577–1584.

[Ubaru et al., 2016] Ubaru, S., Chen, J., and Saad, Y. (2016). Fast estimation of tr(f(A)) via stochastic lanczos quadrature.

[Yao et al., 2016] Yao, J.-g., Fan, F., Zhao, W. X., Wan, X., Chang, E., and Xiao, J. (2016). Tweet timeline generation with determinantal point processes. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 3080–3086. AAAI Press.

[Zhang and Ou, 2016] Zhang, M. J. and Ou, Z. (2016). Block-wise map inference for determinantal point processes with application to change-point detection. In *Statistical Signal Processing Workshop (SSP), 2016 IEEE*, pages 1–5. IEEE.