Composable Core-sets for Determinant Maximization: A Simple Near-Optimal Algorithm

Piotr Indyk [*1] **Sepideh Mahabadi** [*2] **Shayan Oveis Gharan** [*3] **Alireza Rezaei** [*3]

A. Missing Proofs

Proof of Claim 5.5 of the paper. We prove the claim by induction on t, and show that for any j s.t. $j > t$, the point $\Pi(\mathcal{G}'_t)(q'_j)$ can be written as the sum $\sum_{i \le t} \alpha_i q'_i$ such that $|\alpha_i| \le 3^t$.

Base Case. First, we prove the base case of induction, i.e., $t = 1$. Recall that by our assumption, $||q_1|| > 3^k x$, and thus by triangle inequality, we have that $||q'_1|| \ge ||q_1|| - x/k \ge 3^k x - x/k \ge 2x$. Therefore, since q_1 is the vector with largest norm in P, using triangle inequality again, we have that for any $j > 1$,

$$||q'_j|| \le ||q_j|| \le ||q_1|| \le ||q'_1|| + x/k \le (1 + \frac{1}{2k})||q'_1||$$

Therefore we can write $\Pi(\mathcal{G}'_1)(q'_j) = \alpha_1 q'_1$ where $|\alpha_1| \le 2$.

Inductive step. Now, lets assume that the hypothesis holds for $\mathcal{G}'_t$. In particular this means that we can write $\Pi(\mathcal{G}'_t)(q'_{t+1}) = \sum_{i \le t} \beta_i q'_i$ where $|\beta_i| \le 3^t$, and that for a given $j > t+1$, we can write $\Pi(\mathcal{G}'_t)(q'_j) = \sum_{i \le t} \gamma_i q'_i$ where $|\gamma_i|$'s are at most 3^t. Now let $\ell = \mathrm{dist}(q'_{t+1}, \mathcal{G}'_t)$. By triangle inequality, we get that

$$\begin{aligned}
\mathrm{dist}(q_{t+1}, \mathcal{G}_t) &\le \mathrm{dist}(q_{t+1}, q'_{t+1}) && (1)\\
&+ \mathrm{dist}(q'_{t+1}, \Pi(\mathcal{G}'_t)(q'_{t+1}) + && (2)\\
&\mathrm{dist}(\Pi(\mathcal{G}'_t)(q'_{t+1}), \mathcal{G}_t) \\
&\le x/k + \ell + \mathrm{dist}(\sum_{i \le t} \beta_i q'_i, \sum_{i \le t} \beta_i q_i) \\
&\le x/k + \ell + \sum_{i \le t} |\beta_i|\, x/k \\
&\le \ell + 3^t x. && (3)
\end{aligned}$$

Now we consider two case. If $\ell \le 3^t x$ then using the above

$$\mathrm{dist}(q_{t+1}, \mathcal{G}_t) \le 2 \cdot 3^t x \le 3^k x,$$

which contradicts our assumption of $\mathrm{dist}(q_{t+1}, \mathcal{G}_t) > 3^k x$. Otherwise,

$$\begin{aligned}
\mathrm{dist}(\Pi(\mathcal{G}'_{t+1})(q'_j), \mathcal{G}'_t) &\le \mathrm{dist}(q'_j, \mathcal{G}'_t) \le \mathrm{dist}(q_j, \mathcal{G}_t) \\
&\le \mathrm{dist}(q_{t+1}, \mathcal{G}_t) \le 2\ell,
\end{aligned}$$

where the last inequality follows from Equation 1 . Therefore, we can write $\Pi(\mathcal{G}'_{t+1})(q'_j) = \alpha_{t+1} q'_{t+1} - \alpha_{t+1} \Pi(\mathcal{G}'_t)(q'_{t+1}) + \Pi(\mathcal{G}_t)(q'_j)$ where $\alpha_{t+1} \le 2$.

By the hypothesis, we can write $\Pi(\mathcal{G}'_t)(q'_j) = \sum_{i \le t} \gamma_i q'_i$, where $|\gamma_i| \le 3^t$. Since $|\alpha_{t+1}| \le 2$, we can write

$$\begin{aligned}
\Pi(\mathcal{G}'_{t+1})(q'_j) &= \alpha_{t+1} q'_{t+1} + \sum_{i \le t} (\gamma_i - \alpha_{t+1}\beta_i) q'_i \\
&= \sum_{i \le t+1} \alpha_i q'_i \quad \text{where } |\alpha_i| \le 3^{t+1}.
\end{aligned}$$

This completes the proof of the claim.

□

B. Details of Experiments on Local Search vs. Greedy as offline algorithms

Here, we compare the quality of Greedy and Local Search as centralized algorithms on the whole data sets. Figure 1 shows the improvement ratio of the determinant of the solution returned by the Local Search algorithm over the determinant of the solution returned by the Greedy algorithm.

[*] Equal contribution [1] Department of Electrical Engineering and Computer Science, Massachusetts Insitiute of Technology, Cambridge, Massachusetts, USA [2] Toyota Technological Institute at Chicago, Chicago, Illinois, USA [3] Department of Computer Science , University of Washington, Seattle, USA. Correspondence to: Piotr Indyk <indyk@mit.edu>, Sepideh Mahabadi <mahabadi@ttic.edu>, Shayan Oveis Gharan <shayan@cs.washington.edu>, Alireza Rezaei <arezaei@cs.washington.edu>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

On average over all values of k, Local Search improves over Greedy by 13% for GENES data set and 5% for MNIST data set. Figure 2 shows the ratio of the time it takes to run the Local Search and Greedy algorithms as a function of k for both data sets. On average, it takes about 6.5 times more to run the Local Search algorithm.

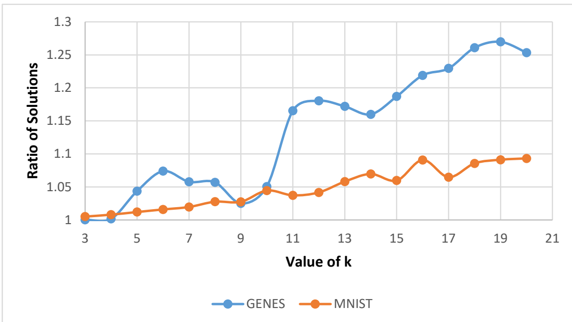

Figure 1. Average improvement of Local Search over Greedy as a function of k.

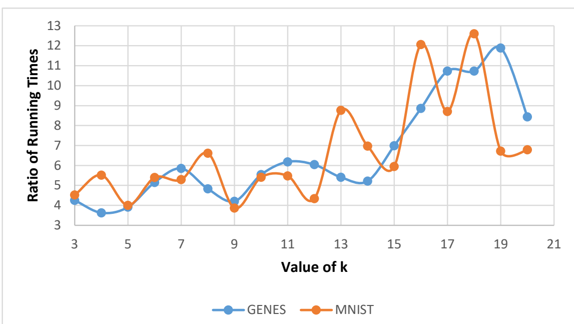

Figure 2. Average ratio of the run time of Local Search over Greedy as a function of k.

C. Details of Experiments for Comparing Local Search vs. the LP-based Algorithm

In this section, we compare the performance of the Local Search algorithm and the LP-based algorithm of (Indyk et al., 2018) for constructing core-sets, i.e., we compare GD/LS with GD/LP. Figure 3 shows how much Local Search improves over the LP-based algorithm. On average this improvement is 7.3%, 1.8% and 1.4% for GENES, MNIST10 and MNIST50 respectively. Moreover, in 78% of all runs, Local Search performed better than Lp-based algorithm, and this improvement can go upto 63%. Figure 4 shows the average ratio of the time to construct core-sets using the LP-based algorithm vs. Local Search.

As it is clear from the graphs, our proposed Local Search algorithm performs better than even the LP-based algorithm which has almost tight approximation guarantees: while picking fewer points in the core-set, in most cases it finds a better solution and runs faster.

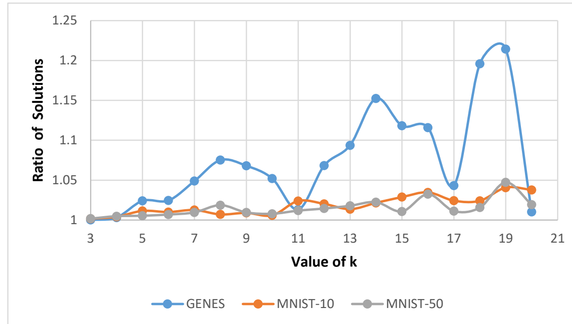

Figure 3. Average improvement of Local Search over LP-based algorithm for constructing core-sets as a function of k.

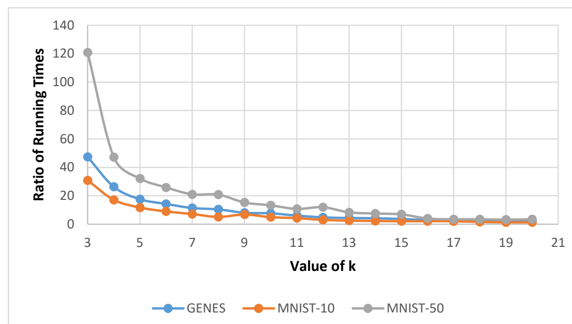

Figure 4. Average ratio of the run time of the optimal algorithm over local search as a function of k.

References

Indyk, P., Mahabadi, S., Gharan, S. O., and Rezaei, A. Composable core-sets for determinant maximization problems via spectral spanners. *arXiv preprint arXiv:1807.11648*, 2018.