Parameterized Binary Matrix Approximation

Petr A. Golovach
(based on the joint work with Fedor V. Fomin and Fahad Panolan)

Department of Informatics, University of Bergen

Algorithmic Tractability via Sparsifiers
Leh, India, 11.08.2019

Matrix approximation

Problem (Matrix Approximation)

Input: *A (binary)* $m \times n$*-matrix* $\mathbf{A} = (a_{ij}) \in \{0,1\}^{m\times n}$.

Task: *Find a (binary)* $m \times n$ *matrix* $\mathbf{B}$ *that satisfies certain conditions and approximates* $\mathbf{A}$.

It is standard to use the *Frobenius* norm (or its square):

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2}.$$

Task: For a given integer k, find $\mathbf{B}$ such that $\|\mathbf{A}-\mathbf{B}\|_F^2 \le k$.

Equivalently, for the binary case, find $\mathbf{B}$ such that $d_H(\mathbf{A},\mathbf{B}) \le k$.

Binary r-Means

Condition: $\mathbf{B}$ has at most r distinct columns.

Problem (Binary r-Means)

Input: *A binary $m \times n$-matrix $\mathbf{A}$ with columns $(\mathbf{a}^1, \ldots, \mathbf{a}^n)$, $r \in \mathbb{N}$ and a nonnegative integer k.*

Task: *Find a partition (some sets may be empty) $\{I_1, \ldots, I_r\}$ of $\{1, \ldots, n\}$ and vectors $\mathbf{c}^1, \ldots, \mathbf{c}^r \in \{0,1\}^m$ (called means) such that*

$$\sum_{i=1}^{r} \sum_{j \in I_i} d_H(\mathbf{c}^i, \mathbf{a}^j) \leq k.$$

Binary r-Means

Equivalently,

Task: Find vectors $\mathbf{c}^1, \ldots, \mathbf{c}^r \in \{0,1\}^m$ (called means) such that

$$\sum_{i=1}^{n} \min_{1 \le j \le r} d_H(\mathbf{c}^j, \mathbf{a}^i) \le k.$$

Equivalently,

Task: Construct a binary matrix $\mathbf{B}$ with at most r pairwise distinct columns from $\mathbf{A}$ by at most k editing operations.

Binary r-Means

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Binary r-Means

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Binary r-Means

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Binary r-Means

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{c}^1 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{c}^2 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{c}^3 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Binary r-Means

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Low GF(2)-Rank Approximation

Condition: The GF(2)-rank of $\mathbf{B}$ is at most r.

Problem (Low GF(2)-Rank Approximation)

Input: *A binary $m \times n$-matrix $\mathbf{A}$, $r \in \mathbb{N}$ and a nonnegative integer k.*

Task: *Find a binary $m \times n$-matrix $\mathbf{B}$ with GF(2)-rank $\leq r$ and $d_H(\mathbf{A}, \mathbf{B}) \leq k$.*

Low GF(2)-Rank Approximation

Equivalently,

Task: Find vectors $\mathbf{c}^1, \ldots, \mathbf{c}^r \in \{0,1\}^m$ such that

$$\sum_{i=1}^{n} \min\{d_H(\mathbf{c}, \mathbf{a}^i) \mid \mathbf{c} \in \mathrm{span}(\mathbf{c}^1, \ldots, \mathbf{c}^r)\} \leq k.$$

Observation: $|\mathrm{span}(\mathbf{c}^1, \ldots, \mathbf{c}^r)| \leq 2^r$.
Equivalently,

Task: Construct a binary matrix $\mathbf{B}$ of rank at most r from $\mathbf{A}$ by at most k editing operations.

Classical complexity

Theorem (Feige, 2014)

BINARY 2-MEANS *is* NP*-complete.*

Theorem (Dan et al., 2015, Gillis and Vavasis, 2015)

Low GF(2)-Rank Approximation *is* NP*-complete for* $r = 1$.

Parameterized Complexity

Parameterized Complexity is a two dimensional framework for studying the computational complexity of a problem.

One dimension is the *input size* $|I|$ and the other is a *parameter* k associated with the input.

A parameterized problem is said to be *fixed parameter tractable* (FPT) if it can be solved in time $f(k) \cdot |I|^{O(1)}$ for some function f.

Parameterized Complexity

Observation

Low GF(2)-Rank Approximation *can be solved in time* $(r+1)^{2k} \cdot (nm)^{O(1)}$.

Parameterized Complexity

Set $\mathbf{B} := \mathbf{A}$.

If $\text{rank}(\mathbf{B}) \leq r$, then return $\mathbf{B}$.

Otherwise, find an $(r+1) \times (r+1)$-submatrix of $\mathbf{B}$ of rank $r+1$ and branch on its elements:

$$\mathbf{B} = \left(\begin{array}{ccc|ccc} b_{11} & \cdots & b_{1r+1} & b_{1r+2} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{r+11} & \cdots & b_{r+1r+1} & b_{r+1r+2} & \cdots & b_{r+1n} \\ \hline b_{r+21} & \cdots & b_{r+2r+1} & b_{1r+2} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mr+1} & b_{mr+2} & \cdots & b_{mn} \end{array}\right)$$

The depth of the search tree is at most k.

Kernels

A *kernelization* algorithm for a parameterized problem Π is a polynomial algorithm that maps each instance (I, k) of Π to an instance (I', k') of Π such that

(i) (I, k) is a yes-instance of Π if and only if (I', k') is a yes-instance of Π, and

(ii) $|I'| + k'$ is bounded by $f(k)$ for a computable function f.

(I', k') is a *kernel* and f is its *size*.

A kernel is *polynomial* if f is polynomial.

Kernelization for Binary r-Means

Theorem

BINARY r-MEANS parameterized by r and k has a kernel of size $O(k^2(k+r)^2)$. Moreover, the kernelization algorithm outputs an instance of BINARY r-MEANS with the matrix that has at most $k+r$ pairwise distinct columns and $O(k(k+r))$ pairwise distinct rows.

Sketch of the proof

Let $(\mathbf{A}, r, k)$ be an instance of BINARY r-MEANS.
$\mathbf{A} = (\mathbf{a}^1, \ldots, \mathbf{a}^n)$

Claim: If $(\mathbf{A}, r, k)$ is a yes-instance, then it has a solution such that the same columns of $\mathbf{A}$ are in the same cluster.

Let $\mathbf{c}^1, \ldots, \mathbf{c}^r$ be means. Then the columns of $\mathbf{A}$ are clustered by selecting a closest mean.

Sketch of the proof

Reduction rules:

- If $\mathbf{A}$ has at most r pairwise distinct columns, then return a trivial yes-instance and stop.
- If $\mathbf{A}$ has at least $r+k+1$ pairwise distinct columns, then return a trivial no-instance and stop.
- If $\mathbf{A}$ has at least $k+2$ columns that are the same, then delete one of them.

We obtain $\mathbf{A}=(\mathbf{a}^1,\ldots,\mathbf{a}^n)$ such that $\mathbf{A}$ contains at most $(k+1)(r+k)$ columns and at most $k+r$ of them are pairwise distinct.

Sketch of the proof

We construct the partition $\mathcal{S} = \{S_1, \ldots, S_t\}$ of $\{1, \ldots, n\}$:

Let $I = \{1, \ldots, n\} \setminus \bigcup_{j=0}^{i-1} S_j$ (we assume that $S_0 = \emptyset$).

- Set $S_i = \{s\}$ for arbitrary $s \in I$ and set $I = I \setminus \{s\}$.
- While there is $j \in I$ such that $d_H(\mathbf{a}^j, \mathbf{a}^h) \leq k$ for some $h \in S_i$, then set $S_i = S_i \cup \{j\}$ and set $I = I \setminus \{j\}$.

$$\mathbf{A} = (\underbrace{\ldots}_{S_1} | \underbrace{\ldots}_{S_2} | \ldots | \underbrace{\ldots}_{S_t})$$

Claim: for every cluster in a solution, the indices of its columns are in the same S_i.

Reduction rule: If $t \geq r + 1$, then return a trivial no-instance and stop.

Sketch of the proof

Let $\mathbf{A}_i = \mathbf{A}[\{1,\ldots,m\}, S_i]$ for $i \in \{1,\ldots,t\}$.

A row of a matrix is *uniform* if all its elements are the same.

Observation: uniform rows are irrelevant for BINARY r-MEANS.

Claim: $\mathbf{A}_i$ has at most $(\ell_i - 1)k \leq (r+k)k$ non-uniform rows, where ℓ_i is the number of pairwise distinct columns of $\mathbf{A}_i$.

For $i \in \{1,\ldots,t\}$, construct $\mathbf{A}_i'$ from $\mathbf{A}_i$ by the deletion of $m-(r+1)k$ uniform rows.

Sketch of the proof

Observation: $A'_1, \ldots, A'_t$ give a Turing kernel.

Return

$$\mathbf{A}' = \left(\begin{array}{c|c|c|c} \mathbf{A}'_1 & \mathbf{A}'_2 & \cdots & \mathbf{A}'_t \\ \hline \mathbb{1}_1 & \mathbb{0}_2 & \cdots & \mathbb{0}_t \\ \hline \mathbb{0}_1 & \mathbb{1}_2 & \cdots & \mathbb{0}_t \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline \mathbb{0}_1 & \mathbb{0}_2 & \cdots & \mathbb{1}_t \end{array} \right)$$

where $\mathbb{0}_i$ and $\mathbb{1}_i$ are $\lceil \frac{k+1}{2} \rceil \times |S_i|$-matrices composed by 0-s and 1-s resp.

Kernelization for Low GF(2)-Rank Approximation

Theorem (Fomin et al. 2018)

LOW GF(2)-RANK APPROXIMATION *parameterized by* r *and* k *admits a kernel such that the output matrix has at most* $(r+1)k$ *row and columns.*

Subexponential algorithms

Theorem

Binary r-Means parameterized by r and k has a kernel of size $O(k^2(k+r)^2)$. Moreover, the kernelization algorithm outputs an instance of Binary r-Means with the matrix that has at most $k+r$ pairwise distinct columns and $O(k(k+r))$ pairwise distinct rows.

Corollary

Binary r-Means can be solved in time $r^{r+k} \cdot (nm)^{O(1)}$.

Observation

Low GF(2)-Rank Approximation can be solved in time $(r+1)^{2k} \cdot (nm)^{O(1)}$.

Binary r-Means

Theorem

BINARY r-MEANS *can be solved in time* $2^{O(\sqrt{rk \log(r+k) \log r})} \cdot nm$.

Sketch of the proof

Let (A, r, k) be an instance of BINARY r-MEANS, $\mathbf{A} = (\mathbf{a}^1, \ldots, \mathbf{a}^n)$.

We apply the kernelization algorithm that either solves the problem or outputs an instance (A, r, k) such that

- A has at most $k + r$ pairwise distinct columns and
- A has at most $O(k(k + r))$ pairwise distinct rows.

Task: Find means $\mathbf{c}^1, \ldots, \mathbf{c}^r \in \{0, 1\}^m$ such that

$$\sum_{i=1}^{n} \min_{1 \le j \le r} d_H(\mathbf{c}^j, \mathbf{a}^i) \le k.$$

Sketch of the proof

We assume that

- the means $\mathbf{c}^1, \ldots, \mathbf{c}^s$, $s \leq r$, are already selected,
- some columns of $\mathbf{A}$ are already assigned to the chosen means and $\mathbf{A}^* = (\mathbf{a}^1, \ldots, \mathbf{a}^p)$ is composed by the remaining columns of $\mathbf{A}$.
- the budget k is adjusted respectively.

If $p \leq r - s$ or

$$\sum_{i=1}^{p} \min\{d_H(\mathbf{a}^i, \mathbf{c}^j) \mid 1 \leq j \leq s\} \leq k,$$

then the problem is solved.

If $s = r$ or $k \leq 0$, then $(\mathbf{A}, r, k)$ is a no-instance.

Sketch of the proof

We guess the minimum distance $h \leq k$ between a new mean $\mathbf{c}^{s+1}$ and a column $\mathbf{a}^i$ in the new cluster.

If

$$d = \min\{d_H(\mathbf{a}^i, \mathbf{c}^j) \mid 1 \leq j \leq s\} \leq h - 1,$$

then we include $\mathbf{a}^i$ in one of the old clusters and set $k := k - d$.

Let $\mathbf{A}^{**} = (\mathbf{a}^1, \ldots, \mathbf{a}^q)$ be the matrix composed by the remaining columns, and let $\ell = O(k(k + r))$ be the number of pairwise distinct rows of $\mathbf{A}^{**}$.

Sketch of the proof

If $q \leq \sqrt{rk \log \ell / \log r}$, we try all possible partitions of the set of columns of $\mathbf{A}^{**}$ into at most $r - s$ clusters by brute force (in fact, into $r - s + 1$ clusters).

We have at most $(r - s + 1)^{\sqrt{rk \log \ell / \log r}}$ or $2^{O(\sqrt{rk \log \ell \log r})}$ possibilities.

Since $\ell = O(k(k + r))$, the running time is $2^{O(\sqrt{rk \log(r+k) \log r})}$.

Sketch of the proof

Let $q > \sqrt{rk \log \ell / \log r}$.

Observation: For a yes-instance, $h \leq k/q \leq \sqrt{k \log r/(r \log \ell)}$.

We branch:

- for every $i \in \{1, \ldots, q\}$ and every $\mathbf{c}^{s+1}$ at distance h from $\mathbf{a}^i$, solve the problem for $\mathbf{c}^1, \ldots, \mathbf{c}^s, \mathbf{c}^{s+1}$ and $\mathbf{A}^{**}$.

Because $\mathbf{A}$ has $O(k(k+r))$ pairwise distinct rows, the number of branches for each column is at most $\ell^{\sqrt{k \log r/(r \log \ell)}} = 2^{O(\sqrt{k/r \log r \log(k+r)})}$.

The depth of the search tree is at most r and the number of leaves is $2^{O(\sqrt{rk \log(r+k) \log r})}$.

Low Rank Approximation

Theorem
LOW GF(2)-RANK APPROXIMATION *can be solved in time* $2^{O(r\sqrt{k\log(rk)})}\cdot nm$.

Task: Find vectors $\mathbf{c}^1,\ldots,\mathbf{c}^r\in\{0,1\}^m$ such that

$$\sum_{i=1}^{n}\min\{d_H(\mathbf{c},\mathbf{a}^i)\mid \mathbf{c}\in \mathrm{span}(\mathbf{c}^1,\ldots,\mathbf{c}^r)\}\leq k.$$

Weaker parameterizations

Theorem (Feige, 2014)

BINARY 2-MEANS *is* NP*-complete.*

Theorem (Dan et al., 2015, Gillis and Vavasis, 2015)

Low GF(2) *is* NP*-complete for* $r = 1$*.*

Binary r-Means parameterized by k

Theorem

BINARY r-MEANS *can be solved in time* $2^{O(k \log k)} \cdot (nm)^{O(1)}$ *(r is a part of the input).*

Idea of the proof

Let $(\mathbf{A}, r, k)$ be an instance of BINARY r-MEANS.

Claim: If $(\mathbf{A}, r, k)$ is a yes-instance, then it has a solution such that the same columns of $\mathbf{A}$ are in the same cluster.

We say that an inclusion maximal set of identical columns of $\mathbf{A}$ is an *initial cluster*.

Let $(\mathbf{A}, r, k)$ be a yes-instance with a given solution.

- A cluster of the solution is *simple* if it is an initial cluster, and
- a cluster is *composite* otherwise.

Highlighting elements of composite clusters

Claim: If $(\mathbf{A}, r, k)$ is a yes-instance, then every solution has at most k composite clusters and at most $2k$ initial clusters are in composite clusters.

We apply the **color coding** technique (Alon, Yuster, and Zwick).

We guess the number s of initial clusters that are in composite clusters in a solution and the number t of composite clusters.

We color the initial clusters uniformly at random by s colors.

If $(\mathbf{A}, r, k)$ is a yes-instance, then the probability that the initial clusters in a solution are colored by distinct colors is

$$\frac{s!}{s^s} \leq \frac{(2k)!}{(2k)^{2k}} \sim e^{-2k}.$$

Highlighting elements of composite clusters

$$\mathbf{A} = \left(\begin{array}{cc|ccc||cc|cc||c||cc|cc} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & \cdots & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & \cdots & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \cdots & 0 & 0 & 1 & 1 \end{array}\right)$$

We are looking for a *colorful* solution, where exactly one initial cluster from each color class is included in a composite cluster.

For each composite cluster, we guess the color classes containing its initial clusters.

The number of guesses is at most $t^s \leq k^{2k}$.

Cluster selection

Problem (Cluster Selection)

Input: *A binary $m \times p$-matrix $\mathbf{A} = (\mathbf{a}^1, \ldots, \mathbf{a}^p)$, a partition $\{I_1, \ldots, I_q\}$ of $\{1, \ldots, p\}$ such that the indices of each initial cluster are in the same element of the partition, and a non-negative integer d.*

Task: *Find a vector $\mathbf{c} \in \{0,1\}^m$ and initial clusters $J_1, \ldots J_q$ such that*

- $J_i \subseteq I_i$ *for* $i \in \{1, \ldots, q\}$,
- $\sum_{i=1}^{q} \sum_{j \in J_i} d_H(\mathbf{c}, \mathbf{a}^j) \leq d$.

Cluster selection

Lemma

Cluster Selection can be solved in time $2^{O(d \log d)} \cdot (pm)^{O(1)}$.

Idea of the proof

$$\mathbf{A} = \left(\begin{array}{c|c|c||c|c|c||c|c}
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \hline
\cdots & 1\,1\,1 & \cdots & \cdots & 0\,0 & \cdots & \cdots & 1\,1 \\ \hline
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \hline
\cdots & 0\,0\,0 & \cdots & \cdots & 1\,1 & \cdots & \cdots & 1\,1 \\ \hline
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots
\end{array}\right)$$

Claim: There are $2^{O(d \log d)} \cdot (dm)^{O(1)}$ subsets of $\{1, \dots, m\}$, where the columns of the selected initial cluster can differ from a mean $\mathbf{c}$, and these subsets can be enumerated in time $2^{O(d \log d)} \cdot (dm)^{O(1)}$.

D. Marx, Closest substring problems with small distances, SIAM J. Comput., 38 (2008), pp. 1382–1410.

Lower bounds

Theorem

BINARY r-MEANS *has no polynomial kernel when parameterized by* k *unless* $\mathrm{NP} \subseteq \mathrm{coNP}/\mathrm{poly}$.

Theorem (Fomin et al., 2017)

LOW GF(2)-RANK APPROXIMATION *is* W[1]*-hard when parameterized by* k.

Our results

- BINARY r-MEANS can be solved in time $2^{O(k \log k)} \cdot (nm)^{O(1)}$.
- BINARY r-MEANS has a kernel of size $O(k^2(k+r)^2)$ when parameterized by k and r.
- BINARY r-MEANS has no polynomial kernel when parameterized by k only unless $\mathrm{NP} \subseteq \mathrm{coNP}/\mathrm{poly}$.

We obtain FPT algorithms for the problems parameterized by r and k that are *subexponential* in k.

- BINARY r-MEANS can be solved in time $2^{O(\sqrt{rk \log r \log (k+r)})} nm$.
- LOW GF(2)-RANK APPROXIMATION can be solved in time $2^{O(r \cdot \sqrt{k \log(rk)})} nm$.

Open problems

- Can BINARY r-MEANS be solved in time $2^{O(k)} \cdot (nm)^{O(1)}$?
- Can BINARY r-MEANS and/or LOW GF(2)-RANK APPROXIMATION be solved in time $2^{f(r)\sqrt{k}} \cdot (nm)^{O(1)}$?
- What can be said about parameterized complexity of MATRIX APPROXIMATION for matrices over different fields and for different measures?
- In particular, what can be said about r-MEANS for matrices over $\mathbb{Z}$ and Frobenius norm?

Thank You!