Publication 4

Kai Puolamäki, Sami Hanhijärvi, and Gemma C. Garriga. 2008. An approximation ratio for biclustering. Information Processing Letters, volume 108, number 2, pages 45-49.

Information Processing Letters 108 (2008) 45–49

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

An approximation ratio for biclustering

Kai Puolamäki, Sami Hanhijärvi *, Gemma C. Garriga

Helsinki Institute for Information Technology, Helsinki University of Technology, P.O. Box 5400, FI-02015 TKK, Finland

ARTICLE INFO

Article history:
Received 13 April 2007
Received in revised form 10 February 2008
Available online 29 March 2008
Communicated by K. Iwama

Keywords:
Approximation algorithms
Biclustering
One-way clustering

ABSTRACT

The problem of biclustering consists of the simultaneous clustering of rows and columns of a matrix such that each of the submatrices induced by a pair of row and column clusters is as uniform as possible. In this paper we approximate the optimal biclustering by applying one-way clustering algorithms independently on the rows and on the columns of the input matrix. We show that such a solution yields a worst-case approximation ratio of $1+\sqrt{2}$ under L_1-norm for 0–1 valued matrices, and of 2 under L_2-norm for real valued matrices.

1. Introduction

The standard clustering problem [8] consists of partitioning a set of input vectors, such that the vectors in each partition (cluster) are close to one another according to some predefined distance function. This formulation is the objective of the popular K-means algorithm (see, for example, [9]), where K denotes the final number of clusters and the distance function is defined by the L_2-norm. Another similar example of this formulation is the K-median algorithm (see, for example, [3]), where the distance function is given by the L_1-norm. Clustering a set of input vectors is a well-known NP-hard problem even for $K=2$ clusters [4]. Several approximation guarantees have been shown for this formulation of the standard clustering problem (see [3,9,2] and references therein).

Intensive recent research has focused on the discovery of homogeneous substructures in large matrices. This is also one of the goals in the problem of *biclustering*. Given a set of N rows in M columns from a matrix X, a biclustering algorithm identifies subsets of rows exhibiting similar behavior across a subset of columns, or vice versa. Note that the optimal solution for this problem necessarily requires to cluster the N vectors and the M dimensions simultaneously, thus the name biclustering. Each submatrix of X, induced by a pair of row and column clusters, is typically referred to as a *bicluster*. See Fig. 1 for a simple toy example. The main challenge of a biclustering algorithm lies in the dependency between the row and column partitions, which makes it difficult to identify the optimal biclusters. A change in a row clustering affects the cost of the induced submatrices (biclusters), and as a consequence, the column clustering may also need to be changed to improve the solution.

Finding an optimal solution for the biclustering problem is NP-hard. This observation follows directly from the reduction of the standard clustering problem (known to be NP-hard) to the biclustering problem by fixing the number of clusters in columns to M. To the best of our knowledge, no algorithm exists that can efficiently approximate biclustering with a proven approximation ratio. The goal of this paper is to propose such an approximation guarantee by means of a very simple scheme.

Our approach will consist of relieving the requirement for simultaneous clustering of rows and columns and instead perform them independently. In other words, our final biclusters will correspond to the submatrices of X induced by pairs of row and columns clusters, found independently with a standard clustering algorithm. We sometimes refer to this standard clustering algorithm as one-way clustering. The simplicity of the solution alleviates us from the inconvenient dependency of rows and columns.

* Corresponding author.
E-mail addresses: Kai.Puolamaki@tkk.fi (K. Puolamäki), Sami.Hanhijarvi@tkk.fi (S. Hanhijärvi), Gemma.Garriga@tkk.fi (G.C. Garriga).

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2008.03.013

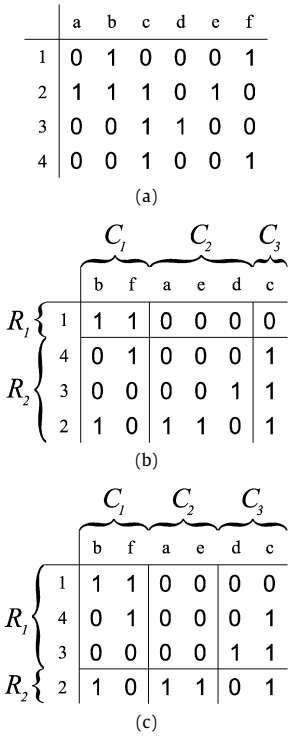

Fig. 1. (a) An example binary data matrix X of dimensions 4×6, with rows and columns labeled with numbers and characters. (b) The optimal biclustering of X consists of $\{R_1^*, R_2^*\} = \{\{1\}, \{2, 3, 4\}\}$ row clusters and $\{C_1^*, C_2^*, C_3^*\} = \{\{b, f\}, \{a, d, e\}, \{c\}\}$ column clusters when using L_1-norm. (c) Biclusters of the data matrix returned by our scheme, that is, using twice an optimal one-way clustering algorithm, once on the 4-row vectors and another on the 6-column vectors, with L_1-norm. Resulting clusterings are $\{R_1, R_2\} = \{\{1, 3, 4\}, \{2\}\}$ for rows and $\{C_1, C_2, C_3\} = \{\{b, f\}, \{a, e\}, \{d, e\}\}$ for columns. For visual clarity, the rows and columns of the original matrix in (a) have been permuted in (b) and (c) by making the rows (and columns) of a single cluster adjacent.

More importantly, the solution obtained with this approach, despite not being optimal, allows for the study of approximation guarantees on the obtained biclusters. Here we prove that our solution achieves a worst-case approximation ratio of $1 + \sqrt{2}$ under L_1-norm for 0–1 valued matrices, and of 2 under L_2-norm for real valued matrices.

Finally, note that our final solution is constructed on top of a standard clustering algorithm (applied twice, once in row vectors and the other in column vectors) and therefore, it is necessary to multiply our ratio with the approximation ratio achieved by the used standard clustering algorithm (such as [3,9]). For clarity, we will lift this restriction in the following proofs by assuming that the applied one-way clustering algorithm provides directly an optimal solution to the standard clustering problem.

1.1. Related work

This basic algorithmic problem and several variations were initially presented in [6] with the name of direct clustering. The same problem and its variations have also been referred to as two-way clustering, co-clustering or subspace clustering. In practice, finding highly homogeneous biclusters has important applications in biological data analysis (see [10] for review and references), where a bicluster may, for example, correspond to an activation pattern common to a group of genes only under specific experimental conditions.

An alternative definition of the basic biclustering problem described in the introduction consists on finding the maximal bicluster in a given matrix. A well-known connection of this alternative formulation is its reduction to the problem of finding a biclique in a bipartite graph [7]. Algorithms for detecting bicliques enumerate them in the graph by using the monotonicity property that a subset of a biclique is also a biclique [1,5]. These algorithms usually have a high order of complexity.

2. Definitions

We assume given a matrix X of size $N \times M$, and integers K_r and K_c, which define the number of clusters partitioning rows and columns, respectively. The goal is to approximate the optimal biclustering of X by means of a one-way row clustering into K_r clusters and a one-way column clustering into K_c clusters.

For any $T \in \mathbb{N}$ we denote $[T] = \{1, \ldots, T\}$. We use $X(R, C)$, where $R \subseteq [N]$ and $C \subseteq [M]$, to denote the submatrix of X induced by the subset of rows R and the subset of columns C. Let Y denote an induced submatrix of X, that is $Y = X(R, C)$ for some $R \subseteq [N]$ and $C \subseteq [M]$. When required by the context, we will also refer to $Y = X(R, C)$ as a bicluster of X and denote the size of Y with $n \times m$, where $n \leqslant N$ and $m \leqslant N$. We use median(Y) and mean(Y) to denote the median and mean of all elements of Y, respectively.

The scheme for approximating the optimal biclustering is defined as follows.

Input: matrix X, number of row clusters K_r, number of column clusters K_c

$$\mathcal{R} = \text{kcluster}(X, K_r),$$

$$\mathcal{C} = \text{kcluster}\big(X^T, K_c\big)$$

Output: a set of biclusters $X(R, C)$, for each $R \in \mathcal{R}$, $C \in \mathcal{C}$

The function kcluster(X, K_r) denotes here an optimal one-way clustering algorithm that partitions the row vectors of matrix X into K_r clusters. We have used X^T to denote the transpose of matrix X.

Instead of fixing a specific norm for the formulas, we use the dissimilarity measure $\mathcal{V}()$ to absorb the norm-dependent part. For L_1-norm, $\mathcal{V}()$ would be defined as $\mathcal{V}(Y) = \sum_{y \in Y} |y\text{-median}(Y)|$, and for L_2-norm as $\mathcal{V}(Y) = \sum_{y \in Y} (y\text{-mean}(Y))^2$. Given Y of size $n \times m$, we further use a special row norm, $\mathcal{V}_R(Y) = \sum_{j=1}^{m} \mathcal{V}(Y([n], j))$, and a special column norm, $\mathcal{V}_C(Y) = \sum_{i=1}^{n} \mathcal{V}(Y(i, [m]))$.

We define the one-way row clustering, given by k-cluster above, as a partition of rows $[N]$ into K_r clusters $\mathcal{R} = \{R_1, \ldots, R_{K_r}\}$ such that the cost function

$$L_R = \sum_{R \in \mathcal{R}} \sum_{j=1}^{M} \mathcal{V}\big(X(R, j)\big) \tag{1}$$

is minimized. Analogously, the one-way clustering of columns $[M]$ into K_c clusters $\mathcal{C} = \{C_1, \ldots, C_{K_c}\}$ is defined such that the cost function

$$L_C = \sum_{i=1}^{N} \sum_{C \in \mathcal{C}} \mathcal{V}\big(X(i, C)\big) \tag{2}$$

is minimized.

The cost of biclustering, induced by the two one-way clusterings above, is

$$L = \sum_{R \in \mathcal{R}} \sum_{C \in \mathcal{C}} \mathcal{V}\big(X(R, C)\big). \tag{3}$$

Notice that we are assuming that the one-way clusterings above, denoted $\mathcal{R}$ on rows and $\mathcal{C}$ on columns, correspond to optimal one-way partitionings on rows and columns, respectively.

Finally, the optimal biclustering on X is given by simultaneous row and column partitions $\mathcal{R}^* = \{R_1^*, \ldots, R_{K_r}^*\}$ and $\mathcal{C}^* = \{C_1^*, \ldots, C_{K_c}^*\}$, that minimize the cost

$$L^* = \sum_{R^* \in \mathcal{R}^*} \sum_{C^* \in \mathcal{C}^*} \mathcal{V}\big(X(R^*, C^*)\big). \tag{4}$$

3. Approximation ratio

Given the definitions above, our main result reads as follows.

Theorem 1. *There exists an approximation ratio of α such that $L \leqslant \alpha L^*$, where $\alpha = 1 + \sqrt{2} \approx 2.41$ for L_1-norm and $X \in \{0,1\}^{N \times M}$, and $\alpha = 2$ for L_2-norm and $X \in \mathbb{R}^{N \times M}$.*

We use the following intermediate result to prove the theorem.

Lemma 2. *There exists an approximation ratio of at most α, that is, $L \leqslant \alpha L^*$, if for any X and for any partitionings $\mathcal{R}$ and $\mathcal{C}$ of X, all biclusters $Y = X(R, C)$, with $R \in \mathcal{R}$ and $C \in \mathcal{C}$, satisfy*

$$\mathcal{V}(Y) \leqslant \frac{1}{2}\alpha\big(\mathcal{V}_R(Y) + \mathcal{V}_C(Y)\big). \tag{5}$$

Proof. First we note that the cost of the optimal biclustering L^* cannot increase when we increase the number of row (or column) clusters. For example, consider the special case where $K_r = N$ (or $K_c = M$). In such case, each row (or column) is assigned to its own cluster and the cost of the optimal biclustering equals the cost of the optimal one-way clustering on columns L_C (or rows L_R). Hence, the optimal biclustering solution is bounded from below by

$$L^* \geqslant \max(L_R, L_C) \geqslant \frac{1}{2}(L_R + L_C). \tag{6}$$

Summing both sides of Eq. (5),

$$\sum_{R \in \mathcal{R}} \sum_{C \in \mathcal{C}} \mathcal{V}(Y)|_{Y = X(R,C)} \leqslant \frac{1}{2}\alpha \sum_{R \in \mathcal{R}} \sum_{C \in \mathcal{C}} \big(\mathcal{V}_R(Y) + \mathcal{V}_C(Y)\big)\big|_{Y = X(R,C)},$$

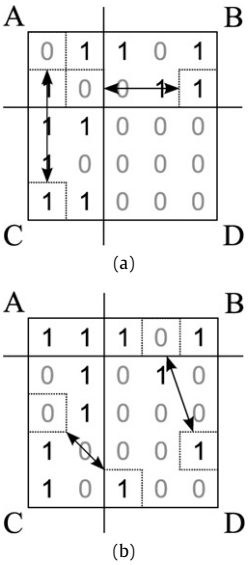

Fig. 2. Examples of swaps performed within bicluster Y for the technical part of the proof in Section 3.1. For clarity, the rows and columns of the bicluster Y have been ordered such that the blocks A, B, C and D are continuous.

and using Eqs. (1), (2) and (3), gives $L \leqslant \frac{1}{2}\alpha(L_R + L_C)$, which together with Eq. (6) implies the approximation ratio of $L \leqslant \alpha L^*$. □

Theorem 1 is proven separately in Sections 3.1 and 3.2 using Lemma 2. Section 3.1 deals with the case of having a 0–1 valued matrix X and L_1-norm distance function, while Section 3.2 deals with real valued matrix X and L_2-norm.

3.1. L_1-norm and 0–1 valued matrix

Consider a 0–1 valued matrix X and L_1-norm. To prove Theorem 1 it suffices to show that Eq. (5) holds for each of the biclusters $Y = X(R, C)$ of X, where $R \in \mathcal{R}$ and $C \in \mathcal{C}$. Therefore, in the following we concentrate on one single bicluster $Y \in \{0,1\}^{n \times m}$.

Without loss of generality, we consider only the case where the bicluster Y has at least as many 0's as 1's. In such case, the median of Y can be safely taken to be zero and the cost $\mathcal{V}(Y) \leqslant \frac{1}{2}nm$ is then fixed to the number of 1's in the matrix. To get the worst case scenario towards the tightest upper bound on α in Eq. (5), we should find first a configuration of 1's such that, given $\mathcal{V}(Y)$, the sum $\mathcal{V}_R(Y) + \mathcal{V}_C(Y)$ is minimized.

Denote by O_R and O_C the sets of rows and columns in Y which have more 1's than 0's, respectively. Denote $A = Y(O_R, O_C)$, $B = Y(O_R, [m] \setminus O_C)$, $C = Y([n] \setminus O_R, O_C)$, $D = Y([n] \setminus O_R, [m] \setminus O_C)$, $n' = |O_R|$ and $m' = |O_C|$. Note that A, B, C and D are simply blocks of bicluster Y, which we need to make explicit in our notation for the proof.

Changing a 0 to 1 in A or a 1 to 0 in D decreases $\mathcal{V}_R(Y) + \mathcal{V}_C(Y)$ by two, while changing a 0 to 1 or 1 to 0 in B or C changes $\mathcal{V}_R(Y) + \mathcal{V}_C(Y)$ by at most one. It follows that *swapping* a 1 in B or C with a 0 in A (see Fig. 2(a)), or swapping a 1 in D with a 0 in A, B or C (see Fig. 2(b)) decreases $\mathcal{V}_R(Y) + \mathcal{V}_C(Y)$ while $\mathcal{V}(Y)$ remains

unchanged. In other words, in a solution that minimizes $\mathcal{V}_R(Y) + \mathcal{V}_C(Y)$ no such swaps can be made. In the remainder of this subsection, we assume that the bicluster Y satisfies this mentioned property.

It follows that

(i) A, B and C are blocks of 1's,
(ii) A is a block of 1's and D is a block of 0's, or
(iii) B, C and D are blocks of 0's.

Denote by $o()$ the number of 1's in a given block. It follows that

$$\mathcal{V}(Y) = o(A) + o(B) + o(C) + o(D) \leqslant \frac{1}{2}nm,$$

$$\mathcal{V}_R(Y) = nm' - o(A) + o(B) - o(C) + o(D) \quad \text{and}$$

$$\mathcal{V}_C(Y) = n'm - o(A) - o(B) + o(C) + o(D).$$

We denote $x = n'/n$, $y = m'/m$, $a = o(A)/(nm)$, $b = o(B)/(nm)$, $c = o(C)/(nm)$ and $d = o(D)/(nm)$ and rewrite Eq. (5) as

$$\alpha = \sup\left(\frac{2\mathcal{V}(Y)}{\mathcal{V}_R(Y) + \mathcal{V}_C(Y)}\right) = 2\sup\left(\frac{a+b+c+d}{x+y-2a+2d}\right),$$

with constraints $a + b + c + d \in [0, \frac{1}{2}]$, $x \in [0, 1]$ $y \in [0, 1]$, as well as

(i) $a = xy$, $b = x(1-y)$, $c = (1-x)y$ and $d \in [0, (1-x)(1-y)]$;
(ii) $a = xy$, $b \in [0, x(1-y)]$, $c \in [0, (1-x)y]$ and $d = 0$; or
(iii) $a \in [0, xy]$ and $b = c = d = 0$.

The optimization problem has two solutions, (i) $x = y = 1 - \sqrt{\frac{1}{2}}$, $a = xy$, $b = x(1-y)$, $c = (1-x)y$ and $d = 0$, and (ii) $x = y = \sqrt{\frac{1}{2}}$, $a = xy$ and $b = c = d = 0$, both solutions yielding $\alpha = 1 + \sqrt{2}$ when exactly half of the entries in the bicluster Y are 1's. This proves Theorem 1 for 0–1 valued matrices and L_1-norm.

Notice that the above proof relies on the fact that the input matrix X has only two types of values. Therefore, the proof does not generalize to real valued matrices.

An example of a matrix with approximation ratio of 2 is given by a $4 \times (4q-1)$ matrix

$$X = \begin{pmatrix} 0\ldots0 & 1\ldots1 & 0\ldots\ldots0 \\ 0\ldots0 & 1\ldots1 & 1\ldots\ldots1 \\ 1\ldots1 & 0\ldots0 & 0\ldots\ldots0 \\ 1\ldots1 & 0\ldots0 & 1\ldots\ldots1 \end{pmatrix}$$

with q columns in the first column group, q columns in the second column group and $2q-1$ columns in the third column group, clustered to two row clusters, $K_r = 2$, and one column cluster, $K_c = 1$, at the limit of large q. The optimal one-way clustering of rows is given by $\mathcal{R} = \{\{1,2\},\{3,4\}\}$, $L = 8q - 2$, and the optimal biclustering of rows by $\mathcal{R}^* = \{\{1,3\},\{2,4\}\}$, $L^* = 4q$.

3.2. L_2-norm and real valued matrix

Consider now a real valued matrix X and L_2-norm. We want to prove Theorem 1 for the real valued biclusters Y of X. To find the approximation ratio, it suffices to show that Eq. (5) holds for each bicluster $Y \in \mathbb{R}^{n\times m}$, which are determined by $Y = X(R, C)$, where $R \in \mathcal{R}$ and $C \in \mathcal{C}$.

Using the definitions of $\mathcal{V}(Y)$, $\mathcal{V}_R(Y)$ and $\mathcal{V}_C(Y)$, we can write

$$\mathcal{V}(Y) = \mathcal{V}_R(Y) + \mathcal{V}_C(Y) - \sum_{i=1}^{n}\sum_{j=1}^{m}\big(Y(i,j) - \overline{Y}(i,j)\big)^2 \leqslant \mathcal{V}_R(Y) + \mathcal{V}_C(Y),$$

where

$$\overline{Y}(i,j) = \text{mean}\big(Y([n],j)\big) + \text{mean}\big(Y(i,[m])\big) - \text{mean}(Y).$$

Hence, Eq. (5) is satisfied for L_2-norm and real valued matrices when $\alpha = 2$.

4. Conclusions

We have shown that approximating the optimal biclustering with independent row- and column-wise standard clusterings achieves a good approximation guarantee. However in practice, standard one-way clustering algorithms (such as K-means or K-median) are also approximate, and therefore, it is necessary to multiply our ratio with the approximation ratio achieved by the standard clustering algorithm (such as presented in [3,9]) to obtain the true approximation ratio of our scheme. Still, our contribution shows that in many practical applications of biclustering, it may be sufficient to use a more straightforward standard clustering of rows and columns instead of applying heuristic algorithms without performance guarantees.

Acknowledgements

We thank Nikolaj Tatti for reading through the manuscript and giving useful comments.

References

[1] G. Alexe, S. Alexe, Y. Crama, S. Foldes, P.L. Hammer, B. Simeone, Consensus algorithms for the generation of all maximal bicliques, Discrete Appl. Math. 145 (1) (2004) 11–21.
[2] D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, in: Proc. of the 18th Annual ACM–SIAM Symposium on Discrete Algorithms, 2007, pp. 1027–1035.
[3] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, V. Pandit, Local search heuristics for k-median and facility location problems, SIAM J. Comput. 33 (3) (2004) 544–562.
[4] P. Drineas, A. Frieze, R. Kannan, S. Vempala, V. Vinay, Clustering large graphs via the singular value decomposition, Mach. Learn. 56 (1–3) (2004) 9–33.
[5] D. Eppstein, Arboricity and bipartite subgraph listing algorithms, Inform. Process. Lett. 51 (4) (1994) 207–211.

[6] J.A. Hartigan, Direct clustering of a data matrix, J. Amer. Statist. Assoc. 67 (337) (1972) 123–129.
[7] D.S. Hochbaum, Approximating clique and biclique problems, J. Algorithms 29 (1) (1998) 174–200.
[8] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, Inc., 1988.
[9] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, A.Y. Wu, A local search approximation algorithm for k-means clustering, Comput. Geometry 28 (2–3) (2004) 89–112.
[10] S.C. Madeira, A.L. Oliveira, Biclustering algorithms for biological data analysis: A survey, IEEE Transactions on Computational Biology and Bioinformatics 1 (1) (2004) 24–45.