Approximation Algorithms for Co-Clustering

CS618 - Final Project Report

K.N.Praveen Kumar (Y5224) (praveenk@)
Rohit Gurjar (Y5383) (rgurjar@)
Fourth Year Undergraduates
Department of Computer Science and Engineering
IIT Kanpur

November 16, 2008

Contents

1 Introduction

Given a set of points and a pairwise distance measure, the objective of clustering is to partition the set into clusters such that points that are close to each other according to the distance measure occur together in a cluster and points that are far away from each other occur in different clusters. Clustering is of fundamental use in many data analaysis applications like information retrieval, data mining etc.

Consider the example of a Boolean matrix, whose rows correspond to movies and the columns correspond to viewers, and an entry is one if and only if the viewer has rated the movie as good. The goal is to cluster both the viewers and movies. One way to accomplish this would be to independently cluster the viewers and movies using the standard notion of clusteringcluster similar viewers and cluster similar movies. However this might fail to capture some subtle structures in the data. To be able to discover such things the clustering objective has to simultaneously retrieve information about both the movies and viewers. This is achieved by co-clustering. [CC00]

1.1 Co-Clustering

Co-clustering is the simultaneous partitioning of the rows and columns of a matrix such that the blocks induced by the row/column partitions are good clusters. In the simplest version of (k, l)-co-clustering, we are given a matrix of numbers and two integers k and l. The goal is partition the rows into k clusters and the columns into l clusters such that the sum-squared deviation from the mean within each block induced by the row-column partitions is minimized.

In the co-clustering problem the data is given in the form of a matrix A in $\mathbb{R}^{m\times n}$. Now, we want to compute a k-partitioning $\mathcal{I} = \{I_1, \ldots, I_k\}$ of the set of rows $\{1, \ldots, m\}$ and an l-partitioning $\mathcal{J} = \{J_1, \ldots, J_l\}$ of the set of columns $\{1, \ldots, n\}$. The two partitionings $\mathcal{I}$ and $\mathcal{J}$ induce clustering index matrices $R \in \mathbb{R}^{m\times k}$, $M \in \mathbb{R}^{k\times l}$ $C \in \mathbb{R}^{l\times n}$, defined as follows: each row in R corresponds to the index vector of the corresponding part in the partition $\mathcal{I}$ that is $R_{i,I} = 1$ if $A_{i\star} \in I$ and 0 otherwise. Similarly the index matrix C is constructed to represent the partitioning $\mathcal{J}$ that is $C_{J,j} = 1$ if $A_{\star j} \in J$ and 0 otherwise.

The clustering error associated with the co-clustering $(\mathcal{I}, \mathcal{J})$ is defined to be the qunatity

$$|||A - RMC|||_p,$$

where M is defined as the matrix in $\mathbb{R}^{k\times l}$ that minimizes

$$M = \arg\min_X |||A - RXC|||_p.$$

Let m_I be the size of the row-cluster I and n_J denote the size of the columns cluster J. By the definition of the $|||.|||_p$, we can write

$$|||A - RMC|||_p = \sum_{I \in \mathcal{I},\ J \in \mathcal{J}} |||A_{IJ} - \mu_{IJ} R_I C_J|||^{1/p}$$

where each $A_{IJ} \in \mathbb{R}^{m_I \times n_J}$, each vector $R_I \in \mathbb{R}^{m_I \times 1}$, and each $\mu_{IJ} \in \mathbb{R}$, vector $C_J \in \mathbb{R}^{1 \times n_J}$. For the $p = 2$ case, the matrix norm $|||.|||_p$ corresponds to the Frobenius norm $||.||_F$, and value μ_{IJ} corresponds to a simple average of the corresponding block.

2 Related Work

The main reference for this work is [ADK08]. In this work the authors give the first constant factor approximation algorithm for (k, l)-co-clustering problem and produces 3α-approximate solution, where α is the approximation factor for the k-means$_p$ problem; for the special case of the Frobenius norm and the constant k, l they obtain a $(\sqrt{2} + \epsilon)$-approximation algorithm by exploiting the geometry of the space and results in the k-means problem. They also show that $(k, 1)$-co-clustering problem can be solved exactly in time $O(mn+m^2k)$ and the (k, n^ϵ)-co-clustering problem is NP-hard, for $k \geq 2$ under the l_1 norm.

Note that k-means$_p$ problem is the one sided clustering problem. There are several publications on approximation algorithms for k-means$_p$ problem in low dimensional euclidean spaces. But their complexity is still open, when the number of clusters is $o(n)$. The authors of the above mentioned reference give an algorithm `Co-Cluster`(A, k, l) which gives an approximate (k, l)-clustering of the matrix A.

Algorithm 1 `Co-Cluster(A,k,l)`

Require: Matrix $A \in \mathbb{R}^{m \times n}$, number of row-clusters k, number of row-clusters l.

Ensure: Two partitions $\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$ of the matrix A that minimizes the above mentioned cost function.

1: Let $\hat{\mathcal{I}}$ be the α-approximate partitioning of the row vectors with k clusters.
2: Let $\hat{\mathcal{J}}$ be the α-approximate partitioning of the column vectors with l clusters.
3: **return** $(\hat{\mathcal{I}}, \hat{\mathcal{J}})$

They also prove the following theorems.

Theorem 1. *Given an α-approximation algorithm for the k-means$_p$ problem, the algorithm `Co-Cluster`(A, k, l) returns a co-clustering that is a 3α-approximation to the optimal co-clustering of A.*

Theorem 2. *Given an α-approximation algorithm for the k-means clustering problem, the algorithm* `Co-Cluster`(A, k, l) *gives a $\sqrt{2}\alpha$-approximate solution to the co-clustering problem with $||.||_F$ objective function(i.e, under the Frobenius norm).*

Lemma 3. *Let $A \in \mathbb{R}^{m\times 1}$ and consider any norm $|||.|||_p$, there is an algorithm that can $(k, 1)$-cluster matrix A optimally in time $O(m^2k)$ and space $O(mk)$.*

Theorem 4. *The problem of finding a (k, l)-co-clustering for a matrix $A \in \mathbb{R}^{m\times n}$ is NP-hard for $l = n^\epsilon$ and for any $k \geq 2$, $\epsilon > 0$, under the ℓ_1 norm.*

3 Contribution of our Project

3.1 Higher dimensional Co-Clustering

As we mentioned the authors of [ADK08] give a simple algorithm that achieves constant approximation w.r.t the optimum co-clustering. In a section on future work in their paper they ask if a version of this algorithm works when the matrix to be clustered has more than two dimensions i.e, what happens when $A \in \mathbb{R}^{m\times n\times o}$.

In this section, we give an algorithm that can approximately find a (p,q,r) Co-Clustering of the matrix $A \in \mathbb{R}^{m\times n\times o}$. We note that this algorithm is similar to the `Co-Cluster`(A, k, l) algorithm given in [ADK08]. As a matter of fact, we use the `Co-Cluster`(A, k, l) algorithm as a subroutine for our algorithm.

Algorithm 2 `Co-Cluster1(A,m,n,r)`

Require: Matrix $A \in \mathbb{R}^{m\times n\times o}$, r is the number of clusters in the third dimension.

Ensure: Partition $\hat{\mathcal{K}}$ of the indices of third dimenson that minimizes the above mentioned cost function.

1: Create matrix $A' \in \mathbb{R}^{mn\times o}$ s.t. for $1 \leq l \leq o$, $1 \leq k \leq mn$

$$A'_{kl} = A_{ijl}$$

where $i = k \mod m$ and $j = \lfloor k/m \rfloor$

2: Get partition $\hat{\mathcal{K}}$ from the one-sided column clustering of A' with r clusters.

3: **return** $(\hat{\mathcal{K}})$

This algorithm gives the partitions $\mathcal{K}$ that minimizes the following cost function:

$$c = \sum_{K\in\mathcal{K}} \sum_{k\in K} \sum_{i,j} |a_{ijk} - \mu_{ijK}|^p$$

where $\mu_{ijK} = \arg\min_X \sum_{k \in K} |a_{ijk} - x|^p$

Algorithm 3 `Co-Cluster2(A,p,q,o)`

Require: Matrix $A \in \mathbb{R}^{m \times n \times o}$, p and q are the number of clusters in the first and second dimensions.

Ensure: Two partitions $\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$ of the indices of first two dimensons that minimizes the above mentioned cost function.

1: Let $\hat{\mathcal{I}}$ be the α-approximate partitioning of the indices in the first dimensions with p clusters i.e., the partitioning resulting from `Co-Cluster1(A,p,n,o)`.

2: Let $\hat{\mathcal{J}}$ be the α-approximate partitioning of the indices in the second dimensions with q clusters i.e., the partitioning resulting from `Co-cluster1(A,m,q,o)`.

3: **return** $(\hat{\mathcal{I}}, \hat{\mathcal{J}})$

This algorithm gives the partitions $\mathcal{I}$, $\mathcal{J}$ that minimizes the following cost function:

$$c = \sum_{\substack{I \in \mathcal{I} \\ J \in \mathcal{J}}} \sum_{\substack{i \in I \\ j \in J}} \sum_{k} |a_{ijk} - \mu_{IJk}|^p$$

where $\mu_{IJk} = \arg\min_X \sum_{\substack{i \in I \\ j \in J}} |a_{ijk} - x|^p$

One can observe that this algorithm is similar to the `Co-Cluster` Algorithm (Algorithm 1). Hence, it is easy to prove that, given an α-approximate k-means solution, the `Co-Cluster2(`A, p, q, o`)` algorithm gives a 3α-approximate solution to the optimal (p,q) co-clustering of the first two dimensions of A. The proof goes in the similar lines as that of the proof for Theorem1, given in [ADK08].

Algorithm 4 `Co-Cluster3(A,p,q,r)`

Require: Matrix $A \in \mathbb{R}^{m \times n \times o}$, p, q and r are the number of clusters in the first second and third dimensions.

Ensure: Three partitions $\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$ and $\hat{\mathcal{K}}$ of matrix A that minimizes the above mentioned cost function.

1: Let $\hat{\mathcal{I}}$ and $\hat{\mathcal{J}}$ be the 3α-approximate partitioning of the indices in the first two dimensions with p, q clusters respectively i.e., the partitioning resulting from `Co-Cluster2(A,p,q,o)`.

2: Let $\hat{\mathcal{K}}$ be the α-approximate partitioning of the indices in the third dimension with r clusters i.e., the partitioning resulting from `Co-cluster1(A,m,n,r)`.

3: **return** $(\hat{\mathcal{I}}, \hat{\mathcal{J}}, \hat{\mathcal{K}})$

This algorithm gives the partitions $\mathcal{I}$, $\mathcal{J}$, $\mathcal{K}$ that minimizes the following cost function:

$$c = \sum_{\substack{I\in\mathcal{I}\\ J\in\mathcal{J}\\ K\in\mathcal{K}}} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - \mu_{IJK}|^p$$

where $\mu_{IJK} = \arg\min_X \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - x|^p$

3.2 Proof of correctness of the algorithm `Co-Cluster3`

Theorem 5. *Given an α-approximation algorithm for the k-means$_p$ problem, the algorithm `Co-Cluster3` returns a co-clustering that is a 5α -approximation to the optimal co-clustering of A.*

Proof. Let $\mathcal{I}^*$, $\mathcal{J}^*$, $\mathcal{K}^*$ be the optimal co-clustering solution. Let $\hat{\mathcal{I}}^*$, $\hat{\mathcal{J}}^*$ be the optimal partitioning of the indices of the first two dimensions(two-sided co-clustering of A). And $\hat{\mathcal{K}}^*$ be the optimal partitioning of the indices of the third dimension(one-sided clustering).

The algorithm `Co-Cluster3` uses approxitmate solution for the two-sided co-clustering of on the first two dimesions and one-sided clustering on the third dimension of A to compute $\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$, $\hat{\mathcal{K}}$. Now,

$$\left(\sum_{\substack{I\in\hat{\mathcal{I}}\\ J\in\hat{\mathcal{J}}}} \sum_{\substack{i\in I\\ j\in J}} \sum_{k} |a_{ijk} - \hat{\eta}_{IJk}|^p \right)^{1/p} \leq 3\alpha \left(\sum_{\substack{I'\in\hat{\mathcal{I}}^*\\ J'\in\hat{\mathcal{J}}^*}} \sum_{\substack{i\in I'\\ j\in J'}} \sum_{k} |a_{ijk} - \hat{\eta}^*_{I'J'k}|^p \right)^{1/p} \tag{1}$$

where $\hat{\eta}_{IJk} = \arg\min_X \sum_{\substack{i\in I\\ j\in J}} |a_{ijk} - x|^p$

and $\hat{\eta}^*_{I'J'k} = \arg\min_X \sum_{\substack{i\in I'\\ j\in J'}} |a_{ijk} - x|^p$

Equation (1) is true because ($\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$) is a 3α-approximate solution resulting from `Co-Cluster2` whose optimal solution is ($\hat{\mathcal{I}}^*$, $\hat{\mathcal{J}}^*$)

$$\left(\sum_{K\in\hat{\mathcal{K}}} \sum_{k\in K} \sum_{i,j} |a_{ijk} - \hat{\sigma}_{ijK}|^p \right)^{1/p} \leq \alpha \left(\sum_{K'\in\hat{\mathcal{K}}^*} \sum_{k\in K'} \sum_{i,j} |a_{ijk} - \hat{\sigma}^*_{ijK'}|^p \right)^{1/p} \tag{2}$$

where $\hat{\sigma}_{ijK} = \arg\min_X \sum_{k\in K} |a_{ijk} - x|^p$

and $\hat{\sigma}^*_{ijK'} = \arg\min_X \sum_{k \in K'} |a_{ijk} - x|^p$

Equation (2) is true because $\hat{\mathcal{K}}$ is a α-approximate solution resulting from `Co-Cluster1` whose optimal solution is $\hat{\mathcal{K}}^*$

For the approximate co-clustering ($\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$, $\hat{\mathcal{K}}$) let μ_{IJK} where $I \in \hat{\mathcal{I}}$, $J \in \hat{\mathcal{J}}$, $K \in \hat{\mathcal{K}}$, be defined as

$$\mu_{IJK} = \arg\min_X \sum_{\substack{i \in I \\ j \in J \\ k \in K}} |a_{ijk} - x|^p \tag{3}$$

For the optimal co-clustering ($\mathcal{I}^*$, $\mathcal{J}^*$, $\mathcal{K}^*$) let μ^*_{IJK} where $I \in \mathcal{I}^*$, $J \in \mathcal{J}^*$, $K \in \mathcal{K}^*$, be defined as

$$\mu^*_{IJK} = \arg\min_X \sum_{\substack{i \in I \\ j \in J \\ k \in K}} |a_{ijk} - x|^p \tag{4}$$

Now, we will show that the co-clustering ($\hat{\mathcal{I}}$, $\hat{\mathcal{J}}$, $\hat{\mathcal{K}}$) will be a 5α-approximate solution. We begin with a relationship between optimal co-clustering $(\mathcal{I}^*, \mathcal{J}^*, \mathcal{K}^*)$ and the optimal two-sided clustering $(\hat{\mathcal{I}}^*, \hat{\mathcal{J}}^*)$.

$$\begin{aligned} \left(\sum_{\substack{I \in \hat{\mathcal{I}}^* \\ J \in \hat{\mathcal{J}}^*}} \sum_{\substack{i \in I \\ j \in J}} \sum_{k} |a_{ijk} - \hat{\eta}^*_{IJk}|^p \right)^{1/p} &\leq \left(\sum_{\substack{I \in \mathcal{I}^* \\ J \in \mathcal{J}^*}} \sum_{\substack{i \in I \\ j \in J}} \sum_{k} |a_{ijk} - \nu^*_{IJk}|^p \right)^{1/p} \\ &\leq \left(\sum_{\substack{I \in \mathcal{I}^* \\ J \in \mathcal{J}^* \\ K \in \mathcal{K}^*}} \sum_{\substack{i \in I \\ j \in J \\ k \in K}} |a_{ijk} - \mu^*_{IJK}|^p \right)^{1/p} \end{aligned} \tag{5}$$

where $\nu^*_{IJk} = \arg\min_X \sum_{\substack{i \in I \\ j \in J}} |a_{ijk} - x|^p$

$$\left(\sum_{K\in\hat{\mathcal{K}}^*}\sum_{k\in K}\sum_{i,j}|a_{ijk}-\hat{\sigma}^*_{ijK}|^p\right)^{1/p} \leq \left(\sum_{K\in\mathcal{K}^*}\sum_{k\in K}\sum_{i,j}|a_{ijk}-\xi^*_{ijK}|^p\right)^{1/p}$$
$$\leq \left(\sum_{\substack{I\in\mathcal{I}^*\\J\in\mathcal{J}^*\\K\in\mathcal{K}^*}}\sum_{\substack{i\in I\\j\in J\\k\in K}}|a_{ijk}-\mu^*_{IJK}|^p\right)^{1/p} \tag{6}$$

where $\xi^*_{ijK} = \arg\min_X \sum_{k\in K}|a_{ijk}-x|^p$

For $(I,J,K)\in\hat{\mathcal{I}}\times\hat{\mathcal{J}}\times\hat{\mathcal{K}}$, Let $\hat{\mu}_{IJK} = \arg\min_X \sum_{\substack{i\in I\\j\in J}}|\hat{\sigma}_{ijK}-x|^p$

Then for all $k\in K$ we have

$$\sum_{\substack{i\in I\\j\in J}}|\hat{\sigma}_{ijK}-\hat{\mu}_{IJK}|^p \leq \sum_{\substack{i\in I\\j\in J}}|\hat{\sigma}_{ijK}-\hat{\eta}_{IJk}|^p$$

which gives

$$\left(\sum_{\substack{i\in I\\j\in J\\k\in K}}|\hat{\sigma}_{ijK}-\hat{\mu}_{IJK}|^p\right)^{1/p} \leq \left(\sum_{\substack{i\in I\\j\in J\\k\in K}}|\hat{\sigma}_{ijK}-\hat{\eta}_{IJk}|^p\right)^{1/p}$$
$$\leq \left(\sum_{\substack{i\in I\\j\in J\\k\in K}}|\hat{\sigma}_{ijK}-a_{ijk}|^p\right)^{1/p} + \left(\sum_{\substack{i\in I\\j\in J\\k\in K}}|a_{ijk}-\hat{\eta}_{IJk}|^p\right)^{1/p} \tag{7}$$

where the last inequality is just application of the triangle inequality.

Then we get

$$
\begin{aligned}
\left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - \mu_{IJK}|^p \right)^{1/p}
&\overset{(a)}{\leq} \left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - \hat{\mu}_{IJK}|^p \right)^{1/p} \\
&\overset{(b)}{\leq} \left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |\hat{\sigma}_{ijK} - a_{ijk}|^p \right)^{1/p} + \left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |\hat{\sigma}_{ijK} - \hat{\mu}_{ijk}|^p \right)^{1/p} \\
&\overset{(c)}{\leq} 2\left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |\hat{\sigma}_{ijK} - a_{ijk}|^p \right)^{1/p} + \left(\sum_{I,J,K} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - \hat{\eta}_{IJk}|^p \right)^{1/p} \\
&\overset{(d)}{\leq} 2\alpha\left(\sum_{K'\in\hat{\mathcal{K}}^*} \sum_{k\in K'} \sum_{i,j} |a_{ijk} - \hat{\sigma}^*_{ijK'}|^p \right)^{1/p} \\
&\quad +3\alpha\left(\sum_{\substack{I'\in\hat{\mathcal{I}}^*\\ J'\in\hat{\mathcal{J}}^*}} \sum_{\substack{i\in I'\\ j\in J'}} \sum_{k} |a_{ijk} - \hat{\eta}^*_{I'J'k}|^p \right)^{1/p} \\
&\overset{(e)}{\leq} 5\alpha\left(\sum_{\substack{I\in\mathcal{I}^*\\ J\in\mathcal{J}^*\\ K\in\mathcal{K}^*}} \sum_{\substack{i\in I\\ j\in J\\ k\in K}} |a_{ijk} - \mu^*_{IJK}|^p \right)^{1/p}
\end{aligned}
$$

where (a) follows from equation(3), (b) is an application of triangular inequality, (c) follows from equation(7), (d) follows from equation(2) and equation(1), (e) follows from equation(6) and equation(5). □

3.3 Hardness of (o(n),o(n)) Co-clustering

Recently, Ben-David and Wulff[DW08] proved that the general (k,l) Monochromatic Bi-clustering is NP hard. Now, the problem of (k,l)-Monochromatic Bi-clustering of a matrix A is the same as the problem of finding a (k,l)-Coclusteing of A under ℓ_1 norm. This can be seen as follows. Following the definition given in [DW08], the monochromatic cost for a given bi-clustering partition $\mathcal{I} = \{I_1, \ldots, I_k\}$ of rows and $\mathcal{J} = \{J_1, \ldots, J_l\}$ of columns is the sum:

$$\mathcal{C} = \sum_{\substack{1 \le s \le k \\ 1 \le s \le l}} \phi(I_s \times J_t)$$

where $\phi(B)$ is the number of entries in a block B that are not equal to the majority value in B.

Since, the problem deals with the case where the entries of A are only from $\{0, 1\}$, if the majority value in the block B is 1, then the median(B) is also 1. In that case,

$$\mathcal{C} = \sum_{\substack{I \in \mathcal{I} \\ J \in \mathcal{J}}} \sum_{\substack{i \in I \\ j \in J}} |a_{ij} - \mu_{median}(I_s \times J_t)|$$

Thus, the Co-clustering problem under ℓ_1 norm is same as Monochromatic Bi-clustering problem. Since the B-clustering problem is proved hard, (o(n),o(n)) Co-clustering problem under ℓ_1 norm should also be NP Hard.

One doubts as to whether this is not true for other ℓ_p norms too. We have studied the proof for the NP hardness of Monochromatic B-Clustering problem given by the before mentioned others and believe that the proof outline may be used to prove the NP-hardness of C-Clustering under ℓ_2 norm. The authors in [DW08] reduce a certain *Corelation Clustering* problem (Which is proved to be NP hard[GG06]), to the monochromatic Bi-clustering problem by using an extra O(N^2) blocks of zeros and ones, where $N = 4n$. We believe the proof structure should hold for the case, where the objective function is defined in terms of ℓ_2 norm. In that case, the $\mu_{median}(B)$ value is replaced by $\mu(B)$ which is the average of the values in the block B. We believe, this should be possible using an extra O(N^2) blocks of zeros and ones, where $N = n^3$. The value of N is chosen such that the average of a block of size O(N^2) is $1 - \epsilon$ such that O($N^2\epsilon$) does not exceed 1/2. This would be major part of our future work. The above proof however may not help us in determining whether Co-Clustering is hard for other ℓ_p norms, where $p \geq 3$.

4 Other hardness results for Co-Clustering

The authors of [ADK08] proved that the co-clustering problem is hard under the ℓ_1 norm for $l = \Omega(n^\epsilon)$. They say that the hardness should hold for any ℓ_p norm, $p \geq 1$. We believe it is easy to prove the hardness for any ℓ_p norm by following the same lines of proof as that for ℓ_1 norm. It would also be interesting to show that it is hard for any combination of k, l. In particular hardness questions for the $(2, 2)$ or the $(O(1), O(1))$ cases are also unresolved. They conjecture that these cases are also hard. It is easy to reduce one-sided $o(n)$-clustering to $(k, o(n))$-co-clustering. However, the complexity results for $o(n)$ one-sided clustering are not known even though there are many approximation algorithms for the same in the literature. So, any significant result on the hardness of $(O(1), O(1))$-co-clustering could be difficult to achieve for general ℓ_p.

References

[ADK08] Aris Anagnostopoulos, Anirban Dasgupta and Ravi Kumar, *Approximation Algorithms for Co-Clustering*, PODS08, June 912, 2008, Vancouver, BC, Canada

[CC00] Y. Cheng and G. M. Church *Biclustering of expression data.*, Proc. of the 8th International Conference on Intelligent Systems for Molecular Biology, pages 93 - 103, 2000.

[DW08] Shai Ben-David, Sharon Wulff , *On The Computational Complexity Of Bi-Clustering.*, Masters Thesis, University of Waterloo, Canada, 2008. `http://uwspace.uwaterloo.ca/bitstream/10012/3900/1/thesis.pdf.`

[GG06] Ioannis Giotis and Venkatesan Guruswami. *Correlation clustering with a fixed number of clusters.* Theory of Computing, 2(13):249266, 2006.