

Consistent Biclustering via Fractional 0–1 Programming

Panos Pardalos, Stanislav Busygin and Oleg Prokopyev

Center for Applied Optimization
Department of Industrial & Systems Engineering
University of Florida

Massive Datasets

- The proliferation of **massive datasets** brings with it a series of special computational challenges. This **data avalanche** arises in a wide range of scientific and commercial applications.
- In particular, microarray technology allows one to grasp simultaneously thousands of gene expressions throughout the entire genome. To extract useful information from such datasets a sophisticated data mining algorithm is required.

Massive Datasets



Abello, J.; Pardalos, P.M.; Resende, M.G. (Eds.),
Handbook of Massive Data Sets, Series: Massive
Computing, Vol. 4, Kluwer, 2002.

Data Representation

- A dataset (e.g., from microarray experiments) is normally given as a rectangular $m \times n$ matrix A , where each column represents a data sample (e.g., patient) and each row represents a feature (e.g., gene):

$$A = (a_{ij})_{m \times n},$$

where the value a_{ij} is the expression of i -th feature in j -th sample.

Major Data Mining Problems

- **Clustering (Unsupervised):** Given a set of samples partition them into groups of similar samples according to some similarity criteria.
- **Classification (Supervised Clustering):** Determine classes of the test samples using known classification of training data set.
- **Feature Selection:** For each of the classes, select a subset of features responsible for creating the condition corresponding to the class (it's also a specific type of **dimensionality reduction**).
- **Outlier Detection:** Some of the samples are not good representative of any of the classes. Therefore, it is better to disregard them while performing data mining.

Major challenges in Data Mining

- Typical noisiness of data arising in many data mining applications complicates solution of data mining problems.
- High-dimensionality of data makes complete search in most of data mining problems computationally infeasible.
- Some data values may be inaccurate or missing.
- The available data may be not sufficient to obtain statistically significant conclusions.

Biclustering

- Biclustering is a methodology allowing for feature set and test set clustering (supervised or unsupervised) simultaneously.
- It finds clusters of samples possessing similar characteristics together with features creating these similarities.
- The required consistency of sample and feature classification gives biclustering an advantage over other methodologies treating samples and features of a dataset separately of each other.

Biclustering

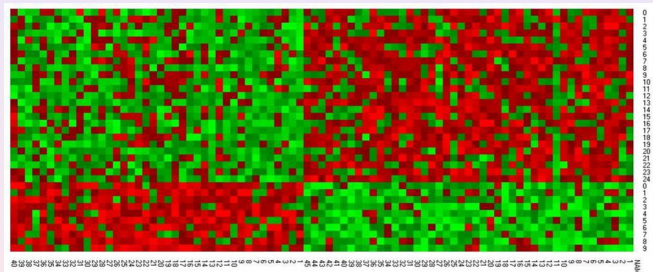


Figure: Partitioning of samples and features into 2 classes.

Survey on Biclustering Methodologies

- **“Direct Clustering” (Hartigan)**
- The algorithm begins with the entire data as a single block and then iteratively finds the row and column split of every block into two pieces. The splits are made so that the total variance in the blocks is minimized.
- The whole partitioning procedure can be represented in a hierarchical manner by trees.
- Drawback: this method does NOT optimize a global objective function.

Survey on Biclustering Methodologies

- **Cheng & Church's algorithm**
- The algorithm constructs one bicluster at a time using a statistical criterion – a low mean squared residue (the variance of the set of all elements in the bicluster, plus the mean row variance and the mean column variance).
- Once a bicluster is created, its entries are replaced by random numbers, and the procedure is repeated iteratively.

Survey on Biclustering Methodologies

- **Graph Bipartitioning**
- Define a bipartite graph $G(F, S, E)$, where F is the set of data set features, S is the set of data set samples, and E are weighted edges such that the weight $E_{ij} = a_{ij}$ for the edge connecting $i \in F$ with $j \in S$. The biclustering corresponds to partitioning of the graph into bicliques.

Survey on Biclustering Methodologies

- Given vertex subsets V_1 and V_2 , define

$$\text{cut}(V_1, V_2) = \sum_{i \in V_1} \sum_{j \in V_2} a_{ij}$$

and for k vertex subsets V_1, V_2, \dots, V_k ,

$$\text{cut}(V_1, V_2, \dots, V_k) = \sum_{i < j} \text{cut}(V_i, V_j)$$

Survey on Biclustering Methodologies

- Biclustering may be performed as

$$\min_{V_1, V_2, \dots, V_k} \text{cut}(V_1, V_2, \dots, V_k),$$

on G or with some modification of the definition of cut to favor balanced clusters.

- This problem is NP -hard, but spectral heuristics show good performance [**Dhillon**]

Biclustering: Applications

- **Biological and Medical:**
 - Microarray data analysis
 - Analysis of drug activity, Liu and Wang (2003)
 - Analysis of nutritional data, Lazzeroni et al. (2000)

Biclustering: Applications

- **Text Mining:** Dhillon (2001, 2003)
- **Marketing:** Gaul and Schader (1996)
- **Dimensionality Reduction in Databases:** Agrawal et al. (1998)
- **Others:**
 - electoral data - Hartigan (1972)
 - currency exchange - Lazzeroni et al. (2000)

Biclustering: Surveys

- S. Madeira, A.L. Oliveira, Biclustering Algorithms for Biological Data Analysis: A Survey, 2004.
- A. Tanay, R. Sharan, R. Shamir, Biclustering Algorithms: A Survey, 2004.
- D. Jiang, C. Tang, A. Zhang, Cluster Analysis for Gene Expression Data: A Survey, 2004.

Definitions

- Data set of n samples and m features is a matrix

$$A = (a_{ij})_{m \times n},$$

where the value a_{ij} is the expression of i -th feature in j -th sample.

- We consider classification of the samples into classes

$$\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_r, \mathcal{S}_k \subseteq \{1 \dots n\}, k = 1 \dots r,$$

$$\mathcal{S}_1 \cup \mathcal{S}_2 \cup \dots \cup \mathcal{S}_r = \{1 \dots n\},$$

$$\mathcal{S}_k \cap \mathcal{S}_\ell = \emptyset, k, \ell = 1 \dots r, k \neq \ell.$$

Definitions

- This classification should be done so that samples from the same class share certain common properties. Correspondingly, a feature i may be assigned to one of the feature classes

$$\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_r, \mathcal{F}_k \subseteq \{1 \dots m\}, k = 1 \dots r,$$

$$\mathcal{F}_1 \cup \mathcal{F}_2 \cup \dots \cup \mathcal{F}_r = \{1 \dots m\},$$

$$\mathcal{F}_k \cap \mathcal{F}_\ell = \emptyset, k, \ell = 1 \dots r, k \neq \ell,$$

in such a way that features of the class \mathcal{F}_k are “**responsible**” for creating the class of samples \mathcal{S}_k .

Definitions

- This may mean for microarray data, for example, strong up-regulation of certain genes under a cancer condition of a particular type (whose samples constitute one class of the data set). Such a simultaneous classification of samples and features is called **biclustering** (or **co-clustering**).

Definitions

Definition

A *biclustering* of a data set is a collection of pairs of sample and feature subsets

$$\mathcal{B} = ((\mathcal{S}_1, \mathcal{F}_1), (\mathcal{S}_2, \mathcal{F}_2), \dots, (\mathcal{S}_r, \mathcal{F}_r))$$

such that the collection $(\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_r)$ forms a partition of the set of samples, and the collection $(\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_r)$ forms a partition of the set of features.

Our Approach: Intuition

- Let us distribute features among the classes of training set such that each feature belongs to the class where its average expression among the training samples is highest.
- Now, if we transpose the matrix, take the feature classification as given, and re-classify the training samples according to highest average expression values in feature classes, will we obtain the same training set classification?
- If yes, we will say that we obtained a **consistent biclustering**.

Consistent Biclustering

- Let each sample be already assigned somehow to one of the classes $\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_r$. Introduce a 0–1 matrix $S = (s_{jk})_{n \times r}$ such that $s_{jk} = 1$ if $j \in \mathcal{S}_k$, and $s_{jk} = 0$ otherwise.
- The sample class *centroids* can be computed as the matrix $C = (c_{ik})_{m \times r}$:

$$C = AS(S^T S)^{-1},$$

whose k -th column represents the centroid of the class \mathcal{S}_k .

Consistent Biclustering

- Consider a row i of the matrix C . Each value in it gives us the average expression of the i -th feature in one of the sample classes. As we want to identify the checkerboard pattern in the data, we have to assign the feature to the class where it is most expressed. So, let us classify the i -th feature to the class \hat{k} with the maximal value $c_{i\hat{k}}$:

$$i \in \mathcal{F}_{\hat{k}} \Rightarrow \forall k = 1 \dots r, k \neq \hat{k} : c_{i\hat{k}} > c_{ik}$$

Consistent Biclustering

- Using the classification of all features into classes $\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_r$, let us construct a classification of samples using the same principle of maximal average expression. We construct a 0–1 matrix $F = (f_{ik})_{m \times r}$ such that $f_{ik} = 1$ if $i \in \mathcal{F}_k$ and $f_{ik} = 0$ otherwise. Then, the feature class centroids can be computed in form of matrix $D = (d_{jk})_{n \times r}$:

$$D = A^T F (F^T F)^{-1},$$

whose k -th column represents the centroid of the class \mathcal{F}_k .

Consistent Biclustering

- The condition on sample classification we need to verify is

$$j \in \mathcal{S}_{\hat{k}} \Rightarrow \forall k = 1 \dots r, k \neq \hat{k} : d_{j\hat{k}} > d_{jk}$$

Consistent Biclustering

Definition

A biclustering \mathcal{B} will be called **consistent** if the following relations hold:

$$i \in \mathcal{F}_{\hat{k}} \Rightarrow \forall k = 1 \dots r, k \neq \hat{k} : c_{i\hat{k}} > c_{ik}$$

$$j \in \mathcal{S}_{\hat{k}} \Rightarrow \forall k = 1 \dots r, k \neq \hat{k} : d_{j\hat{k}} > d_{jk}$$

Consistent Biclustering

Definition

A data set is **biclustering-admitting** if some consistent biclustering for it exists.

Definition

The data set will be called **conditionally biclustering-admitting** with respect to a given (partial) classification of some samples and/or features if there exists a consistent biclustering preserving the given (partial) classification.

Consistent Biclustering

- **A consistent biclustering implies separability of the classes by convex cones.**

Theorem

Let \mathcal{B} be a consistent biclustering. Then there exist convex cones $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_r \subseteq \mathbb{R}^m$ such that all samples from \mathcal{S}_k belong to the cone \mathcal{P}_k and no other sample belongs to it, $k = 1 \dots r$. Similarly, there exist convex cones $\mathcal{Q}_1, \mathcal{Q}_2, \dots, \mathcal{Q}_r \subseteq \mathbb{R}^n$ such that all features from \mathcal{F}_k belong to the cone \mathcal{Q}_k and no other feature belongs to it, $k = 1 \dots r$.

Conic Separability

Proof.

Let \mathcal{P}_k be the conic hull of the samples of \mathcal{S}_k . Suppose $\hat{j} \in \mathcal{S}_\ell$, $\ell \neq k$, belongs to \mathcal{P}_k . Then

$$\mathbf{a}_{\hat{j}} = \sum_{j \in \mathcal{S}_k} \gamma_j \mathbf{a}_j,$$

where $\gamma_j \geq 0$. Biclustering consistency implies that $d_{j\ell} > d_{jk}$, that is

$$\frac{\sum_{i \in \mathcal{F}_\ell} \mathbf{a}_{i\hat{j}}}{|\mathcal{F}_\ell|} > \frac{\sum_{i \in \mathcal{F}_k} \mathbf{a}_{i\hat{j}}}{|\mathcal{F}_k|}$$

Conic Separability

Proof (cont'd).

Plugging the conic representation of a_{ij} , we can obtain

$$\sum_{j \in \mathcal{S}_k} \gamma_j d_{j\ell} > \sum_{j \in \mathcal{S}_k} \gamma_j d_{jk},$$

that contradicts to $d_{j\ell} < d_{jk}$ (also implied by biclustering consistency).

Similarly, we can show that the formulated conic separability holds for feature classes.

Biclustering

- **Supervised Biclustering**
- **Unsupervised Biclustering**

Supervised Biclustering

- One of the most important problems for real-life data mining applications is **supervised classification** of test samples on the basis of information provided by training data.
- A **supervised classification** method consists of two routines, first of which derives classification criteria while processing the training samples, and the second one applies these criteria to the test samples.

Supervised Biclustering

- In genomic and proteomic data analysis, as well as in other data mining applications, where only a small subset of features is expected to be relevant to the classification of interest, the classification criteria should involve dimensionality reduction and feature selection.
- We handle such a task utilizing the notion of consistent biclustering. Namely, we select a subset of features of the original data set in such a way that the obtained subset of data becomes conditionally biclustering-admitting with respect to the given classification of training samples.

Fractional 0–1 Programming Formulation

- Formally, let us introduce a vector of 0–1 variables $\mathbf{x} = (x_i)_{i=1\dots m}$ and consider the i -th feature selected if $x_i = 1$.
- The condition of biclustering consistency, when only the selected features are used, becomes

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{k}} x_i}{\sum_{i=1}^m f_{i\hat{k}} x_i} > \frac{\sum_{i=1}^m a_{ij} f_{ik} x_i}{\sum_{i=1}^m f_{ik} x_i}, \forall j \in \mathcal{S}_{\hat{k}}, \hat{k}, k = 1 \dots r, \hat{k} \neq k.$$

Fractional 0–1 Programming Formulation

- We will use the fractional relations as constraints of an optimization problem selecting the feature set. It may incorporate various objective functions over x , depending on the desirable properties of the selected features, but one general choice is **to select the maximal possible number of features in order to lose minimal amount of information provided by the training set**. In this case, the objective function is

$$\max \sum_{i=1}^m x_i$$

Fractional 0–1 Programming Formulation

- One of the possible fractional 0–1 formulations based on biclustering criterion:

$$\max_{\mathbf{x} \in \mathbb{B}^n} \sum_{i=1}^m x_i,$$

s.t.

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{k}} x_i}{\sum_{i=1}^m f_{i\hat{k}} x_i} \geq (1+t) \frac{\sum_{i=1}^m a_{ij} f_{ik} x_i}{\sum_{i=1}^m f_{ik} x_i}, \quad \forall j \in \mathcal{S}_{\hat{k}}, \hat{k}, k = 1 \dots r, \hat{k} \neq k,$$

where t is a class separation parameter.

Fractional 0–1 Programming Formulation

- Generally, in the framework of fractional 0–1 programming we consider problems, where we optimize a multiple-ratio fractional 0–1 function subject to a set of linear constraints.
- We have a **new class** of fractional 0–1 programming problems, where fractional terms are not in the objective function, but in constraints, i.e. we optimize a linear objective function subject to fractional constraints.
- **How to solve fractionally constrained 0–1 programming problem?**

Linear Mixed 0–1 Formulation

- We can reduce our problem to a linear mixed 0–1 programming problem applying the approach similar to the one used to linearize problems with fractional 0–1 objective function.



T.-H. Wu, A note on a global approach for general 0–1 fractional programming, *European J. Oper. Res.* 101 (1997) 220–223.

Linear Mixed 0–1 Formulation

Theorem

A polynomial mixed 0–1 term $z = xy$, where x is a 0–1 variable, and y is a continuous variable, can be represented by the following linear inequalities:

- (1) $z \leq Ux$;
- (2) $z \leq y + L(x - 1)$;
- (3) $z \geq y + U(x - 1)$;
- (4) $z \geq Lx$,

where U and L are upper and lower bounds of variable y , i.e. $L \leq y \leq U$.

Linear Mixed 0–1 Formulation

- To linearize the fractional 0–1 program we need to introduce new variable y_k

$$y_k = \frac{1}{\sum_{\ell=1}^m f_{\ell k} x_{\ell}}, \quad k = 1, \dots, r.$$

Linear Mixed 0–1 Formulation

- In terms of the new variables fractional constraints are replaced by

$$\sum_{i=1}^m a_{ij} f_{i\hat{k}} x_i y_{\hat{k}} \geq (1 + t) \sum_{i=1}^m a_{ij} f_{ik} x_i y_k$$

Linear Mixed 0–1 Formulation

- Next, observe that the term $x_i y_k$ is present if and only if $f_{ik} = 1$, i.e., $i \in \mathcal{F}_k$. So, there are totally only m of such products, and hence we can introduce m variables $z_i = x_i y_k, i \in \mathcal{F}_k$:

$$z_i = \frac{x_i}{\sum_{\ell=1}^m f_{\ell k} x_{\ell}}, i \in \mathcal{F}_k.$$

Linear Mixed 0–1 Formulation

- In terms of z_i we have the following constraints:

$$\sum_{i=1}^m f_{ik} z_i = 1, \quad k = 1 \dots r.$$

$$\sum_{i=1}^m a_{ij} f_{i\hat{k}} z_i \geq (1+t) \sum_{i=1}^m a_{ij} f_{ik} z_i \quad \forall j \in \mathcal{S}_{\hat{k}}, \quad \hat{k}, k = 1 \dots r, \quad \hat{k} \neq k.$$

$$y_k - z_i \leq 1 - x_i, \quad z_i \leq y_k, \quad z_i \leq x_i, \quad z_i \geq 0, \quad i \in \mathcal{F}_k.$$

Supervised Biclustering

- Unfortunately, while the linearization works nicely for small-size problems, it often creates instances, where the gap between the integer programming and the linear programming relaxation optimum solutions is very big for larger problems. As a consequence, the instance **can not be solved in a reasonable time** even with the best techniques implemented in modern integer programming solvers.
- **HuGE Index Data set: about 7000 features**
- **ALL vs. AML Data Set: about 7000 features**
- **GBM vs. AO data set: about 12000 features**

Heuristic

- If we know that no more than m_k features can be selected for class \mathcal{F}_k , then we can impose

$$x_i \leq m_k z_i, \quad x_i \geq z_i, \quad i \in \mathcal{F}_k.$$

Heuristic

Algorithm 1

1. Assign $m_k := |\mathcal{F}_k|$, $k = 1 \dots r$.
2. Solve the mixed 0–1 programming formulation using the inequalities

$$x_i \leq m_k z_i, \quad x_i \geq z_i, \quad i \in \mathcal{F}_k.$$

instead of

$$y_k - z_i \leq 1 - x_i, \quad z_i \leq y_k, \quad z_i \leq x_i, \quad z_i \geq 0, \quad i \in \mathcal{F}_k.$$

3. If $m_k = \sum_{i=1}^m f_{ik} x_i$ for all $k = 1 \dots r$, go to 6.
4. Assign $m_k := \sum_{i=1}^m f_{ik} x_i$ for all $k = 1 \dots r$.
5. Go to 2.
6. STOP.

Supervised Biclustering

- After the feature selection is done, we perform classification of test samples according to the following procedure.
- If $b = (b_i)_{i=1\dots m}$ is a test sample, we assign it to the class $\mathcal{F}_{\hat{k}}$ satisfying

$$\frac{\sum_{i=1}^m b_i f_{i\hat{k}} x_i}{\sum_{i=1}^m f_{i\hat{k}} x_i} > \frac{\sum_{i=1}^m b_i f_{ik} x_i}{\sum_{i=1}^m f_{ik} x_i}, \quad k = 1 \dots r, \quad \hat{k} \neq k.$$

HuGE index data set: Feature Selection

- A computational experiment that we conducted was on **feature selection for consistent biclustering** of the Human Gene Expression (*HuGE*) Index data set. The purpose of the HuGE project is to provide a comprehensive database of gene expressions in normal tissues of different parts of human body and to highlight similarities and differences among the organ systems.
- The number of selected features (genes) is 6889 (out of 7070).

HuGE index data set: Feature Selection

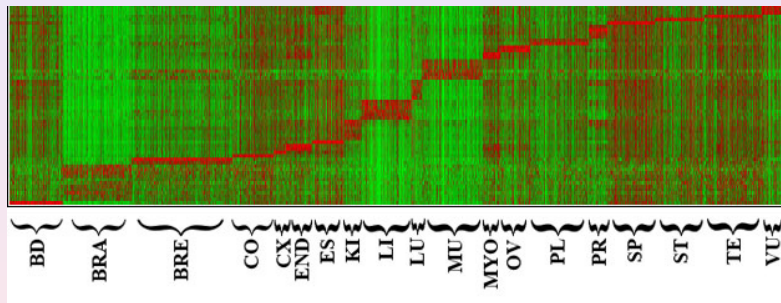


Figure: HuGE Index heatmap.

ALL vs. AML data set

- T. Golub et al. (1999) considered a dataset containing 47 samples from *ALL* patients and 25 samples from *AML* patients. The dataset was obtained with Affymetrix GeneChips.
- Our biclustering algorithm selected 3439 features for class *ALL* and 3242 features for class *AML*. The subsequent classification contained **only one error**: the *AML*-sample 66 was classified into the *ALL* class.
- The SVM approach delivers up to 5 classification errors depending on how the parameters of the method are tuned. The perfect classification was obtained only with one specific set of values of the parameters.

ALL vs. AML data set

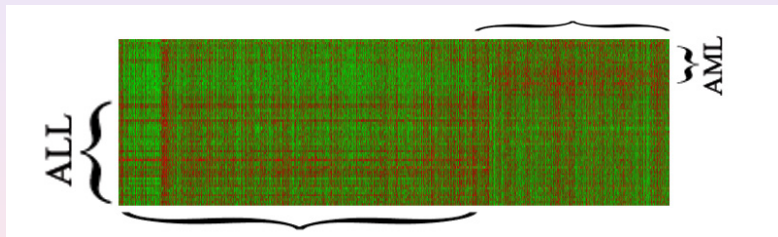


Figure: ALL vs. AML heatmap.

GBM vs. AO data set

- The algorithm selected 3875 features for the class GBM and 2398 features for the class AO. The obtained classification contained only 4 errors: two GBM samples (Brain_NG_1 and Brain_NG_2) were classified into the AO class and two AO samples (Brain_NO_14 and Brain_NO_8) were classified into the GBM class.

GBM vs. AO data set

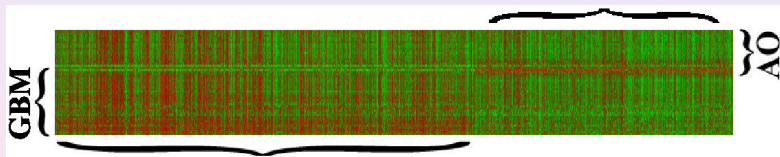




Figure: GBM vs. AO heatmap.

References

-  S. Busygin, P. Pardalos, O. Prokopyev, Feature selection for consistent biclustering via fractional 0–1 programming, *Journal of Combinatorial Optimization*, Vol. 10/1 (2005), pp. 7–21.
-  P.M. Pardalos, S. Busygin, O.A. Prokopyev, “On Biclustering with Feature Selection for Microarray Data Sets,” *BIOMAT 2005 – International Symposium on Mathematical and Computational Biology*, R. Mondaini (ed.), World Scientific (2006), pp. 367–378.

Unsupervised Biclustering

- Suppose we want to assign each sample to one of the classes

$$\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_r.$$

We introduce a 0–1 matrix $S = (s_{jk})_{n \times r}$ such that $s_{jk} = 1$ if $j \in \mathcal{S}_k$, and $s_{jk} = 0$ otherwise.

- We also want to classify all features into classes

$$\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_r.$$

Let us introduce a 0–1 matrix $F = (f_{ik})_{m \times r}$ such that $f_{ik} = 1$ if $i \in \mathcal{F}_k$ and $f_{ik} = 0$ otherwise.

Unsupervised Biclustering

- We have the following constraints on *biclustering consistency*:

$$s_{j\hat{k}} \left(\frac{\sum_{i=1}^m a_{ij} f_{i\hat{k}}}{\sum_{i=1}^m f_{i\hat{k}}} - (1+t) \frac{\sum_{i=1}^m a_{ij} f_{ik}}{\sum_{i=1}^m f_{ik}} \right) \geq 0 \quad \forall j, \hat{k}, k = 1 \dots r, \hat{k} \neq k$$

$$f_{i\hat{k}} \left(\frac{\sum_{j=1}^n a_{ij} s_{j\hat{k}}}{\sum_{j=1}^n s_{j\hat{k}}} - (1+t) \frac{\sum_{j=1}^n a_{ij} s_{jk}}{\sum_{j=1}^n s_{jk}} \right) \geq 0 \quad \forall i, \hat{k}, k = 1 \dots r, \hat{k} \neq k$$

Unsupervised Biclustering

- These constraints are equivalent to

$$\frac{\sum_{i=1}^m a_{ij} f_{i\hat{k}}}{\sum_{i=1}^m f_{i\hat{k}}} - (1 + t) \frac{\sum_{i=1}^m a_{ij} f_{ik}}{\sum_{i=1}^m f_{ik}} \geq -L_j^s (1 - s_{j\hat{k}})$$

$$\frac{\sum_{j=1}^n a_{ij} s_{j\hat{k}}}{\sum_{j=1}^n s_{j\hat{k}}} - (1 + t) \frac{\sum_{j=1}^n a_{ij} s_{jk}}{\sum_{j=1}^n s_{jk}} \geq -L_i^f (1 - f_{i\hat{k}})$$

Unsupervised Biclustering

- L_i^f and L_j^s are large enough constants, which can be chosen as

$$L_j^s = \max_i a_{ij} - \min_i a_{ij}$$

$$L_i^f = \max_j a_{ij} - \min_j a_{ij}$$

Linear Mixed 0–1 Reformulation

Let us introduce new variables

$$u_k = \frac{1}{\sum_{i=1}^m f_{ik}}, \quad k = 1 \dots r.$$

$$v_k = \frac{1}{\sum_{j=1}^n s_{jk}}, \quad k = 1 \dots r.$$

$$z_{ik} = \frac{f_{ik}}{\sum_{\ell=1}^m f_{\ell k}}, \quad i = 1 \dots m, \quad k = 1 \dots r.$$

$$y_{jk} = \frac{s_{jk}}{\sum_{\ell=1}^n s_{\ell k}}, \quad j = 1 \dots n, \quad k = 1 \dots r.$$

Linear Mixed 0–1 Reformulation

$$\sum_{i=1}^m a_{ij} z_{i\hat{k}} - (1+t) \sum_{i=1}^m a_{ij} z_{ik} \geq -L_j^s (1 - s_{j\hat{k}}) \quad \forall j, \hat{k}, k = 1 \dots r, \hat{k} \neq k,$$

$$\sum_{j=1}^n a_{ij} y_{j\hat{k}} - (1+t) \sum_{j=1}^n a_{ij} y_{jk} \geq -L_i^f (1 - f_{i\hat{k}}) \quad \forall i, \hat{k}, k = 1 \dots r, \hat{k} \neq k,$$

$$\sum_{i=1}^m z_{ik} = 1, \quad \sum_{j=1}^n y_{jk} = 1, \quad k = 1 \dots r.$$

$$u_k - z_{ik} \leq 1 - f_{ik}, \quad z_{ik} \leq u_k, \quad z_{ik} \leq f_{ik}, \quad z_{ik} \geq 0, \quad \forall i, k = 1 \dots r.$$

$$v_k - y_{jk} \leq 1 - s_{jk}, \quad y_{jk} \leq v_k, \quad y_{jk} \leq s_{jk}, \quad y_{jk} \geq 0, \quad \forall j, k = 1 \dots r.$$

Linear Mixed 0–1 Reformulation

- The number of new continuous variables is $2r$.
- The number of new 0–1 variables is $(m + n)r$.
- The total number of new variables is

$$2r + (m + n)r$$

Additional Constraints

- Each feature can be selected to at most one class

$$\forall i \sum_{k=1}^r f_{ik} \leq 1$$

- Each sample must be classified at least once

$$\forall j \sum_{k=1}^r s_{jk} \leq 1$$

Additional Constraints

- Each class must contain at least one feature

$$\forall k \sum_{i=1}^m f_{ik} \geq 1$$

- Each class must contain at least one sample

$$\forall k \sum_{j=1}^n s_{jk} \geq 1$$

Objective Function

- We formulate the biclustering problem with feature selection and outlier detection as an **optimization task** and use the objective function to minimize the information loss. In other words the goal is to select as many features and samples as possible while at the same time satisfying constraints on **biclustering consistency**. The objective function may be expressed as

$$\max \quad m \cdot \sum_{k=1}^r \sum_{j=1}^n s_{jk} + n \cdot \sum_{k=1}^r \sum_{i=1}^m f_{ik}$$

Random Data Simulation Results

- We studied the existence of large biclustering patterns in random data sets ($n = 30$ and $m = 30$).
- One would expect that such patterns would be **extremely rare** due to the fact that consistent biclustering criterion is rather strong.

Random Data Simulation Results

- **Surprisingly**, the numerical experiments showed that for a small number of classes ($r \leq 3$) the checkerboard pattern can be obtained on the basis of almost entire data set (in the case of $r = 2$), or at least on the basis of a half of the data set ($r = 3$).

Random Data Simulation Results

- This results questions the general value of unsupervised biclustering techniques **with a small number of classes**. Unless some specific strongly expressed pattern exists in the data, unsupervised biclustering with a small number of classes can find any partitioning of the data set with no relevance to the phenomenon of interest.

Challenges

- This formulation is currently **computationally intractable** for data sets with more few hundred samples/features.
- **New methods for solving fractionally constrained 0–1 optimization problems ?!**

Alternative Computational Approach

Similarly to such clustering algorithms as k -means and SOM, we can try to achieve consistent biclustering by an iterative process.

- 1 Start from a random partition of samples into k groups.
- 2 Put each feature into the class where its average expression value is largest with respect to the partition of samples.
- 3 Put each sample into the class where its average expression value is largest with respect to the partition of features.
- 4 If at least one sample or feature is moved, go to 2.

Alternative Computational Approach

- The convergence of the procedure is not guaranteed, but in some instances it delivers plausible result.
- The procedure cannot perform feature selection and outlier detection explicitly but some of the created clusters may be easily recognized as “junk” if their separation is weak.
- On the HuGE dataset, the procedure clearly designates classes BRA, LI, and MU.

HuGE index data set: Unsupervised Result

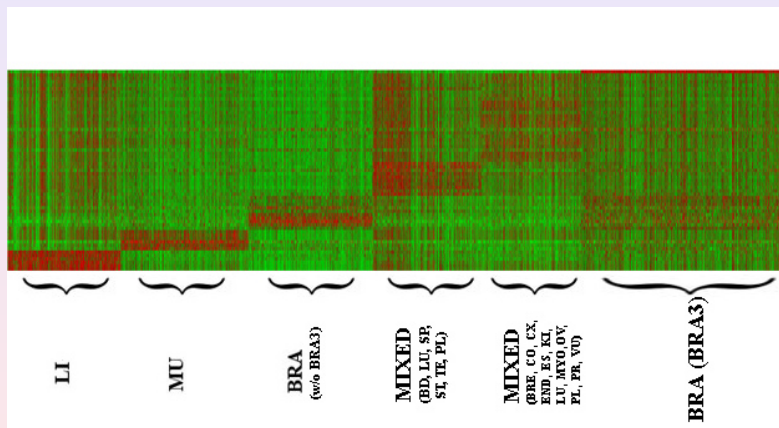


Figure: HuGE Index heatmap.

Conclusions-I

- We proposed a data mining methodology that utilizes both sample and feature patterns, is able to perform feature selection, classification, and unsupervised learning.
- In contrast to other biclustering schemes, consistent biclustering is justified by the conic separation property.

Conclusions-II

- The obtained fractional 0-1 programming problem for supervised biclustering is tractable via a relaxation-based heuristic. The method requires from the user to provide only 1 parameter (t , a class separation parameter), that is particularly attractive for biomedical researchers who are not experts in data mining.
- The consistent biclustering framework is also viable for unsupervised learning, though the fractional 0-1 programming formulation becomes intractable for real-life datasets. Alternative approaches are possible.

Conclusions-III

- A general challenge for data mining research is not to be “fooled by randomness”. That is, revealed patterns should have a negligible probability to appear in random data. Unfortunately, it is not the case for unsupervised clustering into a small number of classes.