
Journal of Combinatorial Optimization, 10, 7–21, 2005

Feature Selection for Consistent Biclustering via Fractional 0–1 Programming*

STANISLAV BUSYGIN busygin@ufl.edu
OLEG A. PROKOPYEV oap4ripe@ufl.edu
Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611

PANOS M. PARDALOS pardalos@ufl.edu
Department of Industrial and Systems Engineering, Biomedical Engineering Program, McKnight Brain Institute, University of Florida, Gainesville, FL 32611

Abstract. Biclustering consists in simultaneous partitioning of the set of samples and the set of their attributes (features) into subsets (classes). Samples and features classified together are supposed to have a high relevance to each other which can be observed by intensity of their expressions. We define the notion of consistency for biclustering using interrelation between centroids of sample and feature classes. We prove that consistent biclustering implies separability of the classes by convex cones. While previous works on biclustering concentrated on unsupervised learning and did not consider employing a training set, whose classification is given, we propose a model for supervised biclustering, whose consistency is achieved by feature selection. The developed model involves solution of a fractional 0–1 programming problem. Preliminary computational results on microarray data mining problems are reported.

Keywords: feature selection, biclustering, classification, supervised learning, microarrays

1. Introduction

Let a data set of n samples and m features be given as a rectangular 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

$$\begin{aligned} &\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_r, \quad \mathcal{S}_k \subseteq \{1 \ldots n\}, \quad k = 1 \ldots r, \\ &\mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots \cup \mathcal{S}_r = \{1 \ldots n\}, \\ &\mathcal{S}_k \cap \mathcal{S}_\ell = \emptyset, \quad k, \ell = 1 \ldots r, \quad k \neq \ell. \end{aligned}$$

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

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

*This research work was partially supported by NSF, NIH and AirForce grants.

$$\mathcal{F}_1 \cup \mathcal{F}_2 \cup \ldots \cup \mathcal{F}_r = \{1 \ldots m\},$$
$$\mathcal{F}_k \cap \mathcal{F}_\ell = \emptyset, \quad k, \ell = 1 \ldots r,\ k \neq \ell,$$

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

Co-clustering of samples and features has been considered in a number of works, among which we should mention biclustering of expression data investigated by Cheng and Church (2000), a paper of I.S. Dhillon on textual biclustering using bipartite spectral graph partitioning (Dhillon, 2001), double conjugated clustering algorithm by Busygin et al. (2002), and spectral biclustering of microarray data by Kluger et al. (2003). However, all these works are dealing with unsupervised learning and do not allow one to incorporate information provided by training data.

The correspondence between classes of samples and features becomes evident once they are sorted according to the classification and represented graphically as a heatmap with a "checkerboard" pattern. In the figure 1, it is easy to identify two classes of samples and

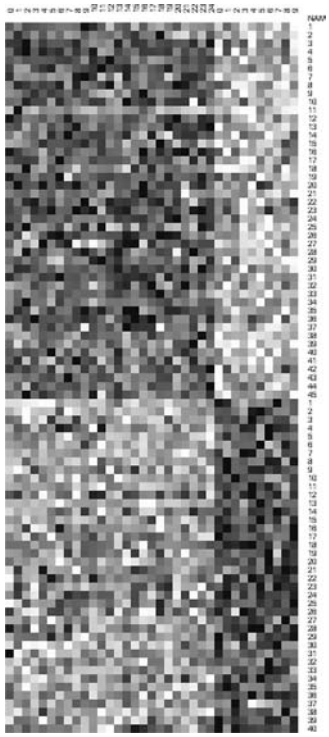

Figure 1. Partitioning of samples and features into 2 classes.

features corresponding to each other by red areas with predominantly red pixels (in the black-and-white figure 1 red pixels correspond to darker ones).

Biclustering has a great significance for biomedical applications. Performing it with high reliability, we are able not only to diagnose conditions represented by sample classes, but also identify features (e.g., genes or proteins) responsible for them, or serving as their markers. We generally understand that the quality of a clustering can be determined by closeness of samples inside classes and their distinguishability between classes according to some appropriate similarity measure. However, how to determine required properties of biclusters, i.e., the pairs $(\mathcal{S}_k, \mathcal{F}_k)$ of the sample and feature subsets that we bind together? In order to answer this question, in this paper we develop the notion of consistency of biclustering and show its application to data mining in biomedicine on two practical data sets.

2. Consistent biclustering

Let each sample be already assigned somehow to one of the classes $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_r$. 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 $C = (c_{ik})_{m\times r}$:

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

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

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 ${c_{i\hat{k}}}^1$:

$$i \in \mathcal{F}_{\hat{k}} \;\Rightarrow\; \forall k = 1 \ldots r, \quad k \neq \hat{k} : \; c_{i\hat{k}} > c_{ik} \tag{2}$$

Now, provided the classification of all features into classes $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_r$, let us construct a classification of samples using the same principle of maximal average expression and see whether we will arrive at the same classification as the initially given one. To do this, 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 $D = (d_{jk})_{n\times r}$:

$$D = A^T F(F^T F)^{-1}, \tag{3}$$

whose k-th column represents the centroid of the class $\mathcal{F}_k$. The condition on sample classification we need to verify is

$$j \in \mathcal{S}_{\hat{k}} \;\Rightarrow\; \forall k = 1 \ldots r, \quad k \neq \hat{k} : \; d_{j\hat{k}} > d_{jk} \tag{4}$$

Let us state now the definition of biclustering and its consistency formally.

Definition 1. A *biclustering* of a data set is a collection of pairs of sample and feature subsets $\mathcal{B} = ((\mathcal{S}_1, \mathcal{F}_1), (\mathcal{S}_2, \mathcal{F}_2), \ldots, (\mathcal{S}_r, \mathcal{F}_r))$ such that the collection $(\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_r)$ forms a partition of the set of samples, and the collection $(\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_r)$ forms a partition of the set of features.

Definition 2. A biclustering $\mathcal{B}$ will be called *consistent* if both relations (2) and (4) hold, where the matrices C and D are defined as in (1) and (3).

We will also say that a data set is *biclustering-admitting* if some consistent biclustering for it exists. Furthermore, 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.

Next, we will show that a consistent biclustering implies separability of the classes by convex cones. Further we will denote j-th sample of the data set by $a_{\cdot j}$ (which is the j-th column of the matrix A), and i-th feature by $a_{i\cdot}$ (which is the i-th row of the matrix A).

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

Similarly, there exist convex cones $\mathcal{Q}_1, \mathcal{Q}_2, \ldots, \mathcal{Q}_r \subseteq \mathbb{R}^n$ such that all features from $\mathcal{F}_k$ belong to the cone $\mathcal{Q}_k$ and no other feature belongs to it, $k = 1 \ldots r$.

Proof: Let $\mathcal{P}_k$ be the conic hull of the samples of class $\mathcal{S}_k$, that is, a vector $x \in \mathcal{P}_k$ if and only if it can be represented as

$$x = \sum_{j \in \mathcal{S}_k} \gamma_j a_{\cdot j},$$

where all $\gamma_j \geq 0$. Obviously, $\mathcal{P}_k$ is convex and all samples of the class $\mathcal{S}_k$ belong to it. Now, suppose there is a sample $\hat{j} \in \mathcal{S}_\ell$, $\ell \neq k$ that belongs to the cone $\mathcal{P}_k$. Then there exists representation

$$a_{\cdot \hat{j}} = \sum_{j \in \mathcal{S}_k} \gamma_j a_{\cdot j},$$

where all $\gamma_j \geq 0$. Next, consistency of the biclustering implies that in the matrix of feature centroids D, the component $d_{\hat{j}\ell} > d_{\hat{j}k}$. This implies

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

Plugging in $a_{i\hat{j}} = \sum_{j \in \mathcal{S}_k} \gamma_j a_{ij}$, we obtain

$$\frac{\sum_{i \in \mathcal{F}_\ell} \sum_{j \in \mathcal{S}_k} \gamma_j a_{ij}}{|\mathcal{F}_\ell|} > \frac{\sum_{i \in \mathcal{F}_k} \sum_{j \in \mathcal{S}_k} \gamma_j a_{ij}}{|\mathcal{F}_k|}$$

Changing the order of summation,

$$\sum_{j\in\mathcal{S}_k}\gamma_j\left(\frac{\sum_{i\in\mathcal{F}_\ell}a_{ij}}{|\mathcal{F}_\ell|}\right)>\sum_{j\in\mathcal{S}_k}\gamma_j\left(\frac{\sum_{i\in\mathcal{F}_k}a_{ij}}{|\mathcal{F}_k|}\right),$$

or

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

On the other hand, for any $j\in\mathcal{S}_k$, the biclustering consistency implies $d_{j\ell}<d_{jk}$, that contradicts to the obtained inequality. Hence, the sample $\hat{j}$ cannot belong to the cone $\mathcal{P}_k$.

Similarly, we can show that the stated conic separability holds for the classes of features. □

It also follows from the proved conic separability that convex hulls of classes are separated, i.e, they do not intersect.

3. 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. In such a setup, a training set of samples is supplied along with its classification known *á priori*, and classification of additional samples, constituting the test set, has to be performed. That is, 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. 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. In this paper, 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.

Assuming that we are given the training set $A=(a_{ij})_{m\times n}$ with the classification of samples into classes $\mathcal{S}_1,\mathcal{S}_2,\ldots,\mathcal{S}_r$, we are able to construct the corresponding classification of features according to (2). Now, if the obtained biclustering is not consistent, our goal is to exclude some features from the data set so that the biclustering with respect to the residual feature set is consistent.

Formally, let us introduce a vector of 0–1 variables $x=(x_i)_{i=1\ldots m}$ and consider the i-th feature selected if $x_i=1$. The condition of biclustering consistency (4), when only the selected features are used, becomes

$$\frac{\sum_{i=1}^m a_{ij}f_{i\hat{k}}x_i}{\sum_{i=1}^m f_{i\hat{k}}x_i}>\frac{\sum_{i=1}^m a_{ij}f_{ik}x_i}{\sum_{i=1}^m f_{ik}x_i},\quad \forall j\in\mathcal{S}_{\hat{k}},\ \hat{k},k=1\ldots r,\quad \hat{k}\neq k. \tag{5}$$

We will use the fractional relations (5) 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

$$\max \sum_{i=1}^{m} x_i \tag{6}$$

The optimization problem (6), (5) is a specific type of *fractional 0–1 programming problem*, which we discuss in the next section.

4. Fractional 0–1 programming

Fractional 0–1 programming problem (or hyperbolic 0–1 programming problem) is defined as follows:

$$\max_{x\in\{0,1\}^m} f(x) = \sum_{j=1}^{n} \frac{\alpha_{j0} + \sum_{i=1}^{m} \alpha_{ji} x_i}{\beta_{j0} + \sum_{i=1}^{m} \beta_{ji} x_i}, \tag{7}$$

where it is usually assumed that for all j and $x \in \{0, 1\}^m$ the denominators in (7) are positive, i.e. $\beta_{j0} + \sum_{i=1}^{m} \beta_{ji} x_i > 0$.

Problem (7) is known to be NP-hard (Prokopyev et al., 2005a). For more information on complexity issues of fractional 0–1 programming problems we refer the reader to Prokopyev et al. (2005a, 2005b).

Applications of *constrained* and *unconstrained* versions of problem (7) arise in numerous areas including but not limited to scheduling (Saipe, 1975), query optimization in data bases and information retrieval (Hansen et al., 1991), and p-choice facility location (Tawarmalani et al., 2002).

Generally, in the framework of fractional 0–1 programming we consider problems, where we optimize a multiple-ratio fractional 0–1 function of type (7) subject to a set of linear constraints. Algorithms for solving problem (7) include linearization techniques (Prokopyev et al., 2005b; Tawarmalani et al., 2002; Wu, 1997), branch and bound methods (Tawarmalani et al., 2002), network-flow (Picard and Queyranne, 1982) and approximation (Hashizume et al.) approaches.

In this paper we define *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. More formally, we define the following problem:

$$\max_{x\in\{0,1\}^m} g(x) = \sum_{i=1}^{m} w_i x_i \tag{8}$$

$$\text{s.t.} \sum_{j=1}^{n_s} \frac{\alpha_{j0}^s + \sum_{i=1}^m \alpha_{ji}^s x_i}{\beta_{j0}^s + \sum_{i=1}^m \beta_{ji}^s x_i} \geq p_s, \quad s = 1, \ldots, S, \tag{9}$$

where S is the number of fractional constraints, and we also assume that for all s, j and $x \in \{0, 1\}^m$ denominators in (9) are positive, i.e. $\beta_{j0}^s + \sum_{i=1}^m \beta_{ji}^s x_i > 0$. This problem is clearly NP-hard since linear 0–1 programming is a special class of problem (8)–(9) if $\beta_{ji}^s = 0$ and $\beta_{j0}^s = 1$ for $j = 1, \ldots, n_s$, $i = 1, \ldots, m$ and $s = 1, \ldots, S$.

A typical approach for solving problem (7) is to reformulate it as a linear mixed 0–1 programming problem, which can be addressed using standard linear programming solvers like CPLEX (ILOG Inc, 2004). For more detailed information on possible linearization methods for fractional 0–1 programming problems we can refer to Prokopyev et al. (2005b), Tawarmalani et al. (2002) and Wu (1997). Fortunately, a similar technique can be also applied to problem (8)–(9).

The linearization approach discussed next is based on a very simple idea:

Theorem 2. *A polynomial mixed* 0–1 *term* $z = xy$, *where* x *is a* 0–1 *variable*, *and* y *is a continuous variable taking any positive value*, *can be represented by the following linear inequalities*: (1) $y - z \leq M - Mx$; (2) $z \leq y$; (3) $z \leq Mx$; (4) $z \geq 0$, *where* M *is a large number greater than* y.

A simple proof of this result can be found in Wu (1997).

Next define a set of new variables y_j^s such that

$$y_j^s = \frac{1}{\beta_{j0}^s + \sum_{i=1}^m \beta_{ji}^s x_i}, \tag{10}$$

where $j = 1, \ldots, n_s$, and $s = 1, \ldots, S$. Since we assume that all denominators are positive, condition (10) is equivalent to

$$\beta_{j0}^s y_j^s + \sum_{i=1}^m \beta_{ji}^s x_i y_j^s = 1. \tag{11}$$

In terms of new variables y_j^s problem (8)–(9) can be rewritten as

$$\max_{x \in \{0,1\}^m} \quad g(x) = \sum_{i=1}^m w_i x_i \tag{12}$$

$$\text{s.t.} \sum_{j=1}^{n_s} \alpha_{j0}^s y_j^s + \sum_{j=1}^{n_s} \sum_{i=1}^m \alpha_{ji}^s x_i y_j^s \geq p_s, \quad s = 1, \ldots, S, \tag{13}$$

$$\beta_{j0}^s y_j^s + \sum_{i=1}^m \beta_{ji}^s x_i y_j^s = 1, \quad j = 1, \ldots, n_s, \quad s = 1, \ldots, S. \tag{14}$$

In order to obtain a linear mixed 0–1 formulations, nonlinear terms $x_i y_j^s$ in (13) and (14) can be linearized introducing additional variables z_{ij}^s and applying the results of Theorem 2. The number of new variables y_j^s and z_{ij}^s is $(m+1)\sum_{s=1}^{S} n_s$.

5. Algorithm for biclustering

To linearize the fractional 0–1 program (6), (5), we should introduce according to (10) the variables

$$y_k = \frac{1}{\sum_{i=1}^{m} f_{ik} x_i}, \ k = 1 \ldots r. \tag{15}$$

Since f_{ik} can take values only zero or one, Eq. (15) can be equivalently rewritten as

$$\sum_{i=1}^{m} f_{ik} x_i \geq 1, \quad k = 1 \ldots r. \tag{16}$$

$$\sum_{i=1}^{m} f_{ik} x_i y_k = 1, \quad k = 1 \ldots r. \tag{17}$$

In terms of the new variables y_k, condition (5) is replaced by

$$\sum_{i=1}^{m} a_{ij} f_{i\hat{k}} x_i y_{\hat{k}} > \sum_{i=1}^{m} a_{ij} f_{ik} x_i y_k \ \forall j \in \mathcal{S}_{\hat{k}}, \quad \hat{k}, k = 1 \ldots r, \quad \hat{k} \neq k. \tag{18}$$

Next, observe that the term $x_i y_k$ is present in (18) if and only if $f_{ik} = 1$, i.e., $i \in \mathcal{F}_k$. So, there are totally only m of such products in (18), and hence we can introduce m variables $z_i = x_i y_k$, $i \in \mathcal{F}_k$ to linearize the system by Theorem 2. Obviously, the parameter M can be set to 1. So, instead of (17) and (18), we have the following constraints:

$$\sum_{i=1}^{m} f_{ik} z_i = 1, \quad k = 1 \ldots r. \tag{19}$$

$$\sum_{i=1}^{m} a_{ij} f_{i\hat{k}} z_i > \sum_{i=1}^{m} a_{ij} f_{ik} z_i \ \forall j \in \mathcal{S}_{\hat{k}}, \quad \hat{k}, k = 1 \ldots r, \quad \hat{k} \neq k. \tag{20}$$

$$y_k - z_i \leq 1 - x_i, \quad z_i \leq y_k, \quad z_i \leq x_i, \quad z_i \geq 0, \quad i \in \mathcal{F}_k. \tag{21}$$

Unfortunately, while the linearization by Theorem 2 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. Hence, we have developed an alternative approach to solving the problem (6), (5) via mixed 0–1 programming, which

is similar by the main idea to the method for solving specific fractional 0–1 programming problems described in Picard and Queyranne (1982).

Consider the meaning of variables z_i. We have introduced them so that

$$z_i = \frac{x_i}{\sum_{\ell=1}^{m} f_{\ell k} x_\ell}, \quad i \in \mathcal{F}_k. \tag{22}$$

Thus, for $i \in \mathcal{F}_k$, z_i is the reciprocal of the cardinality of the class $\mathcal{F}_k$ after the feature selection, if the i-th feature is selected, and 0 otherwise. This suggests that z_i is also a binary variable by nature as x_i is, but its nonzero value is just not set to 1. That value is not known unless the optimal sizes of feature classes are obtained. However, knowing z_i is sufficient to define the value of x_i, and the system of constraints with respect only to the continuous variables $0 \le z_i \le 1$ constitutes a linear relaxation of the biclustering constraints (5). Furthermore it can be strengthened by the system of inequalities connecting z_i to x_i. Indeed, if we know that no more than m_k features can be selected for class $\mathcal{F}_k$, then it is valid to impose:

$$x_i \le m_k z_i, \quad x_i \ge z_i, \quad i \in \mathcal{F}_k. \tag{23}$$

We can prove

Theorem 3. *If x^* is an optimal solution to* (6), (5), *and $m_k = \sum_{i=1}^{m} f_{ik} x_i^*$, then x^* is also an optimal solution to* (6), (19), (20), (23).

Proof: Obviously, x^* is a feasible solution to the new program, so we just have to show that (6), (19), (20), (23) cannot have a better solution. Assume such a solution x^{**} exists. Then,

$$\sum_{i=1}^{m} x_i^{**} > \sum_{i=1}^{m} x_i^*,$$

and, therefore, at least for one $k \in \{1 \dots r\}$,

$$\sum_{i=1}^{m} f_{ik} x_i^{**} > \sum_{i=1}^{m} f_{ik} x_i^*.$$

On the other hand, $x_i \le m_k z_i$, and in conjunction with (19) it implies that

$$\sum_{i=1}^{m} f_{ik} x_i^{**} \le \sum_{i=1}^{m} m_k f_{ik} z_i = m_k = \sum_{i=1}^{m} f_{ik} x_i^*.$$

We have obtained a contradiction and, therefore, x^* is an optimal solution to (6), (19), (20), (23). □

Hence, we can use the following iterative heuristic algorithm for feature selection:

Algorithm 1
1. *Assign* $m_k := |\mathcal{F}_k|$, $k = 1 \dots r$.
2. *Solve the mixed* 0–1 *programming formulation using the inequalities* (23) *instead of* (21).
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.*

Another modification of the program (6), (5) that may result in the improvement of quality of the feature selection is strengthening of the class separation by introduction of a coefficient greater than 1 for the right-hand side of the inequality (5). In this case, we improve (5) by the relation

$$\frac{\sum_{i=1}^{m} a_{ij} f_{i\hat{k}} x_i}{\sum_{i=1}^{m} f_{i\hat{k}} x_i} \geq (1+t) \frac{\sum_{i=1}^{m} a_{ij} f_{ik} x_i}{\sum_{i=1}^{m} f_{ik} x_i}, \tag{24}$$

where $t > 0$ is a constant that becomes a parameter of the method (notice also that doing this we have also replaced the strict inequalities by non-strict ones and made the feasible domain closed). In the mixed 0–1 programming formulation, it is achieved by replacing (20) by

$$\sum_{i=1}^{m} a_{ij} f_{i\hat{k}} z_i \geq (1+t) \sum_{i=1}^{m} a_{ij} f_{ik} z_i \ \forall j \in \mathcal{S}_{\hat{k}}, \quad \hat{k}, k = 1 \dots r, \quad \hat{k} \neq k. \tag{25}$$

After the feature selection is done, we perform classification of test samples according to (4). That is, if $b = (b_i)_{i=1\dots m}$ is a test sample, we assign it to the class $\mathcal{S}_{\hat{k}}$ satisfying

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

6. Preliminary computational results

6.1. ALL vs. AML data set

We applied supervised biclustering to a well-researched microarray data set containing samples from patients diagnosed with *acute lymphoblastic leukemia* (ALL) and *acute myeloid leukemia* (AML) diseases (Golub et al., 1999). It has been the subject of a variety of research papers, e.g. (Ben-Dor et al., 2000, 2001; Weston et al. 2001; Xing and Karp 2001). This data set was also used in the CAMDA data contest (CAMDA, 2001). It is divided into two parts—the training set (27 ALL, 11 AML samples), and the test set (20 ALL, 14 AML samples). According to the described methodology, we performed feature selection

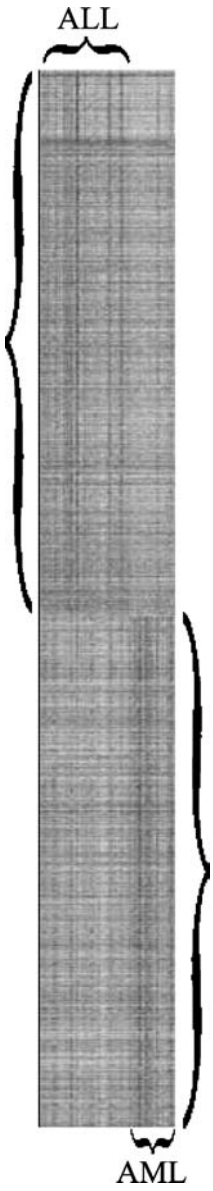

Figure 2. ALL vs. AML heatmap.

for obtaining a consistent biclustering using the training set, and the samples of the test set were subsequently classified choosing for each of them the class with the highest average feature expression. The parameter of separation $t = 0.1$ was used. The algorithm selected 3439 features for class ALL and 3242 features for class AML. The obtained classification

Table 1. HuGE Index biclustering.

Tissue type	Abbreviation	#samples	#features selected
Blood	BD	1	472
Brain	BRA	11	614
Breast	BRE	2	902
Colon	CO	1	367
Cervix	CX	1	107
Endometrium	ENDO	2	225
Esophagus	ES	1	289
Kidney	KI	6	159
Liver	LI	6	440
Lung	LU	6	102
Muscle	MU	6	532
Myometrium	MYO	2	163
Ovary	OV	2	272
Placenta	PL	2	514
Prostate	PR	4	174
Spleen	SP	1	417
Stomach	ST	1	442
Testes	TE	1	512
Vulva	VU	3	186

contains only one error: the AML-sample 66 was classified into the ALL class. To provide the justification of the quality of this result, we should mention that the support vector machines (SVM) approach delivers up to 5 classification errors on the ALL vs. AML data set depending on how the parameters of the method are tuned (Weston et al., 2001). Furthermore, the perfect classification was obtained only with one specific set of values of the parameters.

The heatmap for the constructed biclustering is presented in figure 2.

6.2. HuGE index data set

Another computational experiment that we conducted was on feature selection for consistent biclustering of the Human Gene Expression (*HuGE*) Index data set (HuGE Index.org Website). 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. We refer the reader to Hsiao et al. (2001) for the detailed description of these studies. The data set consists of 59 samples from 19 distinct tissue types. It was obtained using oligonucleotide microarrays capturing 7070 genes. The samples were obtained from 49 human individuals: 24 males with median age of 63 and 25 females with median age of 50. Each sample came from a different individual except

Figure 3. HuGE Index heatmap.

for first 7 BRA samples that were from the different brain regions of the same individual and 5th LI sample, which came from that individual as well. We applied to the data set Algorithm 1 with the parameter of separation $t = 0.1$.

The obtained biclustering is summarized in Table 1 and its heatmap is presented in figure 3. The distinct block-diagonal pattern of the heatmap evidences the high quality of the obtained feature classification. We also mention that the original studies of HuGE Index data set in Hsiao et al. (2001) were performed without 6 of the available samples: 2 KI samples, 2 LU samples, and 2 PR samples were excluded because their quality was too poor for the statistical methods used. Nevertheless, we may observe that none of them distorts the obtained biclustering pattern, which confirms the robustness of our method.

7. Conclusions and future research

We have developed a new optimization framework to perform supervised biclustering with feature selection. It has been proved that the obtained partitions of samples and features of the data set satisfy a conic separation criterion of classification. Though the constructed fractional 0–1 programming formulation may be hard to tackle with direct solving methods, it admits a good linear continuous relaxation. Preliminary computational results show that tightening it iteratively with valid inequalities linking the continuous and 0–1 variables, we are able to obtain a good heuristic solution providing a reliable feature selection and the test set classification based on it. We also note that in contrast to many other data mining methodologies the developed algorithm involves only one parameter that should be defined by the user.

Further research work should reveal more properties relating solutions of the linear relaxation to solutions of the original fractional 0–1 programming problem. This should allow for more grounded choices of the class separation parameter t for feature selection and better solving methods. It is also interesting to investigate whether the problem (6) subject to (5) itself is *NP*-hard.

Note

1. Taking into account that in real-life data mining applications all data are fractional values, whose accuracy is not perfect, we may disregard the case when this maximum is not unique. However, for the sake of theoretical purity we further assume that if the ambiguity in classification occurs, we apply a negligible perturbation to the data set values and start the procedure anew.

References

A. Ben-Dor, L. Bruhn, I. Nachman, M. Schummer, and Z. Yakhini, "Tissue classification with gene expression profiles," *Journal of Computational Biology*, vol. 7, pp. 559–584, 2000.

A. Ben-Dor, N. Friedman, and Z. Yakhini, "Class discovery in gene expression data," in *Proc. Fifth Annual Inter. Conf. on Computational Molecular Biology (RECOMB)*, 2001.

S. Busygin, G. Jacobsen, and E. Krämer, "Double conjugated clustering applied to leukemia microarray data," *SDM 2002 Workshop on Clustering High Dimensional Data and its Applications*, 2002.

Y. Cheng and G.M. Church, "Biclustering of expression data," in: *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology*, 2000, pp. 93–103.

I.S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in: *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD)*, August 26–29, 2001, San Francisco, CA.

P. Hansen, M. Poggi de Aragão, and C.C. Ribeiro, "Hyperbolic 0–1 programming and query optimization in information retrieval," *Math. Program.*, vol. 52, pp. 256–263, 1991.

S. Hashizume, M. Fukushima, N. Katoh, and T. Ibaraki, "Approximation algortihms for combinatorial fractional programming problems," *Mathematical Programming*, vol. 37, pp. 255–267.

L.-L. Hsiao, F. Dangond, T. Yoshida, R. Hong, R.V. Jensen, J. Misra, W. Dillon, K.F. Lee, KE. Clark, P. Haverty, Z. Weng, G. Mutter, M.P. Frosch, M.E. MacDonald, E.L. Milford, C.P. Crum, R. Bueno, R.E. Pratt, M. Mahadevappa, J.A. Warrington, G. Stephanopoulos, G. Stephanopoulos, and S.R. Gullans, "A compendium of gene expression in normal human tissues," *Physiol. Genomics*, vol. 7, pp. 97–104, 2001.

T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, pp. 531–537, 1999.

Y. Kluger, R. Basri, J.T. Chang, and M. Gerstein, "Spectral biclustering of microarray data: Coclustering genes and conditions," *Genome Res*, vol. 13, pp. 703–716, 2003.

J.-C. Picard and M. Queyranne, "A network flow solution to some nonlinear 0–1 programming problems, with applications to graph theory," *Networks*, vol. 12, pp. 141–159, 1982.

O.A. Prokopyev, H.-X. Huang, and P.M. Pardalos, "On complexity of unconstrained hyperbolic 0–1 programming problems," *Oper. Res. Lett*., vol. 33, pp. 312–318, 2005a.

O.A. Prokopyev, C. Meneses, C.A.S. Oliveira, and P.M. Pardalos, "On multiple-ratio hyperbolic 0–1 programming problems," to appear in *Pacific Journal of Optimization*, 2005b.

S. Saipe, "Solving a (0,1) hyperbolic program by branch and bound," *Naval Res. Logist. Quarterly*, vol. 22, pp. 497–515, 1975.

M. Tawarmalani, S. Ahmed, and N. Sahinidis, "Global optimization of 0–1 hyperbolic programs," *J. Global Optim.*, vol. 24, pp. 385–416, 2002.

J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, Feature selection for SVMs. NIPS, 2001.

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

E.P. Xing and R.M. Karp "CLIFF: Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts," *Bioinformatics Discovery Note*, vol. 1, pp. 1–9, 2001.

CAMDA 2001 Conference. `http://bioinformatics.duke.edu/camda/camda01/`.

HuGE Index.org Website. `http://www.hugeindex.org`.

ILOG Inc. CPLEX 9.0 User's Manual, 2004.