Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 269–274

ISBN 978-83-60810-22-4

Extending the definition of β-consistent biclustering for feature selection

Antonio Mucherino‡
‡CERFACS, Toulouse, France.
mucherino@cerfacs.fr

***Abstract*—Consistent biclusterings of sets of data are useful for solving feature selection and classification problems. The problem of finding a consistent biclustering can be formulated as a combinatorial optimization problem, and it can be solved by the employment of a recently proposed VNS-based heuristic. In this context, the concept of β-consistent biclustering has been introduced for dealing with noisy data and experimental errors. However, the given definition for β-consistent biclustering is coherent only when sets containing non-negative data are considered. This paper extends the definition of β-consistent biclustering to negative data and shows, through computational experiments, that the employment of the new definition allows to perform better classifications on a well-known test problem.**

I. Introduction

CLASSIFICATION problems in data mining aim at finding a suitable partition of the samples contained in a certain set of data. Various classification techniques have been proposed over the last years, and they have been applied to various problems arising in applied fields [8], [13]. Recently, a new approach for classification have been proposed in [2], [3], which is based on the concept of *consistent biclustering*. Samples and features of a set of data are organized on the columns and on the rows, respectively, of a matrix, and a partition in biclusters of this matrix (the so-called *biclustering*) can be found with the aim of performing supervised classifications. If the biclustering is *consistent*, then the knowledge acquired by finding the biclustering can be exploited for performing good-quality classifications (see Section II for more details).

When real data are considered, i.e. data obtained by experimental techniques, the matrix representing the whole set of data does not usually admit any consistent biclustering. This may be due to the fact that some of the features which are considered for describing the samples are actually not pertinent to the problem. A way to overcome to this issue is then to remove all these features from the set of data. During this process, however, useful features should not be discarded [2].

Since experimentally obtained data are usually noisy, even small errors introduced in the set of data may cause the loss of the consistency of the found biclusterings. This issue has been firstly addressed in [2] and, successively, the concepts of α-consistent biclustering and β-consistent biclustering have been introduced in [15] with the aim of efficiently managing noisy data and errors. When looking for biclusterings satisfying the α-consistency or the β-consistency property, a larger number of features (depending on the parameters α and β) need to be discarded from the set of data, because all the features that are sensitive to experimental errors are supposed to be identified and removed.

While the α-consistency property helped in the management of errors and noise, biclusterings satisfying the β-consistency property showed instead a weird behavior [12], [15]. While larger α values allowed for finding α-consistent biclusterings in which *better* features were selected, so that better-quality classifications were actually possible by exploiting the biclustering, β-consistent biclusterings with larger β values did not follow this general trend. In fact, the definition of β-consistent biclustering given in [15] is coherent only if the considered set contains non-negative data only. The aim of this paper is therefore to extend the definition of β-consistent biclustering to negative data. The definition given in this paper actually allows for finding correct β-consistent biclusterings even when negative data are available, as it is usually the case when real-life problems are considered.

The rest of the paper is organized as follows. Section II will provide some more details about how to find consistent biclusterings and how to use this knowledge for performing supervised classifications. Section III will briefly describe a recently proposed heuristic for the solution of the combinatorial optimization problem for the identification of consistent biclusterings of training sets. Then, the concept of β-consistent biclustering will be deeply discussed. In Section IV, an extended version of its formal definition will be presented. In Section V, a well-known set of data will be scaled so that it only contains non-negative entries, and an existing algorithm will be employed for finding β-consistent biclusterings. The experiments show that better classifications can be obtained by exploiting these new biclusterings. Results show that the new β-consistent biclusterings allow to obtain classifications with no misclassified elements, as it was instead the case in previous works. Section VI concludes the paper.

II. Consistent Biclustering for Supervised Classifications

Let $A \equiv \{a_{ij}\}$ be an $m \times n$ matrix representing a set of data. The matrix A contains n samples (column by column) and m features (row by row). A *bicluster* is a submatrix of A, which is able to group together a subset of samples (a class S_r) and a subset of features (a class F_r) of the set of data. Finally, a *biclustering*

$$B = \{(S_1, F_1), (S_2, F_2), \ldots, (S_k, F_k)\}$$

is a partition of A in disjoint biclusters which covers A, i.e. the following conditions must be satisfied:

$$\bigcup_{r=1}^{k} S_r \equiv A, \qquad S_\zeta \cap S_\xi = \emptyset \quad 1 \leq \zeta \neq \xi \leq k,$$

$$\bigcup_{r=1}^{k} F_r \equiv A, \qquad F_\zeta \cap F_\xi = \emptyset \quad 1 \leq \zeta \neq \xi \leq k,$$

where $k \leq \min(n, m)$ is the considered number of biclusters [2].

Biclusterings of sets of data are usually searched by unsupervised techniques, where it is supposed that no information about the data is available. The interested reader can refer to [10] for a recent survey. In this approach, it is instead supposed that the set of data A is a training set, i.e. A is a set for which the classification of its samples is already known. The corresponding biclustering is therefore computed by employing a supervised technique, and the found biclustering is then exploited for classifying samples having no known classification.

Let us suppose that A is a training set. Therefore, the classification of its samples in k classes is known. From this classification, it is possible to construct a classification of its features in k classes. The basic idea is to assign each feature to the class $F_{\hat{r}}$ (with $\hat{r} \in \{1, 2, \ldots, k\}$) such that it is mostly expressed (i.e. *it has higher value*), in average, in the class of samples $S_{\hat{r}}$. The reader is referred to [2], [12], [15] for details about this supervised procedure. Note that the same procedure can be inverted and it can be used for finding a classification of the samples of A from a known classification of its features.

By combining the two classifications, the one for the samples of A and the other one for the features of A, a biclustering can be defined for the matrix A. The supervised procedure mentioned above can construct classifications of the samples from classifications of the features, and vice versa. If the biclustering remains unchanged when the supervised procedure is applied, then it is said to be *consistent*. In other words, the biclustering is consistent if the classification of the samples (the features) suffices for correctly reconstructing the biclustering.

Consistent biclusterings can be very useful for performing supervised classifications. Let $\hat{A}$ be a set of data which is not a training set and that it is related to the same classification problem as the set A. No information regarding the classification of the samples in $\hat{A}$ is available, but $\hat{A}$ contains the same features of A and a classification of these features is known because a biclustering for A is available. By using the supervised procedure, then, a classification for the samples of $\hat{A}$ can be found from the known classification of its features. Since the biclustering of A is consistent, the procedure is able to find the correct classification for the samples in A, and therefore it should be able to do most likely the same for the samples in $\hat{A}$ [2].

Let f_{ir} be a binary parameter which indicates if the i^{th} feature belongs to the class F_r of features ($f_{ir} = 1$) or not ($f_{ir} = 0$). Let $x \equiv \{x_1, x_2, \ldots, x_m\}$ be a binary vector of variables, where x_i is 1 if the i^{th} feature of A is selected, and it is 0 otherwise. Let us also indicate with the symbol $A[x]$ the submatrix of A obtained by selecting only the features (rows) of A for which $x_i = 1$.

II.1 Definition
A biclustering for $A[x]$ is consistent if and only if, $\forall \hat{r}, \xi \in \{1, 2, \ldots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$, the following inequality is satisfied [2]:

$$\frac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} > \frac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i}. \tag{1}$$

Note that the two fractions in (1) are used for computing the *centroids* of the considered biclusters (for each sample in $S_{\hat{r}}$, the average over the features beloging to the same class is computed). On the left hand side of (1), the j^{th} component of the centroid of the bicluster $(S_{\hat{r}}, F_{\hat{r}})$ is computed. On the right hand side of (1), the j^{th} component of the centroid of the bicluster $(S_{\hat{r}}, F_\xi)$ is computed. In order to have a consistent biclustering, all components of the centroid of $(S_{\hat{r}}, F_{\hat{r}})$ must have a larger value.

In order to overcome issues related to sets of data containing noisy data, the concepts of α-consistent biclustering and β-consistent biclustering have been introduced in [15]. The basic idea is to artificially increase the margin between the centroids of the different biclusters in the constraints (1). In this way, small variations due to noisy data and errors should not be able to spoil the classifications performed by exploiting the found biclusterings.

II.2 Definition
A biclustering for $A[x]$ is α-consistent if and only if, $\forall \hat{r}, \xi \in \{1, 2, \ldots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$, the following inequality is satisfied [15]:

$$\frac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} > \alpha + \frac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i}, \tag{2}$$

where $\alpha > 0$.

The additive parameter $\alpha > 0$ is used to guarantee that the margin between the centroid of $(S_{\hat{r}}, F_{\hat{r}})$ and any other bicluster concerning $S_{\hat{r}}$ is at least greater than α, independently from the considered data. Similarly, in the case of β-consistent biclustering, a multiplicative parameter β is used.

II.3 Definition
A biclustering for $A[x]$ is β-consistent if and only if, $\forall \hat{r}, \xi \in \{1, 2, \ldots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$, the following inequality is

satisfied [15]:

$$\frac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} > \beta \times \frac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i}, \tag{3}$$

where $\beta > 1$.

Note that different values for the parameters α and β could be used for different components of the centroids. Usually, however, only one value is set up for all the components. More details about α-consistent and β-consistent biclustering are given in Section IV.

In real-life applications, there are usually no biclusterings which are consistent, α-consistent or β-consistent if all the features are selected (this situation corresponds to a binary vector x with all its components equal to 1). As already mentioned in the Introduction, this is consequence of the fact that some of the considered features may not be pertinent. Such features must therefore be removed from the set of data, while the total number of considered features must be maximized in order to lose the minimum amount of information.

To this aim, the following combinatorial optimization problem can be considered:

$$\max_x \left(f(x) = \sum_{i=1}^{m} x_i \right), \tag{4}$$

subject to constraints (1), (2) or (3) depending on the fact that a consistent, α-consistent or β-consistent biclustering, respectively, is searched. These three optimization problems are all NP-hard [9], and different heuristic algorithms have been proposed in order to solve such problems [2], [12], [15]. In previous works, consistent biclusterings have been found for sets of data related to:

- gene expressions of human tissues from healthy and sick (affected by cancer) patients [12], [16];
- patients diagnosed with acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML) diseases [2], [4], [12], [15];
- the Human Gene Expression (HuGE) Index [2], [6], [15];
- wine fermentations [14], [17].

III. A VNS-based Heuristic for Finding Biclusterings

The optimization problems (4)-(1), (4)-(2) and (4)-(3) are combinatorial problems with fractional constraints and binary decision variables. In order to solve these optimization problems, we employ a recently proposed heuristic [12], which is based on a reformulation of these problems as bilevel programs.

Let us introduce new continuous variables

$$y_r,\ r = 1, 2, \ldots, k.$$

Each variable y_r is related to the bicluster (S_r, F_r) of a possible biclustering, and it represents the percentage of features that are selected in that bicluster. The bilevel reformulation can be written for problem (4)-(1) as follows:

$$\min_y \left(g(x,y) = \sum_{r=1}^{k} \left[(1 - y_r) + \sum_{\xi=1:\xi\neq r}^{k} c(x, r, \xi) \right] \right)$$

s.t.:

$$x = \arg\max_x \left(f(x) = \sum_{i=1}^{m} x_i \right)$$

$$\text{s. t.} \begin{cases} \sum_{i=1}^{m} f_{ir} x_i = \lfloor y_r \sum_{i=1}^{m} f_{ir} \rfloor \quad \forall r \in \{1, \ldots, k\} \\ \frac{1}{y_{\hat{r}}} \sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i > \frac{1}{y_\xi} \sum_{i=1}^{m} a_{ij} f_{i\xi} x_i \end{cases}$$

$$\sum_{r=1}^{k} y_r \leq 1,$$

where

$$c(x, \hat{r}, \xi) = \sum_{j \in S_{\hat{r}}} \left| \frac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i} - \frac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} \right|_+,$$

and the symbol $|\cdot|_+$ represents the function which returns its argument if it is positive, and it returns 0 otherwise. Solving this bilevel program is equivalent to solving the original problem (4)-(1). For more details, the reader is referred to [12].

Reformulating a single optimization problem in a bilevel program may not seem convenient, because the complexity of the problem might increase. However, this reformulation allowed the development of an efficient heuristic for the solution of the problem. The inner problem of this bilevel program is linear, and therefore it can be solved by standard methods for linear optimization. The heuristic is inspired by the Variable Neighborhood Search (VNS) [5], [11] and it only acts on the new introduced variables y_r, with $r = 1, 2, \ldots, k$. At each iteration of the algorithm, the inner problem is exactly solved and a set of values for the original variables x_i, with $i = 1, 2, \ldots, m$, is obtained. The main intuition is that the exact solution of the inner problem helps the heuristic in converging towards the desired biclusterings.

Algorithm 1 gives a sketch of this VNS-based heuristic [12]. It is composed by two VNS's which are nested. An adaptive value for the percentage of unselected features, $unsel$, is kept small at the beginning ($unsel \simeq 0$) of the heuristic, and then it increases when no better solutions can be found. In this way, the algorithm firstly tries to find solutions where the number of selected features is high. After, solutions where fewer features are selected are also allowed. For each neighbor of the first VNS, there is a full execution of the second VNS. The neighbors of the second VNS are generated so that the set of variables y_r can be slightly perturbed at the beginning ($range = starting\_range$), and larger perturbations can be performed only when no better solutions can be found by

Algorithm 1 A VNS-based heuristic for feature selection.

let $iter = 0$;
let $x_i = 1$, $\forall i \in \{1, 2, \ldots, m\}$;
let $y_r = \sum_i f_{ir}/m$, $\forall r \in \{1, 2, \ldots, k\}$;
let $y_r^{best} = y_r$, $\forall r \in \{1, 2, \ldots, k\}$;
let $range = starting\_range$;
let $unsel = 0$;
while ((1) unsatisfied and $unsel \leq max\_unsel$) **do**
  while ((1) unsatisfied and $range \leq max\_range$) **do**
    let $iter = iter + 1$;
    solve inner optimization problem (linear & cont.);
    if (constraints (1) unsatisfied) **then**
      increase $range$;
      if (g has improved) **then**
        let $y_r^{best} = y_r$, $\forall r \in \{1, 2, \ldots, k\}$;
        let $range = starting\_range$;
      end if
      let $y_r = y_r^{best}$, $\forall r \in \{1, 2, \ldots, k\}$;
      let r' = random in $\{1, 2, \ldots, k\}$;
      choose randomly $y_{r'}$ in $[y_{r'} - range, y_{r'} + range]$;
      let r'' = random in $\{1, 2, \ldots, k\}$: $r' \neq r''$;
      set $y_{r''}$ so that $1 - unsel \leq \sum_r y_r \leq 1$;
    end if
  end while
  if (constraints (1) unsatisfied) **then**
    increase $unsel$;
  end if
end while

considering the current neighbor. As for all heuristics, there is no guarantee that the biclusterings that are found by the heuristic are the ones with the largest number of selected features. However, multi-start techniques may be for example used for improving the quality of the found solutions.

IV. Extending the Definition of β-consistent Biclustering

The constraints (1) guarantee that all the components of the centroid of $(S_{\hat{r}}, F_{\hat{r}})$ are larger than their homologous components in any other bicluster $(S_{\hat{r}}, F_\xi)$, for any $\xi \in \{1, 2, \ldots, k\}$, with $\xi \neq \hat{r}$. In the case of α-consistent biclustering (see constraints (2)), this requirement is strengthened by introducing a minimum margin between pairs of homologous components of the centroids. This prevents to have the constraints unsatisfied after small variations in the data, and it leads to the following immediate result:

IV.1 Proposition

Any α-consistent biclustering of $A[x]$ (see Definition II.2) is also a consistent biclustering of $A[x]$ (see Definition II.1).

The basic idea behind the β-consistent biclustering is the following. Instead of using an additive parameter α, the multiplicative parameter β is employed, which must be greater than 1. In this case (see constraints (3)), each component of the centroid of $(S_{\hat{r}}, F_{\hat{r}})$ must be larger than β times the value of its homologous component in any other bicluster $(S_{\hat{r}}, F_\xi)$. However, if some of these components are negative, the multiplication by β can give an undesired effect. When this happens, even if the constraints (3) are all satisfied, the original set of constraints (1) may not be satisfied. Therefore, even though the found biclustering may be β-consistent by Definition II.3, it might actually be not even consistent.

The incorrectness of Definition II.3 for β-consistent biclustering is also reflected in the results of the experiments reported in [12], [15]. While the rate of correct classifications constantly increased when α-consistent biclusterings with larger α values were searched, this rate had a *strange* behaviour in correspondence with β-consistent biclusterings with larger β values. After a certain threshold for β, the number of correct classifications performed with the found biclustering started to decrease. Since the considered matrix A contains both positive and negative elements, this phenomenon can most likely be explained by the fact that the found β-consistent biclusterings were actually not consistent.

Consider for example this simple matrix:

$$A = \begin{pmatrix} -2 & 2 \\ -1 & 8 \end{pmatrix}.$$

Suppose that there are two classes of samples and features. Suppose that the two biclusters of the considered biclustering contain, respectively, the element -2 and the element 8 of A. It is very easy to verify that this biclustering is β-consistent with $\beta = 3$, because -2 is greater than $\beta \times (-1)$. However, it is not consistent, because -2 is not greater than -1. This incoherence is due to the fact that A contains negative entries.

The following is the definition of β-consistent biclustering that extends the one given in [15] to matrices A containing negative elements.

IV.2 Definition

A biclustering for $A[x]$ is β-consistent if and only if, $\forall \hat{r}, \xi \in \{1, 2, \ldots, k\}, \hat{r} \neq \xi, j \in S_{\hat{r}}$, the following condition is satisfied:

$$\begin{cases} \dfrac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} > \beta \times \dfrac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i} & \text{if } c > 0 \\ \dfrac{\sum_{i=1}^{m} a_{ij} f_{i\hat{r}} x_i}{\sum_{i=1}^{m} f_{i\hat{r}} x_i} > (2 - \beta) \times \dfrac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i} & \text{if } c < 0 \end{cases} \tag{5}$$

where

$$c = \frac{\sum_{i=1}^{m} a_{ij} f_{i\xi} x_i}{\sum_{i=1}^{m} f_{i\xi} x_i}$$

and $\beta > 1$.

If c is positive, then Definition II.3 is coherent. If c is instead negative, its product by β would produce a decrease of its value. Suppose that $\beta = 1 + \gamma$, with $\gamma > 0$. Then, $\beta \times c$ can be divided in two parts: c itself, and $\gamma \times c$, which represents the *variation* on the original value of c obtained by performing the product. Definition IV.2 is able to correct this variation when c is negative by inverting the sign of $\gamma \times c$. Indeed, when $-\gamma \times c$ is added to c, the multiplicative factor is actually $1 - \gamma$, which corresponds to $2 - \beta$. The following theoretical result can now be stated:

IV.3 Proposition
Any β-consistent biclustering of $A[x]$ (see Definition IV.2) is also a consistent biclustering of $A[x]$ (see Definition II.1).

V. Computational Experiments

The heuristic algorithm discussed in Section III is employed in the following experiments for finding β-consistent biclusterings of sets of data (Definition IV.2) containing a maximal number of selected features. The heuristic has been implemented in AMPL [1], where the ILOG CPLEX11 solver [7] has been used for solving the linear inner problem at each iteration of the VNS-based heuristic. The set of data which is considered in the experiments is related to patients diagnosed with acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML) diseases [4]. All the experiments are carried out on an Intel Core 2 CPU 6400 @ 2.13 GHz with 4GB RAM, running Linux.

The set of data is divided in a training set, which can be used for finding the biclusterings, and a validation set, which can be used for checking the quality of the classifications performed by the exploiting the knowledge acquired by finding the biclusterings (see Section II). The training set contains 38 samples: 27 ALL samples and 11 AML samples. The validation set contains 34 samples: 20 ALL samples and 14 AML samples. The total number of features is 7129. This is a well-known test problem, and experiments on this problem can be found, for example, in [12], [15], where found biclusterings have been employed for performing supervised classifications. In these experiments, the number of misclassifications on the validation set gets higher when the parameter β reaches a certain threshold.

In order to use the heuristic described in Section III without any modification, the training set is scaled in the experiments so that it only contains non-negative data. This practically solves the issue related to negative data, but it does not allow for directly comparing the experiments presented in this paper to those in [12], [15]. In fact, the degree of magnitude of the data changes with scaling, and the values for the parameter β are not comparable anymore. However, the important result which is presented in this paper is that completely correct classifications can be performed with the newly found biclusterings. The β value that allows for obtaining correct classifications in the original scaling of the training set is only a marginal information.

Table I shows the experiments. The total number $f(x)$ of features that are selected in each experiment is reported, together with the number err of misclassifications occurring when the samples of the validation set are classified accordingly with the found β-consistent biclusterings. Moreover, the labels of the misclassified samples (if any) are reported in the last column (each sample is labeled by an integer number between 1 and 34 and by following the ordering in which they are stored in the set of data [4]). Each experiment took no more than 10 minutes of CPU time.

TABLE I
COMPUTATIONAL EXPERIMENTS ON A SET OF SAMPLES FROM PATIENTS DIAGNOSED WITH ALL AND AML DISEASES. THESE ARE THE FIRST EXPERIMENTS IN WHICH THE NUMBER OF MISCLASSIFIED SAMPLES IS 0.

β	$f(x)$	err	*mis. samples*
1.001	7011	2	{3,31}
1.002	6984	2	{3,31}
1.003	6946	1	31
1.004	6702	1	31
1.005	5914	1	31
1.006	5072	1	31
1.007	4524	0	-
1.008	3932	0	-
1.009	3443	0	-
1.010	3033	0	-

When β is rather small, two samples of the validation set, the one labeled by 3 and the one labeled by 31, are misclassified even though the found biclustering is β-consistent. When β increases, fewer samples are misclassified. In particular, when β is equal to 1.007 (notice that this value is strongly related to the new scaling of the training set), and even for larger values, there are no misclassifications on the validation set when the found β-consistent biclustering is employed for performing the classifications. For the very first time, there are no misclassifications on the validation set when its samples are classified accordingly to β-consistent biclusterings of this training set.

VI. Conclusions

This paper extends the definition of β-consistent biclustering previously given in [15]. More accurate classifications can be performed by using this extended definition. The paper presents a theoretical study, which includes a new definition for the β-consistency, as well as an experimental study. A subset of features was found for a well-known classification problem related to gene expression data that allowed for performing classifications with no mistakes.

Acknowledgments

I am grateful to Sonia Cafieri for the fruitful discussions.

References

[1] AMPL, http://www.ampl.com/
[2] S. Busygin, O. A. Prokopyev, P. M. Pardalos, *Feature Selection for Consistent Biclustering via Fractional 0-1 Programming*, Journal of Combinatorial Optimization **10**, 7-21, 2005.
[3] S. Busygin, O. A. Prokopyev, P. M. Pardalos, *Biclustering in Data Mining*, Computers & Operations Research **35**, 2964–2987, 2008.

[4] 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, E. S. Lander, *Molecular Classification of Cancer: Class Discovery and Class Prodiction by Gene Expression Monitoring*, Science **286**, 531–537, 1999.

[5] P. Hansen and N. Mladenovic. *Variable Neighborhood Search: Principles and Applications*, European Journal of Operational Research **130**(3), 449–67, 2001.

[6] L.-L. Hsiao, F. Dangond, T. Yoshida, R. Hong, R. V. Jensen, J. Misra, W. Dillon, K. F. Lee, K.E. Clark, P. Haverty, Z. Weng, G. L. Mutter, M. P. Frosch, M.E. MacDonald, E. L. Milford, C.P. Crum, R. Bueno, R. E. Pratt, M. Mahadevappa, J. A. Warrington, Gr. Stephanopoulos, Ge. Stephanopoulos, S.R. Gullans, *A Compendium of Gene Expression in Normal Human Tissues*, Physiological Genomics **7**, 97-104, 2001.

[7] ILOG, CPLEX, `http://www.ilog.com/products/cplex/`

[8] W. Klosgen, J.M. Zytkow, *Handbook of Data Mining and Knowledge Discovery*, Oxford University Press, 2002.

[9] O. E. Kundakcioglu, P. M. Pardalos, *The Complexity of Feature Selection for Consistent Biclustering*, In: Clustering Challenges in Biological Networks, S. Butenko, P. M. Pardalos, W. A. Chaovalitwongse (Eds.), World Scientific Publishing, 2009.

[10] S. C. Madeira and A. L. Oliveira, *Biclustering Algorithms for Biological Data Analysis: a Survey*, IEEE Transactions on Computational Biology and Bioinformatics **1** (1), 24–44, 2004.

[11] M. Mladenovic and P. Hansen, *Variable Neighborhood Search*, Computers and Operations Research **24**, 1097–1100, 1997.

[12] A. Mucherino, S. Cafieri, *A New Heuristic for Feature Selection by Consistent Biclustering*, arXiv e-print, arXiv:1003.3279v1, March 2010.

[13] A. Mucherino, P. Papajorgji, P. M. Pardalos, *Data Mining in Agriculture*, Springer, 2009.

[14] A. Mucherino, A. Urtubia, *Consistent Biclustering and Applications to Agriculture*, IbaI Conference Proceedings, Proceedings of the Industrial Conference on Data Mining (ICDM10), Workshop on Data Mining and Agriculture (DMA10), Berlin, Germany, 105-113, 2010.

[15] A. Nahapatyan, S. Busygin, and P.M. Pardalos, *An Improved Heuristic for Consistent Biclustering Problems*, In: Mathematical Modelling of Biosystems, R.P. Mondaini and P.M. Pardalos (Eds.), Applied Optimization **102**, Springer, 185–198, 2008.

[16] D. A. Notterman, U. Alon, A.J. Sierk, A. J. Levine, *Transcriptional Gene Expression Profiles of Colorectal Adenoma, Adenocarcinoma, and Normal Tissue Examined by Oligonucleotide Arrays*, Cancer Research **61**, 3124-3130, 2001.

[17] A. Urtubia, J. R. Perez-Correa, A. Soto, P. Pszczolkowski, *Using Data Mining Techniques to Predict Industrial Wine Problem Fermentations*, Food Control **18**, 1512–1517, 2007.