Approximate Bicluster and Tricluster Boxes in the Analysis of Binary Data

Boris G. Mirkin[1,2,⋆] and Andrey V. Kramarenko[1]

[1] National Research University – Higher School of Economics, Moscow, Russia
[2] Department of Computer Science and Information Systems,
Birkbeck University of London, UK
bmirkin@hse.ru, mirkin@dcs.bbk.ac.uk

Abstract. A disjunctive model of box bicluster and tricluster analysis is considered. A least-squares locally-optimal one cluster method is proposed, oriented towards the analysis of binary data. The method involves a parameter, the scale shift, and is proven to lead to "contrast" box bi- and tri-clusters. An experimental study of the method is reported.

Keywords: box, bicluster, tricluster.

1 Introduction

The concept of bicluster emerges when the data relate two different sets of objects to each other so that highly related pairs of subsets, partitions or even hierarchies can be distinguished in each of the sets. This cluster structure was first made explicit by J.Hartigan[2,3] and dubbed as biclustering by B.Mirkin[7]. The concept and corresponding methods gained popularity in several applied areas of which probably the most effective is bioinformatics (see, for example, Madeira and Oliveira[5], Prelic et al.[10]). A somewhat more conservative and mathematically driven approach to establishing relations between a set of objects and a set of attributes was taken in developing the abstract formal model concept (Wille and Ganter[1]). The notion of a formal concept was first developed for binary data matrix $R = (r_{ij}), i \in I, j \in J$, where all r_{ij} are either 1 or 0, which is the case we consider here. A formal concept (V, W), where $V \subseteq I, W \subseteq J$, corresponds to all $r_{ij} = 1$ for $i \in V, j \in W$ in such a way that adding elements to either V or W would break the equation at least at one pair (i, j). This notion is well justified in application to well developed "contexts" R, but seems somewhat rigid when applied to real world datasets. This is why researchers have been trying to relax the notion of formal concepts by admitting some zeros inside the box (V, W) and some unities outside it (see, for example, Pensa and Boullicaut[9]). In this respect, the box clustering approach proposed by Mirkin et al.[6] seems another form of relaxation of the notion of formal concept. Yet the box clustering algorithms proposed in Mirkin et al.[6] are not applicable to the binary data. Therefore, we put a threefold goal for this paper:

[⋆] Corresponding author

S.O. Kuznetsov et al. (Eds.): RSFDGrC 2011, LNAI 6743, pp. 248–256, 2011.

(1) To propose and explore a model and algorithm for biclustering boxes suitable for binary contexts;

(2) Extend it to triclustering of binary data involving three interrelated data objects;

(3) Apply both bi- and tri-clustering algorithms to real world datasets.

2 The Notions of Formal Concept and Box Clustering

A data matrix $R = (r_{ij})$, with $i \in I$ (objects) and $j \in J$ (attributes), such that either $r_{ij} = 1$ or $r_{ij} = 0$ is referred to as a conceptual context. A formal concept is a pair of sets (V, W), that is, a biset, such that $V \subseteq I$, $W \subseteq J$ and

$$r_{ij} = 1 \text{ for all } (i, j) \in V \times W \tag{1}$$

and neither V nor W can be increased without breaking the property (1). The cardinalities will be denoted by #V = n, #W = m.

The condition of all within-entries being non-zero can be too restrictive, especially with noisy data. There have been attempts at modifying both of these conditions by admitting a few zeros inside and most zeros outside (Pensa and Boulicaut[9], Rome and Haralick[11], Ignatov and Kuznetsov[4]).The data recovery clustering can be utilized to address this as well.

A set of box clusters $(\lambda_t, V_t, W_t), t = 1, \ldots, T$, forms a disjunctive box cluster model of data R if

$$r_{ij} = max_{t=1,\ldots,T}\lambda_t v_{it} w_{jt} + \lambda_0 + e_{ij} \tag{2}$$

where e_{ij} are sufficiently small, and $\lambda_0, 0 < \lambda_0 < 1$, plays the role of an intercept in linear data models. This model differs from those of additive bi-clustering since (2) involves the operation of maximization rather than summation. To fit (2) with a relatively small number of boxes, assume λ_0 to be constant and specified before the fitting of the model. Then the model in (2) can be rewritten by putting $r'_{ij} = r_{ij} - \lambda_0$ on the left, so that λ_0 becomes a similarity shift value rather than an intercept.

We apply here the one-by-one fitting strategy (Mirkin[7]) so that each box cluster (λ_t, V_t, W_t) in (1) is found as a most deviant from the "middle", that is, minimizing the residuals in a single cluster model (with a constant λ_0)

$$r'_{ij} = r_{ij} - \lambda_0 = \lambda v_i w_j + e_{ij} \tag{3}$$

with the least squares criterion. In this formulation, $v = (v_i)$ and $w = (w_j)$ are binary membership vectors of V and W, respectively, so that $v_i w_j = 1$ if and only if $(i, j) \in V \times W$ like it is in a formal concept.

Let us initially assume $\lambda_0 = 0$ so that $r'_{ij} = r_{ij}$. Box cluster (λ_t, V_t, W_t) minimizing the least squares criterion

$$L^2 = \Sigma_{ij}(r'_{ij} - \lambda v_i w_j)^2 \tag{4}$$

over real λ and binary v_i, w_j, must lead to optimal λ being equal to the within-box average:

$$\lambda = \Sigma_{i\in V, j\in W} r'_{ij}/nm \tag{5}$$

which is the proportion of ones within the box minus λ_0, and, assuming that the λ is optimal, criterion L^2 in (4) admits the following decomposition:

$$L^2 = \Sigma_{ij} r'^2_{ij} - \lambda^2 nm \tag{6}$$

thus implying the following criterion to maximize

$$g(V, W) = \lambda^2 nm \tag{7}$$

According to (6), this criterion expresses the contribution of the box (V, W) to the data scatter $\Sigma_{ij} r'^2_{ij}$ which is useful to see how closely the box follows the data. On the other hand, criterion (7) combines two contrasting criteria for a box to be optimal: (a) the largest area, (b) the largest proportion of within-box unities. If restricted to a within-box non-zero option, the criterion (7) would lead to the formal concepts of the largest sizes, nm, as the only maximizers.

3 Locally Optimal Box Cluster

As the optimal intensity of a box (V, W) is fully determined by the summary entries within the box formed, we identify the box cluster with just sets V and W, that is, biset (V, W). With no loss of generality assume that it is V' that differs from V, by adding or removing an entity $i^* \in I$, while $W' = W$:

$$Diff(i^*) = [z_i r^2(i^*, W) + 2z_i r(V, W) r(i^*, W) - r^2(V, W)/n]/(m(n + z)) \tag{8}$$

where z_i^* equals 1 if i^* is added to V and -1 if i^* is removed from V, and $r(V, W)$ denotes the sum of all R' entries within box $V \times W$ and $r(i^*, W)$ is the sum of all r'_{i*j} over $j \in W$. A symmetric expression holds for $Diff(j^*)$ at $j^* \in J$.

A local search algorithm can be drawn starting from every entity $i \in V$ (or $j \in W$):

Algorithm Bicluster Box (i)

0. Take $V = i$ and $W = \{\, j \mid r_{ij} = 1\}$ as the starting box.

1. Find $Diff(8)$ for all elements of I and J, take that D^* which is maximum.

2. If D^* is not positive, halt. Otherwise, perform the operation of adding/removing for the corresponding entity and return to Step 1 with the updated box.

The resulting cluster box is provably rather contrast:

Statement 1
If box cluster (V, W) is found with $BBox()$ algorithm then, for any entity outside the box, its average similarity to it is less than the half of the within-box similarity λ; in contrast, for any entity belonging to the box, its average similarity to it is greater than or equal to the half of the within-box similarity λ.

Fitting model (2) can be done by applying algorithm $BBox()$ starting from each of the entities and retaining only different and most contributing solutions. Let us remind that the contribution of a box bicluster is but the value of criterion (7).

The subtracted λ_0 value can be used as a user-defined parameter to control, on average, the box cluster sizes. In our experiments we take λ to be the average value of R, that is, proportion of unities in the matrix.

4 Extending to Triclusters and P-Clusters

The model (2), as well as criteria (4) and (7) and algorithm $BBox$ are easily extended to the case when the data refer to relations between more than two sets. Specifically, one may suggest that there are p different entity sets, $I_1, I_2, \ldots, I_p$ such that relation R is p-ary, that is, corresponds to a subset of Cartesian product $\rho \subseteq I_1 \times I_2 \times \ldots \times I_p$. Then we would consider p-ary boxes. For the sake of simplicity, further on we consider only the case when $p = 3$, so that the three sets are denoted I, J, and K, whereas their subsets, by V, W, and U, respectively.

An extended form of model (2) is

$$r_{ijk} = max_{t=1T} \lambda_t v_{it} w_{jt} u_{kt} + \lambda_0 + e_{ijk} \quad (i \in I, j \in J, k \in K) \tag{9}$$

whereas criteria (4) and (7) lead to

$$L^2 = \Sigma_{ijk}(r'_{ijk} - \lambda v_i w_j u_k)^2 \tag{10}$$

and

$$g(V, W, U) = \lambda^2 nml \tag{11}$$

where n,m, and l are cardinalities of V, W, and U, respectively, and the optimal λ being the average value of all the R' three-way entries. The value of (11) shows contribution of the tricluster to the data scatter.

The value of difference $D(i^*) = g(V', W, U)g(V, W, U)$, where V' differs from V by the state of just one entity $i^* \in I$ so that i^* either belongs to V' if $i^* \notin V$ or does not, if $i^* \in V$, is expressed with a formula analogous to (8):

$$D(i^*) = [r^2(i^*, W, U) + 2z_{i^*} r(V, W, U) r(i^*, W, U) - z_{i^*} r^2(U, V, W)/n]/((n + \ z_{i^*})ml)$$

Here $z_{i^*} = 1$, if i^* is added to V and $z_{i^*} = -1$ otherwise, $r(V, W, U)$ is the sum of all the entries in R' over $(i, j, k) \in V \times W \times U$, and $r(i^*, V, W)$ is the sum of all the r'_{i^*jk} over $j \in W$ and $k \in U$. A symmetric expression holds for the changes in box (V, W, U) over $j^* \in W$ and $k^* \in U$. This leads to the following tricluster finding algorithm.

Algorithm Tricluster Box (i)

1.Take $V = \{\, i \,\}$, $W = \{\, j : r_{ijk} = 1$ for some $k \}$ and $U = \{\, k \mid r_{ijk} = 1$ for some $j \}$ as the starting box.

2. Find $D(i^*)$,$D(j^*)$ and $D(k^*)$ for all $i^* \in I$, $j^* \in J$, and $k^* \in K$ and J, take that of the values D which is maximum, denote it D^*.

3. If D^* is not positive, halt. Otherwise, perform the operation of adding/ removing for the corresponding entity and return to Step 1 with the updated box.

A statement, similar to Statement 1, holds.

Statement 2
If box cluster $B = (V, W, U)$ is found with $TriclusterBox()$ algorithm then, for any entity outside the box, its average similarity to B is less than the half of the within-box similarity λ; in contrast, for any entity belonging to the box, its average similarity to B is greater than or equal to the half of the within-box similarity λ.

5 Experiments

We have run experiments with synthetic data sets, just to see that $BiclusterBox$ is competitive towards both generalized formal concepts algorithms and conventional bicluster algorithms. We also run experiments on three-way data, first, to see if our triclusters are any good, and second, to compare solutions found with tricluster and bicluster algorithms. That is possible if one considers a three-way dataset over $I \times J \times K$ as a two-way dataset over $I \times (J \times K)$, that is, over I and Cartesian product of the two other sets, $J \times K$. We describe here some of the experiments.

Experiment 1. Take a binary 30×15 data table $R0$ comprising three non-overlapping formal concepts from Pensa and Boulicaut[9]. All $R0's$ entries are zeros except for those within three boxes comprising, in respect, first 10 rows (from 1 to 10) and first 5 columns (from 1 to 5), second 10 rows (from 11 to 20) and second 5 columns (from 6 to 10), and third 10 rows (from 21 to 30) and third 5 columns (from 11 to 15), whose all entries are ones. Then this matrix is changed to a matrix R_p by randomly changing its every entry with the probability $p\%, p = 1, 2, \ldots, 40$. Algorithm $BBox(i)$ has been applied to each R_p, with its mean subtracted as λ_0, at each $i \in I$ and $j \in J$, and sets of differing results stored as B_p and D_p. To compare these results with the original concepts in $R0$, we utilized the extension of Jaccard coefficient described in Pensa and Boulicaut[9].

Specifically, given two bisets, (V, W) and (V', W'), ratio of the areas of the rectangles corresponding to their intersection and union is taken as the measure of similarity:

$S((V, W), (V', W')) = |V \cap V'||W \cap W'|/(|V \cup V'||W \cup W'|)$.

so that

$$\sigma(B, B') = \Sigma_i max_j S((V_i, W_i), (V'_j, W'_j))/|B| \quad (12)$$

The averaged results of runs of two versions of approximate $BBox$ algorithm through several rounds of generated matrices $R_p(p = 1, 2, , 40)$ are summarised in Figure 1.

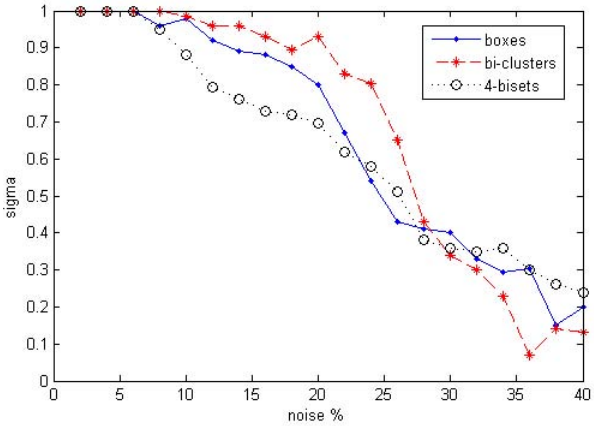

Fig. 1. Graphs of σ measure between the original three concepts and results of *Box* and *Dual* bicluster algorithms (Mirkin[8]) applied to the binary R_p matrix at different levels of random noise, $p = 1, 2, .., 40\%$. The third graph represents the σ values at 4-bisets by Pensa and Boullicaut[9].

Experiment 2. Pre-specified biclusters are set similarly here, yet the error is introduced via additive Gaussian noise so that the data become non-binary and general biclustering algorithms are applicable. We compare *BBox* with the best algorithm *Bimax* according to Prelic et al. [10] (see Figure 2). Here we refer to scoring functions measuring relevance and recovery as those utilized by Prelic et al. [10]. Given two sets of boxes on $I \times J$, B and B', consider expression

$$s_I(B, B') = \Sigma_{(v,w)\in B} max_{(v',w')\in B'} Jac(v, v')/|B| \tag{13}$$

where $Jac(v, v') = |v \cap v'|/|v \cup v'|$, the celebrated Jackard similarity index between sets v and v'. This is referred to as measure of recovery of B by B', and measure of relevance of B versus B'. *BBox* appears to be better than *Bimax* in recovery and worse than that in relevance.

Experiment 3. A binary file of a ternary relation between 250 movies, 738 keywords and 20 genres have been downloaded from Web-site:

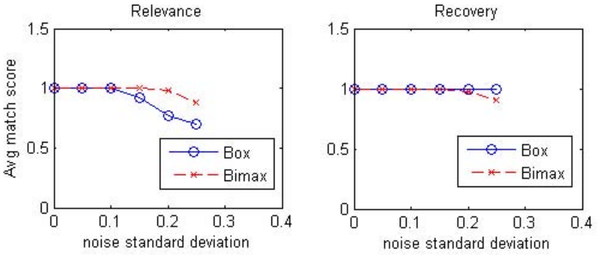

Fig. 2. Comparison of BBox and Bimax on the additive noise data

Table 1. Most contributing triclusters for 250 popular movies

Contrib.%	Movie	Keyword	Genre
28.5	'Star Wars:V-The Empire Strikes Back (1980)' 'Star Wars (1977)' 'Star Wars:VI - Return of the Jedi (1980)'	'Rebel' 'Princess' 'Empire' 'Death Star'	'Adventure' 'Action' 'Sci-Fi' 'Fantasy'
18.9	'12 Angry Men (1957)' 'Double Indemnity (1944)' 'Chinatown (1974)' 'The Big Sleep (1946)' 'Witness for the Prosecution (1957)' 'Dial M for Murder (1954)' 'Shadow of a Doubt (1943)'	'Murder' 'Trial' 'Widow' 'Marriage' 'Private' 'Detective' 'Blackmail' 'Letter'	'Crime' 'Drama' 'Thriller' 'Mystery' 'Film-Noir'
18.0	'The Return of the King (2003)' 'The Fellowship of the Ring (2001)' 'The Two Towers (2002)'	'Ring' 'Middle Earth'	'Adventure' 'Action' 'Fantasy'
18.0	'Terminator 2: Judgment Day(1991) ' 'The Terminator (1984)'	'Future' 'Cyborg'	'Thriller' 'Sci-Fi'

`http://www.imdb.com/chart/top` (accessed on 10 April 2009). In fact, the data involves more than 6500 keywords, but only those present at six or more of the movies have been used in our computations.

$TBox(i)$ algorithm found 84 triclusters containing more than one movie. Most contributing triclusters are in Table 1. Those who know of the movies can see that these are rather tight and meaningful indeed. We also took the data as a two-way table of movies over all possible pairs keyword-genre, to see if those results could be found without developing a novel algorithm. This gave a much smaller number of non-trivial biclusters, just 40 of them. Some of biclusters match the triclusters in Table 1 rather closely. Table 2 presents two of them, just to illustrate the differences. The most important difference is that biclusters are less expressive, in spite of more degrees of freedom: they can take any set of keyword-genre pairs whereas triclusters carry only Cartesian products. Take, for example, "Star Wars" biclusters it comprises just Cartesian product of set of keywords, Princess, Empire, of set of genres Adventure, Action, Sci-Fi, thus missing less trivial keywords "Rebel" and "Death Star" that have been picked up by the tricluster.

A similar observation can be made about tri- and bi-clusters containing "12 Angry Men" movie: the description of the biclusters, in Table 2, even if not a Cartesian product, seems rather dry and trivial in comparison to the description in Table 1.

Experiment 4. This experiment has been carried out with a 200 × 50 × 295 data set C extracted from website: `http://www.automotive.com/index.html` (in November 2010; the dataset is available from the authors on request). This concerns 200 passenger car models along with their 50 possible features. Besides, each car model is associated with a set of car models that are said to be

Table 2. Comparable most contributing biclusters

Contrib.%	Movie	Keyword,Genre
18.0	’Star Wars:V-The Empire Strikes Back (1980)’ ’Star Wars (1977)’ ’Star Wars:VI - Return of the Jedi (1980)’	’Princess, Adventure’ ’Princess, Action’ ’Princess, Sci-Fi’ ’Empire, Adventure’ ’Empire, Action’ ’Empire, Sci-Fi’
10.1	’12 Angry Men (1957)’ ’To Kill a Mockingbird (1962)’ ’Witness for the Prosecution (1957)”	’Murder, Drama’ ’Trial, Crime’ ’Trial, Drama’ ’Trial, Mystery’

comparable to the model under consideration. The set of comparable cars appears to be somewhat greater than the car set, totalling to 295 car models. On comparing triclusters found at C with biclusters found at its two-way, 200 × (50 × 295), version, triclusters versus biclusters show trends similar to those observed at the movie data set as illustrated in Table 3 for a set of budget cars appeared at both of the approaches.

Table 3. A budget car tricluster (columns 1,2 and 3) versus its bicluster namesake (columns 1, 4)

Tricluster: V (Car)	Tricluster: W (Feature)	Tricluster: U (Comparable)
’Ford Fiesta Sdn’ ’Hyundai Accent Sdn’ ’Mazda 2 Hb’	’4-Door’ ’Front Wheel Drive’ ’5 passengers’ ’5-speed Manual’	’Toyota Yaris sedan’ ’Chevrolet Aveo 4-door sedan” ’Honda Fit’
Bicluster intent part for set V (Feature ×Comparable)		
’4-Door,Toyota Yaris sedan’ ’4-Door,Honda Fit’ ’Front Wheel Drive,Toyota Yaris sedan’ ’Front Wheel Drive,Honda Fit’ ’5 passengers,Toyota Yaris sedan’ ’5 passengers,Honda Fit’		

Another curiosity is that in all the most contributing triclusters (V, W, U), the set of comparable cars U does not necessarily cover the set V but is always biased towards more luxury cars than V, as can be seen in Table 3 in column 3 versus column 1.

6 Conclusion

The approximation approach proves flexible enough to develop effective methods for biclustering of binary data and extend them to p-way binary data.

Specifically, a viable algorithm for triclustering has been developed, for the first time in the literature, to our knowledge. In our experiments with two ternary contexts it has proved more effective than the corresponding biclustering procedures.

The approximation methods proposed in this paper can be viewed in two aspects: (a) finding one "best" p-cluster, or (b) filling in a disjunctive model of the p-cluster structure of the given p-ary context. The "p-Box" algorithms developed in the paper depend on a specific entry in the context, thus appear to be rather computationally intensive, and thus not competitive, in the aspect (a). In this regard, we tried to accelerate the repetitive process by applying the algorithm only to those entries that have not been processed in the previous iterations yet. Unfortunately, this leads to poor results and should be further elaborated. If, however, one concentrates on the aspect (b) – revealing the entire cluster structure in a disjunctive way, then we can safely claim that our approach leads to a solution to this problem. It is still time consuming and the future efforts should address this issue.

References

1. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg (1999)
2. Hartigan, J.A.: Direct clustering of a data matrix. Journal of the American Statistical Association 67(337), 123–129 (1972)
3. Hartigan, J.A.: Clustering Algorithms. Wiley, Chichester (1975)
4. Ignatov, D., Kuznetsov, S.: Biclustering methods using lattices of closed subsets. In: Proceedings of 12th National Conference on Artificial Intelligence, Moscow, FML, vol. 1, pp. 175–182 (2010) (in Russian)
5. Madeira, S.C., Oliveira, A.L.: Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 1(1), 24–45 (2004)
6. Mirkin, B., Arabie, P., Hubert, L.: Additive two-mode clustering: the error-variance approach revisited. Journal of Classification 12, 243–263 (1995)
7. Mirkin, B.: Mathematical Classification and Clustering, p. 448. Kluwer, Dordrecht (1996)
8. Mirkin, B.: Two goals for biclustering: "Box" and "Dual" methods (2008) (unpublished manuscript)
9. Pensa, R.G., Boulicaut, J.-F.: Towards fault-tolerant formal concept analysis. In: Bandini, S., Manzoni, S. (eds.) AI*IA 2005. LNCS (LNAI), vol. 3673, pp. 212–223. Springer, Heidelberg (2005)
10. Prelic, A., Bleuler, S., Zimmermann, P., Wille, A., Bühlmann, P., Gruissem, W., Hennig, L., Thiele, L., Zitzler, E.: A Systematic Comparison and Evaluation of Biclustering Methods for Gene Expression Data. Bioinformatics 22(9), 1122–1129 (2006)
11. Rome, J.E., Haralick, R.M.: Towards a formal concept analysis approach to exploring communities on the World Wide Web. In: International Conference on Formal Concept Analysis, Lens, France (2005)