A Scalable Algorithm for Banded Pattern Mining in Multi-dimensional Zero-One Data

Fatimah B. Abdullahi, Frans Coenen, and Russell Martin

The Department of Computer Science, The University of Liverpool, Ashton Street, Liverpool, L69 3BX, United Kingdom
{f.b.abdullahi,coenen,russell.martin}@liverpool.ac.uk

Abstract. A banded pattern in "zero-one" high dimensional data is one where all the dimensions can be organized in such a way that the "ones" are arranged along the leading diagonal across the dimensions. Rearranging zero-one data so as to feature bandedness allows for the identification of hidden information and enhances the operation of many data mining algorithms that work with zero-one data. In this paper an effective ND banding algorithm, the ND-BPM algorithm, is presented together with a full evaluation of its operation. To illustrate the utility of the banded pattern concept a case study using the GB Cattle movement database is also presented.

Keywords: Banded Patterns, Zero-One data, Pattern Mining.

1 Introduction

Zero-one data occurs in many real world datasets, ranging from bioinformatics [3] to information retrieval [6]. The identification of patterns in zero-one data is an important task within the field of data mining, for example association rule mining [1]. In this paper, we study banded patterns in high dimensional zero-one data. Examples illustrating 2D and 3D bandings are presented in Figures 1 and 2. In practice data can typically not be perfectly banded, but in many cases some form of banding can be achieved. This paper presents a novel N-dimensional Banded Pattern Mining algorithm (ND-BPM) for the identification of banded patterns in zero-one ND data. The operation of the ND-BPM algorithm differs from previous work on banded patterns, such as the MBA [12] and BC [16] algorithms, that allowed for the discovery of banding in only 2D data.

	b	c	e	a	d
3	•	•			
1		•	•		
2			•	•	
4				•	•

Fig. 1. 2D Banding Example

	a	c	b	d
2	•	•		
1				
3				

(a)

	a	c	b	d
2				
1		•	•	
3				

(b)

	a	c	b	d
2				
1				
3			•	•

(c)

Fig. 2. 3D Banding Example

L. Bellatreche and M.K. Mohania (Eds.): DaWaK 2014, LNCS 8646, pp. 345–356, 2014.

While the concept of banded matrices has its origins in numerical analysis [17], it has been studied within the data mining community [12,11]. The advantages of banding may be summarized as follows:

1. Banding may be an indication of some interesting phenomena which is otherwise hidden in the data.
2. Working with banded data is seen as preferable from a computational point of view; the computational cost involved in performing certain operations falls significantly for banded matrices leading to significant savings in terms of processing time [10].
3. Related to 2, when a matrix is banded, only the non-zero entries along the diagonal needs to be considered. Thus, when using banded storage schemes the amount of memory required to store the data is directly proportional to the bandwidth. Therefore finding a banding that minimizes the bandwidth is important for reducing storage space and algorithmic speed up [15].

The main issue with the identification of banding in data is the large number of permutations that need to be considered. There has been some research in the context of 2D data focused on minimizing the distance of non-zero entries from the main diagonal of the matrix (bandwidth) by reordering the original matrix [5,10,17]. The current (2D) state-of-the-art algorithm, MBA [13], focuses on identifying banding in binary matrices by flipping zero entries (0s) to one entries (1s) and vice versa, assuming a fixed column permutation.

The rest of this paper is organized as follows. Section 2 discuss related work. A formalism for the banded pattern problem is then presented in Section 3. Section 4 provide an overview of the proposed scoring mechanism and the ND-BPM algorithm is presented in Section 5. Section 6 provides a worked example illustrating the algorithm in the context of 2D. The evaluation of the ND-BPM Algorithm with respect to both 2D and 3D zero-one data is reported in Section 7. Finally, in Section 8 some conclusions are presented.

2 Related Work

From the data analysis perspective banded patterns can occur in many applications, examples can be found in paleontology [4], Network data analysis [7] and linguistics [13]. The property of bandedness with respect to data analysis was first studied by Gemma et al. [12]. They addressed the minimum banding problem by computing how far a 2D data "matrix" is from being banded. The authors in [12] defined the banding problem as: given a binary matrix M, find the minimum number of 0 entries that needs to be modified into 1 entries and the minimum number of 1 entries that needs to be modified into 0 entries so that M becomes fully banded. Gemma et al. fixed the column permutations of the data matrix before executing their algorithm [12]. As noted in the introduction to this paper the current state of the art algorithm is the Minimum Banded Augmentation (MBA) algorithm [13] which uses the principle of assuming "a fixed column permutation" over a given Matrix M. The basic idea is to solve optimally the consecutive one property on the permuted matrix M and then resolve "Sperner conflicts" between each row of the permuted matrix M, by going through all the extra rows and making them consecutive. While it can be argued that the fixed column permutation assumption is not a very realistic assumption with respect to many real world situations, heuristical methods were proposed in [12] to determine a suitable fixed column permutation. Another banding strategy that transposes a matrix is the Barycentric (BC) algorithm that was originally designed for graph drawing and more recently used to reorder binary matrices [16]. The distinction between these previous algorithms and that presented in this paper is that the previous algorithms were all directed at 2D data, while the proposed algorithm operates in 3D. It should also be noted that Bandwidth minimization of binary matrices is known to be NP-Complete [10] as it is related to the reordering of binary matrices [15].

Given the above the MBA and BC algorithms are the two exemplar banding algorithms with which the operation of the proposed ND-BPM algorithm is compared and evaluated as discussed later in this paper (see Section 7).

3 Problem Definition

Let Dim be a set of dimension $\{Dim_1, Dim_2, \ldots, Dim_n\}$. Each dimension comprises a set of k indexes such that $Dim_i = \{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\}$. Thus in 2D space the indexes associated with Dim_1 might be record numbers and the indexes associated with Dim_2 may be attribute value identifiers. In 3D Dim_3 might equate to time and the indexes to discrete time slots, and so on. Note that we will indicate a particular index j belonging to a dimension i using the notation a_{i_j}. Note also that dimensions are not necessarily of equal size. Given a zero-one data set D that corresponds to the data set defined by Dim we can think of this data space in terms of an ND grid with the "ones" indicated by "dots" (ND spheres) and the "zeroes" by empty space. Individual dots can thus be referenced using the ND coordinate system defined by Dim. Such a data space can be "perfectly banded" if there exists a permutation of the indexes such that: (i) $\forall a_{i_j} \in dim_i$ the dots occur consecutively at indexes $\{a, a+1, a+2, \ldots\}$ and the "starting index" for dim_i is less than or equal to the starting index for dim_{i+1}.

4 The N Dimensional Banding Mechanism

The discovery of the presence of banding in a zero-one ND space requires the rearrangement of the indexes in each dimension so as to "reveal" a banding (or at least an approximate banding). This is a computationally expensive task especially in the context of ND space. In the case of the ND-BPM algorithm it is proposed that this be achieved using the concept of banding scores. Given a particular dimension Dim_i each index a_{i_j} will have a banding score BS_{i_j} associated with it. These banding scores are then used to rearrange the ordering of the indexes in Dim_i so that the index with the greatest banding score is listed first. Individual banding scores are calculated by considering dimension pairs.

Thus given two dimensions Dim_p and Dim_q we calculate the banding scores for all $a_{p_j} \in Dim_p$ with respect to Dim_q. We use the notation BS_{pq_j} to indicate the banding score of index a_j in Dim_p calculated with respect to Dim_q as follows:

$$BS_{pq_j} = \frac{\sum_{k=1}^{k=|W|}(|Dim_q| - W_k + 1)}{\sum_{k=1}^{k=|W|}(|Dim_q| - k + 1)} \tag{1}$$

where the set W is the set of Dim_q indexes representing "dots" whose coordinate set feature the index x_{p_j} from Dim_p. However, if $n > 1$ we need to do this for all instances of the Dim_p and Dim_q pairings that can exist across the space. Thus the set of dimensions identifiers, I, that excludes the identifiers for Dim_p and Dim_q. Thus:

$$BS_{pq_j} = \frac{\sum_{i=1}^{i=z} BS_{pq_j} \ for \ Dim_i}{z} \tag{2}$$

where $z = \prod_{i=1}^{i=|D|} |Dim_{I_i}|$.

We can also use the banding score concept as a measure of the goodness of a banding configuration. By first calculating the Dimension Banding Score (DBS) for each dimension p with respect to dimension q (DBS_{pq}) as follows:

$$DBS_{pq} = \frac{\sum_{j=1}^{j=|Dim_p|} BS_{pq_j}}{|Dim_p|} \tag{3}$$

The Global Banding Score (GBS) for the entire configuration is then calculated as follows:

$$GBS = \frac{\sum_{p=1}^{p=n-1} \sum_{q=p+1}^{q=n} DBS_{pq}}{\sum_{i=1}^{i=n-1} n - i} \tag{4}$$

5 N Dimensional Banded Pattern Mining (ND-BPM) Algorithm

The ND-BPM algorithm is presented in algorithm 1. The inputs are (line 3): (i) a zero-one data set D and (ii) the set DIM. The output is a rearranged data space that maximizes GBS. The algorithm iteratively loops over the data space. On each iteration the algorithm attempts to rearrange the indexes in the set of dimensions DIM. It does this by considering all possible dimension pairings pq. For each pairing the BS value for each index j in dimension Dim_p is calculated (line 11) and used to rearrange the dimension (line 13). If a change has been effected a change flag is set to $TRUE$ (line 15) and a DBS value calculated (line 17). Once all pairings have been calculated a GBS_{new} value is calculated (line 20). If GBS_{new} is worse than the current GBS value (GBS_{sofar}), or there has been no change, we exit with the current configuration D (line 23). Otherwise we set D to D', and $GBS_{sofar}$to GBS_{new} and repeat.

Algorithm 1. The ND-BPM Algorithm

Input D, DIM
Output Rearranged data space that serves to maximize GBS
$change = FALSE$
$n = |Dim|$
$GBS_{sofar} = 0$
loop
  $D' = D$
  for $p = 1$ **to** $p = n - 1$ **do**
    for $q = p + 1$ **to** $q = n$ **do**
      for $j = 1$ **to** $|DIM_p|$ **do**
        Calculate BS_{pq_j} using Equations (1) and (2) as appropriate
      end for
      $D'' = D'$ with indexes in Dim_p reordered according to the set BS_{pq_j}
      if $D' \neq D''$ **then**
        change = TRUE
        $D' = D''$
      end if
      Calculate DBS_{pq} using Equation(3)
    end for
    Calculate GBS_{new} using Equation(4)
  end for
  if $change = FALSE$ **or** $GBS_{new} < GBS_{sofar}$ **then**
    exit with current configuration D
  else
    $D = D'$
    $GBS_{sofar} = GBS_{new}$
  end if
end loop

6 Worked Example

To illustrate the operation of the ND-BPM algorithm a worked example, using the 5×4 data space given in Figure 3, is presented here. We commence by calculating the set of scores BS_{1_j} for dimension 1 (Dim_1) to obtain: $BS_{1_1} = 0.9166$, $BS_{1_2} = 0.5000$, $BS_{1_3} = 0.6666$ and $BS_{1_4} = 0.7777$. Using this set of scores the indexes in Dim_1 are rearranged to produce the configuration shown in Figure 4. The Dim_1 banding score is then $DBS_1 = 0.7277$. Next we calculate the set of scores BS_{2_j} for (Dim_2) to obtain: $BS_{2_1} = 0.8888$, $BS_{2_2} = 1.0000$, $BS_{2_3} = 0.6666$, $BS_{2_4} = 0.6666$ and $BS_{2_5} = 0.5555$. Using this set of scores the indexes in Dim_2 are rearranged to produce the configuration shown in Figure 5. The Dim_2 banding score is then $DBS_2 = 0.8221$ and the global banding score is:

$$GBS = \frac{\sum_{p=1}^{p=n-1} \sum_{q=p+1}^{q=n} DBS_{pq}}{\sum_{i=1}^{i=n-1} n - i} = 0.7749$$

	a	b	c	d
1	•		•	
2	•			•
3		•		•
4	•	•		
5		•	•	

Fig. 3. Input data

	a	d	c	b
1	•		•	
2	•	•		
3		•		•
4	•			•
5			•	•

Fig. 4. Input data with Dim_1 rearranged

	a	d	c	b
2	•	•		
1	•		•	
4	•			•
3		•		•
5			•	•

Fig. 5. Input data with Dim_2 rearranged

We repeat the process since changes were made. The set of scores for Dim_1 are now: $BS_{1_1} = 1.0000$, $BS_{1_2} = 0.7777$, $BS_{1_3} = 0.5555$ and $BS_{1_4} = 0.5000$; with this set of scores Dim_1 remains unchanged. However, the Dim_1 banding score DBS_1 is now 0.7944 because of changes to Dim_2 (previously this was 0.7277). The set of scores for Dim_2 are now: $BS_{2_1} = 1.0000$, $BS_{2_2} = 0.8888$, $BS_{2_3} = 0.6666$, $BS_{2_4} = 0.6666$ and $BS_{2_5} = 0.5555$. Again, with this set of scores Dim_2 remains unchanged, thus the configuration shown in Figure 5 remains unchanged. The Dim_2 banding score is now $DBS_2 = 0.8296$ (was 0.8221). The global banding score is now:

$$GBS = \frac{\sum_{p=1}^{p=n-1} \sum_{q=p+1}^{q=n} DBS_{pq}}{\sum_{i=1}^{i=n-1} n - i} = 0.8120$$

On the previous iteration it was 0.7749, however no changes have been made on the second iteration so the algorithm terminates.

7 Evaluation

To evaluate the ND-BPM algorithm its operation was compared with the established MBA and BC algorithms, two exemplar algorithms illustrative of the alternative approaches to identifying banding in zero-one data as described in Section 2. Because MBA and BC were designed to operate using 2D, the evaluation was conducted in these terms. Eight data sets taken from the UCI machine learning data repository [8] were used. The first set of experiments, reported in sub-section 7.1 below, considered the efficiency of the ND-BPM algorithm in comparison with the MBA and BC algorithms. The second set of experiments (Section 7.2) considered the effectiveness of ND-BPM algorithm, again in comparison with the MBA and BC algorithms, with respect to the bandings produced. The third set of experiments, reported in sub-section 7.3 below, considered the effectiveness of banding with respect to a Frequent Itemset Mining (FIM) scenario. To determine the effectiveness of the ND-BPM algorithm with respect to a higher number of dimensions further experiments were conducted using the GB cattle movement database. This is described in Section 7.4.

7.1 Efficiency

To determine the efficiency of the proposed ND-BPM algorithm in the context of 2D and with respect to the MBA and BC algorithms, we recorded the run time required to maximize the banding score GBS in each case. The data sets were normalized and discretized using the LUCS-KDD ARM DN Software[1] to produce the desired zero-one data sets (continuous values were ranged using a maximum of five ranges). Table 1 shows the results obtained. Table 1 presents run-time and the final GBS value obtained in each case. The table also records the number of attributes (after discretization) and the number of records for each data set. From the table it can be observed that there is a clear correlation between the number of records in a dataset and run time as the number of records increases the processing time also increases (this is to be expected). The table also demonstrates that the ND-BPM algorithm requires less processing time than the other two algorithms considered.

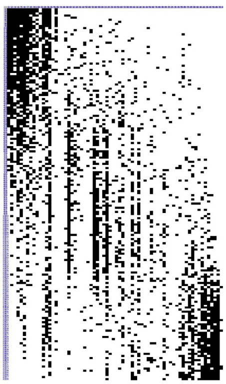

Fig. 6. Banding resulting from ND-BPM algorithm as applied to the Wine dataset ($GBS = 0.7993$)

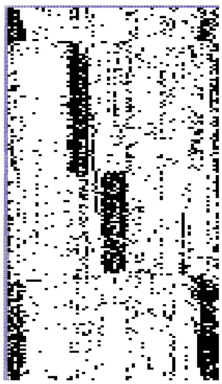

Fig. 7. Banding resulting from MBA algorithm as applied to the Wine dataset ($GBS = 0.7123$)

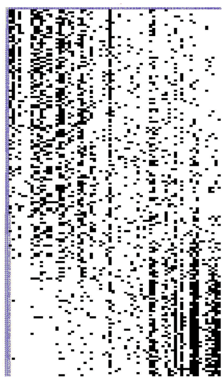

Fig. 8. Banding resulting from BC algorithm as applied to the Wine dataset ($GBS = 0.7021$)

7.2 Effectiveness with Respect to Global Banding Score

It was not possible to identify a perfect banding with respect to any of the UCI data sets, this was to be expected. However, in terms of GBS, Table 1 clearly shows that the proposed ND-BPM algorithm outperformed the previously proposed MBA and BC algorithms (best scores highlighted using bold font). Figures 6, 7 and 8 show the bandings obtained using the wine data sets and the ND-BPM, MBA and BC algorithms respectively. Inspection of these Figures indicates that banding can be identified in all cases. However, from inspection of the figures it is suggested that the banding produced using the proposed

[1] `http://www.csc.liv.ac.uk/$\sim$/frans/KDD/Software/LUCS_KDD_DN_ARM.`

ND-BPM algorithm is better. For example considering the banding produced when the MBA algorithm is applied to the wine dataset (Figure 7) the resulting banding includes dots ("1"s) in the top-right and bottom-left corners while the ND-BPM algorithm does not (it features a smaller bandwidth). When the BC algorithm is applied to the wine dataset (Figure 8) the banding is less dense than in the case of the ND-BPM algorithm.

Table 1. Efficiency Experimental Results (best results presented in bold font), GBS = Global Banding Score, RT = Run time (secs.)

Datasets	# Rec s	# Cols	ND-BPM GBS	MBA GBS	BC GBS	ND-BPM RT	MBA RT	BC RT
annealing	898	73	**0.8026**	0.7305	0.7374	**0.150**	0.260	0.840
heart	303	52	**0.8062**	0.7785	0.7224	**0.050**	0.160	0.170
horsecolic	368	85	**0.8152**	0.6992	0.7425	**0.070**	0.200	0.250
lympography	148	59	**0.8365**	0.7439	0.7711	**0.030**	0.140	0.110
wine	178	68	**0.7993**	0.7123	0.7021	**0.040**	0.150	0.110
hepatitis	155	56	**0.8393**	0.7403	0.7545	**0.050**	0.150	0.090
iris	150	19	**0.8404**	0.8205	0.7516	**0.020**	0.080	0.060
zoo	101	42	**0.8634**	0.7806	0.7796	**0.020**	0.100	0.050

7.3 Effectiveness with Respect to FIM

In addition to being an indicator of some pattern that may exist in zero-one data, banding also has application with respect to increasing the efficiency of algorithms that use matrices or tabular information stored in the form of n-dimensional data storage structures. One example is algorithms that use $n \times n$ affinity matrices, such as spectral clustering algorithms [14], to identify communities in networks (where n is the number of network nodes). Another example is Frequent Itemset Mining (FIM) [1,2] where it is necessary to process large binary valued data collections stored in the form of a set of feature vectors (drawn from a vector space model of the data). To test the effectiveness of the bandings produced as a result of the experiments reported in Sub-section 7.1 above, a FIM algorithm was applied to the banded data sets produced using the ND-BPM algorithm (the TFP algorithm [9] was actually used, but any alternative FIM algorithm would equally well have sufficed). The results are presented in Table 2. From the table it can be seen that FIM is always much more efficient when using banded data than when using non banded data if we do not include the time to conduct the banding. If we include the banding time, in 8 out of the 12 cases, it is still more efficient. Similarly, when the FIM algorithm was applied to the banded data sets produced using the MBA and BC algorithms, it was also observed that FIM was more effecient using banded data than when using non banded data without the banding time, with the banding time in 4 (MBA) and 5 (BC) out of the 12 cases, FIM is still more effecient.

Table 2. FIM runtime with and without banding ($\sigma = 2\%$)

Datasets	#Rows	#Cols	Banding Time(s)	FIM time (s) with Banding	Total	FIM time (s) without Banding
adult	48842	97	346.740	**2.274**	349.014	**5.827**
anneal	898	73	0.150	**0.736**	**0.086**	2.889
chessKRvk	28056	58	95.370	**0.082**	95.452	**0.171**
heart	303	52	0.050	**0.294**	**0.344**	0.387
hepatitis	155	56	0.030	**0.055**	**0.085**	22.416
horseColic	368	85	0.070	**0.899**	**0.969**	1.242
letRecog	20000	106	42.420	**3.004**	45.424	**6.763**
lympography	148	59	0.030	**7.997**	**8.022**	12.658
mushroom	8124	90	14.400	**874.104**	**888.504**	1232.740
penDigits	10992	89	21.940	**2.107**	24.047	**2.725**
waveForm	5000	101	3.030	**119.220**	**122.250**	174.864
wine	178	68	0.010	**0.155**	**0.165**	0.169

7.4 Large Scale: Cattle Movement Database

To illustrate the utility of the proposed ND-BPM algorithm, the authors have applied the algorithm to a 3 dimensional data set constructed from the GB Cattle movement data base. The GB cattle movement database records all the movements of cattle registered within or imported into Great Britain. The database is maintained by the UK Department for Environment, Food and Rural Affairs ($DEFRA$). For the analysis reported in this work, data sets for the months of January to December 2003 to 2006, for one county (Lancashire in Great Britain), was used. Each record comprises: (i) Animal Gender, (ii) Animal age, (iii) the cattle breed type, (iv) sender location in terms of easting and northing grid values, (v) the type of the sender location, (vi) receiver location in terms of eastings and northings grid values, (vii) receiver location type and $(viii)$ the number of cattle moved. Discretization and Normalization processes were used to convert the input data into the desired zero-one format. As a result the GB dataset comprised 80 items distributed over four dimensions: records, attributes, easting values and northing values. For ease of understanding and so that results can be displayed in a 2D format only three dimensions were considered at any one time (records, attributes and eastings; and records, attributes and northings).

The results obtained are presented in Tables 3 and 4. The tables record the number of attributes (after discretization) representing attribute information, the number of records and the number of slices used to represent the discretized sender eastings and northings. The tables also record the run-times required by the algorithms in order to maximize the global banding score GBS and the final GBS value arrived at in each case.

Figures 9 and 11 shows the sampled data before banding and Figures 10 and 12 shows the sampled data after banding using a subset of the data for the month of January 2003. Inspection of the figures indicates that banding can clearly be identified. More specifically, there are certain movement patterns, that can be identified

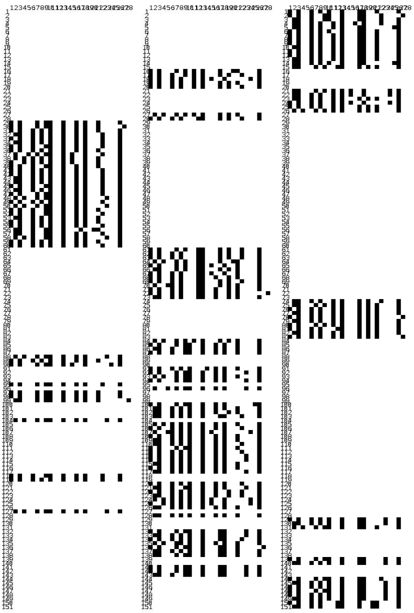

Fig. 9. January Raw data, with Eastings as 3rd dimension, before Banding

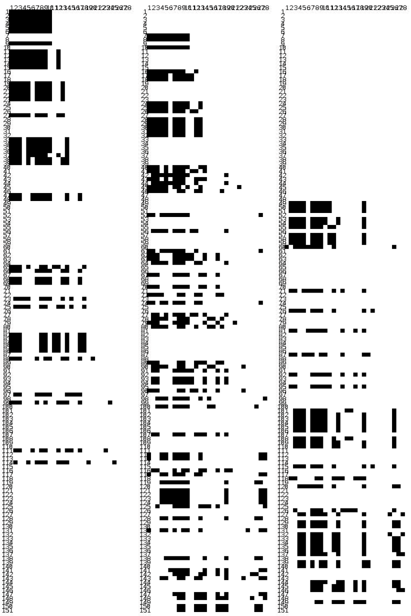

Fig. 10. Data set from Figure 9 after Banding

Table 3. Experimental Results for GB Summary for Easting locations

Years	Datasets	# Recs	# Attrs	# Slices	GBS	Run time
2003	Jan-Dec	167919	70	10	0.3902	2485.99
2004	Jan-Dec	217566	72	10	0.2823	5475.51
2005	Jan-Dec	157142	72	10	0.3093	2114.09
2006	Jan-Dec	196290	72	10	0.3075	3856.83

Table 4. Experimental Results for GB Summary for Northing locations

Years	Datasets	# Recs	# Attrs	# Slices	GBS	Run time
2003	Jan-Dec	167919	70	10	0.4239	2393.50
2004	Jan-Dec	217566	72	10	0.3101	4786.66
2005	Jan-Dec	157142	72	10	0.3632	1232.09
2006	Jan-Dec	196290	72	10	0.3525	4162.71

from the generated banding. For example, from Figure 10, it can be observed that male cattle breeds are moved more often in the east of the country than in the west. Similarly, from Figure 12, it can be observed that male cattle of (age = 1) are more frequently moved in the north than in the south of the country.

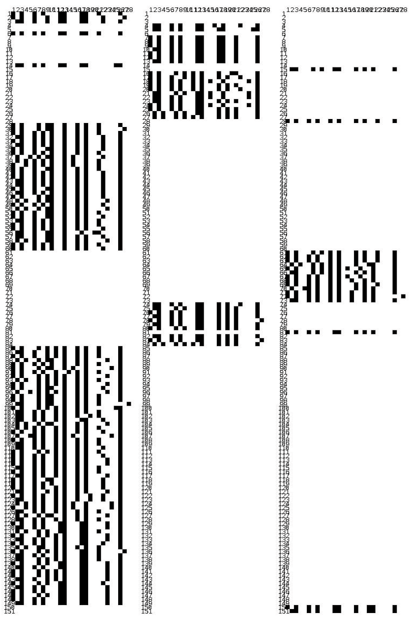

Fig. 11. January Raw data, with Northings as 3rd dimension, before Banding

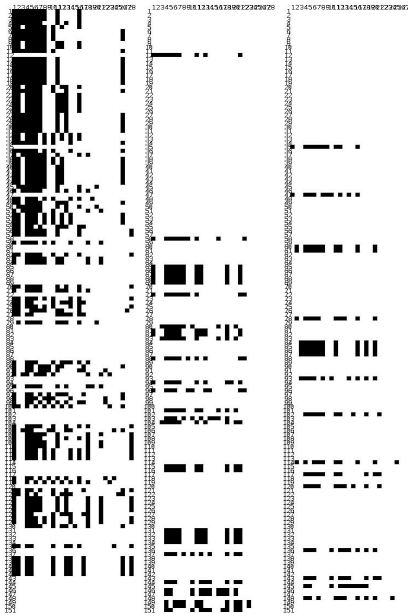

Fig. 12. Data set from Figure 11 after Banding

8 Conclusions

In this paper the authors have described an approach to identifying bandings in zero-one data using the concept of banding scores. More specifically the ND-BPM algorithm has been presented. This algorithm operates by iteratively rearranging the items associated with individual dimensions according to the concept of banding scores. The operation of the ND-BPM algorithm was compared with the operation of the MBA and BC algoithms in the context of 2D using eight data sets taken from the UCI machine learning repository. In the context of 3D, it was tested using sample data taken from the GB cattle movement database for the months of January to December 2003 to 2006. The reported evaluation established that the proposed approach can reveal banded patterns within zero-one data reliably and with reasonable computational efficiency and able to handle even higher dimensions in reasonable time. The evaluation also confirmed that, at least in the context of FIM, efficiency gains can be realized using the banding concept. For future work the authors intend to extend their research to address situations where we seek to establish banding with respect to a subset of the available dimensions (maintaining the position of indexes in the other dimensions). Whatever the case, the authors have been greatly encouraged by the results produced so far, as presented in this paper.

References

1. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: SIGMOD 1993 pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings 20th International Conference on Very Large Data Bases (VLDB 1994), pp. 487–499 (1994)
3. Alizadeh, F., Karp, R.M., Newberg, L.A., Weisser, D.K.: Physical mapping of chromosomes: A combinatorial problem in molecular biology. Algorithmica 13, 52–76 (1995)
4. Atkins, J., Boman, E., Hendrickson, B.: Spectral algorithm for seriation and the consecutive ones problem. SIAM J. Comput. 28, 297–310 (1999)
5. Aykanat, C., Pinar, A., Catalyurek, U.: Permuting sparse rectangular matrices into block-diagonal form. SIAM Journal on Scientific Computing 25, 1860–1879 (2004)
6. Baeza-Yates, R., RibeiroNeto., B.: Modern Information Retrieval. Addison-Wesley (1999)
7. Banerjee, A., Krumpelman, C., Ghosh, J., Basu, S., Mooney, R.: Model-based overlapping clustering. In: Proceedings of Knowledge Discovery and DataMining, pp. 532–537 (2005)
8. Blake, C.L, Merz, C.J.: Uci repository of machine learning databases (1998), `http://www.ics.uci.edu/~mlearn/MLRepository.htm`
9. Coenen, F.P., Goulbourne, G., Leng, P.: Computing association rules using partial totals. In: Siebes, A., De Raedt, L. (eds.) PKDD 2001. LNCS (LNAI), vol. 2168, pp. 54–66. Springer, Heidelberg (2001)
10. Cuthill, A.E., McKee, J.: Reducing bandwidth of sparse symmentric matrices. In: Proceedings of the 1969 29th ACM National Conference, pp. 157–172 (1969)
11. Fortelius, M., Puolamaki, M.F.K., Mannila, H.: Seriation in paleontological data using markov chain monte method. PLoS Computational Biology, 2 (2006)
12. Garriga, G.C., Junttila, E., Mannila, H.: Banded structures in binary matrices. Knowledge Discovery and Information System 28, 197–226 (2011)
13. Junttila, E.: Pattern in Permuted Binary Matrices. Ph.D. thesis (2011)
14. Von Luxburg, U.A.: A tutorial on spectral clustering. Statistical Computation 17, 395–416 (2007)
15. Mueller, C.: Sparse matrix reordering algorithms for cluster identification. Machune Learning in Bioinformatics (2004)
16. Mäkinen, E., Siirtola, H.: The barycenter heuristic and the reorderable matrix. Informatica 29, 357–363 (2005)
17. Rosen, R.: Matrix bandwidth minimisation. In: ACM National conference Proceedings, pp. 585–595 (1968)