Matrix Reordering Based on PQR Trees, Binarization and Smoothin

Bruno Figueiredo Medina*, Celmar Guimarães da Silva**

School of Technology - University of Campinas

ABSTRACT

Providing good permutations of rows and columns of a matrix is necessary in order to enable users to understand underlying patterns hidden on its data. A previous work provided a preliminary version of an algorithm based on matrix binarization and PQR trees for reordering quantitative data matrices. This work proposes an improved version of this algorithm. It uses smoothing as an intermediary step for enhancing the capability of finding good permutations, without modifying original matrix data. The work exemplifies the potential of this technique by reordering scrambled synthetic matrices with underlying patterns.

Keywords: Matrix reordering, Smoothing, PQR Tree, Reordering Algorithms.

1 INTRODUCTION

Several areas use matrix reordering in order to highlight patterns and/or to facilitate analyses of a data set [5]. Given that there is a large number of possible permutations of matrix rows and columns, some reordering algorithms try to speed up the process of choosing a good permutation, such as: Barycenter Heuristic [6], Cluster Analysis, Multidimensional Scaling, Eigen Decomposition, Graph-Theoretic Dimension Reduction [11], Traveling Salesman Problem solvers (TSP) [1], Correspondence Analysis [3], Elliptical Seriation [2], among others.

Our research team developed two reordering algorithms based in PQR trees [10] for reordering binary matrices: PQR Sort [9] and PQR Sort with Sorted Restrictions [8] (or PQR Sort SR, for short). Recently, we presented an algorithm called Multiple Binarization that uses some steps of these two algorithms for quantitative matrix reordering [7]. This paper presents how smoothing techniques and a slight adjust on the use of PQR trees may be applied to Multiple Binarization for improving reordering quality of its output.

The paper is organized as follow: Section 2 presents briefly concepts of PQR tree, noise smoothing and Multiple Binarization algorithm; Section 3 presents our developed method; and Section 4 concludes the paper and presents future works.

2 RELATED WORKS

Our work is based in a data structure called PQR tree and reordering algorithms. Furthermore, it is related with noise smoothing. These issues are explained hereafter.

2.1 PQR Tree

PQR tree is a rooted tree that represents possible permutations of a universe set U. These permutations obey an input restriction set, where a restriction is a set of elements which should be consecutive on the resultant permutations. This tree has four node types: P, Q, R and leaves, as follow [10]:

- Leaves are the elements of U;
- A P node enables any permutation of its children;
- A Q node enables only reversal of its children;
- An R node is similar to P node, but occurs only when its children may not be permuted in order to obey simultaneously all restrictions.

One of permutations represented by PQR tree is its frontier (i.e., their leaves, read from left to right), which is easy to obtain.

It is important highlighting that a PQR tree does not change when one adds to it a restriction whose elements are leaves with a common R node as their ancestor.

2.2 Multiple Binarization Algorithm

Multiple Binarization [7] (or MB, for short) is a method created to reorder quantitative matrices. Its name is related to one of its steps, which consists on creating multiple binary matrices used to obtain restrictions sets and to create two PQR trees (one for rows and one for columns).

Therefore, taking as input a quantitative data matrix *M*, to be reordered, and a set of delimiter values $D = \{d_1, d_2, \ldots, d_k\}$, the reordering is made as following:

1. Create k binary matrices $M^1, M^2, \ldots, M^k$, where $M^k(r,c)$ = 1 if $M(r,c) > d_k$, otherwise $M^k(r,c)$ = 0;
2. For each M^i, create two restriction sets: one representing row restrictions and other column restrictions. A row restriction contains row labels $r_1, r_2, \ldots, r_n$ if there is at least one possible value of c such as $M(r_i,c)=1$ for all $1 \leq i \leq n$. A column restriction may be defined similarly to a row restriction.
3. Define two union sets, one for rows and other for columns. Row union set have all row restrictions of each binary matrix obtained in Step 2, without repetition. Column union set have all column restrictions of each binary matrix obtained in Step 2, without repetition.
4. Sort row union set and column union set in ascending order of restriction size, generating two ordered lists.
5. Create two PQR trees, one for rows and other for columns, using as input the row and column restriction lists (Step 4), respectively.
6. Reorder rows and columns of M according to the frontier of both PQR trees.
7. Return the reordered version of M created in the previous step.

In this algorithm, Steps 2 and 4 belong to the intermediate process of the PQR Sort and PQR Sort SR algorithms, respectively.

2.3 Smoothing

Smoothing is an Image Processing technique which aims to reduce noise and enhance image quality. Gonzalez and Woods [4] present some smoothing techniques, from which we used the mean filter with a 3 × 3 box filter as a first approach to our problem. Briefly, it consists on creating a new matrix S, based on a matrix M, such as

$$S(r,c) = \frac{1}{9}\sum_{s=-1}^{1}\sum_{t=-1}^{1} M(r+s, c+t);$$

i.e., it calculates *S(r,c)* as the mean value of a 3 × 3 neighborhood centered on *M(r,c)*. In this formulation, values outside *M* borders, when consulted, should be considered as 0.

* E-mail address: bruno.medina@pos.ft.unicamp.br

** E-mail address: celmar@ft.unicamp.br

3 WORK IN PROGRESS

We perceived that Multiple Binarization algorithm has some difficulties in producing “good” reordered matrices in presence of noise, but returns good results where noise level is low or absent. Therefore, we present a new reordering algorithm which mixes Multiple Binarization method and mean filter in order to overcome this difficulty. We propose to remove noises (with mean filter) that become visible after a first reordering process, and then to reorder the smoothed matrix. After that, we reorder the original matrix according to row and column ordering of the smoothed one.

Additionally, we observed that the early creation of R nodes in the PQR trees of this method potentially hampers to find a matrix reordering that evidences an implicit pattern in the data set. In this sense, we opted for discarding restrictions whose insertion at a PQR tree creates an R node.

Our new algorithm (which we call Smoothed Multiple Binarization - or SMB, for short) has the following steps:

1. Reorder the input matrix by Multiple Binarization algorithm, with a slight difference in its Step 5: if adding a restriction to the tree results in a new R node, then discard this tree and use the last tree, which has no R nodes.
2. Smooth the reordered matrix with a mean filter;
3. Repeat Step 1 in the smoothed matrix.
4. Use rows and columns order generated by Step 3 to reorder input matrix, and return it as the result.

Figure 1 exemplifies these steps.

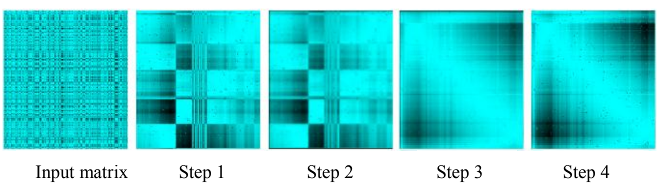

Figure 1: Example of reordering by SMB on a Band Matrix.

Figure 2 exemplifies SMB and MB outputs. We compared them to the output of other two algorithms: classical Multidimensional Scaling (MDS) (which Wilkinson points out as a good reordering algorithm [11]) and Barycenter Heuristic (BH) [6]. This figure shows matrices created according to some of Wilkinson's canonical data patterns (Circumplex, Band, Equi and Simplex) [11], with size 650 × 1000, cells with real values between 0 and 100 and noise ratio of 1% in the data set. We used the following delimiter values for each algorithm and pattern:

- For **MB and SMB**: Circumplex: {45, 50, 55, 60, 99}, Band: {45,50,55,60,65,70,75,80,85,90,95,99}, Equi: {10, 15, 20, 25, 30, 35, 40, 45, 50, 60, 70, 80, 90, 95}, Simplex: {50, 40};
- For **BH**: Circumplex: {99}, Band: {80}, Equi:{50}, Simplex: {50}.

Comparing these images, we observed that SMB was the unique to evidence Circumplex pattern, while MDS and BH try to create a Band pattern in this case. On Band and Simplex patterns, MDS and BH produced the best observed results, followed by SMB; MB returned a poor result. SMB and MDS provided the best results for Equi, at a cost of defining many delimiters for SMB (which impacts SMB performance).

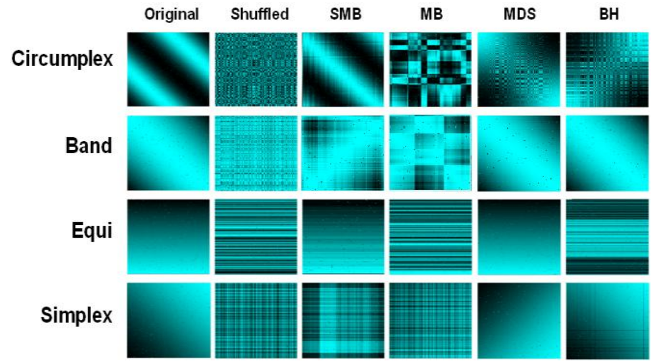

Figure 2: Reordering of the canonical Pattern by algorithms: SMB, MB, MDS and BH.

4 CONCLUSION

We conclude that Smoothed Multiple Binarization has a potential to produce better results than our previous proposal. We also conclude that its result on reordering Circumplex pattern was closer to the original matrix than the result of other methods; MDS and BH results could lead to misinterpretation of the dataset.

Future works include comparing our new algorithm with other reordering algorithms, in terms of statistically measured output qualities and execution time, using MRA tool [9].

REFERENCES

[1] M. Ankerst et al. Similarity Clustering of Dimensions for an Enhanced Visualization of Multidimensional Data. *Proceedings of IEEE Symposium on Information Visualization*, pages 52-60, 1998.

[2] C-H. Chen. Generalized Association Plots: Information Visualization Via Iteratively Genereated Correlation Matrices. *Statistica Sinica*, Volume 12, pages 7-29, 2002.

[3] N. Elqmvist et al. Zame: Interactive Large-Scale Graph Visualization. *Proceeding of IEEE Pacific Visualization Symposium*, March 2008.

[4] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. 3ª Edition, Prentice Hall, 2010.

[5] I. Liiv. Seriation and Matrix Reordering: An Historical Overview. *Wiley Interscience*, 2010.

[6] E. Mäkinen and H. Siirtola. The Barycenter Heuristic and the Reorderable matrix. *Informaton Visualization*, pages 32-48, 2005.

[7] B. F. Medina and C. G. Silva. Using PQR trees for reordering quantitative matrix. Proceedings of *Computer Graphics, Visualization, Computer Vision and Image Processing - MCCSIS*, July 2014.

[8] M. F. Melo. *Improvement of PQR-Sort algorithm for binary matrices reordering*. Master's thesis. School of Technology, University of Campinas, 2012.

[9] C. G. Silva et al. PQR-Sort - Using PQR-Tree for binary matrix reorganization. *Journal of the Brazilian Computer Society*, 20:3, January 2014.

[10] G.P. Telles and J. Meidanis. Building PQR-Tree in almost-linear time. *Eletronic Notes in Discrete Mathematics,* 19:33-39, January 2005.

[11] L. Wilkinson. Patterns Analysis. *The Grammar of Graphics, Second Edition*, pages 523-529, 2005.