Clustering of documents from a two-way viewpoint

Simona Balbi, Raffaele Miele, Germana Scepi

Dipartimento di Matematica e Statistica, Università "Federico II" di Napoli
I-80127 Napoli - Italy

Abstract

Methods for high-dimensional data clustering represents a prolific research area in data mining, encouraging a large quantity of provisional solutions. In text mining and in the analysis of gene expression data, the idea of bidimensional clustering arose, in the sense of finding clusters of documents characterized by cluster of terms (and analogously, clusters of genes and clusters of different experimental conditions). Although we are often more interested in clustering one way of our data structure, however co-clustering seems to be convenient (both from an interpretative and a computational viewpoint). Here we try to frame the problem in a multidimensional data analysis perspective, referring to classic association and/or prediction indexes for contingency tables. Following previous works, we propose the use of a predictability index, Goodman and Kruskal τ_b, dealing with documents-by-terms tables. After a quick review of the wide literature related to two-way clustering, mainly developed in microarray analysis, we propose a new algorithm belonging to the genetic family, based on the optimization of the predictability index τ_b. We present the results of our proposal applied to well-known-in-literature data sets to test the effectiveness of our co-clustering algorithm in practice.

Keywords: two-way clustering, Goodman&Kruskal τ_b, Genetic Algorithms

1. Introduction

Creating categories and classifying objects in categories is the basis of knowledge. From an operational viewpoint, with respect to the huge quantity of data nowadays available in any field, it makes necessary to develop methods and techniques for making sense of data, in a Knowledge Discovery in Data Base perspective, i.e. mapping low-level data into more compact, more abstract, and more useful forms.

High-dimensional data clustering and related problems is a prolific research area in data mining. Therefore a large quantity of algorithms and procedures have been proposed. In text mining, clustering techniques are fundamental tools for reducing the huge amount of textual data to be explored. Moreover, the usual bag-of-words coding, and the common reference to the vector space model, immediately shows the usefulness of multidimensional data analysis tools, for this aim (and the development of Latent Semantic Analysis, Deerwester et al., 1990, seems a confirmation).

Here we try to frame the problem in a multidimensional data analysis perspective, first of all discussing how to statistically represent the data structure to be analyzed, and how to choose a proper similarity measure, and the index for determining the element belonging to a cluster.

All those elements are necessary for proposing an appropriate co-clustering algorithm which has to be applied for the analysis of large unstructured databases, as usual dealing with documents.

In this paper, we consider the problem of simultaneous clustering, or co-clustering, of documents and words (section 2). In particular, we deal with the choice of a criterion to be optimized during the co-clustering procedure (section 3). Therefore, we propose a new co-clustering algorithm (section 4) belonging to the genetic family, based on the local optimization of Goodman and Kruskal index, τ_b (1954) , for solving a text categorization task. We test the performance of our algorithm on two well know data sets: Medline (1033 medical abstracts) and Cranfield (1399 aeronautical systems abstracts), downloadable at the site ftp://ftp.cs.cornell.edu/pub/smart.

2. Co-clustering approaches

Two-way clustering, *co-clustering* or biclustering are clustering methods where the rows and columns of the data matrix are clustered simultaneously. The original idea was finding clusters of similar elements possessing similar clusters of features. The first reference to both clustering rows and columns is Hartigan (1972), proposing what was later named a "sequential approach", that means clustering successively and independently the rows and the columns of the starting matrix (see Berkhin, 2006, for a survey from a data mining perspective).

However, it is often desirable to co-cluster or simultaneously cluster both dimensions of a matrix by exploiting the duality between rows and columns. There are many advantages in a simultaneous rather than a sequential approach (Van Mechelen et al., 2004).

First, the mathematical structures or models as implied by several simultaneous clustering methods cannot be reduced to a simple concatenation or addition of constituent row and column clustering. Several simultaneous clustering methods imply the optimization of an overall objective function that cannot be reduced to a simple combination of constituent row and column objective functions.

Secondly, and more importantly, a simultaneous clustering may highlight the association between the row and column clustering that appears from the data analysis as a linked clustering.

Furthermore, simultaneous approaches allow the researcher to characterize the nature of the interaction or of the dependence structure between rows and columns, as implied by the data.

Finally, as Dhillon et al. (2003) suggest, even if we are interested in clustering along one dimension of the contingency table, when dealing with sparse and high-dimensional data (as it is always the case when we deal with word-document matrices), it turns out to be beneficial to employ co-clustering.

In literature, there are very heterogeneous simultaneous clustering methods proposed by different authors. Given that for all available two-mode clustering methods the implied row and column clustering are of the same type, three families of methods will be distinguished: a) imply row and column partitioning, b) nested row and column clustering and c) overlapping row and column clustering.

In the following we motivate our choice to limit our interest to the class of methods that imply row/column partitions, and focus our attention to contingency tables.

Partitions consist of a certain number of non-empty, nonintersecting clusters that span the full set under consideration. All these methods imply a partition (I_1, ..., I_r, ..., I_R) of rows (*R*), a partition (J_1, ..., Jc, ..., J_C) of columns (*C*) and a data clustering that is a partition of *RxC* as obtained by fully crossing the row and column partitioning. Therefore the rows and columns of the data matrix are permuted such that all row and column clusters consist of neighbouring elements.

In this family of methods, it is possible to distinguish among deterministic, stochastic and procedural approach on the basis of level of modelling and criterion of optimization. Here we focus our attention on the deterministic approach.

Generally, deterministic methods are block modelling approaches based on the assumption that after an appropriate permutation of rows and columns of the starting matrix, the reconstructed data take the form of a block diagonal matrix. The procedure looks for an optimal approximating block diagonal matrix where the loss in inertia is minimal.

In the special case of contingency tables, this means maximize the association between the row and column classes, where the strength of association is captured by the classical χ^2 measure:

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J}\frac{\left(f_{ij} - f_{i.}f_{.j}\right)^2}{f_{i.}f_{.j}} \tag{1}$$

with f_{ij} the relative frequency in the ij cell, $f_{i.}$ (and $f_{.j}$) denoting the marginal row (column) frequencies.

Similarly, we may consider the block frequencies w_{rc} for all data clusters, with associated χ^2 value:

$$\chi^2 = \sum_{r=1}^{R}\sum_{c=1}^{C}\frac{\left(w_{rc} - w_{r.}w_{.c}\right)^2}{w_{r.}w_{.c}} \tag{2}$$

The optimal partitioning minimizes the difference between Equations (1) and (2), or, equivalently, maximizes the χ^2 (Greenacre, 1988) in (2).

Bock (2003a; 2003b) has shown that the loss function (2) is a member of a broad family of loss functions, involving a convex function of the partition class centroids and various measures can be used instead of the χ^2 as a criterion, such as, for instance, the Kullback-Leibler information measure.

In all this approaches a symmetric two-mode clustering method is applied.

3. Co-clustering words and documents from a statistical perspective

As clearly stated in Dhillon (2001), although when we are interested in one-way clustering, i.e. either document or word clustering, the common theme among existing algorithms is to cluster documents based upon their word distributions while word clustering is determined by co-occurrence in documents. This duality suggests to bring the problem in a co-clustering scheme: clusters of words introduce “context” in clustering documents.

In order to clearly define the frame of our proposal, we start by solving some preliminary questions from a statistical perspective.

First of all, how to structure the data. Here we adopt Lebart et al. (1998) viewpoint: in analysing a corpus, the statistical unit is given by the occurrence of a word in a document. Therefore, the data structure to be dealt with is the peculiar contingency table cross-classifying *words by documents*. This choice has some consequences in terms of metrics and criterion. Following again Lebart et al. (1998), in a Correspondence Analysis frame, distances between rows (or columns) are computed using the so called χ^2-metrics, a peculiar weighted Euclidean metrics.

Let **T** be the contingency table cross-classifying *D* documents d_i and *K* terms t_k with general element f_{ik} ($i = 1, \dots, D$; $k = 1, \dots, K$), the distance d between two documents d_i and *d'* is given by:

$$\mathrm{d}(d_i, d_{i'}) = \sum_{k=1}^{k} \frac{1}{f_{.k}} \left(\frac{f_{ik}}{f_{i.}} - \frac{f_{i'k}}{f_{i'.}} \right)^2 \tag{3}$$

And, analogously, for terms *k* and *k'*:

$$\mathrm{d}(t_k, t_{k'}) = \sum_{i=1}^{D} \frac{1}{f_{i.}} \left(\frac{f_{ik}}{f_{.k}} - \frac{f_{ik'}}{f_{.k'}} \right)^2 \tag{4}$$

(Co-)clustering in a contingency table may be seen from two different viewpoints: we can define a similarity measure for rows (and columns) of the table, as it is usual when we deal with individual-variable matrices. Or we can choose a criterion to be optimised during the clustering procedure, let us say referring to the behaviour of a proper association measure. It is well-known that this two approaches are the two sides of a coin, from an analytical viewpoint. But from an algorithmic perspective, things can be completely different, in terms of computational burden. When we have huge data sets, working on dissimilarity or distance matrices can be very expensive, in procedures re-computing the dissimilarity (distance) matrices at each step, when two elements are merged.

Therefore if we only need to re-compute the chosen index, our algorithm is more efficient. Additionally, it is sometimes possible to introduce well-known results related to the behaviour of an index well established in literature.

In the latter approach, it is important the choice of the criterion. Following a co-clustering approach, as described in section 2, we think that it is better to choose Goodman and Kruskal τ_b as criterion, enhancing the idea of minimizing the prediction error.

3.1. The choice of τ_b

The choice of the index is strictly connected to the peculiar definition of "association" proper for our research interest. With a clustering aim, we stress the "causality" running in one direction, therefore asymmetrical indexes have to be preferred.

Here we propose the use of Goodman and Kruskal's τ_b. Suppose that a population is cross-classified with respect to two classifications: variable I with *I* categories and variable J with *J* categories. Goodman and Kruskal proposed τ_b as a measure of predictability, that is a measure for quantifying how much the knowledge of J helps in predicting the I categories. Indeed, if we want to predict a value of I and we don't have any information on J, we consider the marginal distribution of I and the prediction is based on the marginal frequency. But, if we have some information on J, the prediction can be improved by looking at the conditional distribution of I, given J. Therefore, the prediction will be the value of the conditional distribution.

In our co-clustering approach, we are particularly interested in the predictability measured by τ_b which computes the decrease in the prediction error when the prediction is based on the conditional distribution instead of the marginal distribution.

Let **F** be a contingency table, with general element f_{ij}, the relative frequency of the joint distribution of the polytomy I with *I* categories in row (*i*=1, …, *I*) and the polytomy J with *J* categories in column (*j*=1, …, *J*), trying to guess *i*, knowing *j*:

$$\tau_b = \frac{\sum_i \sum_j f_{ij}^2 / f_{.j} - \sum_j f_{i.}^2}{1 - \sum_i f_{i.}^2} = \frac{\sum_i \sum_j \frac{\left(f_{ij} - f_{i.} f_{.j}\right)^2}{f_{.j}}}{1 - \sum_i f_{i.}^2} \tag{5}$$

This index takes values between 0 (if independence) and 1 (if the knowledge of *j* completely determines *i*, for any *i* and *j*). It is indeterminate if all f_{ij}*'s* but one, are 0 (and this can be the case with very sparse matrices, as in text mining, before pre-processing of the analysed corpus).

Being the τ_b determinator the quantity decomposed during a non symmetrical correspondence analysis, we can refer to analytical results for knowing the effects on τ_b, by merging two categories (Balbi, 1994).

4. The proposed co-clustering algorithm

4.1. Genetic Algorithms

Genetic Algorithms (GA) are stochastic procedures that provide a random-search based alternative to traditional optimization methods by using powerful search techniques to locate near optimal (and, sometimes, optimal) solutions in complex optimization problems. They can be briefly described as stochastic algorithms whose search methods model some natural phenomena based on genetic inheritance and natural selection. GA perform multidirectional search by maintaining a population of potential solutions and assuring knowledge formation and exchange between the directions (Michalewicz, 1996). The potential solutions to a problem evolve to a better-fit group of solutions in the sense of an objective function. At each generation the better solutions reproduce, while the relatively bad solutions eventually die off. GA's have been successfully applied to many real world optimization problems like scheduling processes and the travelling salesman problems. GA's have the following properties:

- Work with encoding of the parameters (chromosomes).
- Search by means of a population of potential solutions.
- Use an evaluation (fitness) function that does not require the calculation of derivatives.
- Search stochastically.

A genetic algorithm is described by in Fig. 1:

```
procedure genetic algorithm

    begin
        choose a coding to represent variables
        t ← 0
        initialise population P(t)
        evaluate population P(t)
        while (not termination condition)  do
            t ← t +1
            select P(t) from P(t-1)
            alter P(t) with crossover and mutation
            evaluate P(t)
        end
    end
```

Figure 1: A genetic algorithm

In order to use a Genetic Algorithm (GA) it is necessary to define:

- A genetic representation for parameters of the problem
- An evaluation (fitness) function
- genetic operators (crossover, mutation) altering the population.

Values for parameters used by the algorithm (population size, numbers of generations, probabilities to apply genetic operators, selective pressure, etc.).

For most applications (in particular when the objective function and the search domain are not too complex) the most critical point is the one related to the encoding of the parameters, while the others can usually be easily accomplished or chosen by empirical evidence.

4.2. A Genetic Algorithm for extracting co-clusters

The co-clustering problem can be seen in the following way: given a lexical table with D documents and K words we want to find some checkerboard structure like the one shown in Fig. 2.

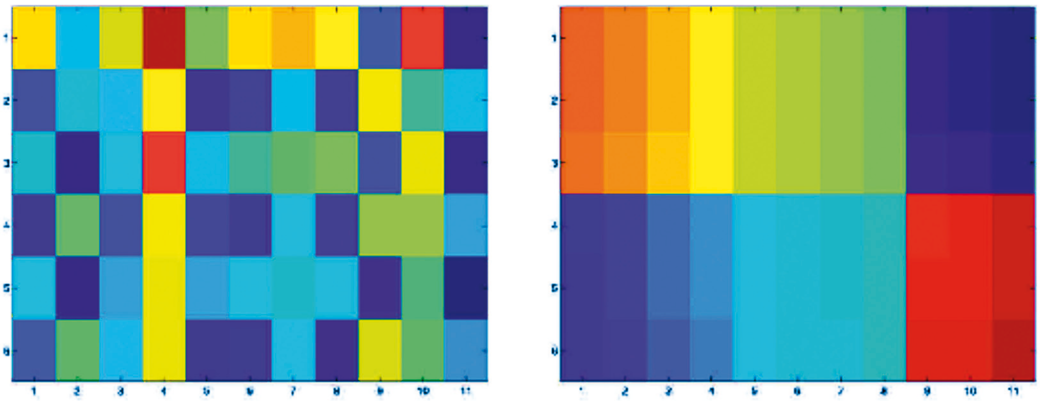

Figure 2: Representing a co-clustering procedure

This representation partitions the lexical table in the following way: cells with the same greyscale belong to a group (a co-cluster) that maximizes some quality measure (i.e. a criterion). A way to represent a co-cluster could be making use of a couple of binary strings $S^r = \{i_1, i_2, \dots, i_r\}$ and $S^c=\{i_1, i_2,..., i_c\}$ whose element say if an element (a document or a word) belong to the cluster or not.

Under these conditions, finding a set of co-clusters means looking for a set of different strings representing subsets of the initial matrix. In this way a Genetic Algorithm could be used to look for co-clusters that maximize the measure of association chosen before, in our case Goodman and Kruskal's τ_b.

Following this approach, the proposed algorithm looks for one cluster at time. More than one co-cluster are found by running the algorithm many times.

In order to avoid the finding of the same solution more than once, a taboo list is written for keeping in memory all the clusters found in the previous steps.

If we are looking for non overlapping co-clusters, solutions containing an already assigned element, are discarded. In case of overlapping solutions, any time a candidate contains a predefined amount of rows and columns in common with a co-cluster that has been previously found, only the one with the highest fitness is kept and the other is discarded. The second option (to get overlapping solutions) has not been implemented yet.

The algorithm consists of the following steps :

- Define *k* (number of co-clusters to be found) and ε (precision threshold)
- Load the data matrix and perform, if necessary, any pre-processing operations
- Initialise the initial population (the set of candidate solutions) randomly
- While [fitness(actual best) – fitness(best of past iteration)] > ε
  - Calculate the fitness measure (the τ_b) on all the candidate solutions
  - Select the set of solutions to use to generate the next population
  - Obtain the new solutions by applying mutation and crossover on the selected candidates
  - Discard all the elements, according to the taboo list and the searching criterion (overlapping, non overlapping)
- End while
- Add the found solution to taboo list
- Return to step 2 until a stopping criterion is met (number of clusters to be found).

The algorithm has been implemented in Java language and runs on a PC-workstation with 8 GB of memory.

5. An experiment on MEDLINE and CRANFIELD data

MEDLINE (Medical Literature Analysis and Retrieval System Online) is a bibliographic database of life sciences and biomedical information. It includes bibliographic information on articles from academic journals covering medicine, nursing, pharmacy, dentistry, veterinary medicine and health care. MEDLINE also covers much of the literature in biology and biochemistry, as well as fields such as molecular evolution. MEDLINE is freely available on the Internet and searchable via PubMed and NLM's National Center for Biotechnology Information's Entrez system. The database contains more than 18 million records from approximately 5,000 selected publications covering biomedicine and health from 1950 to the present. MEDLINE uses Medical Subject Headings (MeSH) for information retrieval. Engines designed to search MEDLINE (such as Entrez and PubMed) generally use a Boolean expression combining MeSH terms, words in abstract and title of the article, author names, date of publication, etc. Both Entrez and PubMed allow also to find articles similar to a given one based on a mathematical scoring system that takes into account the similarity of word content of the abstracts and titles of two articles.

Cranfield data base contains over 6,000 bibliographic references to staff publications since 1996 and covers journal articles, conference papers, books, book chapters, reports, etc., concerning with Aerodynamics. It is updated monthly.

We conducted experiments on the available document collections Medline and Cranfield ftp://ftp.cs.cornell.edu/pub/smart, usually adopted for testing algorithms developed both for clustering and for retrieving information. The Medline document collection contains a total of 1,033 documents indexed by 14,569 terms. So it forms a term-by-document matrix of size 14,569 × 1,033 with rank 1,033. The Cranfield data collection contains 1,398 documents indexed by 11,058 terms. Hence it forms a term-by-document matrix of size 11,058 × 1,398 with rank 1,398.

For testing our algorithm, we merge the two collections, considering 200 documents for each collection. The joint vocabulary consists of 7,121 terms. After removing hapaxes, words with one or two characters, numbers, and a list of commonly suggested stopwords and words occurring with a frequency higher than 400, our final dataset is a collection of 400 documents and 3,395 terms. Therefore our term by document matrix has a size equal to 400 × 3,395 = 1,358,000 cells.

We ran our algorithm on this dataset by fixing the number of cluster to search to 2 (the original data sets).

All the Genetic Algorithms parameters have been tuned by simulation:

- Population size: 400
- Crossover type: single-point crossover
- Crossover rate: 0.6
- Mutation rate: 0.04

Since the documents come from two distinct set and we know which are the true class labels (MEDLINE and CRANFIELD) it is possible to produce a confusion matrix, which is shown in Tab. 1.

	Medline	Cranfield
C_1	171	29
C_2	37	163

Table 1: Our confusion matrix

Where C1 and C2 are the two clusters that the algorithm was looking for. These very preliminary results show that the algorithm is able to discriminate data coming from two different corpora and the same behavior have been encountered on similar situations. Further experiments have to be carried out in order to properly compare our proposal with other ones in literature. However, the required amount of iterations is still very high and this is due to the computationally intensive nature of the Genetic Algorithm.

6. Conclusions and further developments

This paper proposes a new algorithm for co-clustering textual data by means of a predictability index, Goodman and Kruskal τ_b, in order to stress the different role played by the two ways of the lexical table, when we are interested in categorizing documents.

Our experimental results have shown a good performance of the algorithm in bi-partitioning documents in clustering belonging to two well-known-in-literature data sets.

However, the algorithm has slow convergence and therefore some work can be directed in its parallel implementation. Furthermore it can be important to use local search-based tuning of the solutions in order to enhance the Genetic Algorithm performances.

From a textual data analysis viewpoint, pre-processing the textual data should be carefully performed, together with a clever thinking to the proper weighting system to be applied to words. Further developments will be devoted to deepen those preliminary steps. Moreover careful comparisons of the performance of our algorithm and other proposals in literature, under different conditions, could be helpful for refining our proposal. By inverting the asymmetry of our index, we think that clustering words with respect to documents can represent an interesting development, useful in introducing quantitative approaches to qualitative survey tools, e.g. for analyzing focus groups results.

Finally**,** we aim to introduce an index for evaluating the goodness of obtaining clustering or respect an external classification (in the sense of external validity) or by generalizing internal indexes proposed in the frame of one-way clustering.

References

Balbi S. (1994). Influence and stability in non symmetrical correspondence analysis. *Metron*, 3/4, 52, pp. 111-128.

Balbi S. (1995). Confidence regions in factorial representations for textual data with non symmetrical correspondence analysis. In Bolasco, S., et al, editors, *JADT1995*, CISU, Roma, 2, pp. 5-12.

Berkhin P. (2006). A Survey of Clustering Data Mining Techniques. In Kogan, Nicholas and Teboulle, editors, *Grouping Multidimensional Data Recent Advances in Clustering*, Berlin-Heidelberg, Springer, pp. 25-71.

Bock H.H. (2003a). Two-way clustering for contingency tables: maximizing a dependence measure. In Schader, M. et al, editors, *Between data science and applied data analysis. Studies in Classification, Data Analysis, and Knowledge Organization*, Heidelberg, Springer, pp.143-154.

Bock H.H. (2003b). Convexity based clustering criteria: theory, algorithm and applications in statistics. *Statistical Methods and Applications*, 12: 293-318.

Deerwester S., Dumais S., Furnas G.W., Landauer T.K. and Harshman R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41/6: 391-407.

Dhillon I.S. (2001). Coclustering documents and words using Bipartite Spectral Graph Partitioning. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, San Francisco, California, pp. 269-274.

Dhillon I.S., Mallela S. and Modha D.S. (2003). Information-theoretic co-clustering. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (SIGKDD '03). ACM Press.

Goodman L.A. and Kruskal W.H. (1954). Measures of association for cross classifications. *Journal of the American Statistical Association*, 49: 732-764.

Greenacre M.J. (1988). Clustering the rows and columns of a contingency table. *Journal of Classification*, 5: 39-51.

Hartigan J.A. (1972). Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67: 123-129.

Lebart L., Salem A. and Berry L. (1998). *Exploring textual data*. Dordrecht: Kluwer Academic.

Michalewicz Z. (1996). *Genetic algorithms + data structures = evolution programs*. Berlin-Heidelberg: Springer, Gmbh & Co.

van Mechelen I., Bock H.-H. and Boeck P. de (2004). Two-mode clustering methods: a structured overview. *Statistical Methods in Medical Research*, 13: 363-394.