STATISTICS IN TRANSITION-new series, Summer 2012
Vol. 13, No. 2, pp. 335—342

ENSEMBLE APPROACH FOR CLUSTERING OF INTERVAL-VALUED SYMBOLIC DATA

Marcin Pełka[1]

ABSTRACT

Ensemble approach has been applied with a success to regression and discrimination tasks [see for example Gatnar 2008]. Nevertheless, the idea of ensemble approach, that is combining (aggregating) the results of many base models, can be applied to cluster analysis of symbolic data.
The aim of the article is to present suitable ensemble clustering based on symbolic data. The empirical part of the paper presents results simulation studies (based on artificial data sets with known cluster structure) of ensemble clustering based on co-occurrence matrix for symbolic interval-valued data, compared with single clustering method. The results are compared according to corrected Rand index.

Key words: Ensemble clustering; interval-valued symbolic data.

1. Introduction

Ensemble techniques based on aggregating information (results) from different models have been applied with a success in context of supervised learning (discrimination and regression). The ensemble techniques are applied in order to improve the accuracy and stability of classification algorithms (Breinman 1996).

Ensemble clustering means combining (aggregating) N base clustering results (models) $P_1,\ldots,P_N$ into one model P^* with k^* clusters (see: Fred and Jain 2005).

Recently several studies on combination method have established a new area in classical taxonomy. Nevertheless, the idea of ensemble approach, that is combining (aggregating) the results of many base models, can be applied to cluster analysis of symbolic data.

There are several proposals of applying the idea of ensemble approach in the context of clustering – aggregation of results of different clustering algorithms,

[1] Department of Econometrics and Computer Science, Wroclaw University of Economics. E-mail: marcin.pelka@ue.wroc.pl.

receiving different partitions by resampling the data, applying different subsets of variables, applying a given algorithm many times with different values of parameters or different initializations.

2. Symbolic data

Symbolic objects, unlike classical objects, can be described by many different symbolic variable types. Bock and Diday have defined five different symbolic variable types (Bock and Diday 2000, p. 2) – see table 1 for examples of symbolic variables:

1) single quantitative value,
2) categorical value,
3) quantitative value of interval type,
4) set of values or categories (multivalued variable),
5) set of values or categories with weights (multivalued variable with weights),
6) modal interval-valued variable proposed in Billard and Diday (Billard and Diday 2006).

Regardless of their type symbolic variables also can be the following (Bock and Diday 2000, p. 2):

1) taxonomic – which present prior known structure,
2) hierarchically dependent – rules which decide if a variable is applicable or not have been defined,
3) logically dependent – logical rules which affect variable's values have been defined.

Table 1. Examples of symbolic variables

Symbolic variable	Realizations	Variable type
preferred price of a new car (in PLN)	<25000; 36000>, <28000; 37000>, <30000; 50000>, <33000; 58000>, <65000; 80000>, <66000; 90000>	interval-valued (non-disjoint)
engine capacity	<1000; 1200>, (1200; 1400>, (1400; 1600>, (1600; 1800>, (1800; 2000>, (2000; 2200>	interval-valued (disjoint)
colour	{green, black, yellow, red, purple, blue}	multivalued
preferred brand of a car	{60% Honda, 35% Toyota, 5% Audi} {40% Honda, 20% Skoda, 20% Toyota, 20% Audi} {80% Audi, 15% Opel, 5% Toyota}	multivalued with weights

Source: Own research.

There are two main symbolic objects types:

1. First order objects (simple objects, individuals) – single respondent, product, company, etc., described by symbolic variable types. This objects are individuals that are symbolic by their nature.

2. Second order objects (aggregate objects, super individuals) – more or less homogeneous classes, groups of individuals described by symbolic variables.

3. Ensemble clustering methods

There are two main approaches that can be applied in ensemble learning for symbolic interval-valued data (see: Gathemi *et al.* 2009; De Carvalho *et al.* 2012; Hornik 2005):

1. Clustering algorithm for multiple relational matrices – proposed by De Carvalho *et al.* 2012. This approach is based on different distance matrices. Those distance matrices can be obtained by applying different distance measures, or subsets of variables or subsets of objects. Distance matrices are used to calculate relevance weight vectors. Relevance weight vectors and distance matrices are then applied to cluster a set of objects into k clusters.
2. Clustering ensemble that apply consensus functions in clustering ensembles. There are five main consensus functions that are applied in clustering ensemble.

Hypergraph partitioning which assumes that clusters can be represented as hyperedges on a graph. Their vertices correspond to the objects to be clusters. Each hyperedge describes a set of objects belonging to the same cluster. The problem of consensus clustering is reduced to finding the minimum-cut of a hypergraph (Gathemi *et. al.* 2009, p. 638; Strehl and Gosh 2002). Different adaptations of hypergraph partitioning have been proposed by Strehl and Gosh (2002), Fern and Brodley (2004), Ng *et al.* (2002).

The main idea of the **voting approach** is to permute cluster labels in such a way that best agreement between the labels of two partitions is obtained. All the partitions from the cluster ensemble must be relabelled according to a fixed reference partition. This reference partition can be taken from the ensemble or from a new clustering of the data set. Fisher and Buhman, and Dudoit and Fridlyand have presented a combination of partitions by relabeling and voting (Gathemi *et al.* 2009, p. 639).

Mutual information approach assumes that the objective function of a clustering ensemble can be formulated as the mutual information between the empirical probability distribution of labels in the consensus partition and the labels in the ensemble. In this approach usually a generalized definition of mutual information is applied – for example in Topchy *et al.* (2003). Luo *et al.* (2006) have introduced consensus scheme via genetic algorithm based on information theory. Azimi *et al.* (2007) have proposed clustering ensemble method which generates a new feature space from initial clustering outputs (Gathemi *et al.* 2009, p. 640).

In the **finite mixture model** approach the main assumption is that the output labels are modelled as random variables drawn from probability distribution described as a mixture of multinomial component densities. The objective of consensus clustering is formulated as a maximum likelihood estimation. Usually the expectation maximization algorithm (EM) is used to solve the maximum likelihood problem. Such approach is presented by Topchy *et al.* (2004), Analoui and Sadighian (2006) (Gathemi *et al.* 2009, p. 641).

The **co-association based functions** operate on the co-association (co-occurrence) matrix. Numerous clustering methods can be applied to co-association matrix to obtain the final partition. By applying different clustering methods, resampling the data, different subsets of variables, or the same clustering with different values of parameters or initializations we obtain N partitions (each can have different number of clusters) of set E (set of objects to be classified):

$$\begin{aligned} P^1 &= \left\{C_1^1, C_2^1, \ldots, C_{k_1}^1\right\}, \\ &\vdots \\ P^N &= \left\{C_1^N, C_2^N, \ldots, C_{k_N}^N\right\} \end{aligned} \tag{1}$$

The algorithm of ensemble clustering that uses co-association matrix can be described as follows (Fred and Jain 2005, p. 848):

a) obtain different base partitions,

b) build the co-association matrix (co-occurrence matrix). The main idea of this matrix is that objects belonging to the same clusters ("natural clusters") are likely to be co-located in the same clusters in different partitions. The elements of the co-association matrix are defined as follows:

$$C(i,j) = \frac{n_{ij}}{N}, \tag{2}$$

where: i, j – pattern (objects) numbers, n_{ij} – number of times pattern (i, j) is assigned to the same cluster among N partitions, N – total number of partitions,

c) apply the co-association matrix as the data matrix for some classical clustering method – like single-link, average, *k*-means or pam,

d) choose the best partition. Fred and Jain (2002) propose to apply "lifetime" criterion in the case of hierarchical clustering methods. They define lifetime as the value of threshold values on the dendrogram that leads to the identification of k clusters – their suggestion is to look for the highest value of this threshold.

Also other methods that will lead to identification of the final number of clusters can be applied – for example Baker & Hubert, Hubert & Levine, Russeeuw's silhouette cluster quality indices (see for example Gatnar and Walesiak 2004, p. 342-343 for details).

4. Results of simulation studies

In order to compare the results of single clustering method (single model) with results of ensemble clustering the adjusted Rand index was applied in the case of single clustering method. In the case of ensemble clustering average ensemble accuracy (that is based on adjusted Rand index) is applied. Average ensemble accuracy can be defined as follows:

$$A_{agr} = \frac{1}{K}\sum_{k=1}^{K} AR\left(P_k^{agr}, P'\right), \tag{3}$$

where: K – number of ensembles, AR – adjusted Rand index, P_k^{agr} – classification on the base of k-th ensemble, P' – known class labels.

The individual accuracy is defined as follows:

$$A_i = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{J}\sum_{j=1}^{J} AR\left(P_k^j, P'\right), \tag{4}$$

where: J – number of ensemble members, P_j^k – classification on the base of j-th member of k-th ensemble.

To compare the results of single clustering methods with results of ensemble clustering four different artificial data sets where generated (models are obtained by applying culsterSim and mlbench packages of R software):

1. **Data set I** – 120 symbolic objects in three elongated clusters described by two interval-valued variables. The observations are independently drawn from bivariate normal distribution with means $(0, 0), (1.5, 7), (3, 14)$ and covariance matrix Σ $(\sigma_{jj} = 1, \sigma_{jl} = -0.9)$.

2. **Data set II** – 120 symbolic objects divided into five clusters in three dimensions that are not well separated. The observations are independently drawn from multivariate normal distribution with means equal to: $(5, 5, 5), (-3, 3, -3), (3, -3, 3), (0, 0, 0), (-5, -5, -5)$, and covariate matrix Σ, where $\sigma_{jj} = 1\,(1 \le j \le 3)$, and $\sigma_{jl} = 0.9\ (1 \le j \ne l \le 3)$.

To obtain symbolic interval data for data sets I and II the data were generated for each model twice into sets *A* and *B* and minimal (maximal) value of $\left\{x_{ij}^A, x_{ij}^B\right\}$ is treated as the beginning (the end of interval).

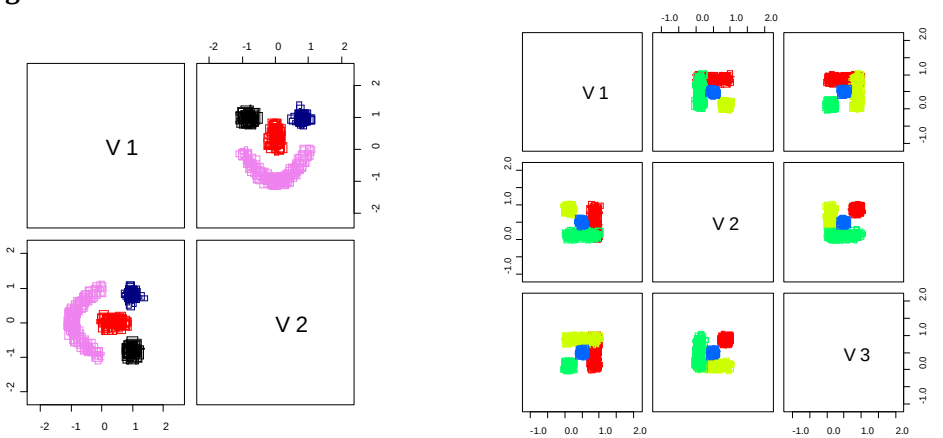
3. **Data set III** – is an adaptation of well-known cuboids data set (from mlbench package). Four clusters in three dimensions.

4. **Data set IV** – is an adaptation of well-known smiley data set (from mlbench package). Four clusters in two dimensions.

In order to build interval-valued variables from mlbench cuboids and smiley data sets the data obtained from mlbench package is treated as the "seed" of a rectangle. Each rectangle is therefore a vector of two intervals defined by:

$([z_j - \gamma_j / 2, z_j + \gamma_j / 2])$, where z_j – is the value of variable for j-th variable, γ_j – is the width and the height of the rectangle for j-th variable. The value γ_j is drawn randomly from the interval $[0, 1]$ for each variable. The figure 1 presents data sets III and IV.

Figure 1. Data sets III and IV



a) b)

a) – data set IV (smiley); b) – data set III (cuboids)

Source: own computations in R software.

To determine the final number of clusters Rousseeuw's Silhouette, Baker & Hubert, Hubert & Levine cluster quality indices were used (Ichino-Yaguchi distance measure was applied). The most common result was taken into consideration. Results of clustering with application of single model and ensemble clustering results for each data set (with application of adjusted Rand index) are presented in table 2.

Table 2. Results of clustering for four data sets

Clustering approach	**Data set I**		**Data set II**		**Data set III**		**Data set IV**	
	Number of clusters	Rand index	Number of clusters	Rand index	Number of clusters	Rand index	Number of clusters	Rand index
Single method:								
- single link	2	1	2	0.1744	11	0.3314	10	0.2212
- average link	2	1	2	0.3961	3	0.2943	2	0.1627
- pam	2	1	2	0.3786	3	0.3171	2	0.2302

Table 2. Results of clustering for four data sets (cont.)

Clustering approach	Data set I		Data set II		Data set III		Data set IV	
	Number of clusters	Rand index	Number of clusters	Rand index	Number of clusters	Rand index	Number of clusters	Rand index
Ensemble approach: - different clustering methods applied; number of clusters chosen at random from the interval [2; 15]	2	1	5	1	4	0.8266	4	0.8457

Source: Own research with application of R software.

5. Final remarks

Ensemble clustering methods that were developed to deal with classical data situation can be quite easily adapted to symbolic data situation. Ensemble clustering methods based on the co-association (co-occurrence) matrix can be applied to cluster symbolic interval-valued data.

Symbolic interval-valued data often tends to form not well-separated clusters of many different shapes. Single clustering methods (hierarchical, divisive or iterative) not always can detect correct number of clusters. Ensemble approach in clustering can be a solution to these problems.

For the purposes of simulation studies a R script was written by author. It allows co-occurrence matrix to be built and applied as the data matrix for any suitable clustering method.

Simulation studies have shown that ensemble clustering based on co-association matrix achieves better results (in terms of adjusted Rand index) than single clustering methods – especially when dealing not typical cluster structures, or not-well separated clusters.

The most important aims for future work are: comparing ensemble clustering based on co-association matrix with other ensemble clustering approaches, do more simulation studies on ensemble learning for symbolic data.

REFERENCES

BILLARD, L., DIDAY E. (2006). Symbolic Data Analysis: Conceptual Statistics and Data Mining, Wiley, Chichester.

BOCK, H.-H., DIDAY, E. (red.) (2000). Analysis of symbolic data. Explanatory methods for extracting statistical information from complex data, Springer Verlag, Berlin-Heidelberg.

BREIMAN, L. (1996). Bagging predictors, Machine Learning, 24(2), p. 123-140.

DE CARVALHO, F.A.T., LECHEVALLIER, Y., DE MELO, F.M. (2012). Partitioning hard clustering algorithms based on multiple dissimilarity matrices, Pattern Recognition, 45(1), p. 447-464.

FERN, X.Z., BRODLEY, C.E. (2004). Solving cluster ensemble problems by bipartite graph partitioning, Proceedings of the 21st International Conference on Machine Learning, Canada.

FRED, A.L.N., JAIN, A.K. (2005). Combining multiple clustering using evidence accumulation, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 27, p. 835-850.

GAHEMI, R., SULAIMAN, N., IBRAHIM, H., MUSTAPHA, N. (2009). A survey: Clustering ensemble techniques [in:] Proceedings of World Academy of Science, Engineering and Technology, Vol. 38, p. 636-645.

GATNAR, E. (2008). Podejście wielomodelowe w zagadnieniach dyskryminacji i regresji, Wydawnictwo Naukowe PWN, Warszawa.

GATNAR, E., WALESIAK, M. (red.) (2004). Metody statystycznej analizy wielowymiarowej w badaniach marketingowych, Wydawnictwo AE, Wrocław.

HORNIK, K. (2005). A clue for cluster ensembles, Journal of Statistical Software, 14, 65-72.

NG, A., JORDAN, M., WEISS, Y. (2002). On spectral clustering: analysis and an algorithm, [In:] T. Dietterich, S. Becker, Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14, MIT Press, 849-856.

STREHL, A., GHOSH, J. (2002). Cluster ensembles – A knowledge reuse framework for combining multiple partitions, Journal of Machine Learning Research, 3, p. 583-618.