STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume L, Number 1, 2005

CORE BASED INCREMENTAL CLUSTERING

GABRIELA ŞERBAN AND ALINA CÂMPAN

Abstract. Clustering is a data mining activity that aims to differentiate groups inside a given set of objects, with respect to a set of relevant attributes of the analyzed objects. Generally, existing clustering methods, such as *k-means* algorithm, start with a known set of objects, measured against a known set of attributes. But there are numerous applications where the attribute set characterizing the objects evolves. We propose in this paper an incremental, *k-means* based clustering method, *Core Based Incremental Clustering (CBIC)*, that is capable to re-partition the objects set,when the attributes set increases. The method starts from the partitioning into clusters that was established by applying *k-means* or *CBIC* before the attribute set changed. The result is reached more efficiently than running *k-means* again from the scratch on the feature-extended object set. Experiments proving the method's efficiency are also reported.

Keywords: Data Mining, clustering, k-means.

1. Introduction

Unsupervised classification, or clustering, as it is more often referred as, is a data mining activity that aims to differentiate groups (classes or clusters) inside a given set of objects. The inferring process is carried out with respect to a set of relevant characteristics or attributes of the analyzed objects. The resulting groups are to be built so that objects within a cluster to have high similarity with each other and low similarity with objects in other groups. Similarity and dissimilarity between objects are calculated using metric or semi-metric functions, applied to the attribute values characterizing the objects.

A large collection of clustering algorithms is available in the literature. [5] and [6] contain comprehensive overviews of existing techniques.

A well-known class of clustering methods is the one of the partitioning methods, with representatives such as the *k-means* algorithm or the *k-medoids* algorithm.

Received by the editors: March 15, 2005.

2000 *Mathematics Subject Classification.* 62H30, 68U35.

1998 *CR Categories and Descriptors.* 62H30 [**Statistics**]: Multivariate analysis – *Classification and discrimination; cluster analysis*; 68U35 [**Computer science**]: Computing methodologies and applications – *Information systems (hypertext navigation, interfaces, decision support, etc.)*;

Essentially, given a set of n objects and a number $k, k \leq n$, such a method divides the object set into k distinct and non-empty partitions. The partitioning process is iterative and heuristic; it stops when a "good" partitioning is achieved. A partitioning is "good", as we said, when the intra-cluster similarities are high and inter-cluster similarities are low.

Generally, these methods start with a known set of objects, measured against a known set of attributes. But there are numerous applications where the object set is dynamic, or the attribute set characterizing the objects evolves. Obviously, for obtaining in these conditions a partitioning of the object set, the clustering algorithm can be applied over and over again, beginning from the scratch, every time when the objects or attributes change. But this can be unefficient. What we want is to propose an incremental, *k-means* based clustering method, named *Core Based Incremental Clustering (CBIC)*, that is capable to efficiently re-partition the objects set, when the attributes set increases with one new attribute. The method starts from the partitioning into clusters that was established by applying *k-means* or *CBIC* before the attribute set changed. The result is reached more efficiently than running *k-means* again from the scratch on the feature-extended object set.

2. Formal Problem Study

Let $\{O_1, O_2, \ldots, O_n\}$ be the set of objects to be classified. Each object is measured with respect to a set of m initial attributes and is described therefore by a m-dimensional vector $O_i = (O_{i1}, \ldots, O_{im}), O_{ik} \in \Re, 1 \leq i \leq n, 1 \leq k \leq m$. Usually, the attributes associated to objects are standardized, in order to ensure an equal weight to all of them ([6]).

Let $\{K_1, K_2, \ldots, K_p\}$ be the set of clusters discovered in data by applying the *k-means* algorithm. Each cluster is a set of objects, $K_j = \{O_1^j, O_2^j, \ldots, O_{n_j}^j\}, 1 \leq j \leq p$. The centroid (clusters mean) of the cluster K_j is denoted by f_j, where $f_j = \left(\frac{\sum\limits_{k=1}^{n_j} O_{k1}}{n_j}, \ldots, \frac{\sum\limits_{k=1}^{n_j} O_{km}}{n_j} \right)$.

The measure used for discriminating objects can be any *metric* function, d. We used the *Euclidian distance*: $d(O_i, O_j) = d_E(O_i, O_j) = \sqrt{\sum\limits_{l=1}^{m} (O_{il} - O_{jl})^2}$.

The measured set of attributes is afterwards extended with one new attribute, the $(m+1)$ or last attribute. After extension, the objects' vectors become $O'_i = (O_{i1}, \ldots, O_{im}, O_{i,m+1}), 1 \leq i \leq n$.

We want to analyze the problem of recalculating the objects grouping into clusters, after object extension and starting from the current partitioning. We want to obtain a performance gain in respect to the partitioning from scratch process.

We start from the fact that, at the end of the initial clustering process, all objects are closer to the centroid of their cluster than to any other centroid. So, for any cluster j and an object $O_i^j \in K_j$, inequality (1) holds.

$$d_E(O_i^j, f_j) \leq d_E(O_i^j, f_r), 1 \leq r \leq p, r \neq j. \tag{1}$$

We denote by $K_j', 1 \leq j \leq p$ the set containing the same objects as K_j, after the extension. By $f_j', 1 \leq j \leq p$ we denote the mean (center) of the set K_j'. These sets $K_j', 1 \leq j \leq p$ will not necessarily represent clusters after the attribute-set extension. The newly arrived attribute can change the objects arrangement into clusters, formed so that the intra-cluster similarity to be high and inter-cluster similarity to be low. But there is a considerable chance, when adding one or few attributes to objects, and the attributes have equal weights and normal data distribution, that the old arrangement in clusters to be close to the actual one. The actual clusters could be obtained by applying the *k-means* classification algorithm on the set of extended objects. But we try to avoid this process and replace it with one less expensive but not less accurate. With these being said, we agree, however, to continue to refer the sets K_j' as clusters.

We therefore start by taking as reference point the previous partitioning in clusters and study in which conditions an extended object $O_i^{j}{}'$ is still correctly placed in its cluster K_j'. For that, we express the distance of $O_i^{j}{}'$ to the center of its cluster, f_j', compared to the distance to the center f_r' of any other cluster K_r'.

Theorem 1. When inequality (2) holds for an extended object $O_i^{j}{}'$ and its cluster K_j'

$$O_{i,m+1} \geq \frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} \tag{2}$$

then the object $O_i^{j}{}'$ is closer to the center f_j' than to any other center $f_r', 1 \leq r \leq p, r \neq j$.

Proof

We prove below this statement.

$$d^2(O_i^{j}{}', f_j') - d^2(O_i^{j}{}', f_r') = d^2(O_i^j, f_j) + \left(\frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} - O_{i,m+1}\right)^2 - d^2(O_i^j, f_r) - \left(\frac{\sum\limits_{k=1}^{n_r} O_{k,m+1}}{n_r} - O_{i,m+1}\right)^2.$$

Using the relation in (1), we have:

$$d^2(O_i^{j\prime}, f_j') - d^2(O_i^{j\prime}, f_r') \le \left(\frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} - O_{i,m+1}\right)^2 - \left(\frac{\sum\limits_{k=1}^{n_r} O_{k,m+1}}{n_r} - O_{i,m+1}\right)^2 \Leftrightarrow$$

$$d^2(O_i^{j\prime}, f_j') - d^2(O_i^{j\prime}, f_r') \le \left(\frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} - \frac{\sum\limits_{k=1}^{n_r} O_{k,m+1}}{n_r}\right) \cdot \left(\frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} + \frac{\sum\limits_{k=1}^{n_r} O_{k,m+1}}{n_r} - 2 \cdot O_{i,m+1}\right).$$

If the relation in (2) holds for $O_i^{j\prime}$, then the inequality above becomes:

$$d^2(O_i^{j\prime}, f_j') - d^2(O_i^{j\prime}, f_r') \le -\left(\frac{\sum\limits_{k=1}^{n_j} O_{k,m+1}}{n_j} - \frac{\sum\limits_{k=1}^{n_r} O_{k,m+1}}{n_r}\right)^2 \Leftrightarrow$$

$$d^2(O_i^{j\prime}, f_j') - d^2(O_i^{j\prime}, f_r') \le 0.$$

Because all the distances are non-negative, it results that

$$d(O_i^{j\prime}, f_j') \le d(O_i^{j\prime}, f_r').$$

3. The *Core Based Incremental Clustering* Algorithm

We will use the property enounced in the previous paragraph in order to identify inside each cluster $K_j', 1 \le j \le p$ those objects that have a considerable chance to remain stable in their cluster. We will use these *cluster cores* as seed for clustering.

Definition 1. We denote by $Core_j = \{O_i^{j\prime} | O_i^{j\prime} \in K_j', d(O_i^{j\prime}, f_j') \le d(O_i^{j\prime}, f_r'), 1 \le r \le p, r \neq j\}$ the set of all objects in K_j' that are closer to f_j' than to any other center f_r'. We denote by $CORE$ the set $\{Core_j, 1 \le j \le p\}$ of all clusters cores.

All objects in $Core_j$ will surely remain together in the same group if clusters do not change. This will not be the case for all core objects, but for most of them.

We give next the Core Based Incremental Clustering algorithm.

We mention that the algorithm stops when the clusters from two consecutive iterations remain unchanged or the number of steps performed exceeds the maximum number of iterations allowed.

```
Algorithm Core Based Incremental Clustering is
Input: - the set $X = \{O_1, \ldots, O_n\}$ of m-dimensional objects previously
         clustered,
       - the set $X' = \{O_1', \ldots, O_n'\}$ of (m+1)-dimensional extended objects
         to be clusterized, $O_i'$ has the same first m components as $O_i$,
       - the metric $d_E$ between objects in a multi-dimensional space,
       - p, the number of desired clusters,
       - $F = \{F_1, \ldots, F_p\}$ the previous partitioning of objects in $X$.
       - $noMaxIter$ the maximum number of iterations allowed.
Output: - the re-partitioning $F' = \{F_1', \ldots, F_p'\}$ for the objects in $X'$
```

```
Begin
  For all clusters $F_j \in F$
     Calculate $Core_j = \{O_i^{j'} \in F_j'$ that satisfies inequality (2)$\}$
     $F_j' := Core_j$
     Calculate $f_j'$ as the mean of objects in $Core_j$
  EndFor
  While ($F'$ changes between two consecutive steps) and
        (there were not performed $noMaxIter$ iterations) do
     For all clusters $F_j'$ do
        $F_j' := \{O_i' \mid \forall f_r'\ d(O_i', f_j') \leq d(O_i', f_r')\}$
     EndFor
     For all clusters $F_j'$ do
        $f_j' :=$ the mean of objects in $F_j'$
     EndFor
  EndWhile
End.
```

4. Results and Evaluation

In this section we present some experimental results obtained after applying the CBIC algoritm described in section 3.

For this purpose, we had used a programming interface for non-hierarchical clustering described in ([1]). We have to mention that using this interface we can simply develop non-hierarchical clustering applications for different kind of data (objects to be clusterized). As it is shown in our experiments, the objects to be clusterized are very different (patients, wine instances).

As a case study, for experimenting our theoretical results described in section 2 and for evaluating the performance of the CBIC algorithm, we consider some experiments that are briefly described in the following subsections.

We have to mention that all data were taken from the website at "http://www.cormactech.com/neunet".

As a quality measure we take the movement degree of the core objects. More stable they are, better was the decision to choose them as cores for the incremental clustering process. We express the *core stability factor* as:

$$CSF(CORE) = \frac{\sum\limits_{j=1}^{p} \frac{|Core_j|}{no\ of\ clusters\ where\ the\ objects\ in\ Core_j\ ended}}{\sum\limits_{j=1}^{p} |Core_j|} \tag{3}$$

The worst case is when each object in $Core_j$ ends in a different final cluster, and this happens for every core in CORE. The best case is that every $Core_j$ remains compact and it is found in a single final cluster. So, the limits between which CSF varies are given below, where the higher the value of CSF is, better was the cores choise:

$$\frac{p}{\sum\limits_{j=1}^{p} |Core_j|} \leq CSF(CORE) \leq 1 \tag{4}$$

4.1. **Experiment 1. Cancer.** The breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

The objects to be clusterized in this experiment are patients: each patient is identified by 9 attributes [2].The attributes have been used to represent instances. Each instance has one of 2 possible classes: benign or malignant. In this experiment are 457 patients (objects).

The attribute information used in the "cancer" experiment is shown in Table 1.

Table 1. Attribute information in the "cancer" experiment

	Attribute	Domain
1.	Clump Thickness	1 - 10
2.	Uniformity of Cell Size	1 - 10
3.	Uniformity of Cell Shape	1 - 10
4.	Marginal Adhesion	1 - 10
5.	Single Epithelial Cell Size	1 - 10
6.	Bare Nuclei	1 - 10
7.	Bland Chromatin	1 - 10
8.	Normal Nucleoli	1 - 10
9.	Mitoses	1 - 10

4.2. **Experiment 2. Dermatology.** The file for this experiment was obtained from the website at "http://www.corma-ctech.com/neunet".

The objects to be clusterized in this experiment are also patients: each patient is identified by 34 attributes, 33 of which are linear valued and one of them is nominal. There are 366 objects (patients).

The aim of the clustering process is to determine the type of Eryhemato-Squamous Disease [3].

The differential diagnosis of erythemato-squamous diseases is a real problem in dermatology [7]. They all share the clinical features of erythema and scaling, with very little differences. The diseases in this group are psoriasis, seboreic dermatitis, lichen planus, pityriasis rosea, cronic dermatitis, and pityriasis rubra pilaris.

Usually a biopsy is necessary for the diagnosis but unfortunately these diseases share many histopathological features as well. Another difficulty for the differential diagnosis is that a disease may show the features of another disease at the beginning stage and may have the characteristic features at the following stages.

Patients were first evaluated clinically with 12 features. Afterwards, skin samples were taken for the evaluation of 22 histopathological features. The values of the histopathological features are determined by an analysis of the samples under a microscope.

In the dataset constructed for this domain, the family history feature has the value 1 if any of these diseases has been observed in the family, and 0 otherwise. The age feature simply represents the age of the patient. Every other feature (clinical and histopathological) was given a degree in the range of 0 to 3. Here, 0 indicates that the feature was not present, 3 indicates the largest amount possible, and 1, 2 indicate the relative intermediate values.

4.3. **Experiment 3. Wine.** The file for this experiment was obtained from the website at "http://www.corma-ctech.com/neunet".

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines [4].

The objects to be clusterized in this experiment are wine instances: each is identified by 13 attributes. There are 178 objects (wine instances).

We have to mention that all attributes in this experiment are continuous.

4.4. **Results.** In this section we present comparatively the results obtained after applying the CBIC algorithm for the experiments described in the above subsections.

TABLE 2. The comparative results

Experiment	No. of objects	No. of attributes (m+1)	No. of iterations for m+1 attributes	No. of iterations for m attributes	No. of iterations for m+1 attributes using CBIC	The cores' stability factor CSF(CORE)
Cancer	457	9	13	10	8	0.804347826
Dermatology	366	34	7	11	5	0.713114754
Wine	178	13	4	6	3	1.0

From Table 2 we observe that using the CBIC algorithm the number of iterations for finding the solution is smaller, and also the cores' stability factor, CSF(CORE), is high.

5. Conclusions and Future Work

We proposed in this paper a new method for adapting a clustering when the attribute set describing the objects increases by one. The experiments on different data sets prove that the result is reached more efficiently using the proposed method than running *k-means* again from the scratch on the feature-extended object set.

Further works can be done in the following directions:

- to experiment the theoretical results in the case in which more attributes (that characterize the objects) are added;
- how can the theoretical results described for non-hierarchical clustering be applied/generalized for other clustering techniques.

References

[1] Şerban, G.: "A Programming Interface for Non-Hierarchical Clustering", Studia Universitatis "Babeş-Bolyai", Informatica, XLX(1), 2005, to appear.

[2] Wolberg, W., Mangasarian, O.L.: "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp 9193–9196.

[3] Demiroz, G., Govenir, H. A., Ilter, N.: "Learning Differential Diagnosis of Eryhemato-Squamous Diseases using Voting Feature Intervals", Artificial Intelligence in Medicine.

[4] Aeberhard, S., Coomans, D., de Vel, O.: "THE CLASSIFICATION PERFORMANCE OF RDA" Tech. Rep. no. 92–01, 1992, Dept. of Computer Science and Dept. of Mathematics and Statistics, James Cook University of North Queensland.

[5] Jain, A., Dubes, R, "Algorithms for Clustering Data", Prentice Hall, Englewood Cliffs, New Jersey, 1998.

[6] Han, J., Kamber, M. "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2001.

[7] http://www.cormactech.com/neunet, "Discover the Patterns in Your Data", CorMac Technologies Inc, Canada.

Babeş Bolyai University, Cluj Napoca,Romania
E-mail address: gabis@cs.ubbcluj.ro

Babeş Bolyai University, Cluj Napoca, Romania
E-mail address: alina@cs.ubbcluj.ro