DISCRETIZATION BASED ON CLUSTERING METHODS

Daniela Joiţa
Titu Maiorescu University, Bucharest, Romania
daniela.joita@utm.ro

Abstract. *Many data mining algorithms require as a pre-processing step the discretization of real-valued data. In this paper we review some discretization methods based on clustering. We describe in detail the algorithms of discretization of a continuos real-valued attribute using the hierarchical graph clustering methods.*

Keywords: discretization, agglomerative clustering, divisive clustering

1. INTRODUCTION

For a given data mining problem one might want to use different data mining techniques and do a cross-validation to find the "best" data mining solution. Since many data mining techniques often require that the attributes of the data sets are discrete and, one would like to try, probably, all the techniques that apply to the problem, it is very important to have algorithms for discretization of continuous data attributes. Also, given that most of the experimental data are continuous, not discrete, the discretization of the continuous attributes is indeed an important issue.

There is a large variety of discretization methods. Dougherty et al. (1995) [4] present a systematic survey of all the discretization method developed by that time. They also make a first classification of discretization methods based on three directions: global vs. local, supervised vs. unsupervised, static vs. dynamic.

In [4] five discretization methods were compared: two unsupervised global methods (equal width and equal frequency interval), two supervised global methods (1RD (Holte 1993) and Fayyad & Irani's (1993) entropy minimisation)), and a supervised local method (the classification algorithm C4.5).

This paper is organized as follows. In the next section we present the definition of discretization of a continuous attribute. In Section 3 we present the clustering concept, the classification of clustering methods as they appear in the speciality literature and some important clusterization methods: k-means, least squares Fisher and hierarchical methods. In Section 4 we describe the algorithms of discretization of a real-valued attribute using the hierarchical graph clustering methods. The paper also contains a conclusion section.

2. DISCRETIZATION

A discrete data attribute can be seen as a function whose range is a finite set, while a continuous data attribute as a function whose range is an infinite totally ordered set, usually an interval. To discretize a continuous data attribute means to find a partition of the range of that attribute into a finite number of intervals [3].

Usually, the discretization process consists of two steps[3]. First, the number of discrete intervals needs to be chosen. Even though there are discretization methods which determine this number in the discretization process, this step is done usually by the user either by some heuristic techniques or by running the discretization technique for different number of intervals and deciding what is the best choice by using a criterion. Second, the cut points must be determined, which is often done by a discretization algorithm itself.

Let give a formal definition of discretization [2]. Let A be an attribute of a finite data set D. Let n be the number of examples in D. We denote by *adom (A)* the set of all values of the attribute A in the data set D, called the *active domain of A* and by $a=(a_1,a_2,\ldots,a_n)$ the vector of all values of the attribute A for all n examples. To discretize the numeric attribute A means to find a partition of *adom(A)*. This implies to determine the cut points $t_0,t_1,\ldots,t_k$ with $t_0<t_1<\ldots<t_k$ such that the set $\{P_1,P_2,\ldots,P_k\}$ forms a partition of *adom(A)*, where P_i is defined by $P_i=\{a\in adom(A): t_{i-1}\le a<t_i\}$ for $i=\overline{0,k-1}$ and $P_k=\{a\in adom(A): t_{k-1}\le a\le t_k\}$ and $t_0=\min adom(A)$ and $t_k=\max adom(A)$.

After the discretization is perfomed, the attribute A is replaced by the discretized attribute A^{disc} whose values are defined as follows:

$A^{disc}=(a^{disc}{}_1,a^{disc}{}_2,\ldots,a^{disc}{}_n)$, $a^{disc}{}_j=i$ iff $a_j\in P_i$ for $j=\overline{1,n}$.

Therefore each value of the attribute A which falls in P_i is replaced by i.

3. CLUSTERING METHODS

Clustering is an important step in the process of data mining. Roughly speaking, clustering means grouping data in clusters, in groups which contain similar data. In other words, given a set of data examples (points) and a similarity measure (distance), the purpose of clustering is to search for similar points and group them into clusters such that the distance between examples within cluster is as small as possible and the distance between clusters is as large as possible. Due to the large databases that are used in the data mining process, the grouping of data that behave similarly in the same group and therefore dividing the databases in smaller groups is computational eficient.

In [8] is presented the following classification of the clustering problems: hard clustering and fuzzy clustering. In hard clustering, a data point belongs to one and only one cluster, while in fuzzy clustering, a data point may belong to two or more clusters with some probabilities. In this paper we will illustrate only the hard clustering methods. These also can be divides into two categories: hierarchical algorithms which create a sequence of nested clusters until the desired number of clusters is found and partitional algorithms that create an one-level partition of the data examples.

The problem of choosing the right number of clusters is very important for all the clustering methods. In practice, usually one runs the clustering algorithm for several different number of clusters and finds the "best" number based on some measure of "goodness" of clustering.

Next we will illustrate few clustering methods.

3.1. K-means clustering method

The *k-means clustering method* remains one of the most popular clustering method. This algorithm has been identified by the IEEE International Conference on Data Mining (ICDM) in December 2006 as a top 10 algorithm, being considered among the most influential data mining algorithms in the research community[15]. The algorithm has been discovered by several researchers across different disciplines, among which, most notably mentioned in the above survey paper, Lloyd (1957, 1982) , Forgy (1965) [6], Friedman and Rubin (1967), and MacQueen (1967) [13].

Given k the desired number of clusters, the algorithm has two steps: the initialization step, in which k examples are chosen randomly as the initial centers of the k clusters and the data points are assigned each to the closest cluster, and the iteration step, in which the centers of the clusters are computed as the average points of all the data points in the cluster and the other data points are reassigned each to the closest cluster until the number of reassingments is less than a small constant. The algorithm is designed such that the objective sum of squares function over the partition of the data points into the clusters *1, 2, …, k* :

$$SSD = \sum_{i=1}^{k} \sum_{x \in cluster\, i} d(x, C_i)^2$$

gets minimized where C_i is the center of the cluster i, and d is the distance measure.

3.2. Least-squares Fisher method

The goal of the least-squares Fisher method described in the one-dimensional case, is the minimization of the same objective sum of squares function *SSD*. A partition of the data points that minimizes *SSD* is called the least-squares partition. To find this partition, Fisher proves that it has to be a contigous partition [7] i.e. if x, y, z are three data points ordered such that $x < y < z$ and x and z are in the same cluster, let say i, then y is also in the same cluster i. The method is based on the following lemma[7]:

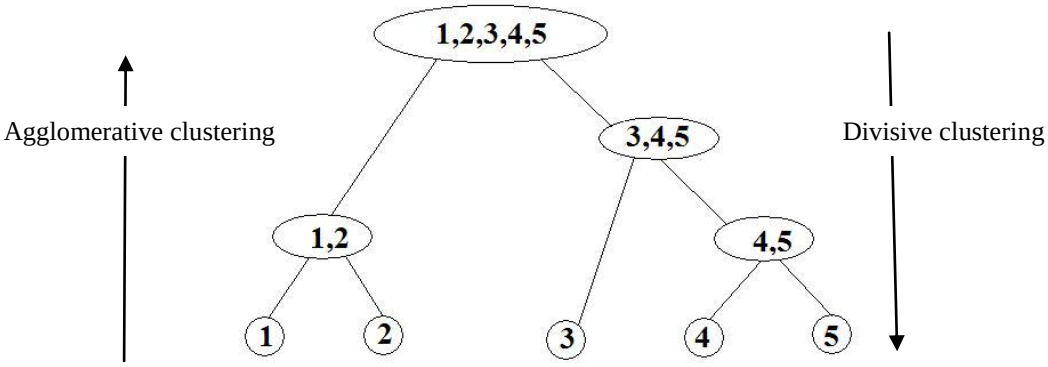
Fisher's Suboptimization Lemma: *If A_1:A_2 denotes a partition of a set A into two disjoint subsets A_1 and A_2, if P_1^* denotes a least squares partition of A_1 into G_1 subsets and if P_2^* denotes a least squares partition of A_2 into G_2 subsets; then, of the class of subpartitions of A_1:A_2 employing G_1 subsets over A_1 and G_2 subsets over A_2 a least subpartition is P_1^*: P_2^*.*

3.3. Hierarchical methods

Hierarchical methods can be divided into two groups: agglomerative and divisive. In the agglomerative case the clustering starts with every data point in one cluster and at each stage the „best" two clusters to be grouped together are determined until the desired number of clusters is found. In the divisive case, the clustering starts with all data points in one big cluster and at each stage the „best" cluster to be divided into smaller clusters is chosen and the process continues until the desired number of clusters is determined. Hierarchical clustering

methods are reprezented usually by a diagram called the dendogram which shows how the clusters are modified over the clustering process.

Dendogram example:



For agglomerative methods, the „best" two clusters to be grouped together are chosen such that the distance between them is as small as possible, i.e. we choose *R* and *S* the best clusters such that

$$D(R,S) = \min_{r,s\ clusters} D(r,s)$$

The reason there are few agglomerative methods, not just one, is because the distance between clusters can be calculated differently. Some of the most used hierarchical methods are the graph methods: the single, complete, average, weighted average linkage methods and the geometric methods: the centroid method, the Ward's method, and the median method. The distance between clusters in the case of graph methods is computed according to the formulas:

1. *The Single-link Method:*

$$D(r,s) = \min_{x\in r,\, y\in s} d(x,y)$$

2. *The Complete-link Method:*

$$D(r,s) = \max_{x\in r,\, y\in s} d(x,y)$$

3. *The Group Average Method:*

$$D(r,s) = \frac{1}{|r||s|}\sum_{x\in r,\, y\in s} d(x,y)$$ unde $|r|$ = number of data points in the cluster *r*

For divisive methods, the „best" cluster to be divided is the one that contains the farthest two data points. Therefore if we define the diameter of a cluster *r*: $diam(r) = \max_{x,y\in r} d(x,y)$ and we denote the two farthest points min *r* and max *r*, then we choose *R* to be the cluster that will be divided to be such that $diam(R) = \max_{r\ cluster} diam(r)$. The division of the best cluster *R* into two clusters R_1 and R_2 is made in the following way: $R_1 = \{x \in R \mid d(x,\min R) < d(x,\max R)\}$ and $R_2 = \{x \in R \mid d(x,\min R) \geq d(x,\max R)\}$.

4. DISCRETIZATION BASED ON CLUSTERING

In [1], Anderberg suggests three clustering methods to be used for discretization: one-dimensional hierarchical linkage methods, Ward's hierarchical method and the least squares Fisher method. In [8] the discretization using the *k*-means algorithm, and least squares method are presented in detail.

In [11] we present an unsupervised static discretization method based on the k-means clustering method, different from the classic one by the fact that the values of the attribute need no sorting before the discretization. We propose a technique of choosing the initial centers of the clusters designed specifically for the clustering of a one–dimensional vector of real valued data, to be used in the process of discretization of a single attribute, which avoids the *O(n log n)* time requirement for sorting the data points.

In this paper we describe the discretization based on hierarchical clustering. With the above notations, $a=(a_1,a_2,...,a_n)$ being the vector of all values of the attribute *A* for all *n* examples, let $X=\{x_1,x_2,...,x_n\}$ be the set of all examples ordered using a sorting algorithm (Quicksort, for example), therefore $x_1 \le x_2 \le ... \le x_n$. Since the goal is also the minimization of the objective sum of squares function *SSD*, accordingly to the Fisher Lemma[7], we can assume that the partition of *X* in clusters has to be a contigous partition i.e. if *x*, *y*, *z* are three data points ordered such that $x < y < z$ and *x* and *z* are in the same cluster, let say *i*, then *y* is also in the same cluster *i*. Therefore our purpose is to determine the cut points $t_0,t_1,...,t_k$ with $t_0 < t_1 < ... < t_k$ such that the set $\{P_1,P_2,...,P_k\}$ forms a partition of *adom(A)=[x_1,x_n]* where P_i is defined by $P_i=\{a \in adom(A): t_{i-1} \le a < t_i\}$ for $i=\overline{0,k-1}$ and $P_k=\{a \in adom(A): t_{k-1} \le a \le t_k\}$ and $t_0 = x_1$ and $t_k = x_n$.

Let $d(\cdot,\cdot)$ be the distance function between two data points and $D(\cdot,\cdot)$ be the distance between two clusters and let *k* be the desired number of clusters.We will describe the algoritms of discretization based on the two types of hierarchical clustering.

Input: Vector of real valued data $x=(x_1,x_2,...,x_n)$ with $x_1 \le x_2 \le ... \le x_n$ and the number of clusters to be determined *k*.
Goal: Our goal is to find a partition of the data in *k* distinct clusters.
Output: The set of cut points $t_0,t_1,...,t_k$ with $t_0 < t_1 < ... < t_k$ that defines the discretization of the *[x_1 , x_n]*.

4.1. The discretization method based on the agglomerative clustering can be described in the following way:

The Agglomerative Algorithm:

m = *n* // *m* = current number of clusters
for *i* = 1 to *n* do
  $C_i = \{x_i\}$
  min C_i = max $C_i = x_i$
endfor
while *m* > *k*
  Determine *j* such that $D(C_j,C_{j+1}) = \min_{i=1}^{n-1} D(C_i,C_{i+1})$.
  // Group together the clusters C_j and C_{j+1}:
    $C_j = C_j \cup C_{j+1}$, $\max C_j = \max C_{j+1}$.
  // Renumber the clusters:
  for *i = j + 1* to *k – 1* do
    $C_i = C_{i+1}$, min C_i = min C_{i+1}, max C_i = max C_{i+1}
  endfor
  m = m - 1
endwhile
// Determination of the cut points
$t_0 = x_1$
for *i* = 1 to *k-1* do
  $t_i = \frac{\max C_i + \min C_{i+1}}{2}$
endfor
$t_k = x_n$

The distance between clusters will be computed depending on the clustering method:

1. *The Single-link Method:* $D(C_i,C_{i+1}) = d(\max C_i, \min C_{i+1})$
2. *The Complete-link Method:* $D(C_i,C_{i+1}) = d(\min C_i, \max C_{i+1})$
3. *The Group Average Method:* $D(r,s) = \frac{1}{|r||s|} \sum_{x \in r, y \in s} d(x,y)$

4.2 The discretization method based on the divisive clustering can be described in the following way:

The Divisive Algorithm:

$C_1 = \{x_1, x_2, \ldots, x_n\}$
$m = 1$ // m = current number of clusters
while $m < k$
    Determine C_j such that $diam(C_j) = \max_{i=1,m} diam(C_i)$.
    // Renumber the clusters:
    for $i = m$ downto $j+2$ do
        $C_i = C_{i-1}$, min C_i = min C_{i-1}, max C_i = max C_{i-1}
    endfor
    // Divide the cluster C_j into two clusters:
    $R_1 = \{x \in C_j \mid d(x, \min C_j) < d(x, \max C_j)\}$
    $R_2 = \{x \in C_j \mid d(x, \min C_j) \geq d(x, \max C_j)\}$.
    $C_j = R_1$, $C_{j+1} = R_2$
    $m = m + 1$
endwhile
// Determination of the cut points
$t_0 = x_1$
for i = 1 to $k-1$ do
    $t_i = \frac{\max C_i + \min C_{i+1}}{2}$
endfor
$t_k = x_n$

5. CONCLUSION

We presented the discretization techniques based on the hierarchical clustering algorithms.The sorting of the data is required before the application of the clustering. Future work may include testing the technique against other discretization methods of the same type and of different type.

BIBLIOGRAPHY

1. Anderberg, M., *Cluster Analysis for Applications*, , Academic Press, New York, 1973

2. Butterworth R., *Contributions to Metric Methods in Data Mining*, Doctoral Dissertation, University of Massachusetts Boston, 2006

3. Cios, K., Pedrycz, W., Swiniarski, R., Kurgan, L., *Data Mining A Knowledge Discovery Approach*, Springer , 2007

4. Dougherty, J., Kohavi, R., Sahami, M., *Supervised and Unsupervised Discrimination of Continuous Features*. In: Proceedings of the12th International Conference, Morgan Kaufman, (1995) p. 194-202

5. Fayyad, U. M., & Irani, K. B. *On the Handling of Continuous-valued Attributes in Decision Tree Generation*, Machine Learning 8, (1992). p. 87-102.

6. Forgy E.W., *Cluster analysis of multivariate data: efficiency vs. interpretability of classifications*, Biometrics 21 (1965) , p.768–769.

7. Fisher W. , *On Grouping for Maximum Homogeneity*, Americal Statistical Association Journal (1958), p.789-798

8. Gan, G., Ma, C., Wu, J., *Data Clustering: Theory, Algorithms, and Applications*, ASA-SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia, ASA, Alexandria, VA, 2007.

9. Hartigan, J., Wong, M., *A k-means Clustering Algorithm*, Applied Statistics 28(1979) p.100–108.

10. Ho, K., Scott P., *Zeta: A Global Method for Discretization of Continuous Variables*, In: Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining (1997), p. 191-194

11. Joiţa D., *Unsupervised static discretization methods in data mining*, Conferinţa internaţională "Educaţie şi creativitate pentru o societate bazată pe cunoaştere", Universitatea Titu Maiorescu, secţiunea Ştiinţa şi tehnologia informaţiei, pag. 29-32, Ediţia a II-a, Bucureşti, 20-22 noiembrie, 2008

12. Kurgan, L., Cios, K., *CAIM discretization algorithm*, IEEE Transactions on Knowledge and Data Engineering, (2004) 16, no.2, p.145-153

13. MacQueen J, *Some methods for classification and analysis of multivariate observations*. Proceedings of the 5th Berkeley Symposium on Mathematics, Statistics and Probability 3 (1967), p. 281–297.

14. Witten, I., Eibe, F., *Data Mining. Practical Machine Learning Tools and Techniques*, Second edition, Morgan Kaufman, 2005

15. Wu**,** X**.** et al**,** *Top 10 Algorithms in Data Mining*, Knowledge Information Systems (2008) 14, p.1–37