Discretization of Continuous Attributes

Fabrice Muhlenbach
EURISE, Université Jean Monnet - Saint-Etienne, France

Ricco Rakotomalala
ERIC, Université Lumière - Lyon 2, France

INTRODUCTION

In the data-mining field, many learning methods — such as association rules, Bayesian networks, and induction rules (Grzymala-Busse & Stefanowski, 2001) — can handle only discrete attributes. Therefore, before the machine-learning process, it is necessary to re-encode each continuous attribute in a discrete attribute constituted by a set of intervals. For example, the age attribute can be transformed in two discrete values representing two intervals: less than 18 (a minor) and 18 or greater. This process, known as *discretization,* is an essential task of the data preprocessing not only because some learning methods do not handle continuous attributes, but also for other important reasons. The data transformed in a set of intervals are more cognitively relevant for a human interpretation (Liu, Hussain, Tan, & Dash, 2002); the computation process goes faster with a reduced level of data, particularly when some attributes are suppressed from the representation space of the learning problem if it is impossible to find a relevant cut (Mittal & Cheong, 2002); the discretization can provide nonlinear relations — for example, the infants and the elderly people are more sensitive to illness. This relation between age and illness is then not linear — which is why many authors propose to discretize the data even if the learning method can handle continuous attributes (Frank & Witten, 1999). Lastly, discretization can harmonize the nature of the data if it is heterogeneous — for example, in text categorization, the attributes are a mix of numerical values and occurrence terms (Macskassy, Hirsh, Banerjee, & Dayanik, 2001).

An expert realizes the best discretization because he can adapt the interval cuts to the context of the study and can then make sense of the transformed attributes. As mentioned previously, the continuous attribute "age" can be divided in two categories. Take basketball as an example; what is interesting about this sport is that it has many categories: "mini-mite" (under 7), "mite" (7 to 8), "squirt" (9 to 10), "peewee" (11 to 12), "bantam" (13 to 14), "midget" (15 to 16), "junior" (17 to 20), and "senior" (over 20). Nevertheless, this approach is not feasible in the majority of machine-learning problem cases because there are no experts available, no a priori knowledge on the domain, or, for a big dataset, the human cost would be prohibitive. It is then necessary to be able to have an automated method to discretize the predictive attributes and find the cut-points that are better adapted to the learning problem.

Discretization was little studied in statistics — except by some rather old articles considering it as a special case of the one-dimensional clustering (Fisher, 1958) — but from the beginning of the 1990s, the research expanded very quickly with the development of supervised methods (Dougherty, Kohavi, & Sahami, 1995; Liu et al., 2002). Lately, the applied discretization has affected other fields: An efficient discretization can also improve the performance of discrete methods such as the association rule construction (Ludl & Widmer, 2000a) or the machine learning of a Bayesian network (Friedman & Goldsmith, 1996).

In this article, we will present the discretization as a preliminary condition of the learning process. The presentation will be limited to the global discretization methods (Frank & Witten, 1999), because in a local discretization, the cutting process depends on the particularities of the model construction — for example, the discretization in rule induction associated with genetic algorithms (Divina, Keijzer, & Marchiori, 2003) or lazy discretization associated with naïve Bayes classifier induction (Yang & Webb, 2002). Moreover, even if this article presents the different approaches to discretize the continuous attributes, whatever the learning method may be used, in the supervised learning framework, only discretizing the predictive attributes will be presented. The cutting of the attributes to be predicted depends a lot on the particular properties of the problem to treat. The discretization of the class attribute is not realistic because this pretreatment, if effectuated, would be the learning process itself.

BACKGROUND

The discretization of a continuous-valued attribute consists of transforming it into a finite number of intervals and to re-encode, for all instances, each value on this attribute by associating it with its corresponding interval. There are many ways to realize this process.

One of these ways consists of realizing a discretization with a fixed number of intervals. In this situation, the user must choose the appropriate number a priori: Too many intervals will be unsuited to the learning problem, and too few intervals can risk losing some interesting information. A continuous attribute can be divided in intervals of equal width (see Figure 1) or equal frequency (see Figure 2). Other methods exist to constitute the intervals based on the clustering principles, for example, *k-means clustering discretization* (Monti & Cooper, 1999).

Nevertheless, for supervised learning, these discretization methods ignore an important source of information: the instance labels of the class attribute. By contrast, the supervised discretization methods handle the class label repartition to achieve the different cuts and find the more appropriate intervals. Figure 3 shows a situation where it is more efficient to have only two intervals for the continuous attribute instead of three: It is not relevant to separate two bordering intervals if they are composed of the same class data. Therefore, the supervised or unsupervised quality of a discretization method is an important criterion to take into consideration.

Another important criterion to qualify a method is the fact that a discretization either processes on the different attributes one by one or takes into account the whole set of attributes for doing an overall cutting. The second case, called multivariate discretization, is particularly interesting when some interactions exist between the different attributes. In Figure 4, a supervised discretization attempts to find the correct cuts by taking into account only one attribute independently of the others. This will fail: It is necessary to represent the data with the attributes X1 and X2 together to find the appropriate intervals on each attribute.

MAIN THRUST

The two criteria mentioned in the previous section — unsupervised/supervised and univariate/multivariate — will characterize the major discretization method families. In the following sections, we use these criteria to distinguish the particularities of each discretization method.

Univariate Unsupervised Discretization

The simplest discretization methods make no use of the instance labels of the class attribute. For example, the equal width interval binning consists of observing the values of the dataset to identify the minimum and the maximum values observed and to divide the continuous attribute into the number of intervals chosen by the user (Figure 1). Nevertheless, in this situation, if uncharacteristic extreme values exist in the dataset ("outliers"), the range will be changed, and the intervals will be misappropriated. To avoid this problem, divide the continuous attribute into intervals containing the same number of instances (Figure 2): This method is called the equal frequency discretization method.

The unsupervised discretization can be grasped as a problem of sorting and separating intermingled probability laws (Potzelberger & Felsenstein, 1993). The existence of an optimum analysis was studied by Teicher (1963) and Yakowitz and Spragins (1968). Nevertheless, these methods are limited in their application in data mining due to too strong of statistical hypotheses seldom checked with real data.

Univariate Supervised Discretization

To improve the quality of a discretization in supervised data-mining methods, it is important to take into account the instance labels of the class attribute. Figure 3 shows the problem of constituting intervals without the information of the class attribute. The intervals that are the better adapted to a discrete machine-learning method are the *pure* intervals containing only instances of a given class. To obtain such intervals, the supervised discretization methods — such as the state-of-the-art method Minimum Description Length Principle Cut (MDLPC) — are based on statistical or information-theoretical criteria and heuristics (Fayyad & Irani, 1993).

In a particular case, even if one supervised method can give better results than another (Kurgan & Krysztof, 2004), with real data, the improvements of one method

Figure 1. Equal width discretization

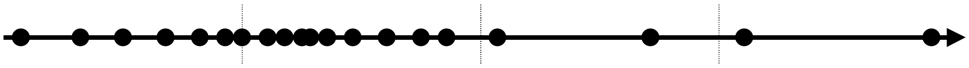

Figure 2. Equal frequency discretization