HAL
open science

Discretization of Continuous Attributes

Fabrice Muhlenbach, Ricco Rakotomalala

To cite this version:

Fabrice Muhlenbach, Ricco Rakotomalala. Discretization of Continuous Attributes. John Wang. Encyclopedia of Data Warehousing and Mining, Idea Group Reference, pp.397-402, 2005. hal-00383757v2

HAL Id: hal-00383757

https://hal.science/hal-00383757v2

Submitted on 13 May 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Discretization of Continuous Attributes

Fabrice Muhlenbach and Ricco Rakotomalala

Fabrice MUHLENBACH

Affiliation: Laboratoire Hubert Curien (ex-EURISE), Université Jean Monnet – Saint-Etienne

Address: Université de Lyon,
CNRS, UMR 5516
Laboratoire Hubert Curien
F-42023, Saint-Etienne, FRANCE

Phone: +33 (0) 477 915 808

Fax: +33 (0) 477 251 817

Email: fabrice.muhlenbach@univ-st-etienne.fr

Ricco RAKOTOMALALA

Affiliation: ERIC, Université Lumière – Lyon 2

Address: Laboratoire ERIC
Université Lumière – Lyon 2 – Bât. L
5 avenue Pierre Mendès-France
69676 Bron Cedex
FRANCE

Phone: +33 (0) 478 772 414

Fax: +33 (0) 478 772 375

Email: ricco.rakotomalala@univ-lyon2.fr

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 –like association rules, Bayesian networks, 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 and more (of age). 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 non-linear relations –e.g., the infants and the elderly people are more sensitive to illness. This relation between age and illness is then not linear– and that is why many authors propose to discretize the data even if the learning method can handle continuous attributes (Frank and Witten, 1999). Lastly, discretization can harmonize the nature of the data if it is heterogeneous –e.g., 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 then he can make sense of the transformed attributes. As mentioned before, the continuous attribute "age" can be divided in two categories ("less than 18" and "18 and more"). Take basketball as an example, what is interesting about this sport is that there are 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 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 nineties the research expanded very quickly with the development of supervised methods (Dougherty, Kohavi & Sahami, 1995; Liu, Hussain, Tan & Dash, 2002). Lately, the applied discretization has affected other fields: an efficient discretization can also improve the performance of discrete methods like the association rule construction (Ludl & Widmer, 2000a) or the machine learning of a Bayesian network (Friedman & Goldsmith, 1996).

In this chapter, the discretization will be presented 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 –e.g., 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 chapter presents the different approaches to discretize the continuous attributes, whatever the learning method may be used– in the supervised learning framework, only to 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 since 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 in realizing a discretization with a fixed number of intervals. In this situation, the user must a priori choose the appropriate number: 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 (figure 1) or equal frequency (figure 2). Other methods exist to constitute the intervals, for example based on the clustering principles –e.g., *K-means clustering discretization* (Monti & Cooper, 1999).

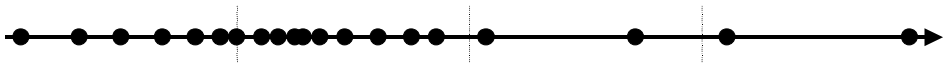

Figure 1 Equal Width Discretization

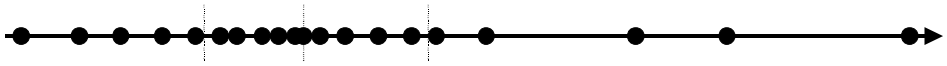

Figure 2 Equal Frequency Discretization

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. The figure 3 shows a situation where it is more efficient to have only 2 intervals for the continuous attribute instead of 3: 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 a overall cutting. The second case, called “multivariate discretization”, is particularly interesting when some interactions exist between the different attributes. On 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.

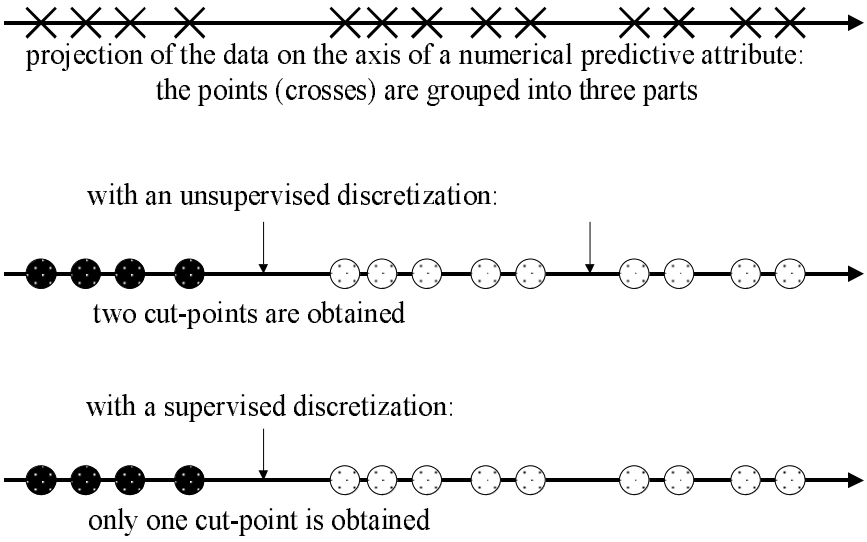

Figure 3 Supervised and Unsupervised Discretizations

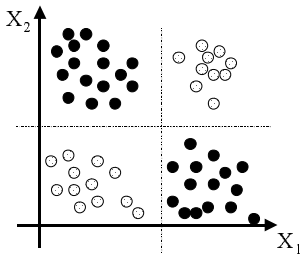

Figure 4 Interaction between the attributes X_1 and X_2

MAIN THRUST OF THE CHAPTER

The two criteria –unsupervised/supervised and univariate/multivariate– mentioned in the previous section will characterize the major discretization method families. In the following section, these criteria will be used 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 uncharacteric extreme values exist in the dataset ("outliers"), the range will be changed and the intervals will be misappropriate. To avoid this problem, it is possible to divide the continuous attribute in intervals containing the same number of instances (figure 2): this method is called *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 (1963). Nevertheless, these methods are limited in their application in data mining due to too strong 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. The 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 *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) however, with real data, the improvements of one method compared to the others supervised methods are insignificant. Moreover, the performance of a discretization method is difficult to estimate without a learning algorithm. In addition, the final results can arise from the discretization processing, the learning processing or the combination of both. Because the discretization is realized in an ad hoc way, independently of the learning algorithm characteristics, there is no guarantee that the interval cut will be optimal for the learning method. Only a little work showed the relevance and the optimality of the global discretization for very specific classifier such as naive Bayes (Hsu, Huang & Wong, 2003; Yang & Webb, 2003). The supervised discretization methods can be distinguished depending on the way the algorithm proceeds: bottom-up (each value represents an interval and they are merged progressively to constitute the appropriate number of intervals) or top-down (the whole dataset represents an interval and it is progressively cut to constitute the appropriate number of intervals). However they are no significant performance differences between these two latest approaches (Zighed, Rakotomalala & Feschet, 1997).

In brief, the different supervised (univariate) methods can be characterized by: (1) the particular statistical criterion used to evaluate the "purity" degree of an interval, (2) the top-down or bottom-up strategy to find the cut-points used to determining the intervals.

Multivariate Unsupervised Discretization

Association rules are an unsupervised learning method that needs discrete attributes. For such a method, the discretization of a continuous attribute can be realized in an univariate way but also in a multivariate way. In the latter case, each attribute is cut in relation to the other

attributes of the database, this approach can then provide some interesting improvements when unsupervised univariate discretization methods do not yield satisfactory results.

The multivariate unsupervised discretizations can be performed by clustering techniques using all attributes globally. It is also possible to consider each cluster obtained as a class and improve the discretization quality by using (univariate) supervised discretization methods (Chmielewski & Grzymala-Busse, 1996).

An approach called *multi-supervised discretization* (Ludl & Widmer, 2000a) can be seen as a particular unsupervised multivariate discretization. This method starts with the temporary univariate discretization of all attributes. Then, the final cutting of a given attribute is based on the univariate supervised discretization of all others attributes previously and temporarily discretized. These attributes play the role of a class attribute one after another. Finally, the smallest intervals are merged. For supervised learning problems, a paving of the representation space can be done by cutting each continuous attribute into intervals. The discretization process consists in merging the bordering intervals in which the data distribution is the same (Bay, 2001). Nevertheless, even if this strategy can introduce the class attribute in the discretization process, it can not give a particular role to the class attribute and can induce the discretization to non-impressive results in the predictive model.

Multivariate Supervised Discretization

When the learning problem is supervised and the instance labels are scattered in the representation space with interactions between the continuous predictive attributes (as presented on figure 4), the methods previously seen will not give satisfactory results. *HyperCluster Finder* is a method that will fix this problem by combining the advantages of the supervised and

multivariate approaches (Muhlenbach & Rakotomalala, 2002). This method is based on clusters constituted as sets of same class instances that are closed in the representation space. The clusters are identified on a multivariate and supervised way: First, a neighborhood graph is built by using all predictive attributes to determine which instances are close to others; Second, the edges connecting two instances belonging to different classes are cut on the graph to constitute the clusters; Third, the minimal and maximal values of each relevant cluster are used as cut-points on each predictive attribute. The intervals found by this method have the characteristic to be "pure" on a pavement of the whole representation space even if the purity is not guaranteed for an independent attribute; It is the combination of all predictive attribute intervals that will provide pure areas in the representation space.

FUTURE TRENDS

Today, the discretization field is well studied in the supervised and unsupervised cases for an univariate process. However there is little work in the multivariate case; There is a related problem in the feature selection domain, which needs to be combined with the aforementioned multivariate case. This should bring improved and more pertinent progress. It is virtually certain that better results can be obtained for a multivariate discretization if all attributes of the representation space are relevant for the learning problem.

CONCLUSION

In a data mining task, for a supervised or unsupervised learning problem, the discretization turned out to be an essential preprocessing step on which will depend the performance of the learning algorithm which uses discretized attributes.

Many methods, supervised or not, multivariate or not, exist to perform this pretreatment, more or less adapted to a given dataset and learning problem. Furthermore, a supervised discretization can also be applied in a regression problem, when the attribute to be predicted is continuous (Ludl & Widmer, 2000b). The choice of a particular discretization method depends (1) on its algorithmic complexity (complex algorithms will take more computation time and will be unsuited to very large datasets), (2) its efficiency (the simple unsupervised univariate discretization methods are inappropriate to complex learning problems), and (3) its appropriate combination with the learning method using the discretized attributes (a supervised discretization is better adapted to supervised learning problem). For the last point, it is also possible to improve significantly the performance of the learning method by choosing an appropriate discretization, for instance a fuzzy discretization for the naive Bayes algorithm (Yang & Webb, 2002). Nevertheless it is unnecessary to employ a sophisticated discretization method if the learning method does not benefit from the discretized attributes (Muhlenbach & Rakotomalala, 2002).

REFERENCES

Bay, S.D. (2001). Multivariate Discretization for Set Mining. *Knowledge and Information Systems, 3(4)*, 491-512.

Chmielewski M.R., & Grzymala-Busse J.W. (1994). Global Discretization of Continuous Attributes as Preprocessing for Machine Learning. *Proceedings of the 3rd International Workshop on Rough Sets and Soft Computing*, 294-301.

Divina F., Keijzer M., & Marchiori E. (2003). A Method for Handling Numerical Attributes in GA-based Inductive Concept Learners. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2003)*, 898-908.

Dougherty K., Kohavi R., & Sahami M. (1995). Supervised and Unsupervised Discretization of Continuous Features. *Proceedings of the 12th International Conference on Machine Learning (ICML-1995)*, 194-202.

Fayyad U.M., & Irani K.B. (1993). The Attribute Selection Problem in Decision Tree Generation. *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1022-1027.

Fisher W.D. (1958). On Grouping for Maximum Homogeneity. *Journal of American Statistical Society, 53*, 789-798.

Frank E., & Witten I. (1999). Making Better Use of Global Discretization. *Proceedings of the 16th International Conference on Machine Learning (ICML-1999)*, 115-123.

Friedman N., & Goldszmidt M. (1996). Discretization of Continuous Attributes while Learning Bayesian Networks from Mixed Data. *Proceedings of 13th International Conference on Machine Learning (ICML-1996)*, 157-165.

Grzymala-Busse J.W., & Stefanowski J. (2001). Three Discretization Methods for Rule Induction, *International Journal of Intelligent Systems, 16*, 29-38.

Hsu H., Huang H., & Wong T. (2003). Implication of Dirichlet Assumption for Discretization of Continuous Variables in Naïve Bayes Classifiers. *Machine Learning, 53(3)*, 235-263.

Kurgan L., & Krysztof J. (2004). CAIM Discretization Algorithm. *IEEE Transactions on Knowledge and Data Engineering, 16(2)*, 145-153.

Liu H., Hussain F., Tan C., & Dash M. (2002). Discretization: An Enabling Technique. *Data Mining and Knowledge Discovery, 6(4)*, 393-423.

Ludl M., & Widmer G. (2000a). Relative Unsupervised Discretization for Association Rule Mining. *Principles of Data Mining and Knowledge Discovery, 4th European Conference (PKDD-2000)*, 148-158.

Ludl M., & Widmer G. (2000b). Relative Unsupervised Discretization for Regression Problem. *Proceedings of 11th European Conference on Machine Learning (ECML-2000)*, 246-253.

Macskassy S.A., Hirsh H., Banerjee A., & Dayanik A.A. (2001). Using Text Classifiers for Numerical Classification. *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-2001)*, 885-890.

Mittal A., & Cheong L. (2002). Employing Discrete Bayes Error Rate for Discretization and Feature Selection Tasks. *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM-2002)*, 298-305.

Monti S., & Cooper G.F. (1999). A latent variable model for multivariate discretization. *The Seventh International Workshop on Artificial Intelligence and Statistics*.

Muhlenbach F., & Rakotomalala R. (2002). Multivariate Supervised Discretization, a Neighborhood Graph Approach. *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM-2002)*, 314-321.

Potzelberger K., & Felsenstein K. (1993). On the Fisher Information of Discretized Data. *The Journal of Statistical Computation and Simulation, 46(3 & 4)*, 125-144.

Teicher, H. (1963). Identifiability of finite mixtures. *Ann. Math. Statist., 34*, 1265-1269.

Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *Ann. Math. Statist., 39*, 209-214.

Yang Y., & Webb G. (2002). Non-Disjoint Discretization for Naïve-Bayes Classifiers. *Proceedings of the 19th International Conference on Machine Learning (ICML-2002)*, 666-673.

Yang Y., & Webb G. (2003) On Why Discretization Works for Naive-Bayes Classifiers. *Proceedings of the 16th Australian Joint Conference on Artificial Intelligence (AI-03)*, 440-452.

Zighed D., Rakotomalala R., & Feschet F. (1997). Optimal Multiple Intervals Discretization of Continuous Attributes for Supervised Learning. *Proceedings of the 3rd International Conference on Knowledge Discovery in Databases (KDD-1997)*, 295-298.

TERMS AND THEIR DEFINITION

Discrete / Continuous attributes: An attribute is a quantity describing an example (or "instance"), its domain is defined by the attribute type which denotes the values taken by an attribute. An attribute can be "discrete" (or "categorical", indeed "symbolic") when the number of values is finite. A "continuous" attribute corresponds to real numerical values (for instance, a measurement). The discretization process transforms an attribute from continuous type to discrete type.

Instances: An instance is an example (or "record") of the dataset; it is often a row of the data table. Instances of a dataset are usually seen as a sample of the whole population (the universe). An instance is described by its attribute values that can be continuous or discrete.

Cut-points: A cut-point (or "split-point") is a value that divides an attribute into intervals. A cut-point has to be included in the range of the continuous attribute to discretize. A discretization process can produce none or several cut-points.

Number of intervals: The number of intervals corresponds to the different values of a discrete attribute resulting from the discretization process. The number of intervals is equal to the number of cut-points plus one. The minimum number of intervals of an attribute is equal to one and the maximum number of intervals is equal to the number of instances.

Supervised / Unsupervised: A supervised learning algorithm searches a functional link between a class-attribute (or "dependent attribute", or "attribute to be predicted") and predictive attributes (the descriptors). The supervised learning process aims to produce a predictive model that is as accurate as possible. In an unsupervised learning process, all attributes play the same role; the unsupervised learning method tries to group instances in clusters where instances in the same cluster are similar, and instances in different clusters are dissimilar.

Univariate / Multivariate: An univariate (or "monothetic") method processes a particular attribute independently of the others. A multivariate (or "polythetic") method processes all attributes of the representation space, so it can fix some problems related to the interactions between the attributes.

Representation space: The representation space is formed with all attributes of learning problem. In the supervised learning, it consists of the representation of the labeled instances in a multidimensional space where all predictive attributes play the role of a dimension.

ACKNOWLEDGEMENT

Edited with the aid of Christopher Yukna.