A Suitability Study of Discretization Methods for Associative Classifiers

Kavita Das
SoS in Computer Science & IT,
Pt. Ravishankar Shukla University,
Raipur, C.G., India

O. P. Vyas
Prof. & Program Coordinator (S/W Engg.)
IIIT, Allahabad, U.P., India

ABSTRACT

Discretization is a popular approach for handling numeric attributes in machine learning. The attributes in the datasets are both nominal and continuous. Most of the Classifiers are capable to be applied on discretized data. Hence, pre-processing of continuous data for converting them into discretized data is a necessary step before being used for the Classification Rule Mining approaches. Recently developed Associative Classifiers like CBA, CMAR and CPAR are almost equal in accuracy and have outperformed traditional classifiers. The distribution of continuous data into discrete ranges may affect the accuracy of classification. This work provides a comparative study of few discretization methods with these new classifiers. The target is to find some suitable discretization methods that are more suitable with these associative classifiers.

General Terms

Discretization, Associative Classification

Keywords

ARM, CRM, CBA, CMAR, CPAR, CADD, USD, ChiMerge, MDLP, ID3, EWD, EFD

1. INTRODUCTION

Discretization has emerged into most important preprocessing task for Classification Rule Mining in Machine Learning. Discretization is the process of converting continuous data into intervals or groups of data values having some similarity or closeness of values. Each interval is mapped to a discrete (categorical, nominal, symbolic) symbol. Discretization is a peripheral but integrated part of pattern learning phase of data mining, especially Classification Rule Mining (CRM). The target of Classification may affect the choice of method of training decision tree, and may further determine the criteria and methodology of discretization. Discretization is generally applied before or during the training of decision tree or learning process.

Discretization has been classified in various ways [3, 8] –

a) Supervised and Unsupervised methods
b) Local and Global methods
c) Static and Dynamic methods
d) Error-based and entropy-based methods
e) Top-down and Bottom-up methods

Supervised discretization methods use predetermined groups or class labels in the data for deciding the interval boundaries. *Unsupervised* methods use similarity or closeness of values to determine a group or class. The classes are not predetermined. *Local* discretization methods carry out discretization during Decision tree building process. They produce partitions that are applied to localized regions of the instance space. *Global* discretization methods involve discretizing each numeric attribute before building classifier. They use the entire value domain of a numeric attribute for discretization and are less prone to variance in estimation for small fragmented data. *Static* methods determine the number of partitions for each attribute independent of the other features. *Dynamic* methods conduct the search through the space of possible k partitions for all features simultaneously, thereby capturing interdependencies in feature discretization. *Error-based methods* select a discretized data as optimum if it produced minimum errors than the other candidates of discrete data by application of a classification method. *Entropy-based* methods uses class information entropy of candidate intervals to select the threshold interval boundaries. *Top-down* methods start off with one big interval containing all values and recursively finds cut points for intervals until certain criteria are reached. *Bottom-up* methods initially consider a number of intervals determined by the set of boundary points and then recursively combine adjacent intervals until certain stopping criteria are reached.

In data mining, the technique of Association rule mining (ARM) aims at finding frequently occurring data items by using minimum support and minimum confidence constraints. It thereby discovers associations existing among data items without any predetermined target. The technique of Classification rule mining (CRM) aims to build a classifier for database records to predict them to some predefined classes. In Associative Classification, the integration of ARM and CRM is done. The integration has proved to be efficient in comparison to conventional classifiers. It focuses on a special subset of association rules whose right-hand-side is restricted to the label class attribute. This subset of rules are referred to as the *class association rules* (CARs). Then CRM is applied on the CARs to generate Classifier or Classification Rule set. Like most of the data mining approaches, Associative Classification are also applied to discretized data. So, discretization of continuous features in the dataset is an essential step for the classification.

The objective of this work is to evaluate the performance of some discretization methods using few efficient Associative Classification approaches. Several datasets have been discretized using each of the discretization methods and then accuracy is observed empirically with each of the Associative Classifiers. The results show the combinations of discretization methods and Associative Classification methods that might be suitable for future applications.

In this paper, issues of discretization are discussed in section 2. In section 3, the discretization methods are described. Description about Associative Classification techniques used is given in section 4. In section 5, our experiment is presented. Conclusion is given in section 6.

2. ISSUES OF DISCRETIZATION

The real data have various characteristics such as small or big size, large number of attributes, various data types and value ranges, continuous, unbounded and high speed data stream, changing data distribution, etc. In discretization process, several issues are desirable in order to achieve good classification results. Some of the issues are being discussed here.

2.1 Purity of Intervals

During discretization, when the values of an attribute are distributed into the intervals with majority of instances (ideally all the instances) belonging to a single class. This purity of intervals is hard to achieve. Some instances may fall into neighbourhood intervals having a different class in majority. When such a discretized dataset is used for generating decision tree or rule set by a classifier, the accuracy of the classifier is affected adversely depending on the number of misplaced items in the intervals. A discretization method that ensures higher probability of generating correctly binned intervals will always be supportive in delivering higher accuracy of classification. Hence, high intra-interval uniformity and high inter-interval difference are desirable for purity of intervals.

2.2 Hierarchy of Intervals

If a discretization method can generate intervals for various degrees of purity may generate pure intervals with general level classes and also generate smaller sub-intervals with specific level classes and may have lesser purity. The method can provide hierarchical structure of discrete values of the continuous attribute. Such a discretization output may provide an alternate way of adjusting classification rules to provide higher accuracy and realistic rules. It also provides an option to restrict number of intervals for discretization of continuous values.

2.3 Modifiability

As the dataset pertaining to an application grows or varies with time or space, the discretization interval boundaries may shift a little. Hence, it is useful that the discretization algorithms may modify or update an existing discretization criteria or structure easily. This will provide a long time durability of pure intervals in a real classification cases. Also the accuracy of classification will not degrade with change in trend of values in the dataset.

2.4 Information Preservation

Any multi-attribute dataset generally consists of associations or relation among its attributes. Discretization is effectively a way of systematic summarization of large dataset. During this pre-processing, some attributes participating in an association may get intervals with different classes. This will cause loss of information about attribute-relationship prevailing between the attributes. Such a pre-processing may lead to leakage of accuracy and incompleteness in the classification rule set. Hence, discretization with criteria for information preservation is desirable.

2.5 Computation Complexity

Due to algorithmic nature of discretization approaches that are fulfilling multiple criteria and are applied on multiple attributes in the dataset, the requirement of time and memory efficiency is obvious. An algorithm with high complexity of time and space degrades in performance gradually with the size of the dataset. Such algorithm will not be useful for online and other dynamic classification environment demanding high speed or having big dataflow.

3. METHODS OF DISCRETIZATION

Though there are several methods of discretization with new emerging concepts and are promising avenue of research, this work primarily aims at choosing a suitable discretization method that could be adopted currently with an associative classifier. The discretization approaches that have been considered are hereby described below. Each discretization method first sorts the values of a numeric attribute of a training set into ascending order.

3.1 Binning

It is the simplest method of discretization to distribute continuous attribute into a specified number of bins. The attribute values are kept into bins of equal width or equal frequency intervals. Each bin is associated with a distinct discrete value.

3.1.1 Equal Width Discretization (EWD)

This method [3] divides the range of values between minimum and maximum values of an attribute into k number of equal sized intervals, where k is user’s given choice. It is a global, unsupervised and static method.

3.1.2 Equal Frequency Discretization (EFD)

This method [3] divides all the values between minimum and maximum values of an attribute into k number of intervals, each containing same number of items. It is a global, unsupervised and static method.

The computation complexity of above methods is O(n). They do not ensure purity of intervals or information preservation and is sensitive to the value of k. They may be suitable for uniformly distributed data with expert knowledge of value of k.

3.2 Bottom-Up Merging

Bottom-up merging process starts with an initial set of intervals. Then the adjacent intervals are considered for testing their similarity. If they have similarity above a threshold level, they are merged into a single new interval. This process is repeated again and again with the new set of intervals until all the intervals become dissimilar significantly.

3.2.1 ChiMerge

It [6] uses a heuristic method of merging adjacent intervals after testing the frequency of classes represented in each interval with Chi-Square test. It starts by creating initial discretization with each example as an interval. Merging of adjacent intervals continues until χ^2 values of all pairs of intervals exceed a threshold. At the end, all adjacent intervals are considered significantly different by the χ^2 independence test.

The formula used for computing χ^2 is:

$$\chi^2 = \sum_{i=1}^{m}\sum_{j=1}^{k} \frac{(Aij-Eij)^2}{E_{ij}}$$

Where:

m = 2(no. of intervals being compared)
k = number of classes
A_{ij} = number of examples in ith interval, jth class
R_i = number of examples in ith interval
C_j = number of examples in jth class
N = total number of examples
E_{ij} = expected frequency of $A_{ij} = \frac{R_i * C_j}{N}$

It is a global, supervised and static method.

3.3 Entropy Based Discretization Methods

This method [3, 4] uses minimal entropy heuristic for discretizing continuous attributes. They either find binary cuts or multiple cuts recursively on continuous data based on value of minimum entropy value at a point of values. Information gain or Minimum Description Length Principle or predefined number of cuts is used as criteria to stop for finding more cut points. Entropy is one of the most commonly used discretization measures. This method may provide highly pure intervals and hierarchical output. It includes MDLP and ID3 methods.

3.3.1 ID3

It [9] is a decision tree induction that uses Shannon's entropy measure. It constructs an inductive decision tree by selecting a feature if its branching results in the overall minimum entropy at the next layer of the decision tree. ID3 employs a greedy search to find the potential cut-points within the existing range of continuous values. The cut-point with the lowest entropy is chosen to split the range into two parts. Splitting continuous within each part until a stopping criterion is satisfied. It is a binary splitting approach. It is a local, supervised and dynamic method.

3.3.2 MDLP

This method [4] provides a principled way of a stopping criterion for splitting an interval. It is an entropy minimization heuristic. It uses class information entropy for finding suitable cut-points in a continuous feature. It is repeatedly applied in existing partitions until a threshold limit is reached. The entropy measure used is Minimum Description Length Principle (MDLP). It is formulated as the cost or length of message, induced by a cut point, needed to communicate an instance for finding its class label. If length of message before cut-point > length of message after cut-point, the cut-point is accepted.

It is computationally complex as it requires N-1 evaluations for each attribute (N = number of values for the attribute) and N is generally large. It is a local, supervised and static method.

3.4 Class-Attribute Dependent Discretizer

This method, abbreviated as CADD [2], is based on the concept of Class-Attribute dependency. It seeks to maximize the dependency relationship between the class variable and continuous valued attribute. It uses a 2-D matrix of frequency of instances of each class in each interval, called quanta matrix. It is used to calculate the estimated joint probability of the event that an object belongs to a particular class while an attribute value falls in a particular range.

The Class-Attribute mutual information between the class variable c_s and the attribute interval boundaries $A_j \in [e_{r-1}, e_r]$ with its quanta set Q_j is given as:

$$I(C{:}A_j) = \sum\sum p_{sr} \log \frac{p_{sr}}{p_{s+} \,.\, p_{+r}}$$

p_{sr} = joint probability that the object belongs to c_s
p_{+r} = marginal probability that $A_j \in e_r$
p_{s+} = probability that class is c_s

It is a local, supervised and static method.

This method is highly combinatorical and its global version would be highly expensive. Hence, a heuristic based "local optimization" is used with three main steps: interval initialization, interval improvement and interval reduction. The initialization step is sorting of unique values of a real-valued attribute in training set and estimation of initial default number of intervals as user input or second order probability estimation. Improvement involves alternating the initial quanta matrix with small increments or decrements at interval boundaries. Reduction combines the statistically insignificant intervals.

3.5 Unparameterized Supervised Discretization

This discretization, abbreviated as USD [5], aims to obtain maximum goodness of the intervals, that is, preserve the information within original continuous attribute. Goodness of interval is defined as the relationship between the goals and errors of this interval. It is expressed as

$$\text{Goodness}(I_i) = \frac{\text{goals}(I_i)}{1 + \text{errors}(I_i)}$$

The goodness can vary depending on the penalty per error. This method needs no user input parameters. The method calculates the initial intervals with a simple discretization method and then maximizes the purity of intervals. This also makes the number of intervals very high. Then it combines the intervals if goodness of union of two intervals is greater than the average of goodness of the intervals. It is a local, supervised and static method.

4. TECHNIQUES OF ASSOCIATIVE CLASSIFICATION

Associative Classification [12] consists of three steps:
1. Generation of frequent item sets or association rules.
2. Selection of all the class association rules (CARs),

3. Building a classifier based on the generated CARs.

Very efficient methods of Associative Classification, the CBA [7], CMAR [13] and CPAR [14] are used in this work.

4.1 Classification Based on Associations (CBA)

This [7] is the first algorithm that integrates ARM and CRM. It uses the apriori approach to discover the association rules. Best association rules having any of targeted rules are selected based on confidence, support and size of antecedent. These rules are pruned using "pessimistic error rate". Finally the rule list is generated using a variation of "cover" principle. For prediction of a new case, it uses a set of related rules by evaluating the correlation among them.

4.2 Classification Based on Multiple Association Rules (CMAR)

CMAR algorithm [13] uses the FP-growth approach to find association rules. The classification rules are stored in a prefix tree data structure, known as a CR-tree. For classifying a new object, the subset of classification rules matching the new object observed at their class labels. In the case where all rules have a common class, CMAR simply assigns that class to the test object. In cases the classes of the accumulated rules are not identical, the rules are divided into separate groups based on their class values and the effects of every group are compared to identify the strongest one. The strength of the groups is measured by the weighted $\chi 2$. The class of the strongest group is assigned to the object.

4.3 Classification Based on Predictive Association Rules (CPAR)

It [14] is a greedy associative classification approach. The best rule condition is measured by FOILgain of the rules generated among the available ones in the dataset. FOILgain is used to measure the information gained from adding a condition to the current rule. Once the condition is identified, the weights of the positive examples associated with it are reduced by a multiplying factor, and the process repeats until all positive examples in the training data set are covered.

In the rule-generation process, CPAR is capable of deriving not only the best condition but also all similar ones. Hence, it includes the rules with similar gains.

5. EXPERIMENT

In order to study empirically the performance of various discretization approaches on Classification Rule Mining, some easily available discretization methods are selected. KEEL [1], a data mining software tool was used for discretizing the datasets. The datasets[1] are also available with this tool. The datasets we used are as shown in Table 1.

Table 1. The datasets

S.no.	Datasets	Size	Classes	Attributes
1	Iris	150	3	4
2	Wine	143	3	13
3	Glass	191	6	10
4	Cleve	268	5	13
5	haberman	275	2	35
6	Breast	614	2	10
7	Pima	692	2	8

The classification approaches chosen are associative classifiers: CBA, CMAR and CPAR. The discretization methods taken into consideration are: Equal-width discretizer (EWD), Equal-frequency discretizer (EFD), ID3, Unparamettrized Supervised discretization (USD), Class Attribute Dependent Discretization (CADD), ChiSquare discretization (Chi) and MDLP based Discretization. The number of discrete intervals generated by various discretizers is shown in Table 2.

Among them EWD and EFD are unreliable methods though many research works have shown that they have performed quite good. ID3 and USD generate a lot of discrete intervals. Hence, EWD, EFD, ID3 and USD were not included for classification. The three methods CADD, Chi and MDLP were actually used for the comparative study.

Table 2. No. of Discrete intervals by different discretizers

S. no	Datasets	ID3	USD	CA DD	CHI	MD LP
1	Iris	61	26	23	35	15
2	Wine	614	401	88	132	38
3	Glass	658	462	34	91	29
4	cleve	341	184	72	75	25
5	haberman	81	38	7	29	6
6	Breast	81	40	20	65	31
7	Pima	797	418	14	81	18

The performances of the three associative classifiers on different datasets discretized with the chosen three discretization methods are given in Table 3, Table 4 and Table 5 and graphically shown in Figure 1, Figure 2 and Figure 3.

Table 3. Performance with CADD

Data sets	CBA		CMAR		CPAR	
	Accu racy	No. of Rules	Accu racy	No. of Rules	Accu racy	No. of Rules
Iris	29.33	2	88.66	16	29.33	5
Wine	26.76	8	0	0	47.89	7
Glass	0	0	0	0	30.84	7
cleve	52.99	5	0	0	47.01	54
haber	70.07	2	0	0	0	0
Breas	75.57	2	0	0	0	0
Pima	69.08	2	0	0	0	0
Aver	46.257		12.666		22.153	

Figure 1. Accuracy with CADD

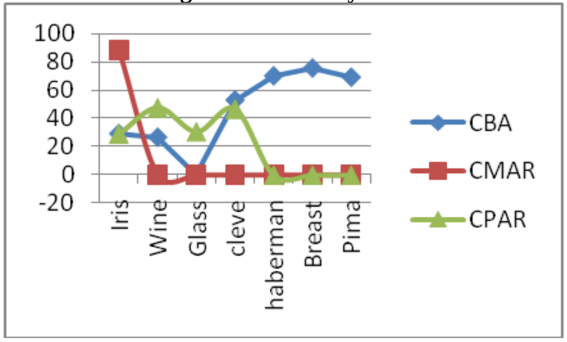


Table 4. Performance with Chi

Data sets	CBA		CMAR		CPAR	
	Accu racy	No. of Rules	Accu racy	No. of Rules	Accu racy	No. of Rules
Iris	33.33	2	48.65	1	32	1
Wine	0	3	67.85	7	47.89	7
Glass	0	0	0	0	36.84	0
cleve	54.48	15	53.14	52	57.46	52
haber	70.07	2	48.64	1	56.93	1
Breas	94.46	16	68.36	66	95.77	66
Pima	68.79	9	23.54	1	69.36	1
Aver	45.87		44.31		56.60	

Figure 2. Accuracy with Chi

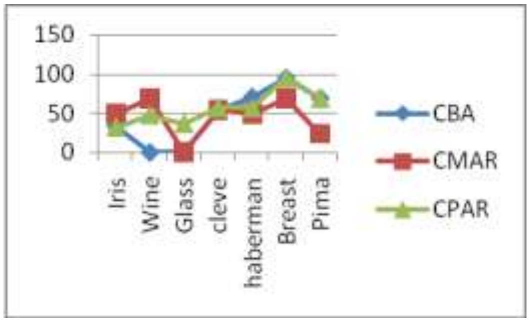
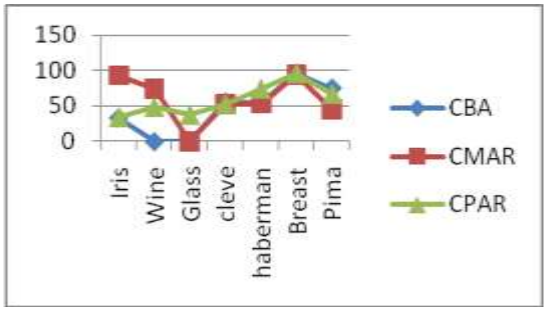


Table 5. Performance with MDLP

Data sets	CBA		CMAR		CPAR	
	Accu racy	No. of Rules	Accu racy	No. of Rules	Accu racy	No. of Rules
Iris	33.33	2	93.33	16	33.33	4
Wine	0	2	74.29	79	47.89	3
Glass	0	0	0	0	36.84	4
cleve	55.97	14	53.14	168	52.24	11
haber	54.01	2	53.46	4	73.72	2
Breas	96.42	17	94.46	112	95.77	11
Pima	75.43	10	44.94	28	66.47	11
Aver age	45.02 3		59.08 9		58.03 7	

Figure 3. Accuracy with MDLP

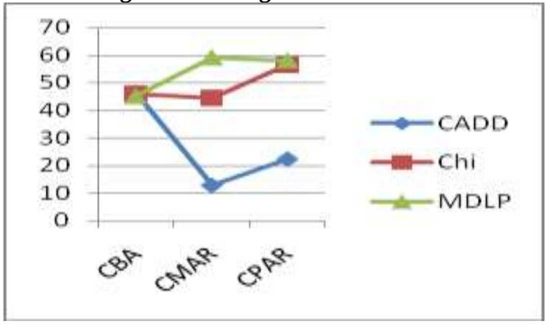


Now, average accuracies of CADD, Chi, MDLP on CBA, CMAR & CPAR are given in Table 6 and shown graphically in Figure 4.

Table 6. Average accuracies

	CBA	CMAR	CPAR
CADD	46.26	12.66	22.15
Chi	45.87	44.31	56.61
MDLP	45.02	59.1	58.02

Figure 4. Average accuracies



From the above comparisons, we can observe the performances of the Associative classifiers w.r.t. each discretizer and also the comparative performances of the discretizers with all the associative classifiers.

6. CONCLUSION

This is a study of a set of discretization approaches in order to evaluate and find methods having suitability with the associative classifiers CBA, CMAR, CPAR in producing better accuracies in Classification Rule Mining. It is found that recently discretization methods are tending to achieve purity of intervals and information preservation.

[1] In each dataset the *data-10-3tra.dat* dataset is used in the experiment.

Also it can be observed that CPAR is consistently giving better performance in terms of accuracy and number of generated rules. The discretization method CADD showing poor generation of discrete data as many times the classifiers are failing to generate any rules whereas MDLP is showing better accuracy averagely with all the associative classifiers. ChiMerge is equivalent to MDLP with two classifiers CBA and CPAR but have very low accuracy with CMAR.

7. REFERENCES

[1] Alcala-Fdez, J., Sanchez, L., et al. 2009. KEEL: A software tool to assess evolutionary algorithms for data mining problems. Soft Computing - A Fusion of Foundations, Methodologies and Applications: Springer Berlin / Heidelberg, pp.307-318. Available from anonymous ftp.

[2] Ching, J. Y., Wong Andrew, K. C., and Chan, Keith K. C. 1995. Inductive Learning from Continuous and Mixed-Mode Data. IEEE Transactions on Pattern Analysis and Machine Intelligence

[3] Dougherty, J., Kohavi, R. and Sahami, N. 1995 Supervised and Unsupervised Discretization of continuous Features. Machine Learning, 14th IJCAI, 1995, 108-121.

[4] Fayyad, U. M. and Irani, K. B. 1993. Multi-Interval Discretization of Continuous Valued Attributes for Classification Learning. 13th IJCAI, vol. 2., Chambery, France, 28.8.-2.9.93, Morgan Kaufmann, 1022–1027

[5] R. Giraldez, J. S., Aguilar-ruiz, et al. 2002. Discretization Oriented to Decision Rules Generation. Frontiers in Artificial Intelligence and Applications.

[6] Kerber, R. 1992. ChiMerge: Discretization of Numeric Attributes. In proceedings of tenth National Conference on Artificial Intelligence. 123-128

[7] Liu, B., Hsu, W., and Ma, Y. 1998. Integrating Classification and Association Rule Mining. In proceedings of the KDD, 1997, New York, 80-86, 1998

[8] Perner, P., Trautzsch, S. 1998. Multi-Interval Discretization Methods for Decision Tree Learning. LNCS 1451, Springer Verlag, 475-482

[9] Quinlan, J. R. 1986. Induction of Decision Trees. Machine Learning, 1: 81-106

[10] Quinlan, J. R., Cameron-Jones, R. M. 1993. FOIL: A midterm report. In proceedings of European Conference on Machine Learning, Vienna, Austria

[11] Thabtah, F., Cowling, P. and Peng, Y. 2005. A Study of Predictive Accuracy for Four Associative Classifiers. Journal Of Digital Information Management

[12] Thabtah, F., Cowling, P. and Peng, Y. 2006. Multiple Labels Associative Classification. Knowledge and Information Systems. Vol. 9, No. 1. 109-129

[13] Wenmin, L., Jiawei, H. and Pei, J. 2001. CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules. In proceedings of ICDM. 369-376

[14] Xiaoxin, Y. and Han, J. 2003. CPAR: Classification based on Predictive Association Rules. In proceedings of SIAM International Conference on Data Mining, San Fransisco, CA. 331-335.