A Comparison of the Discretization Approach for CST and Discretization Approach for VDM

Omar A. A. Shiba
Faculty of Science, Sebha University- Libya
E-mail: abumoad99@gmail.com

Abstract- The preprocessing data is the most important task in data mining steps. Discretization process is known to be one of the most important data preprocessing tasks in data mining. Presently, many discretization methods are available. Discretization methods are used to reduce the number of values for a given continuous attributes by dividing the range of the attribute into intervals. Discretization makes learning more accurate and faster. This paper will investigate the effects of discretization of continuous attributes in a dataset to the performance of Case Slicing Technique as a classification method and compared it with another classification approach which will be a generic version of the VDM algorithm" the discretized value difference metric (DVDM).

Keywords- data mining, discretization, classification, slicing, preprocessing.

I. INTRODUCTION

Data mining classification algorithms developed to produce a solution to improving the efficiency of systems includes a lot of data. Classification is a widely used technique in various fields, including data mining whose goal is classify a large dataset into predefined classes. In a dataset the attributes may be continuous, categorical and binary. However, most attributes in real world are in continuous form [1].

This paper will investigate the effects of discretization of continuous attributes in a dataset to the performance of two classification methods, a generic version of the VDM algorithm" the discretized value difference metric (DVDM) and the improved discretization approach of Case Slicing Technique (CST) [2]. The aim is to show that discretization will improve the accuracy of CST classifier more than the VDM classifier. The datasets to be used are the "Iris Plants Dataset (IRIS), Credit Card Applications (CRX), Hepatitis Domain (HEPA) and Cleveland Heart Disease (CLEV) from the UCI Machine Learning Repository [3]. All datasets have been discretized and feed to classification algorithms.

The rest of this paper is organized as follows: the next section presents some preprocessing data tasks. Section 3 presents experimental results that compare the performance of the algorithms. Finally the conclusion is presented in section 4.

II. DATA PREPROCESSING

In this section some preprocessing data tasks such as discretization, weighting attributes, similarity computation etc**.** that needed by attribute selection approach will be discussed.

A. Identify attributes Types

Attributes can be continuous or discrete. A continuous (or continuously-valued) attribute uses real values. A discrete attribute is one that can have only a fixed set of values, and can be either symbolic or linear.

A linear discrete attribute can have only discrete set of linear values. It can be argued that any value stored in a computer is discrete at some levels. The reason continuous attributes are treated differently is that they can have so many different values that each value may appear only rarely (perhaps only once). This causes problems for algorithms such as Value Difference Metric (VDM) and Case Slicing Technique (CST) discussed in this Paper. These algorithms depend on testing two values for equality but these values will rarely be equal, though they may be quite close to each other.

A symbolic (or nominal) attribute is a discrete attribute whose values are not in any linear order. So, the continuous attributes have to be transformed into discrete values. This process is called discretization, which will be discussed in subsection C.

B. Weighting attributes

Most of the classification algorithms developed until now somehow try to measure the relative importance of an attribute in classification compared to the others, and use this knowledge while classifying objects.

Some of them try to determine the relevant and irrelevant attributes in a set of attributes in order to take relevant attributes into account more than the irrelevant ones, and some others attempt to assign weights according to the degree of relevance on the classification of instances.

In this stage of the Case Slicing Technique, the basic version of conditional probabilities has been used to measure the importance of each attribute in the classification. The weight of each attribute has been calculated to classify the new case by using this statistical approach shown in Eq.1 and Eq. 2. High weight values are assigned to attributes that are highly correlated with the given class. The conditional probabilities have been used because it gives a real weight for each attribute based on its repetition in the dataset and based on the class attribute for each case.

$$w(i_a) = P(C \mid i_a) \qquad (1)$$

$$P\left(C \mid i_a\right) = \frac{\left| instances \quad containing \quad i_a \wedge class = C \right|}{\left| instances \quad containing \quad i_a \right|} \qquad (2)$$

where the weight for attribute *a* of a class *c* is the conditional probability that a case is a member of *c* given the value of *a*. Fig.1 gives the pseudo code for doing this.

```
Let c be the output class of case i
$N_{a,v,c} = N_{a,v,c} + 1$        {# of value v of attribute a with output class c}
$N_{a,v} = N_{a,v} + 1$                {# of value v of attribute a}
    For each value v (of attribute a)  do
        For each class c  do
            If $N_{a,v} = 0$
                $P_{a,v,c} = 0$
            Else
                $p_{a,v,c} = N_{a,v,c}/N_{a,v}$
        Endfor
    Endfor
```

Fig. 1: Pseudo code for *assign weights to attributes {using conditional probability} P*(*C* | i_a)

C. *Discretization*

In data mining, discretization process is known to be one of the most important data preprocessing tasks. Most of the existing machine learning algorithms is capable of extracting knowledge from databases that store discrete attributes. If the attribute are continuous, the algorithms can be integrated with a discretization algorithms which transform them into discrete attribute.
Discretization methods are used to reduce the number of values for a given continuous attributes by dividing the range of the attribute into intervals [4],[5]. Discretization makes learning more accurate and faster.

Discretization as used in this paper and in the machine learning literature in general is a process of transforming a continuous attribute value into a finite number of intervals and associating each interval with a discrete, numerical value. The usual approach for learning tasks that use mixed-mode (continuous and discrete) data is to perform discretization prior to the learning process [6],[7],[8].

The discretization process first finds the number of discrete intervals, and then the width, or the boundaries for the intervals, given the range of values of a continuous attribute. More often than not, the user must specify the number of intervals, or provide some heuristic rules to be used [9].

A variety of discretization methods have been developed in recent years. Some models that have used the Value Difference Metrics (VDM) or variants of it [10],[11],[12] have discretized continuous attributes into a somewhat arbitrary number of discrete ranges, and then treated these values as nominal (discrete unordered) values. The discretization method proposed by Randall and Tony [13] and has been extended by Payne and Edwards [14] as shown in Eq.3.

$$v = disc_a(x) = \begin{cases} x, & if\ a\ is\ discrete \\ s, & if\ x \geq max_a,\ else \\ 1, & if\ x \leq min_a,\ else \\ \lfloor (x - min_a)/w_a \rfloor + 1 & \end{cases} \qquad (3)$$

where $$w_a = \frac{|max_a - min_a|}{s} \qquad (4)$$

where *max*$_a$ and *min*$_a$ are the maximum and minimum values, respectively which occur in the training set for attribute *a* and *s* is an integer supplied by the user. Unfortunately, there is currently little guidance as to what value should be assigned to *s*. Current research is examining more sophisticated techniques for determining good values of *s*, such as cross-validation, or other statistical methods [13].

When using the slicing approach, continuous values are discretized into *s* equal-width intervals (though the continuous values are also retained for later use), where *s* is an integer supplied by the user. The width w_a of a discretized interval for attribute *a* is given by Eq. 4.

In this Paper, we propose a new alternative, which is to use discretization in order to collect statistics and determine good values of $P(C \mid i_a)$ for continuous values occurring in the training set cases, but then retain the continuous values for later use. A generic version of the VDM algorithm, called the *discretized value difference metric* (DVDM) will be used for comparisons with extensions proposed in this paper. Thus, differences between DVDM and the new function can be applied to the original VDM algorithm or other extensions.

The discretized value *v* of a continuous value *x* for attribute *a* is an integer from 1 to *s*, and is given by Eq. 5. Fig.2 shows the pseudo code for the discretization of continuous values in the proposed approach.

$$v = disc_a(x) = \begin{cases} \dfrac{\lceil (x - min_a) \rceil}{w_a} & if\ attribute\ a\ is\ continuous \\ x & if\ attribute\ a\ is\ discrete \end{cases} \qquad (5)$$

```
Let x be the input value for attribute a of case i
    $v = disc_a(x)$          {which is just x if a is discrete}
     $w_a = abs(max_a - min_a)/s$

{The width of a discretized interval for attribute a}

{where max and min are the maximum and minimum value, respectively, which occur in the training
set for attribute a},
{The discretized value v of a continuous value x for attribute a is an integer from 1 to s and s is
determine by the user.}
If a is continuous then
        $v = disc_a(x) = \lceil x - \min_a \rceil / w_a$
  Else
        $v = disc_a(x) = x$
Endif
```

Fig. 2: Pseudo code for discretize *continuous values*

III. EXPERIMENTAL RESULTS

In this section the results of several practical experiments are presented to compare the proposed approach with DVDM approach. In the experiments, we applied the discretization algorithm as the preprocessing step of CST classifier on some selected real world problems.

A. Empirical Results

In this section the comparison of the discretization approach of the DVDM and the improved discretization approach of Case Slicing Technique is presented on some selected datasets. In this comparison four datasets has been selected that are Iris Plants Dataset (IRIS), Credit Card Applications (CRX), Hepatitis Domain (HEPA) and Cleveland Heart Disease (CLEV). All datasets have been discretized and feed to classification algorithms, the results of the comparison are shown in Table 1.

Table 1: Classification accuracy of CST against DVDM based on discretization approach

Methods / *Datasets*	DVDM	CST
IRIS	92.00 %	99.30 %
CRX	83.04 %	97.80 %
HEPA	80.58 %	99.00 %
CLEV	79.86 %	96.00 %

In Table 1 the comparison between the proposed discretization approach for Case Slicing Technique and discretization approach for VDM has been done on four types of datasets.

The discretization result of CST and VDM have been forwarded into CST classifier for classification to make sure that the result obtained from each technique is based on discretization approach used during classification. List of the result of classification accuracy shown in Table 1. CST gave better classification accuracy compared with DVDM classification accuracy. Fig.3 shows the difference in classification accuracy of CST against DVDM based on discretization approaches.

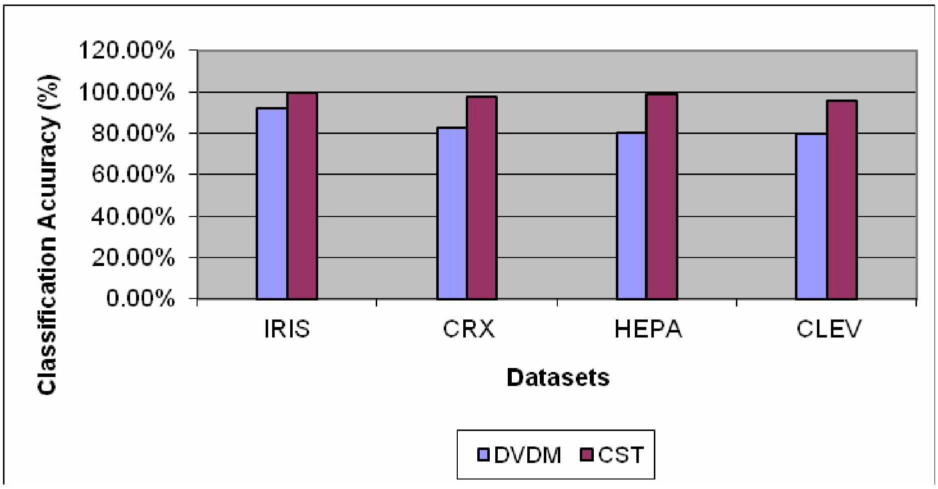

Fig. 3 : Difference in classification accuracy of CST against DVDM

VI. CONCLUSION

This paper has presented a comparison between the proposed discretization approach for Case Slicing Technique (CST) and discretization approach for VDM. The discretization result of CST and VDM have been forwarded into CST classifier for classification to make sure that the result obtained from each technique is based on discretization approach used during classification. The experiments show that using the proposed discretization approach for (CST) indeed improves the accuracy of classification. It gave very high percentage of classification accuracy.

REFERENCES

[1]. Mehmet Hacibeyoglu, Ahmet Arslan, Sirzat Kahramanli. "Improving Classification Accuracy with Discretization on Datasets Including Continuous Valued Features" , *World Academy of Science, Engineering and Technology, 2011, 54.*

[2]. Shiba, O., Sulaiman, M.N., Mamat, A. & Ahmad, F. "An Efficient and Effective Case Classification Method Based On Slicing", *International Journal of The Computer, the Internet and Management Vol. 14.No.2 (May - August, 2006) pp15-23*

[3]. Murphy, P.M. "UCI Repositories of Machine Learning and Domain Theories",1997,URL: /http://www.ics.uci.edu/~mlearn/MLRepository.html. (Accessed on 10 Jan. 2012).

[4]. Kurgan. L and Cios, K.J.: "Discretization Algorithm that Uses Class-Attribute Interdependence Maximization", *Proceedings of the International Conference on Artificial Intelligence (IC-AI 2001)*,pp.980-987, Las Vegas, Nevada**.**

[5]. Ratanamahatana, C. A. "CloNI: Clustering of N-Interval Discretization", *Proceedings of the 4th International Conference on Data Mining Including Building Application for CRM & Competitive Intelligence, Rio de Janeiro, Brazil 2003.*

[6]. Catlett, J. "On Changing Continuous Attributes into Ordered Discrete Attributes". *Proceedings of the European Working Session on Learning,1991 (pp. 164-178), Berlin: Springer-Verlag.*

[7]. Dougherty, J., Kohavi, R. & Sahami, M. "Supervised and Unsupervised Discretization of Continuous Features". *Proc. of the 12th International Conference on Machine Learning, 1995, (pp. 194-202), San Francisco, CA: Morgan Kaufmann.*

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

[9]. Ching, J., Wong, A. & Chan, K. "Class-Dependent Discretization for Inductive Learning from Continuous and Mixed Mode Data". *IEEE Transactions on Pattern Analysis and Machine Intelligence, 17 (7), 641-651. 1995.*

[10]. Cost, S. & Salzberg, S. "A Weighted Nearest Neighbor Algorithm for Learning With Symbolic Features". *Machine Learning, 10, 57-78. 1993.*

[11]. Rachlin, J., Simon, K., Salzberg, S. & David, W. "Towards a Better Understanding of Memory-Based and Bayesian Classifiers". *In Proceedings of the 11th International Machine Learning Conference, 1994, (pp. 242-250), New Brunswick, NJ: Morgan Kaufmann.*

[12]. Mohri, T. & Tanaka, H. "An Optimal Weighting Criterion of Case Indexing for Both Numeric and Symbolic Attributes". *Workshop, (Technical Report ws-94-01), 01, 1994, (pp.123-127), Menlo Park, CA: AIII press.*

[13]. Randall, D.W. & Tony, R.M. "Value Difference Metrics for Continuously Valued Attributes". *In Proceedings of the International Conference on Artificial Intelligence, Expert Systems and Neural Networks, 1996 (pp. 11-14).*

[14]. Payne, T.R. & Edwards, P. "Implicit Feature Selection With the Value Difference Metric". *ECAI. 13th European Conference on Artificial Intelligence Edited by Henri Prade, 1998, (pp. 450-454), Brighton, UK: Published by John Wiley & Sons, Ltd.*