A PRELIMINARY STUDY ON SELECTING THE OPTIMAL CUT POINTS IN DISCRETIZATION BY EVOLUTIONARY ALGORITHMS

Salvador García[1], Victoria López[2], Julián Luengo[3], Cristóbal J. Carmona[1] and Francisco Herrera[2]
[1]*Department of Computer Science, University of Jaén, Jaén, Spain*
[2]*Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain*
[3]*Department of Civil Engineering, University of Burgos, Burgos, Spain*
sglopez@ujaen.es, vlopez@decsai.ugr.es, jluengo@ubu.es, ccarmona@ujaen.es, herrera@decsai.ugr.es

Keywords: Discretization, Evolutionary algorithms, Decision trees, Bayesian learning, Classification.

Abstract: The Discretization, as a data preprocessing technique, has played an important role in many areas such as artificial intelligence, data mining and machine learning. In this paper, we propose the use of evolutionary algorithms to select a subset of cut points that defines the best possible discretization scheme of a data set. First, we identify the boundary points for each input attribute and then we establish the individual representation as the joining of all of them, forming bit-strings based chromosomes. In addition, we consider an inconsistency based fitness function for measuring the quality of the chromosomes during the evolutionary cycle. The CHC model is adopted as evolutionary approach, showing that it can bring higher accuracy to the discretization process. The proposal has been compared with other state-of-the-art and recent discretizers on 20 real data sets and the experiments show that our proposed algorithm generates competitive discretization schemes in terms of accuracy, for both C4.5 and Naive Bayes classifiers, but using a lower number of cut points.

1 INTRODUCTION

In the successful application of most of machine learning tools, the quality of the databases is very influential. Therefore, data preparation is a crucial research topic for this (Pyle, 1999). Discretization, as one of the basic data reduction techniques, has received increasing research attention in recent years (Yang et al., 2010) and has become one of the most broadly used preprocessing technique that is applied in machine learning. The discretization process converts continuous attributes into discrete ones by yielding intervals in which the attribute value can reside instead of singleton values, and by associating a discrete, numerical value with each interval (Liu et al., 2002).

Existing discretization techniques can be classified into two main categories, top-down (splitting) and bottom-up (merging). Top-down methods (Kurgan and Cios, 2004) start from the initial interval and recursively split it into smaller intervals, while bottom-up mechanisms begin with the set of single value intervals and iteratively merge adjacent intervals (Kerber, 1992). A good taxonomy can be found in (Liu et al., 2002), where discretizers are also categorized in Static/Dynamic, Univariate/Multivariate, Supervised/Unsupervised, Global/Local and Direct/Incremental. Among others, classical and well known discretizers are ChiMerge (Kerber, 1992), Zeta (Ho and Scott, 1997) and Chi2 (Liu and Setiono, 1997). Some recent proposed techniques are CAIM (Kurgan and Cios, 2004), MODL (Boullé, 2006) and PKID (Yang and Webb, 2009).

Evolutionary Algorithms (EAs) have been used for data preparation with promising results (Freitas, 2002). In discretization, few approaches can be found in the literature. The most important development in this area was done in (Flores et al., 2007), where an estimation of distribution algorithm is used for optimizing a Naive Bayes wrapper based discretizer.

In this contribution, we attempt to use EAs for optimal cut points selection in discretization. Our objective is to maximize the accuracy of the subsequent classification process and also to minimize the number of cut points required. For this, we perform an evolutionary selection of boundary points (Elomaa and Rousu, 1999) by using binary chromosome representation and an inconsistency based fitness function.

We compare our approach with other discretizers considering two classifiers, C4.5 and Naive Bayes, which are considered two of the most influential data mining algorithms (Wu and Kumar, 2009). The empirical study consists of 20 real data sets, 15 discretizers for comparison and analysis based on non-parametric statistical testing (Sheskin, 2007).

The rest of the contribution is organized as follows: Section 2 gives a brief summary of basic concepts regarding discretization. In Section 3, the evolutionary selection of cut points is explained. In Section 4, we provide the experimentation framework, the results obtained and an analysis over them. Finally, Section 5 concludes the paper.

2 DISCRETIZATION

Let TR be a training set with N instances which consists of pairs $(x_i, y_i), i = 1, \ldots, N$, where x_i defines an input vector of attributes and y_i defines the corresponding class label. Each of the N instances has M input attributes, M_n are numerical and M_c are categorical or nominal.

A refers to any of the M_n continuous attributes in the data set. A discretization algorithm partitions the continuous attribute A into k_A discrete and disjoint intervals:

$$D_A = \{[d_0, d_1], (d_1, d_2], \ldots, (d_{k_A-1}, d_{k_A}]\} \quad (1)$$

where d_0 is the minimal value, d_{k_A} is the maximal value and $d_i < d_{i+i}$, for $i = 0, 1, \ldots, k_A - 1$. Such a discrete result D_A is called a discretization scheme on attribute A and $P_A = \{d_1, d_2, \ldots, d_{k_A-1}\}$ is the set of cut points of attribute A. Hence, the joint discretization for all attributes defines the complete set of cut points P:

$$P = \bigcup_{A=0}^{M_n} P_A \quad (2)$$

If the discretizer is *univariate*, it chooses to search an optimal P_A for each attribute independently, whereas a *multivariate* discretizer attempts to find the best complete P. Few discretizers are proposed as multivariate due to the high complexity of the resulted search space. The search space is defined by the set of all the *candidate cut points* for each attribute, which is basically all the different numerical values registered in TR, considering each attribute separately.

In order to alleviate the complexity reducing the initial search space, the *boundary point* concept is defined. The set of *boundary point* is a subset of candidate points. Let a sequence S of examples be sorted by the value of a numerical attribute A. The set of *boundary points* is defined as follows:

- The maximum value in S is a boundary point.
- A value $T \in Dom(A)$ is a boundary point if and only if there exists a pair of examples $u, v \in S$, having different classes, such that $val_A(u) = T < val_A(v)$; and there does not exist another example $w \in S$ such that $val_A(u) < val_A(w) < val_A(v)$.

Thus, the set of boundary points for attribute A is denoted as BP_A, and BP denotes for the complete set of them. It is proved that optimal splits always fall on boundary points for most of the evaluation measures used (Elomaa and Rousu, 1999). Hence, substantial reductions in time consumption can be obtained, since only the boundary points need to be considered as candidate cut points.

3 EVOLUTIONARY SELECTION OF CUT POINTS

The selection of cut points in discretization can be considered as a search problem in which EAs can be applied. Our approach will be denoted by Evolutionary Cut Points Selection for Discretization (ECPSD). We take into account two important issues: the specification of the representation of the solutions and the definition of the fitness function.

- *Representation*: The search space associated is constituted by all the possible subsets of BP. This is accomplished by using a binary representation. A chromosome consists of $|BP|$ genes (one for each boundary cut point in BP) with two possible states: 0 and 1. If the gene is 1, its associated cut point is included in P, which is represented by the chromosome. If it is 0, this does not occur.
- *Fitness Function*: Let P be a subset of cut points selected from BP and be coded by a chromosome. We define a fitness function as the aggregation of two sub-objectives, namely the inconsistency of the data and the minimization of the number of cut points.

$$Fitness(P) = \alpha \cdot Inconsistency + (1 - \alpha) \cdot \frac{|P|}{|BP|} \quad (3)$$

where $|P|$ is the number of cut points currently selected in the chromosome, $|BP|$ is the total number of boundary points and α is the weight factor which is specified as input parameter. The *inconsistency* is a supervision-based measure used

to compute the number of unavoidable errors produced in the data set. An unavoidable error is one associated to two examples with the same values for input attributes and different class labels. In general, data sets with continuous attributes are consistent, but when a discretization scheme is applied over the data, an inconsistent data set may be obtained. The desired inconsistency level that a discretizer should obtain is 0.0. In our case, the inconsistency measure is computed as the summation of inconsistency instances present in the training set divided by the total number of instances.

The objective of the EA is to minimize the fitness function defined; therefore, to obtain consistent discretization schemes with the minimum possible number of cut points, thus enhancing its simplicity.

As the evolutionary computation method, we have used the CHC model (Eshelman, 1990). CHC is a classical evolutionary model that introduces different features to obtain a trade-off between exploration and exploitation; such as incest prevention, reinitialization of the search process when it becomes blocked and the competition among parents and offspring into the replacement process. During each generation the CHC develops the following steps:

- It uses a parent population of the same size as the original to generate an intermediate population, which are randomly paired and used to generate the potential offspring.
- Then, a survival competition is held where the best chromosomes from the parent and offspring populations are selected to form the next generation, keeping the fixed size of the population.

CHC also implements an heterogeneous recombination using HUX, a special recombination operator. HUX exchanges half of the bits that differ between parents, where the bit position to be exchanged is randomly determined. CHC also employs a method of incest prevention. Before applying HUX to the two parents, the Hamming distance between them is measured. Only those parents who differ from each other by some number of bits (mating threshold) are mated. The initial threshold is set at $L/4$, where L is the length of the chromosomes. If no offspring are inserted into the new population then the threshold is reduced by one.

No mutation is applied during the recombination phase. Instead, when the population converges or the search stops making progress (i.e., the difference threshold has dropped to zero and no new offspring are being generated which are better than any member of the parent population) the population is reinitialized to introduce new diversity to the search. The chromosome representing the best solution found over the course of the search is used as a template to reseed the population. Reseeding of the population is accomplished by randomly changing 35% of the bits in the template chromosome to form each of the other new chromosomes in the population. The search is then resumed.

Table 1: Summary description for classification data sets.

Data Set	#Ex.	#Atts.	#Num.	#Nom.	#Cl.
appendicitis	106	7	7	0	2
autos	205	25	15	10	6
bands	539	19	19	0	2
bupa	345	6	6	0	2
cleveland	303	13	13	0	5
contraceptive	1,473	9	9	0	3
crx	690	15	6	9	2
dermatology	366	34	34	0	6
ecoli	336	7	7	0	8
flare-solar	1066	9	9	0	2
glass	214	9	9	0	7
haberman	306	3	3	0	2
iris	150	4	4	0	3
mammographic	961	5	5	0	2
newthyroid	215	5	5	0	3
saheart	462	9	8	1	2
specfheart	267	44	44	0	2
tae	151	5	5	0	3
wine	178	13	13	0	3
wisconsin	699	9	9	0	2

Table 2: Parameters of the discretizers and classifiers.

Method	Parameters
C4.5	pruned tree, confidence = 0.25, 2 examples per leaf
Chi2	inconsistency threshold = 0.02
ChiMerge	confidence threshold = 0.05
FUSINTER	$\alpha = 0.975$, $\lambda = 1$
MODL	optimized process type
ECPSD	population = 50, eval. = 10,000 $\alpha = 0.5$

4 EXPERIMENTAL FRAMEWORK AND RESULTS

This section describes the methodology followed in the experimental study which compares the proposed technique with other discretization algorithms. We will explain the configuration of the experiment: used data sets and parameters of the discretizers. The discretizers involved in the comparison are: Ameva (González-Abril et al., 2009), CACC (Tsai et al.,

Table 3: Average accuracy obtained for C4.5.

	Ameva	*CACC*	*CAIM*	*Chi2*	*ChiMerge*	*DIBD*	*E-Width*	*Ext-Chi2*	*FUSINTER*	*Hellinger*	*Khiops*	*Mod-Chi2*	*MODL*	*PKID*	*Zeta*	*ECPSD*
appendicitis	0.8336	0.8336	0.8336	0.8127	0.8336	0.8418	0.7836	0.8018	0.8236	0.8509	0.8427	0.7855	0.8309	0.8018	0.8236	**0.8518**
autos	0.7549	0.7500	0.7263	0.7599	0.7474	0.7784	0.7308	0.6765	0.7937	0.7998	**0.8096**	0.7897	0.7033	0.7670	0.7349	0.7456
bands	**0.6700**	0.6605	0.6458	0.6624	0.6346	0.6605	0.6493	0.5491	0.5975	0.6363	0.6197	0.6642	0.6569	0.6197	0.6382	0.6327
bupa	**0.6807**	0.6265	0.6065	0.5822	0.6361	0.6689	0.5886	0.6173	0.6198	0.6192	0.6323	0.5704	0.5889	0.5789	0.5742	0.6724
cleveland	0.5574	0.5146	0.5484	**0.5778**	0.5482	0.5644	0.5681	0.5445	0.5640	0.5551	0.5580	0.5471	0.5675	0.5345	0.5346	0.5545
contraceptive	0.4909	0.4970	0.5105	0.5010	0.5506	0.4290	0.4943	0.5187	0.5221	0.4990	0.4840	0.5045	0.5290	0.4875	0.5316	**0.5595**
crx	0.8551	0.8522	0.8739	0.8594	0.8667	0.8696	0.8478	0.8681	**0.8797**	0.8652	0.8768	0.8768	0.8507	0.8522	0.8667	0.8594
dermatology	0.9535	0.9533	0.9318	0.9070	0.9424	0.9178	0.9291	0.9506	0.9508	0.9318	0.9152	**0.9589**	0.9508	0.9454	0.9369	0.9289
ecoli	0.7082	0.7915	0.7469	0.7291	0.7708	0.7859	0.6940	0.7175	0.7503	0.7202	0.7205	0.7381	0.7530	0.6603	**0.7921**	0.7378
flare	0.6782	0.6782	0.6782	0.6764	0.6782	0.5525	0.6754	0.6633	0.6735	0.6754	0.6754	0.6754	**0.6792**	0.6754	0.6754	0.6754
glass	0.5305	0.3557	0.6761	0.6812	0.6814	0.6348	0.6427	0.6927	0.6679	0.5938	0.7015	0.6280	0.6950	0.5788	0.6349	**0.7038**
haberman	0.7413	0.7349	**0.7512**	0.7353	0.7347	0.7219	0.7256	0.7353	0.7251	0.7319	0.7219	0.7353	0.7087	0.7353	0.7512	0.7188
iris	0.9333	0.9333	0.9333	0.9467	0.9333	0.7867	0.9467	**0.9533**	0.9400	0.8467	0.9267	0.9333	0.9400	0.9267	0.9333	0.9400
mammographic	0.8159	0.8221	0.8284	0.8210	**0.8325**	0.7856	0.8044	0.8304	0.8263	0.8221	0.8252	0.8200	0.8179	0.8117	0.8273	0.8023
newthyroid	0.9253	0.9253	0.9351	0.9210	**0.9444**	0.9022	0.8649	0.8749	0.9165	0.9169	0.9303	0.9398	0.9305	0.9398	0.9353	0.9400
saheart	0.6991	0.6537	0.7055	0.7036	0.7080	0.6818	0.6885	**0.7121**	0.6450	0.6906	0.6689	0.6971	0.6409	0.6580	0.7011	0.6905
specfheart	**0.8050**	0.7642	0.7785	0.7530	0.7829	0.7678	0.7681	0.7947	0.7496	0.7905	0.7718	0.7752	0.7531	0.7942	0.7869	0.7604
tae	0.4454	0.4838	0.4583	0.5296	0.5433	0.3725	0.4854	0.5296	0.5113	**0.5629**	0.5233	0.5296	0.4579	0.4708	0.5171	0.5442
wine	0.9382	0.9265	0.9101	0.8252	0.9042	**0.9490**	0.8922	0.6343	0.9379	0.9036	0.8641	0.9268	0.8984	0.7974	0.9212	0.8706
wisconsin	0.9371	0.9371	0.9385	0.9513	0.9514	**0.9557**	0.9528	0.9470	0.9456	0.9542	0.9456	0.9471	0.9442	0.9384	0.9456	0.9427
MEAN	*0.7477*	*0.7347*	*0.7508*	*0.7468*	***0.7612***	*0.7313*	*0.7366*	*0.7306*	*0.7520*	*0.7483*	*0.7507*	*0.7521*	*0.7448*	*0.7287*	*0.7531*	*0.7566*

2008), CAIM (Kurgan and Cios, 2004), Chi2 (Liu and Setiono, 1997), ChiMerge (Kerber, 1992), DIBD (Wu et al., 2007), E-Width (Liu et al., 2002), Ext-Chi2 (Su and Hsu, 2005), FUSINTER (Zighed et al., 1998), Hellinger (Lee, 2007), Khiops (Boullé, 2004), Mod-Chi2 (Tay and Shen, 2002), MODL (Boullé, 2006), PKID (Yang and Webb, 2009) and Zeta (Ho and Scott, 1997).

The classifiers used are C4.5 (Quinlan, 1993) and Naive Bayes (Cios et al., 2007). Implementations of these algorithms can be found under the KEEL data mining tool (Alcalá-Fdez et al., 2009).

4.1 Experimental Framework

Performance of the algorithms is analyzed by using 20 data sets taken from the UCI Machine Learning Database Repository (Frank and Asuncion, 2010). The main characteristics of these data sets are summarized in Table 1. For each data set, it shows the number of examples (#Ex.), the total number of attributes (#Atts.), which some of the could be numerical (#Num.), or nominal (#Nom.) and the number classes (#Cl). The data sets considered are partitioned using the *ten fold cross-validation (10-fcv)* procedure. The parameters of the discretizers and classifiers are presented in Table 2.

4.2 Results and Analysis

Tables 3 and 4 show the accuracy in test data obtained by C4.5 and Naive Bayes, respectively, when using different discretization approaches. The best case in each data set is highlighted in bold. Table 5 shows the average cut points selected by each discretizer.

Observing the mentioned tables, we can make the following analysis:

- In terms of accuracy, ECPSD is the second best option when using C4.5 as classifier. When considering Naive Bayes, ECPSD can be situated in half of the set of methods in terms of accuracy.
- The number of cut points yielded by ECPSD is the lowest in average.
- A remarkable case in point is observed in the *contraceptive* data set. ECPSD only requires 4 cut points to offer the best accuracy with C4.5 and Naive Bayes. In some data sets, i.e. *appendicitis* and *flare*, the proposal provides a very good trade-off of accuracy and number of cut points.

We have included a second type of analysis accomplishing a statistical comparison of methods over multiple data sets. The non-parametric Wilcoxon Signed-Ranks Test (Sheskin, 2007) is used for conducting pairwise comparison between our proposal and the rest of the techniques. Table 6 collects the results offered by the Wilcoxon test. This table is divided into three parts, each one associated with columns: In the first and second parts, the measure of accuracy classification in test is used for C4.5 and Naive Bayes, respectively. In the third part, we accomplish the Wilcoxon test by using as performance measure the number of cut points produced by the discretizers. In each part, our proposed method is compared against N_a rows where N_a is the number of discretizers compared in this study. In each one of the cells it can appear three symbols: +, = or -. They represent that the proposal outperforms (+), is similar (=) or is worse (-) in performance than the discretizer which appears in the column (Table 6). The value in brackets is the p-value obtained in the comparison and the level of significance considered is $\alpha = 0.10$.

- Statistically, no method can be considered better than our proposal ECPSD in accuracy. It also outperforms four discretizers: *E-Width* and *PKID* for C4.5, *DIBD* for Naive Bayes and *Ext-Chi2* for

Table 4: Average accuracy obtained for Naive Bayes.

	Ameva	*CACC*	*CAIM*	*Chi2*	*ChiMerge*	*DIBD*	*E-Width*	*Ext-Chi2*	*FUSINTER*	*Hellinger*	*Khiops*	*Mod-Chi2*	*MODL*	*PKID*	*Zeta*	*ECPSD*
appendicitis	**0.8800**	**0.8800**	0.8709	0.8618	0.8518	**0.8800**	0.8109	0.8018	0.8418	0.8609	0.8709	0.8518	0.8609	0.8609	0.8618	0.8709
autos	0.6729	0.7114	0.6497	0.5803	0.6603	0.6142	0.6333	0.6156	0.6580	0.6139	0.6219	0.6488	0.7228	**0.7269**	0.6379	0.6054
bands	**0.7255**	0.7031	0.6643	0.7124	0.6715	0.6493	0.6567	0.5992	0.7013	0.6457	0.6810	0.7235	0.7013	0.6865	0.6513	0.6160
bupa	0.6599	0.6397	0.6169	0.6519	0.6130	0.6130	0.5853	0.6316	0.6427	0.6246	0.6321	0.6376	0.6293	0.6277	0.5943	**0.6902**
cleveland	0.5778	0.5610	0.5612	0.5781	0.5449	0.5276	0.5812	0.5746	0.5580	0.5646	0.5615	0.5482	0.5477	0.5544	0.5743	**0.5874**
contraceptive	0.5064	0.5186	0.4963	0.5004	0.5255	0.4535	0.5085	0.5126	0.5132	0.5099	0.5099	0.5024	0.5241	0.5092	0.5179	**0.5595**
crx	0.8551	0.8406	0.8609	0.8478	0.8638	0.8507	0.8609	0.8304	0.8449	**0.8667**	0.8507	0.8420	0.8290	0.8522	0.8638	0.8565
dermatology	0.9810	0.9811	0.9755	0.9454	0.9755	0.9536	0.9728	0.9837	0.9783	0.9729	0.9511	**0.9865**	0.9838	0.9782	0.9783	0.9453
ecoli	0.8127	0.8275	0.8094	0.8215	0.8213	0.8094	0.8160	0.7919	0.8332	0.8068	0.7974	0.7948	0.8274	0.8038	**0.8335**	0.8098
flare	0.6557	0.6576	0.6557	0.6548	0.6539	0.5488	0.6632	0.6558	0.6576	0.6539	0.6641	0.6529	0.6632	0.6548	0.6679	**0.6754**
glass	0.4647	0.3557	0.7034	0.7061	0.6736	0.6534	0.6606	0.7109	0.6878	0.6432	0.6432	0.7199	**0.7417**	0.7246	0.6439	0.6832
haberman	0.7478	0.7351	0.7352	0.7220	0.7251	0.7219	0.7385	0.6990	0.7382	0.7384	0.7348	0.7220	0.7057	0.7282	**0.7545**	0.7252
iris	0.9333	0.9400	0.9400	0.9467	0.9400	0.7667	0.9133	0.9400	0.9467	0.8467	0.9200	0.9333	**0.9600**	0.9200	0.9267	0.9400
mammographic	0.8148	0.8158	0.8262	0.8252	0.8086	0.7919	0.7950	0.8252	0.8315	0.8263	0.8294	0.8263	**0.8346**	0.8325	0.8283	0.8023
newthyroid	0.9535	0.9535	0.9582	0.9437	0.9673	0.9537	0.9210	0.8749	**0.9725**	0.9115	0.9355	0.9582	0.9675	0.9675	0.9675	0.9446
saheart	0.6582	0.6689	0.7035	0.6709	**0.7249**	0.6604	0.6991	0.6968	0.6451	0.6840	0.6670	0.6777	0.6364	0.6756	0.7100	0.6818
specfheart	0.7644	0.7530	0.7682	0.7645	0.7345	0.7272	0.6893	0.7608	0.7756	0.7566	**0.7789**	0.7496	0.7454	0.7752	0.7681	0.7717
tae	0.5113	0.5042	0.4904	0.5504	0.5046	0.3658	0.4988	0.5238	0.4771	0.5113	0.5029	**0.5571**	0.4779	0.4904	0.5038	0.5242
wine	**0.9830**	0.9771	0.9775	0.8134	0.9830	0.9830	0.9549	0.6510	0.9663	0.9039	0.9771	0.9598	0.9608	0.9663	0.9719	0.9268
wisconsin	0.9671	0.9671	0.9671	0.9628	0.9700	0.9728	0.9714	0.9714	0.9742	0.9728	**0.9742**	0.9714	0.9714	0.9728	0.9657	0.9427
MEAN	*0.7563*	*0.7495*	*0.7615*	*0.7530*	*0.7606*	*0.7248*	*0.7465*	*0.7326*	*0.7622*	*0.7457*	*0.7552*	*0.7632*	*0.7646*	***0.7654***	*0.7611*	*0.7579*

Table 5: Average number of cut points.

	Ameva	*CACC*	*CAIM*	*Chi2*	*ChiMerge*	*DIBD*	*E-Width*	*Ext-Chi2*	*FUSINTER*	*Hellinger*	*Khiops*	*Mod-Chi2*	*MODL*	*PKID*	*Zeta*	*ECPSD*
appendicitis	8.30	8.60	8.00	24.00	8.00	8.00	8.00	**1.00**	16.70	8.00	17.80	35.00	64.80	64.00	8.00	8.20
autos	79.30	263.30	76.20	10.90	75.80	66.50	76.00	**4.70**	42.40	76.00	63.80	27.20	109.10	183.80	76.00	15.70
bands	34.80	67.80	20.00	27.00	19.80	17.80	52.00	**11.50**	78.70	58.00	79.10	53.40	63.80	235.40	20.00	25.60
bupa	10.70	30.40	7.00	41.80	7.00	8.40	13.00	17.30	33.10	13.00	32.00	111.80	48.60	91.10	**7.00**	22.50
cleveland	53.10	165.30	37.90	16.10	33.10	21.10	34.10	**13.70**	16.40	50.00	28.20	34.70	27.30	90.00	53.00	22.30
contraceptive	19.00	13.40	16.00	42.50	16.00	8.00	35.80	39.70	23.80	54.00	40.90	41.20	16.20	53.00	19.00	**4.00**
crx	9.40	261.80	7.00	4.90	6.50	12.50	28.00	**3.40**	44.80	31.00	36.80	92.20	211.60	114.20	7.00	12.10
dermatology	171.00	43.00	101.90	22.40	72.90	29.10	74.40	28.90	46.40	167.90	67.20	37.40	42.20	107.80	171.00	**7.70**
ecoli	49.30	12.20	37.90	34.60	37.20	15.00	37.00	72.90	**11.70**	41.90	22.00	62.50	23.70	82.00	49.30	20.00
flare	10.20	11.40	10.00	14.70	10.00	**2.00**	12.00	8.30	7.50	39.70	8.40	13.90	7.20	13.00	10.00	3.00
glass	55.00	**1.00**	55.00	32.90	52.80	42.20	48.00	18.00	14.10	55.00	10.00	51.10	35.30	96.40	55.00	28.10
haberman	6.60	10.90	4.00	46.90	4.00	3.20	4.00	43.50	11.20	4.00	14.20	46.60	11.00	36.00	4.00	**3.00**
iris	9.00	8.00	9.00	11.70	9.00	8.50	9.00	18.10	10.10	9.00	17.00	27.30	10.70	44.60	9.00	**3.30**
mammographic	6.10	6.40	6.00	50.20	6.00	6.10	19.00	42.10	12.60	31.00	22.60	47.50	13.60	43.00	6.00	**2.50**
newthyroid	11.00	10.00	11.00	9.10	11.00	9.80	11.00	**5.20**	14.60	11.00	19.80	16.30	17.70	65.70	11.00	5.80
saheart	12.70	278.40	9.00	32.70	9.00	16.50	25.00	**3.70**	50.80	25.00	53.90	54.10	385.60	145.70	9.00	33.90
specfheart	63.40	142.60	45.00	21.40	44.90	42.40	45.00	**7.90**	124.30	45.00	201.00	30.90	50.30	616.90	45.00	27.10
tae	15.70	50.00	9.00	66.90	9.00	**4.60**	9.00	52.80	15.20	11.00	15.20	70.40	30.60	33.10	11.00	13.30
wine	27.00	19.40	27.00	4.90	26.90	26.80	27.00	**2.50**	44.20	27.00	48.10	13.80	61.60	157.00	27.00	15.80
wisconsin	10.00	10.00	10.00	7.90	10.00	23.90	30.90	16.30	22.60	46.00	51.50	20.40	26.00	71.20	10.00	**3.10**
MEAN	*33.08*	*70.70*	*25.35*	*26.18*	*23.45*	*18.62*	*29.91*	*20.58*	*32.06*	*40.18*	*42.48*	*44.39*	*62.85*	*117.20*	*30.37*	***13.85***

both classifiers.

- The Wilcoxon test confirms that ECPSD is better than all the other discretizers, except *DIBD* and *Ext-Chi2* when considering the number of cut points.
- The preferred option considering the trade-off accuracy/simplicity can be established as the ECPSD. It needs to make a lower number cut points than the discretizers which are similar in accuracy, and outperforms in accuracy the discretizers with similar performance in simplicity.

Table 6: Wilcoxon's test results over accuracy for C4.5 and Naive Bayes and number of cutpoints.

Discretizer	Acc. C4.5	Acc. NB	Num. CP
Ameva	= (0.92)	= (1.00)	**+ (0.003)**
CACC	= (0.27)	= (0.90)	**+ (0.001)**
CAIM	= (0.65)	= (1.00)	**+ (0.019)**
Chi2	= (0.24)	= (0.47)	**+ (0.009)**
ChiMerge	= (1.00)	= (1.00)	**+ (0.029)**
DIBD	= (0.24)	**+ (0.05)**	= (0.187)
E-Width	**+ (0.08)**	= (0.43)	**+ (0.001)**
Ext-Chi2	**+ (0.08)**	**+ (0.08)**	= (0.385)
FUSINTER	= (0.42)	= (1.00)	**+ (0.001)**
Hellinger	= (0.96)	= (0.15)	**+ (0.001)**
Khiops	= (0.16)	= (0.54)	**+ (0.001)**
Mod-Chi2	= (1.00)	= (1.00)	**+ (0.000)**
MODL	= (0.25)	= (1.00)	**+ (0.000)**
PKID	**+ (0.02)**	= (1.00)	**+ (0.000)**
Zeta	= (0.84)	= (1.00)	**+ (0.014)**

5 CONCLUDING REMARKS

The purpose of this contribution is to present a proposal named Evolutionary Cut Points Selection Algorithm for discretization. The results shows that our proposal allows us to yield good discretization schemes which have been tested over C4.5 and Naive Bayes. We have checked that the proposal obtains very competitive results, because it often requires a lower number cut points than the yielded by the compared discretizers and outperforms in accuracy those

with similar simplicity capabilities.

As future work, we will improve the computationally demand required by our approach in order to tackle larger data sets.

ACKNOWLEDGEMENTS

This work was supported by the Research Projects TIN2011-28488 and TIC-6858.

REFERENCES

Alcalá-Fdez, J., Sánchez, L., García, S., del Jesus, M. J., Ventura, S., Garrell, J. M., Otero, J., Romero, C., Bacardit, J., Rivas, V. M., Fernández, J. C., and Herrera, F. (2009). KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3):307–318.

Boullé, M. (2004). Khiops: A statistical discretization method of continuous attributes. *Machine Learning*, 55:53–69.

Boullé, M. (2006). MODL: A bayes optimal discretization method for continuous attributes. *Machine Learning*, 65(1):131–165.

Cios, K. J., Pedrycz, W., Swiniarski, R. W., and Kurgan, L. A. (2007). *Data Mining: A Knowledge Discovery Approach*. Springer.

Elomaa, T. and Rousu, J. (1999). General and efficient multisplitting of numerical attributes. *Machine Learning*, 36:201–244.

Eshelman, L. J. (1990). The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. In *FOGA*, pages 265–283.

Flores, J. L., Inza, I., and Larra (2007). Wrapper discretization by means of estimation of distribution algorithms. *Intelligent Data Analysis*, 11(5):525–545.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository.

Freitas, A. A. (2002). *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer-Verlag New York, Inc.

González-Abril, L., Cuberos, F. J., Velasco, F., and Ortega, J. A. (2009). Ameva: An autonomous discretization algorithm. *Expert Systems with Applications*, 36:5327–5332.

Ho, K. M. and Scott, P. D. (1997). Zeta: A global method for discretization of continuous variables. In *KDD*, pages 191–194.

Kerber, R. (1992). Chimerge: Discretization of numeric attributes. In *National Conference on Artifical Intelligence American Association for Artificial Intelligence(AAAI92)*, pages 123–128.

Kurgan, L. A. and Cios, K. J. (2004). CAIM discretization algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 16(2):145–153.

Lee, C.-H. (2007). A hellinger-based discretization method for numeric attributes in classification learning. *Knowledge-Based Systems*, 20:419–425.

Liu, H., Hussain, F., Tan, C. L., and Dash, M. (2002). Discretization: An enabling technique. *Data Mining and Knowledge Discovery*, 6(4):393–423.

Liu, H. and Setiono, R. (1997). Feature selection via discretization. *IEEE Transactions on Knowledge and Data Engineering*, 9:642–645.

Pyle, D. (1999). *Data preparation for data mining*. Morgan Kaufmann Publishers Inc.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc.

Sheskin, D. J. (2007). *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC.

Su, C.-T. and Hsu, J.-H. (2005). An extended chi2 algorithm for discretization of real value attributes. *IEEE Transactions on Knowledge and Data Engineering*, 17:437–441.

Tay, F. E. H. and Shen, L. (2002). A modified chi2 algorithm for discretization. *IEEE Transactions on Knowledge and Data Engineering*, 14:666–670.

Tsai, C.-J., Lee, C.-I., and Yang, W.-P. (2008). A discretization algorithm based on class-attribute contingency coefficient. *Information Sciences*, 178:714–731.

Wu, Q., Bell, D. A., Prasad, G., and McGinnity, T. M. (2007). A distribution-index-based discretizer for decision-making with symbolic ai approaches. *IEEE Transactions on Knowledge and Data Engineering*, 19:17–28.

Wu, X. and Kumar, V., editors (2009). *The Top Ten Algorithms in Data Mining*. Chapman & Hall/CRC Data Mining and Knowledge Discovery.

Yang, Y. and Webb, G. I. (2009). Discretization for naive-bayes learning: managing discretization bias and variance. *Machine Learning*, 74(1):39–74.

Yang, Y., Webb, G. I., and Wu, X. (2010). Discretization methods. In *Data Mining and Knowledge Discovery Handbook*, pages 101–116. Springer.

Zighed, D. A., Rabaséda, S., and Rakotomalala, R. (1998). FUSINTER: a method for discretization of continuous attributes. *International Journal of Uncertainty, Fuzziness Knowledge-Based Systems*, 6:307–326.