Input space reduction for Rule Based Classification

MOHAMMED M MAZID, A B M SHAWKAT ALI, KEVIN S TICKLE
School of Computing Science
Central Queensland University
AUSTRALIA.
E-mail: {m.mazid, s.ali, k.tickle@cqu.edu.au}

Abstract: - Rule based classification is one of the most popular way of classification in data mining. There are number of algorithms for rule based classification. C4.5 and Partial Decision Tree (PART) are very popular algorithms among them and both have many empirical features such as continuous number categorization, missing value handling, etc. However in many cases these algorithms takes more processing time and provides less accuracy rate for correctly classified instances. One of the main reasons is high dimensionality of the databases. A large dataset might contain hundreds of attributes with huge instances. We need to choose most related attributes among them to obtain higher accuracy. It is also a difficult task to choose a proper algorithm to perform efficient and perfect classification. With our proposed method, we select the most relevant attributes from a dataset by reducing input space and simultaneously improve the performance of these two rule based algorithms. The improved performance is measured based on better accuracy and less computational complexity. We measure Entropy of Information Theory to identify the central attribute for a dataset. Then apply correlation coefficient measure namely, Pearson's, Spearman and Kendall correlation utilizing the central attribute of the same dataset. We have conducted a comparative study using these three most popular correlation coefficient measures to choose the best method. We have picked datasets from well known data repository UCI (University of California Irvine) database. We have used box plot to compare experimental results. Our proposed method has showed better performance in most of the individual experiment.

Key words: Classification, C4.5, PART, Entropy, Pearson's Correlation, Spearman Correlation, Kendall Correlation.

1 Introduction

Classification is one of the most significant areas in data mining. It is also known as pattern recognition, discrimination or prediction. Classification algorithms extract patterns by using data files with a set of labeled training examples. Classification algorithms are in the supervised learning group because they build a classifier/model based on supplied classes. It uses classifiers to predict classes. A classifier is a global model which generates a concise and eloquent description for each class by using attributes of data files [1]. A classifier is computed with decision functions $f(\mathbf{x}_i, \alpha_i) = y_i, \alpha_i \in \Lambda, \forall \langle \mathbf{x}_i, y_i \rangle \in D$ where D is a dataset with I independently identically distributed samples: $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_l, y_l)$; samples are set of feature vectors with length m; binary class $y_i \in \{+1, -1\}$ is the target value and Λ is a set of abstract parameters [2]. Classification algorithms have made significant inroads in the fields of bioinformatics, medical diagnosis, weather prediction, fraud detection, loan risk prediction, customer segmentation, target marketing, text classification, engineering fault detection and more. Because classification covers such a wide range of data mining, researchers have discovered many approaches such as rule-based classification, induction and covering approach, associative and instance-centric approaches, genetic algorithm based approaches, probability theory, etc. Each approach has at least one or more popular algorithms, for instance C4.5 [3], PART (Partial Decision Trees) [4], SVM (Support Vector Machine) [5], NN (Neural Networks) [6], Naive Bayes [7], etc. Foremost objective of this research is to reduce input space before classification task. Insignificant factors should be discarded from dataset while mining process as irrelevant factors could affect on final outcome[8].

In this research, we have chosen two most popular rule based classification algorithms namely C4.5 and PART. In the following section, we have discussed briefly about rule based classification and those two algorithms i.e. C4.5 and PART with their features, limitations, etc. Section 3 is about few past research conducted by various researchers on these two algorithms. Section 4 reveals our proposed solution. Section 5 explores our experimental design and finally conclusion is in section 6.

2 Rule-based classification

Rule-based classifier is one of the popular rule induction

methods. It follows some heuristic steps to generate rules. First it split data file into training and testing in predefine manner. Then build a model from the training data set and extract a set of high quality rules from it. These rules are then used for prediction of the class of unlabeled instances of test data set. Some of the rule based classifiers are C4.5 [3], Incremental Reduced Error Pruning(IREP) [9], First Order Inductive Learner(FOIL) [10], Classification based on Predictive Association Rules(CPAR) [11], etc. Several researches have been performed on rules based classification. According to those researches, rules based classification performs well in categorical databases [11-12] and sparse high dimensional databases especially with the context documentation classification [13-14]. In rule based classification approach, rules are generated using many heuristic methods for pruning the search space. Rule selection is based on sequential database covering paradigm [15]. Rule selection is based on sequential database covering paradigm. This strategy could mislead final rule selection. Such as, the final rule set might not be the globally best rules for some instances in training dataset[16]. It could produce worse result in case of large data file. When data files are really large, handling of search space for pruning becomes very difficult with this approach.

Rule based classifiers like FOIL [10], CPAR [11] and RIPPER [17] utilize sequential covering methodology. In this methodology rules are extracted one at a time. Every time, a rule is learned from examples by applying various heuristics formula (for instance information gain) to sort out the best attributes (or variables or literals) and this rule is covered by those examples. Then examples are removed before a new rule is extracted [18]. In this methodology, learning a rule set to a sequence becomes simpler by implementing the search of individual rules. But in many times the concluding rules are not assured the best rules in dataset in spite of following the heuristic steps and sequential covering. As instances are removed periodically each time, calculation of information gain is performed on incomplete data. As a consequence, attributes selected to extract more rules might not be any globally optimal attribute. Specifically databases with large number of classes, this algorithm have to apply various numbers of times. As a result this algorithm is not efficient for multi class database.

In this research, we have choose two most popular rule based classification algorithms those are C4.5 [3] and PART [4]. Numbers of researches have conducted on these two rules. Following two sections are regarding their features, limitations, etc.

2.1 C4.5

C4.5 is a popular decision tree based algorithm to solve data mining task. Professor Ross Quinlan from University of Sydney has developed C4.5 in 1993 [3]. Basically it is the advance version of ID3 algorithm, which is also proposed by Ross Quinlan in 1986 [19]. C4.5 is decision tree based algorithm. The decision tree-growing algorithm basically emphasis on pointing which attribute to test at each node in the tree [20]. C4.5 has additional features such as handling missing values, categorization of continuous attributes, pruning of decision trees, rule derivation and many others. C4.5 constructs a very big tree by considering all attribute values and finalizes the decision rule by pruning. It uses a heuristic approach for pruning based on the statistical significance of splits. Basic construction of C4.5 decision tree is as follows[21].

- The root nodes are the top node of the tree. It considers all samples and selects the attributes that are most significant.
- The sample information is passed to subsequent nodes, called 'branch nodes' which eventually terminate in leaf nodes that give decisions.
- Rules are generated by illustrating the path from the root node to leaf node.

Dealing huge data with computational efficiency is one of the major challenges for C4.5 users. Most of the time, it is very difficult to handle data file when dimensionality expands enormously during process for rule generation. As C4.5 uses decision tree, it needs to consider some other issues such as depth of the decision tree, handling of continuous attributes, method of selection measure to adopt significant attributes, dealing of missing values, etc. Following section illustrates about some features of C4.5 algorithm.

2.1.1 Features of C4.5 Algorithm

There are several features of C4.5. Some features of C4.5 algorithm are discussed below.

- **Continuous Attributes Categorization:** Earlier versions of decision tree algorithms were unable to deal with continuous attributes. 'An attribute must be categorical value' was one of the preconditions for decision trees [21]. Another condition is 'decision nodes of the tree must be categorical' as well. Decision tree of C4.5 algorithm illuminates this problem by partitioning the continuous attribute value into discrete set of intervals which is widely known as 'discretization'. For instance, if a continuous attribute C needs to be processed by C4.5 algorithm, then this algorithm creates a new Boolean attributes C_b so that it is true if $C<b$ and false otherwise [22]. Then it picks values by choosing a best suitable threshold.

• **Handling Missing Values:** Dealing with missing values of attribute is another feature of C4.5 algorithm. There are several ways to handle missing attributes. Some of these are Case Substitution, Mean Substitution, Hot Deck Imputation, Cold Deck Imputation, Nearest Neighbour Imputation [22]. However C4.5 uses probability values for missing value rather assigning existing most common values of that attribute. This probability values are calculated from the observed frequencies in that instance. For example, let A is a Boolean attribute. If this attribute has six values with A=1 and four with A=0, then in accordance with Probability Theory, the probability of A=1 is 0.6 and the probability of A=0 is 0.4. At this point, the instance is divided into two fractions: the 0.6 fraction of the instances is distributed down the branch for A=1 and the remaining 0.4 fraction is distributed down the other branch of tree. As C4.5 split dataset to training and testing, the above method is applied in both of the datasets. In a sentence we can say that, C4.5 uses most probable classification which is computed by summing the weights of the attributes frequency.

2.1.2 Limitations of C4.5 Algorithm

Although C4.5 one of the popular algorithms, there are some shortcomings of this algorithm. Some limitations of C4.5 are discussed below.

• **Empty branches:** Constructing tree with meaningful value is one of the crucial steps for rule generation by C4.5 algorithm. In our experiment, we have found many nodes with zero values or close to zero values. These values neither contribute to generate rules nor help to construct any class for classification task. Rather it makes the tree bigger and more complex.

• **Insignificant branches:** Numbers of selected discrete attributes create equal number of potential branches to build a decision tree. But all of them are not significant for classification task. These insignificant branches not only reduce the usability of decision trees but also bring on the problem of over fitting.

• **Over fitting:** Over fitting happens when algorithm model picks up data with uncommon characteristics[23]. This cause many fragmentations is the process distribution. Statistically insignificant nodes with very few samples are known as fragmentations [24]. Generally C4.5 algorithm constructs trees and grows it branches 'just deep enough to perfectly classify the training examples'. This strategy performs well with noise free data. But most of the time this approach over fits the training examples with noisy data. Currently there are two approaches are widely using to bypass this over-fitting in decision tree learning [1]. Those are:

- If tree grows very large, stop it before it reaches maximal point of perfect classification of the training data.
- Allow the tree to over-fit the training data then post-prune tree.

2.2 PART

The PART algorithm was developed by Frank and Witten [4]. PART is acronym of Partial Decision Tree. This algorithm produce rules by recurrently generating partial decision trees from data set. That is why this algorithm is called PART (Partial Decision Trees). PART is the extended version of C4.5 [3] and RIPPER [17]. One of the common properties of these algorithms is utilization of decision tree to generate rule set. However PART does not perform global optimization like other two algorithms. Global optimization technique builds the tree and then generates rules from the tree. Finally it simplifies rules from generated rules. This technique is not suitable for huge data sets. Rather PART applies "separate-and-conquer" to overcome deficiency of its antecedent algorithms [25]. In this strategy, one rule is generated at a time. Then it removes the instances covered by that rule and iteratively induces further rules for the remaining instances until none is left. In a multi-class setting this automatically leads to an ordered list of rules. An ordered list of rules is a type of classifier that is termed as 'decision list'. It differs from the standard approach in the way that each rule is created. To generate a single rule, a pruned decision tree is built for the current set of instances. Tree nodes with the largest coverage are made into a rule and the tree is discarded. This avoids hasty global generalization. PART is an ordinary decision tree that contains branches to undefined sub-trees [4].

2.2.1 Features of PART Algorithm

As mentioned earlier that PART is an extended version of C4.5 [3] and RIPPER [17], it has many features of these two algorithm. However, one of the considerable differences from them is rule generation technique. C4.5 [3] and RIPPER [17] generate each rule in two phases. On the other hand, PART by pass extensive pruning by generating one rule at a time [4]. This saves huge time in case of large dataset. Moreover, PART does not construct a complete decision tree rather it builds a partial decision tree. This is another plus point of saving time. PART produces each rule from leaf node of the partial decision tree that has largest coverage of the tree. As a result, accuracy of most of the rules is generally higher than other two algorithms. PART deals with missing value which is feature not available in many

algorithms. In a brief, PART produces rules with better accuracy and consuming less time, compare to its predecessor such as C4.5 and RIPPER.

2.2.2 Limitations of PART Algorithm

PART is one of the efficient classification algorithms. Generate rules are meaningful and accurate than antecedent algorithms. Although PART seems to be better performer than others, there are some issues need to consider. Such as:

- Most of the time algorithms require that the class attribute is categorical
- Tends to perform well if a few highly relevant input attributes exist but less so if complex interactions between input attributes exist
- Over sensitive to irrelevant attributes and unstable performance when noise in training data
- Suffer from fragmentation problem if there are many relevant attributes
- Splitting criterions ranks possible splits based on their immediate descendants and may overlook effects of combinations of attributes

Numbers of researches have been conducted for improvement of these two algorithms. In the following section we have presented a few of them.

3 Recent Works

C4.5 is one of the most widely use algorithm for inductive inference because of its efficiency and comprehensive features. As a result, data miners have proposed several techniques for betterment of this algorithm. In this section, we are going to discuss few recent works. Polat and Gunes [26] have offered 'one against all approach' with C4.5. They have conducted experiment with three famous data set namely Dermatology, Image segmentation, Lymphography from UCI. In their experiment they have found excellent accuracy against other algorithms. But did not mention regarding time and the performance against other type of database. In many cases, algorithms are biased by the nature of data files [27]. Jiang and Yu [28] have proposed a hybrid algorithm based on outlier detection and C4.5. They have worked with imbalance data to make them balance using outlier detection then implement C4.5 algorithm. Their proposed algorithm shows good accuracy relatively to other algorithm namely C4.5 and RIPPER [17]. But differences of accuracy with other algorithms are not considerably high according to their experiment result. Computational time is not mentioned in this paper as well. Yu and Ai [29] have worked for classification of Remote Sensing (RS) data using rough set and C4.5 algorithm. Their algorithm performs well on that specific data type. Yang [30] has used hierarchical clustering to limit the decision tree to binary tree to improve traditional C4.5 algorithm. The author's algorithm successfully trim down the number of leaf nodes and improve accuracy. In our proposed improvement of C4.5, we use Entropy and Correlation Coefficients. We use box plot to compare the significance of accuracy and time. As we mentioned earlier that PART algorithm is derived form C4.5 and RIPPER. This algorithm also has many limitations that we discussed in Section 4.2.

However none of those are complete solution of this problem. So in this research we have proposed two tools to reduce the input space of data. The first tool is Entropy of Information Theory and the second is Correlation Coefficient. In this study, we have examined some famous datasets from the UCI Repository [31]. The details of the data sets description is provided in Table 1 and Table 3. A Java based machine learning tool Weka3.4 [32] is used to perform the experiment. The machine configuration is Intel Core2 Duo CPU 2.33GHz and 4GB RAM.

4 Proposed Method

Basic focus of our experiment is to reduce the input space of a data file, roll back the processing time and boost up the percentage of classification accuracy. To do so, we propose popular measurement of Information Theory the Entropy. Entropy finds out the average uncertainty of collection of data. We have used it to find out the central point of the data file. After getting the central point, we have applied the correlation coefficient to choose significant attributes in the data files. Then we have applied C4.5 algorithm on chosen significant attributes. There are brief discussions on the Entropy and three types of correlation coefficient in the following sections.

4.1 Entropy

Information theory (IT) is a widely used topic for computer scientists, cognitive scientists, data miners, statisticians, biologists, and engineers. In information theory, entropy measures the uncertainty among random variables in a data file. Claude E. Shannon [33] has developed the idea of entropy of random variables. He introduced the beginnings of information theory and the modern age of Ergodic theory. Entropy and related information provides the long term behaviour of random processes that are very useful to analyse data.

The behaviour of random process is also a key factor for developing the coding for information theory. Entropy is a measurement of average uncertainty of collection of data when we do not know the outcome of an information source. That means it's a measurement of how much information we do not have. This also indicates the average amount of information we will receive from outcome of an information source.

Let X is an attribute, p is each element and j is position of each element of X then calculation for entropy is

$$H(X) = \sum_{j=1}^{k} p_j \log_2 \frac{1}{p_j}$$

$$= -\sum_{j=1}^{k} p_j \log_2 p_j \quad \ldots\ldots\ldots (1)$$

Larger value $H(X)$ indicates that attribute X is more random. On the other hand, attribute with smaller $H(X)$ value implies less random i.e. this attribute is more significant for the data mining. The value of the entropy attains its minimum 0, when all other p_j's are 0. The value reaches its maximum $\log_2 k$, when all p_j's are equal to 1/k.

4.2 Correlation coefficient

Correlation coefficient is one of the major statistical tools to analysis sets of variables and determines their relationships. So that user can make decisions on the basis of provided information by correlation coefficients. Thus it saves millions even billions of dollars for businessman, reduces enormous time for researchers and scale down effort for many other working person in various profession. Researchers have worked on this tool to improve its efficiency by introducing different way of calculation. Among different correlation coefficients, we have chosen three most popular one which are Pearson's, Spearman's and Kendall correlation coefficients. In the following section we have describe briefly about those.

4.2.1 Pearson correlation coefficient

Pearson's correlation coefficient is developed by Karl Pearson [34]. It measures the linear relationship between two variables by comparing their strength and direction. Relationship between two variables is expressed by -1 to +1. If the variables are perfectly linear related by an increasing relationship, the Correlation Coefficient gains the maximum value i.e. +1. On the other hand, if the variables are perfectly linear related by a decreasing relationship, the correlation value gains -1. And a value of 0 expresses that the variables are not linear related by each other. In general, if the correlation coefficient is greater than 0.8, it expresses strong correlation between variables.

Let X and Y are interval or ratio variables. They are normal distribution and their joint distribution is bivariate normal. So the formula of Pearson's Correlation Coefficient is:

$$r_{xy} = \frac{\sum XY - \frac{(\sum X)(\sum Y)}{n}}{\sqrt{(SS_X)(SS_Y)}} \qquad OR$$

$$r_{xy} = \frac{\sum XY - \frac{(\sum X)(\sum Y)}{n}}{\sqrt{\left[\left(\sum X^2 - \frac{(\sum X)^2}{n_x}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{n_y}\right)\right]}}$$

…….. (2)

Where
ΣX is sum of all the X scores.
ΣY is sum of all the Y scores.
ΣX^2 is square of each X score and then sum of them.
ΣY^2 is square of each Y score and then sum of them.
ΣXY is multiply of each X score by its associated Y score and then add of the resulting products together. This is also called cross product.
n refers to the number of "pairs" of data

4.2.2 Spearman's rank correlation coefficient

Spearman's correlation [35] uses nonparametric method to measure the correlation between variable. It describes the relationship of arbitrary monotonic function of two variables. This correlation does not need frequency distribution of the variables for calculation. Assumption of linear relationship between variable is not required in this correlation. Generally Spearman correlation coefficient is denoted by the Greek letter ρ (rho). It performs well with testing the null hypothesis off the relationship. The range of value of Spearman's correlation coefficient is -1 to +1.

In order to compute the Spearman rank correlation coefficient, the two variables (X and Y) are converted to ranks. A rank is assigned according with the position of value into a sort serried of values. In assignment of rank process, the lowest value had the lowest rank and the highest value has the highest rank. When there are two equal values for two different compounds, the associated rank had equal values and is calculated as

means of corresponding ranks. Then we need to calculate the difference between two ranks. Let d is the difference of two ranks and n is the total pair of variables, the formula of Spearman's correlation coefficient is:

$$\rho = 1 - \frac{6 \times \sum d_i^2}{n(n^2 - 1)} \quad \text{……...…….. (3)}$$

4.2.3 Kendall's rank correlation coefficients

Kendall correlation coefficient [36] is also uses nonparametric method for correlation measure. It is also regarded as Spearman rank correlation coefficient. Spearman correlation is calculated from variables' rank rather Kendall correlation is associated with probability calculation. Kendal Correlation coefficient is denoted with the Greek letter τ (tau). Kendall-tau uses concordant or discordant values. The range of value of Kendall correlation coefficient is -1 to +1.

Let X and Y be the pair of measured and estimated inhibitory activity. Kendall tau coefficient is defined as

$$\tau = \frac{n_c - n_d}{\frac{1}{2} n(n-1)} \quad \text{……...…….. (4)}$$

Where n_c is concordant value, n_d is discordant value and n is total number of instance.

5 Experimental Design

To perform our experiment, we have calculated entropy using Matlab [37]. We choose the attribute with minimum entropy value. According to entropy property, we nominate that attribute as the central attribute of the database. Then we find out Pearson's, Spearman and Kendall correlation coefficient based on the central attribute using Matlab. Finally we have applied C4.5 algorithm for Experiment 1 and PART algorithm for Experiment 2 with WEKA [31]. WEKA provides different types of test options to classify data files such as use training set, supplied test set, cross validation and percentage split. We choose 10 fold cross validation if number of instance less than or equal to 1000. In case of more than 1000 instance, we have split data file to 70% training and 30% testing data.

5.1 Experiment 1

Experiment 1 is carried out on popular rule based algorithm C4.5. We compare C4.5 with our proposed approach in this experiment. The following two sections describe about data and experimental outcome.

5.1.1 Data Description

We have experiment on 8 data files. All these data files are picked up from popular UCI [31] data repository. Table 1 shows the details of those files.

Table 1: Data files properties

Data file Name	Total Instances	Total Attribute *(before improved method applied)*	Total Attribute *(after improved method applied)*
optdigits	5620	65	34
waveFormNoise	5000	41	23
vehicle	846	19	13
ionosphere	351	35	19
Sonar	208	61	33
Glass	214	10	6
wpbc	199	34	21
parkinson	195	23	15

5.1.2 Experimental Outcome

Table 2 shows the comparison of modelling time and accuracy among original C4.5, improved Pearson's, improved Spearman and improved Kendall C4.5 algorithm. Improved Pearson's shows the supremacy in modelling time and accuracy for each data file except 'glass'. But improved Spearman C4.5 shows tremendous performance for that specific data file. It is said that Spearman correlation coefficient and Kendall correlation coefficient are similar type of correlation coefficient. However improved Spearman is more consistence than Kendall according to Box Plot analysis in Figure 1.

5.1.3 Result Analysis

We have used Box Plot [38], a visual representation of statistical technique with five number analyses to analyse our experimental data. We have applied Matlab [37] to construct the box plot. Figure 1 reflects about comparison among original C4.5 and our improved C4.5 algorithms. According to Box Plot illustration of Figure 1, the median line of box for C4.5 algorithm is at 76%. On the other hand, median line for improved C4.5 with Pearson's, Spearman and Kendall correlation coefficients are 81%, 79% and 80% respectively.

Table 2 : Comparisons of original C4.5 and three Improved C4.5

Data file Name	C4.5		Improved C4.5 (Pearson's)		Improved C4.5 (Spearman)		Improved C4.5 (Kendall)	
	Modelling Time	*Accuracy*	*Modelling Time*	*Accuracy*	*Modelling Time*	*Accuracy*	*Modelling Time*	*Accuracy*
ionosphere	0.03	80.19%	0.02	**92.31%**	0.02	91.45%	0.02	92.02%
waveFormNoise	0.67	84.36%	0.41	**86.96%**	0.41	83.96%	0.36	83.82%
wpbc	0.02	70.35%	0.001	74.37%	0.001	74.37%	0.02	**75.88%**
optdigits	1.27	90.69%	0.72	**92.74%**	0.69	91.38%	0.72	91.38%
vehicle	0.05	72.46%	0.02	**73.26%**	0.02	71.62%	0.02	71.62%
glass	0.22	61.54%	0.02	60.00%	0.02	**69.23%**	0.02	60.0%
sonar	0.03	71.15 %	0.03	**72.65%**	0.03	71.83%	0.03	71.83%
parkinsons	0.02	80.51 %	0.02	**86.15%**	0.001	85.64%	0.02	84.62%

In regards of dispersion of data, inter-quartile ranges (both upper quartile and lower quartile) are also obtained superior value of box plot. As average performance of all algorithms are good, there are no potential outliers in this graphical chart. However pattern of skewness is not straightforward and not symmetrical for all algorithms. Improved C4.5 with Pearson's correlation coefficient has smaller values with low skew as it has longer whisker at the bottom of the box.

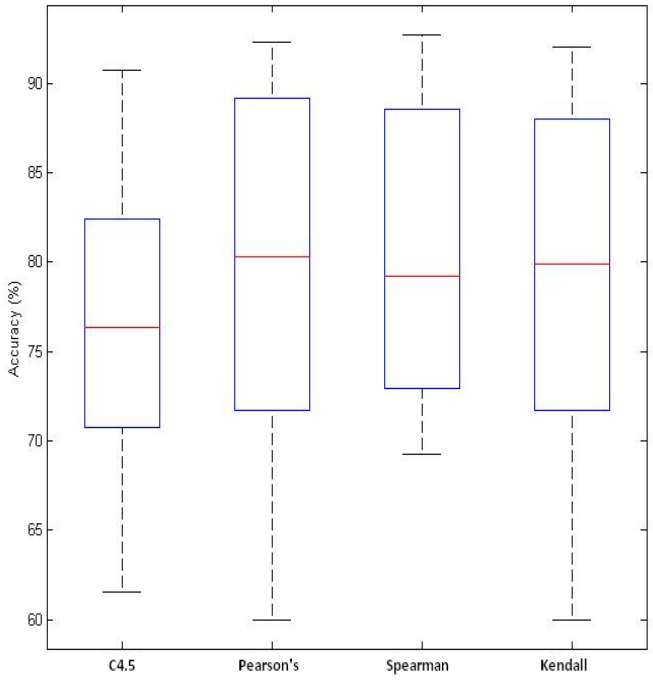

Figure 1: Box plot analysis of accuracy among algorithms

But the box itself is symmetrical which contain the middle 50% of accuracy experimental data of the improved Pearson's correlation coefficient algorithm. This box also obtains highest value of upper quartile among all the algorithms in our experiment. Whiskers of improved C4.5 with Spearman correlation coefficient are symmetrical. Moreover this box appears to be upper-skew, because the line marking of median is towards the bottom of the box. Thus the box indicates that accuracy of this algorithm has more upper values then lower. The box plot reflects that the nature of improved C4.5 with Pearson's and Kendall correlation coefficient are all most similar except a bit long whisker on top of Kendall. On the whole, general C4.5 algorithm has longer whiskers and relatively smaller box in the Figure 1 which indicates that performance of this algorithm is stagnant within a certain range. Whereas other improved C4.5 algorithms proposed in this paper are significantly better than the original C4.5 algorithm.

Figure 2 reveals comparison of processing time among C4.5 and improved C4.5 algorithms. At a glance we can explicate that our proposed C4.5 algorithms takes less processing time than original C4.5. There is an outlier for each box in the plot. Generally outlier appears in case of unusual properties of datasets. Among eight datasets, there is a relatively large data file compare to others and that file need more processing time. Original C4.5 has the highest value (1.27 sec) than other improved C4.5 algorithms. However, nowadays high performance computer, super computer, etc. are available for users. which lessen processing timing tremendously.

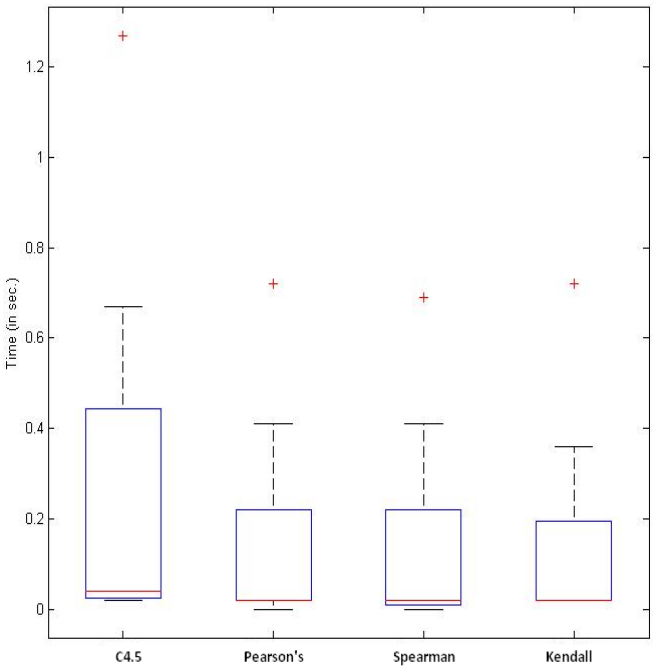

Figure 2: Box plot analysis of processing time among algorithms

5.2 Experiment 2

Experiment 2 is carried out on popular rule based algorithm PART. We have compared PART with our proposed approach in this experiment. We have followed the same procedure that we have used for Experiment 1. We have chosen some datasets that we have applied in Experiment 1 and some are different to examine the variation of our proposed approach. The following two sections describe about data and experimental outcome of this experiment.

5.2.1 Data Description

We have experiment on 7 data files. All these data files are picked up from popular UCI [31] data repository. Table 3 shows the details of those files.

Table 3: Data Files Properties

Data file name	**Total Instances**	**Total Attribute** ***(before improved method applied)***	**Improved Attributes** ***(after improved method applied)***
pendigits	10992	17	9
waveFormNoise	5000	41	24
mfeat-factors	2000	217	78
ionosphere	351	35	21
Glass	214	10	6
wpbc	199	34	21
parkinsons	195	23	15

5.2.2 Experimental Outcome

Table 4 shows the comparison of modeling time and accuracy among original PART, improved Pearson's, improved Spearman and improved Kendall PART algorithm. Improved Pearson's show the supremacy in modeling time and accuracy for each data file except 'parkinsons ' and 'ionosphere'. In terms of time comparision improved PART Person's performs always better or at least similar to other approaches. Details of result analysis are discussed in the following section.

Table 4: Comparisons of original PART and three Improved PART

Data file Name	**PART**		**Improved PART(Pearson's)**		**Improved PART (Spearman)**		**Improved PART (Kendall)**	
	Modelling Time	***Accuracy***	***Modelling Time***	***Accuracy***	***Modelling Time***	***Accuracy***	***Modelling Time***	***Accuracy***
parkinsons	0.03	81.02%	0.01	85.64%	0.02	**87.18%**	0.02	84.10%
wpbc	0.02	71.67%	0.02	**75.00%**	0.02	74.37%	0.02	74.37%
Glass	0.02	61.54%	0.02	**65.89%**	0.02	64.81%	0.02	64.81%
optdigits	3.95	92.82%	3.03	**95.30%**	3.34	93.11%	3.34	93.11%
mfeat-factors	2.94	89.80%	1.2	**93.80%**	1.9	91.80%	1.9	91.80%
waveFormNoise	1.61	84.53%	0.81	**85.90%**	0.77	85.16%	0.77	85.16%
ionosphere	0.09	83.02%	0.05	89.74%	0.001	**91.16%**	0.001	90.59%

5.2.3 Result Analysis

In this section we analysis our experiment result with Box plot. Figure 3 reflects about comparison among original PART and our improved PART algorithms. According to Box Plot illustration of Figure 3, the median line of box for PART algorithm is at 83%. On the other hand, median line for improved PART with Pearson's, Spearman and Kendall correlation coefficients are 86%, 87% and 84% respectively. In regards of dispersion of data, inter-quartile ranges (both upper quartile and lower quartile) are also obtained superior value of box plot. As average performance of all algorithms are good, there are no potential outliers in this graphical chart. However pattern of skewness is not straightforward and not symmetrical for all algorithms. Improved PART with Pearson's correlation coefficient has smaller values with low-skew as it has longer whisker at the bottom of the box. But the box itself is almost symmetrical which contain about middle 50% of

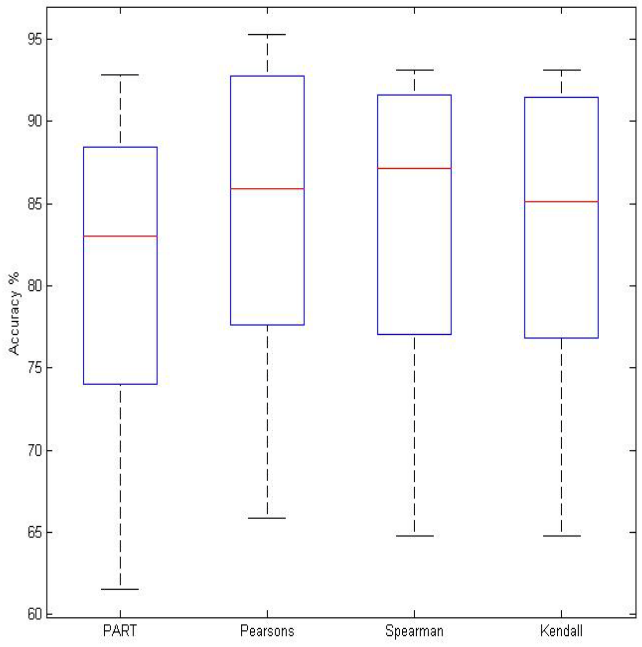

Figure 3: Box plot analysis of accuracy among algorithms

accuracy experimental data of the improved Pearson's correlation coefficient algorithm. This box also obtains highest value of upper quartile among all the algorithms in our experiment.

The box plot reflects that the nature of improved PART with Pearson's and Kendall correlation coefficient are all most similar except median lines of boxes. Median line of improved C4.5 with Spearman is toward top of the box which reflects that this box is lower skewed i.e. this algorithm has lower values then upper. Whereas Kendall contains almost a symmetrical box. Whiskers of improved PART with Spearman correlation coefficient and Kendall correlation coefficient are not symmetrical. Both of them have almost similar properties to Pearson's except a bit short whiskers at the top On the whole, general PART algorithm has longer whiskers and median line is relatively lower in the box according to Figure 3. This indicates that performance of this original C4.5 algorithm is relatively low than our proposed approach in terms of accuracy measure.

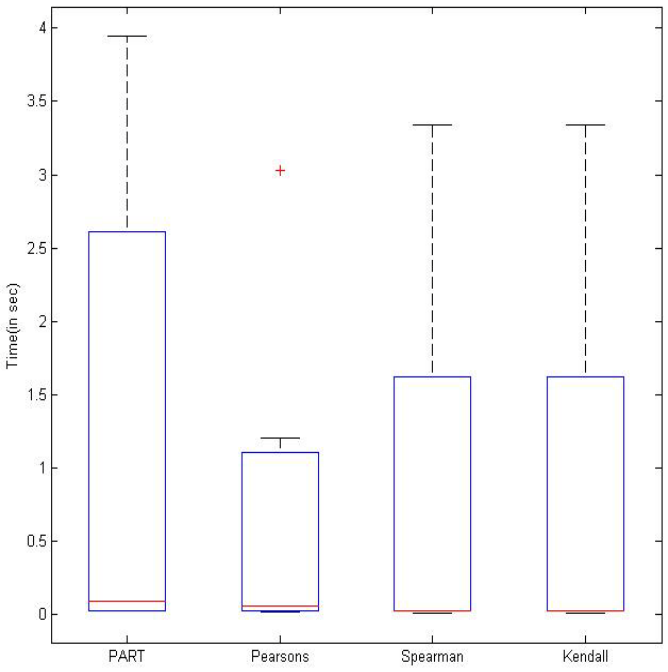

Figure 4: Box plot analysis of processing time among algorithms

Figure 4 reveals comparison of processing time among PART and improved PART algorithms. At a glance we can explicate that our proposed PART algorithms takes less processing time than original PART. There is an outlier for improved Pearson's PART because of variations of time. However, nowadays high performance computer, super computer, etc. are available for users that lessen processing timing tremendously.

6 Conclusion

In this research, we discuss briefly about rule based classification algorithm and two most popular algorithms that are PART and C4.5. We conducted two experiments for each algorithm to compare their performance in terms of produced rule's accuracy and

rule generation time. We implement our approach to improve their performance. In our approach we propose Entropy and three different correlation coefficients to pick the best approach that we offer. The main objective is to boost up the classification accuracy and simultaneously roll back timing to build a classification model. We have emphasized reducing input space using entropy and several correlation coefficients formulas. Experiment 1 is conducted with C4.5 algorithm and our proposed approach. Individually each improved C4.5 (i.e. our proposed approach) is performing better than original C4.5 in every test case. Although difference of performance varies from data file to data file. Improved Pearson's C4.5 is most consistent among three improved C4.5. Between improved Spearman C4.5 and improved Kendall C4.5, Spearman shows the better performance in Experiment 1. Experiment 2 is an evaluation of PART algorithm with our proposed approach i.e. Improved C4.5 with three correlation coefficient. In case of PART, our proposed method performed better then original PART as well. Among three improved PART, improved Pearson's PART is superior as well. Improved Spearman PART has performed well for two datasets out of seven datasets. Box plot analysis also reveals that our improved approach with Entropy and Correlation Coefficient is always performing better than original C4.5 and PART algorithm. In a brief, we can conclude that by reducing input space with Entropy and Correlation Coefficient, there will be significant improvement in C.45 and PART algorithm. The future issue of this research is to implement this approach in a new intrusion data. We will also investigate that why some data are performing well with improved Pearson's approach and some data are not. At the same time we need to explore the reason of performing good or bad for other two improved approaches with Spearman and Kendall.

References:

[1] S. Ali and K. A. Smith, On learning algorithm selection for classification, *Applied Soft Computing Journal,* vol. 6, pp. 119-138, 2006.

[2] A. B. M. S. Ali, Automated support vector learning algorithms, PhD Thesis, Monash University, Victoria, Australia, 2005.

[3] J. R. Quinlan, *C4. 5: programs for machine learning*. San Mateo, CA: Morgan Kaufmann, 2003.

[4] I. H. Witten and E. Frank, Generating accurate rule sets without global optimization, *in Proceedings of the Fifteenth International Conference*, San Francisco, CA, 1998.

[5] V. N. Vapnik, *The nature of statistical learning theory*: Springer Verlag, Heidelberg, DE, 1995.

[6] B. Müller*, et al.*, *Neural networks: an introduction*: Springer Verlag, 1995.

[7] R. Agrawal and R. Srikant, Fast algorithms for mining association rules, *in The 20th Int'l Conference on Very Large Databases*, Santiago, Chile, 1994.

[8] S. Shin*, et al.*, Development of a data mining methodology using robust design, *WSEAS Transactions on Computers,* vol. 5, pp. 852-857, 2006.

[9] J. Furnkranz and G. Widmer, Incremental reduced error pruning, *in Proceedings of the 11th Annual Conference on Machine Learning*, New Brunswick, NJ, 1994, pp. 70–77.

[10] J. R. Quinlan and R. M. Cameron-Jones*, FOIL: A midterm report*, Machine Learning: ECML-93. vol. 667/1993, ed: Springer Berlin / Heidelberg, 1993, pp. 1-20.

[11] X. Yin and J. Han, CPAR: Classification based on predictive association rules, in *Proceedings of the SDM*, San Francisco, CA,, 2003, p. 331.

[12] B. Liu*, et al.*, Integrating classification and association rule mining, *Knowledge Discovery and Data Mining,* pp. 80–86, 1998.

[13] M. L. Antonie and O. R. Zaïane, Text document categorization by term association, in *ICDM*, 2002, p. 19.

[14] C. Apté*, et al.*, Towards language independent automated learning of text categorization models, in *Proceedings of SIGIR*, 1994, p. 30.

[15] S. Ramos*, et al.*, Use of Data Mining Techniques to Characterize MV Consumers and to Support the Consumer-Supplier Relationship, *in Proceedings of the 6th WSEAS International Conference on Power Systems*, Lisbon, Portugal, 2006, pp. 296-301.

[16] J. Wang and G. Karypis, On mining instance-centric classification rules, *IEEE Transactions on Knowledge and Data Engineering,* pp. 1497-1511, 2006.

[17] W. W. Cohen, Fast effective rule induction, *In Proceedings of the Twelfth International Conference on Machine Learning* Chambery, France., 1993, pp. 115–123.

[18] A. Weijters and J. Paredis, Rule induction with a genetic sequential covering algorithm (GESECO), 2000, pp. 245–251.

[19] J. Quinlan, Induction of decision trees, *Machine Learning,* vol. 1, pp. 81-106, 1986.

[20] M. E. El-Telbany, Mining the classification rules: the egyptian rice diseases as case study, *in Proceedings of the 4th WSEAS International Conference on Telecommunications and Informatics*, Prague, Czech Republic, 2005.
[21] A. B. M. S. Ali and S. A. Wasimi, Data Mining: Methods and Techniques, ed. Victoria, Australia: *Thomson Publishers*, 2007.
[22] M. Singh. (2010, 18 March). *How to Handle Missing Values*. Available: http://www.articlesbase.com/information-technology-articles/how-to-handle-missing-values-538449.html#.
[23] A. Du and B. Fang, Machine Learning Algorithms in Chinese Web Filtering: Problems, Evaluation and Enhancement, *WSEAS Transactions on Information Science and Applications,* vol. 1, pp. 1072-1078, 2004.
[24] J. Han and M. Kamber, *Data mining: concepts and techniques*: Morgan Kaufmann, 2006.
[25] G. Bagallo and D. Haussler, Boolean feature discovery in empirical learning, *Machine Learning,* vol. 5, pp. 71-99, 1990.
[26] K. Polat and S. Güne, A novel hybrid intelligent method based on C4. 5 decision tree classifier and one-against-all approach for multi-class classification problems, *Expert Systems with Applications,* vol. 36, pp. 1587-1592, 2009.
[27] M. Mazid*, et al.*, "Finding a unique Association Rule Mining algorithm based on data characteristics," in *International Conference on Electrical & Computer Engineering*, Dhaka, Bangladesh, 2008, pp. 902-908.
[28] S. Jiang and W. Yu, A Combination Classification Algorithm Based on Outlier Detection and C4. 5, in *Springer*, 2009, p. 511.
[29] M. Yu and T. H. Ai, Study of RS data classification based on rough sets and C4. 5 algorithm, 2009, p. 10.
[30] X. Y. Yang, Decision tree induction with constrained number of leaf node, 2009.
[31] C. Blake and M. C.J. (2007, 10 February 2010). *UCI Repository of machine learning databases*. Available: http://archive.ics.uci.edu/ml/.
[32] I. Witten and E. Frank, Data mining: practical machine learning tools and techniques with Java implementations, *ACM SIGMOD Record,* vol. 31, pp. 76-77, 2002.
[33] C. Shannon, A mathematical theory of communication, *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 5, pp. 3-55, 2001.
[34] K. Pearson, Notes on the history of correlation, *Biometrika,* vol. 13, pp. 25-45, 1920.
[35] C. Spearman, The proof and measurement of association between two things, *The American journal of psychology,* pp. 441-471, 1987.
[36] M. Kendall, *Rank correlation methods*. New York: Hanfer Publishing Co., 1955.
[37] Matlab, *Statistics Toolbox User's Guide*, 6.2 ed. USA: The MathWorksInc, 2008.
[38] J. Tukey, Exploratory data analysis, 1977.