Attribute selection for classificatory analysis using a probabilistic approach

Amir Ahmad
Solid State Physics Laboratory,
Timarpur, Delhi., India-110054
email:amir_ahmed/sspl@ssplnet.org

Lipika Dey
Department of Mathematics,
Indian Institute of Technology,
Hauz Khas, New Delhi – 110016.
email:lipika@maths,iitd,ernet.in

Abstract

Dimensionality reduction is one of the key data analysis steps. Besides increasing the speed of computation, eliminating insignificant attributes from data enhances the quality of knowledge extracted from the data. In this paper we have proposed an efficient, conditional probability based method for computing the significance of attributes. The algorithm is highly scalable and can simultaneously rank all the attributes. The proposed method can be used to analyze pre-classified data by exploiting the attribute-to-class and class-to-attribute co-relations. The effectiveness of the approach is established through the analysis of various large test data sets. The method can be extended to extract classificatory knowledge from the data.

1. INTRODUCTION

Analysis of past data for extracting classificatory knowledge often provide excellent insights which can pave the way for better decision making to be applied in the future. One of the central problems of classificatory knowledge extraction is that of dimensionality or the selection of an appropriate subset of features which can preserve the classificatory knowledge. Feature selection aims at identifying a small set of significant subset of attributes from data that can produce good classification results at a reduced time. Presence of irrelevant, erroneous or redundant attributes in the data, also affect the quality of knowledge extracted. An attribute is redundant if it does not contribute anything new to the knowledge that is extracted without it. In such cases, normally the redundant attributes are co-related to other attributes. Pruning the feature set can also lead to simpler, more understandable models [2,3] for prediction.

In this paper, we have proposed an efficient methodology for analyzing pre-classified data sets and determine the significant attributes in the set by ranking them. Unlike some of the classical models, which typically look at all possible subsets of attributes to decide on an appropriate set, our method works independently on each attribute and computes the significance as a function of its classificatory power.

Knowledge engineering has traditionally looked at ranking of attributes and application of extracted classificatory knowledge for prediction as two different problems. We have however extended the task of attribute significance computation itself to classificatory knowledge extraction, which is then used for prediction. Unlike most of the algorithms for computation of significance of an attribute which vary exponentially with the data base size, the computation of significance using the proposed algorithms, varies linearly with the size of the data base. Hence the algorithm is highly scalable.

In the next section, we have provided a brief overview of some of the related work in this area. Sections 3 and 4 describe the design principles of the proposed algorithm followed by an analysis of its complexity. In section 5 we have presented performance evaluation measures obtained on some well known data sets. We have also provided some comparative measures on efficiency of the proposed scheme, wherever such data was available for other methods.

2. FEATURE SELECTION AND PREDICTION: A SURVEY

Feature selection attempts to select a subset of relevant descriptive attributes from pre-classified data. The minimal subset of attributes for which the classification accuracy remains the same as that of the original set of attributes, is termed as a **"reduct"**. If the feature set size is N then there are 2^N possible subsets. Some feature selection methods perform a complete search over this state space for the optimal subset [3]. The Importance Score technique reported in [5] is an example of such a method which tests the predictive accuracy of each feature on the basis of positive examples classified correctly by the feature in a training set. The minimum number of significant features that can achieve good classification accuracy is then selected by repeatedly reducing the number of features and observing its effect on the classification result. Obviously, the complexity of these feature selection methods is very high. Besides, reducts may not exist for all data sets.

Alternatively, heuristic methods are employed to find subsets of attributes, which do not necessarily preserve the classification knowledge accurately, but also do not deteriorate it significantly [2, 3]. A well known alternative to the exhaustive search procedure is the Branch and Bound algorithm presented by Narendra and Fukunaga in [4]. This is a top-down algorithm with back-tracking. In this method a monotonic criterion function is associated to feature subsets. A search tree is constructed in which the leaves represent subsets of features while the root represents the set of all features. Principal Component Analysis is a well-established method, which is used to find the most significant attributes

in a data set [5]. However, principal component analysis cannot be extended to symbolic attributes.

Decision tree based classification methods like ID3 algorithm uses an information theoretic based approach to compute the significance of attributes and use them to construct a decision tree which checks for attribute values, in the order of decreasing significance of attributes, for prediction of class decision. The problem with this approach lies in the fact that each time the most significant attribute is eliminated form the data set, the significance of the remaining attributes have to be recomputed with respect to the new scenario. Besides, the decision tree reflects the training data set accurately and does not necessarily perform well while used for prediction of new data. Hence decision trees are pruned to make them practically applicable. The full complexity of decision tree induction then turns out to be $O(mn \log n) + O(n (\log n)^2)$, where m is the number of attributes and n is the number of samples.

HCV [3] is a heuristic attribute-based induction algorithm that is based on the extension matrix approach. In this approach, the positive examples (PE) of a specific class in a given example set are divided into intersecting groups and then a set of strategies are adopted to find a heuristic conjunctive formula in each group which covers all of the group's positive examples and none of the negative examples (NE). The covering formula is represented in the form of variable-valued logic for PE against NE. This is done in low-order polynomial time. This algorithm being very efficient for prediction, we have used the time analysis of prediction results obtained by this algorithm as the basis for comparing our prediction results.

Soft computing paradigms like neural networks and genetic algorithms have also been successfully applied for classificatory knowledge extraction [6]. Genetic algorithms help in accumulating information about an unknown search space to bias future search into promising subspaces. However, genetic algorithms can be designed only with a substantial amount of domain knowledge.

3. DETERMINING SIGNIFICANCE OF ATTRIBUTES

Ours is a conditional probability based approach originating from the following observations. If an attribute is significant, then there is a strong possibility that on changing the value of the attribute for an element, the categorization of the element would also change. Alternatively, given that the class decision for two elements in the data set is different, the values of a significant attribute should also be different for these two. We therefore compute the significance of an attribute as a **two way** function of its association to the class decision. For each attribute A_i, we compute Æ (A_i), which captures the cumulative effect of all its possible values and their effects on class decisions. Æ(A_i) represents the overall **attribute-to-class association** for A_i. Next, we take note of how an attribute's values change with a change in the class decision. We capture this effect in the quantity Œ (A_i) for the attribute A_i. This represents the **class-to-attribute association** for every attribute. An attribute is really **significant** if both attribute-to-class association and class-to-attribute association for the attribute are high. While most of the methods like those using a decision tree or principal component analysis do use the first kind of association , the second type of association is not utilized by them.

3.1. COMPUTING Æ( ) FOR ALL ATTRIBUTES

Let U be the collection of data elements and let A_1, A_2, ... A_g be the attributes which describe the elements of this data set. We assume that the elements of U are members of m different classes denoted by natural numbers 1, 2, ..., m. Let J represent the set of all classes i.e. J={1,2,3, ... m}.

To compute the overall association of A_i to the different classes, let us assume that it can take k different symbolic values. We use the notation A_i^r to denote the r^{th} attribute value of A_i. The notation $A_i^{\sim r}$ is used to denote a value of A_i which is not equal to A_i^r. This is a short hand notation for all values not equal to A_i^r, and can actually take (k-1) different values.

We introduce a set of notations which we will use hereafter.

- w is a proper subset of J

P_i^r (w) denotes the probability that elements of U with i^{th} attribute value equal to A_i^r belong to classes contained in w. This can be computed from U using frequency counts.

$P_i^{\sim r}$ (~w) denotes the probability that elements not having the i^{th} attribute value equal to A_i^r (i.e. elements with i^{th} attribute value equal to anything other than A_i^r) do not belong to classes contained in w. This can also be computed from U using frequency counts.

Our first observation is that if an attribute value is very significant, then both P_i^r (w) and $P_i^{\sim r}$(~w) are high. This implies that objects with i^{th} attribute(A_i) value equal to A_i^r and those with $A_i^{\sim r}$ ***classify to different groups of complementary classes***.

We term the quantity P_i^r (w) + $P_i^{\sim r}$ (~w) as the **separating power of A_i^r with respect to** w. This quantity reaches a maximum, when both the terms individually reach their maxima. Since there are $(2^m - 1)$ possible values of w, we associate with each value A_i^r , the subset w_i^r, which yields the maximum value for the summation (P_i^r(w) + $P_i^{\sim r}$ (~w)). Since w_i^r yields the maximum value for the above quantity, this subset can be said to have the strongest association to the value A_i^r .

Definition 3.1.1: The subset w = w_i^r that maximizes the term (P_i^r(w) + $P_i^{\sim r}$ (~w)) is termed as the **support set** for the value A_i^r.

Definition 3.1.2: The quantity (P_i^r(w_i^r) +$P_i^{\sim r}$(~w_i^r)) is defined as the **discriminating power** of an attribute value A_i^r. We use the symbol ϑ_i^r to denote the discriminating power of an attribute value A_i^r.

An attribute will be significant if all it's values have high discriminating power.

Definition 3.1.3: The **attribute-to-class association** of an attribute A_i, denoted by **Æ(A_i)**, is a function of the mean of the discriminating powers of all possible values of an attribute A_i. We restrict its value between 0.0 and 1.0.

To compute $P_i^r(w)$ for any w, we note that

$$P_i^r(w) = \sum_{n \in w} P(n \mid A_i^r),$$

since an element can belong to exactly one class contained in w. $P(n \mid A_i^r)$ denotes the conditional probability that an element belongs to class n given that the value for its i^{th} attribute is A_i^r. This can be directly computed for any given preclassified data set.

Now, for any i, for any r and for any w, we have $0 \leq P_i^r(w) \leq 1$, and $0 \leq P_i^{\sim r}(\sim w) \leq 1$. The value of 0 is obtained when none of the elements of U with i^{th} attribute value A_i^r belong to any class contained in w. The value of 1 indicates that w contains all the classes that elements of U with i^{th} attribute value A_i^r, belong to. Hence, $0 <= (P_i^r(w) + P_i^{\sim r}(\sim w)) <= 2$.

We use a linear incremental approach to find the support set w_i^r for each attribute value A_i^r of attribute A_i. Let **n** denote the total number of elements in the data set. For each class $t \in J$, let ***N(t)*** denote the number of elements belonging to class ***t***. Let $\mathbf{T}_i^r$ denote the total number of elements in the data set having A_i^r as the value for A_i. Let $\mathbf{M}_i^r(t)$ denote the number of elements that belong to class ***t*** and have attribute value A_i^r for A_i. Then $P(t/A_i^r)$ is given by $\mathbf{M}_i^r(t)/\mathbf{T}_i^r$.

To find the subset w_i^r, starting with an empty w_i^r, we add ***t*** to w_i^r, if $\mathbf{P(t/A_i^r)}$ is greater than $\mathbf{P(t/A_i^{\sim r})}$. It can be theoretically proved that this approach indeed gives the maximizing subset which is the support set for the attribute value A_i^r.

The linear approach to finding the support set proceeds as follows. If the conditional probability of an element belonging to class ***t*** is higher with a given attribute value $\mathbf{A_i^r}$ than with the values $\mathbf{A_i^{\sim r}}$, then ***t*** will be included in w_i^r, while it will be included in $(\sim w_i^r)$, if it is the other way round. Obviously, no class can belong to both w_i^r and $(\sim w_i^r)$. Thus, when all the classes ***t*** are taken care of, w_i^r accumulates those classes which occur more frequently in association to the value $\mathbf{A_i^r}$ for $\mathbf{A_i}$, while $(\sim w_i^r)$ accumulates those classes which occur more frequently in association with $\mathbf{A_i^{\sim r}}$.

Now, to compute $P(t/A_i^{\sim r})$, we need the proportion of elements which belong to class ***t*** but does not have attribute value A_i^r for A_i, out of all the elements of the data set. The quantity $(N(t) - \mathbf{M}_i^r(t))$ denotes the number of elements which belong to class ***t*** but does not have attribute value A_i^r for A_i. The total number of elements in the data set which does not have the attribute value A_i^r for A_i is given by $(\mathbf{n} - \mathbf{T}_i^r)$. Thus the required conditional probability $P(t/A_i^{\sim r})$ is given by

$$P(t/A_i^{\sim r}) = (N(t) - \mathbf{M}_i^r(t)) / (\mathbf{n} - \mathbf{T}_i^r).$$

The entire data set has to be scanned only once to compute all the conditional probabilities that are required. Thus the computation of the support set is linear in terms of the number of training samples.

We will now show that the value of $(P_i^r(w_i^r) + P_i^{\sim r}(\sim w_i^r))$ will lie between 1.0 and 2.0. The maximum value of $(P_i^r(w_i^r) + P_i^{\sim r}(\sim w_i^r))$ is obviously 2.0. Let w denote any proper subset of J and w′ its complement. Let the value of $P_i^r(w)$ be **x**. Then $P_i^r(w')$ has value $(1 - \mathbf{x})$. Let $P_i^{\sim r}(w)$ have value **y.** Then $P_i^{\sim r}(w')$ has value $(1-\mathbf{y})$. Now, the separating power of A_i^r with respect to w, is denoted by $(P_i^r(w) + P_i^{\sim r}(w'))$ and is given by $(\mathbf{x} + 1 - \mathbf{y})$, while the separating power of A_i^r with respect to w′, is given by $(P_i^r(w') + P_i^{\sim r}(w))$ which is equal to $(1-\mathbf{x} + \mathbf{y})$.

- If $\mathbf{x} > \mathbf{y}$, then $(\mathbf{x}+1-\mathbf{y}) > 1.0$ which implies that the separating power of A_i^r with respect to w will be greater than 1.0.
- If $\mathbf{y} > \mathbf{x}$, then $(1-\mathbf{x}+\mathbf{y}) > 1.0$ which implies that the separating power of A_i^r with respect to w′ will be greater than 1.0.
- If $\mathbf{x}=\mathbf{y}$, then both the above quantities are equal to 1.0.

Thus, it is always possible to find a subset of J with respect to which the separating power of A_i^r is at least 1.0. Hence the separating power of A_i^r will lie between 1.0 and 2.0.

We have already introduced the term $Æ(A_i)$ in definition 3.1.2, which denotes the attribute-to-class association for attribute A_i. We now define $Æ(A_i)$, for A_i with k different attribute values as follows:

$$Æ(A_i) = (1/k * \sum_{r=1,2---k} \vartheta_i^r) - 1.0.$$

The subtraction of 1.0 in the above formula ensures that the value $Æ(A_i)$ lies between 0 and 1.0.

The following implementation realizes the steps discussed earlier. The function **max_sep_probability()** finds the maximizing set w_i^r for a particular attribute value $\mathbf{A_i^r}$. The algorithm **A** is used to compute $Æ(A_i)$.

Function *max_sep_probability* **($\mathbf{A_i^r}$, D)**

begin

Input – D – the data set and A_i^r - a specific attribute value

Output - w_i^r and $\vartheta_i^r = P_i^r(w_i^r) + P_i^r(\sim w_i^r)$

m - number of classes in D

/* $P(t/A_i^r)$ – conditional probability of class t given A_i^r, as calculated form D

$P(t/A_i^{\sim r})$ – conditional probability of class t given $A_i^{\sim r}$, as calculated from D */

$\vartheta_i^r = 0.0$;

$w_i^r = \phi$;

for(t=1;t<=m;t++)

{ if ($P(t/A_i^r) > P(t/A_i^{\sim r})$) /* A greater proportion of elements with value A_i^r belong to class t than those with values $A_i^{\sim r}$ so $t \in w_i^r$ */

{ $w_i^r = w_i^r + \{t\}$;

$\vartheta_i^r = \vartheta_i^r + P(t/A_i^r)$;

}

else

{ /* $t \in \sim w_i^r$ */

$\vartheta_i^r = \vartheta_i^r + P(t/A_i^{\sim r})$;

}}

end;

Algorithm *Æ* for computing *Æ*(A_i)

begin

Input : D - the set of all pre-classified data elements described with attributes A_1 , A_2 , _ _ _ _ _ _ _ _ _ , A_g where A_1 , A_2 , _ _ _ _ _ _ _ _ _ , A_g are symbolic attributes.

Output : Æ(A_1), Æ(A_2), … , Æ(A_g)

Step 1: For each attribute A_i repeat steps 2 to 5

Step 2 : Æ(A_i) = 0.0 (initialize)

Step 3 : For each of the r values A_i^r of A_i,

Æ(A_i) = Æ(A_i) + ***max_sep_probability*** (A_i^r , D)

Step 4: Æ(A_i) = Æ(A_i) / ξrξ; /* ξrξ denotes the number of values A_i takes */

Step 5: Æ(A_i) = Æ(A_i) – 1.0;

end;

3.2. COMPUTING Œ( ) FOR ALL ATTRIBUTES

Œ(A_i) finds the association between the attribute A_i and various class decisions, by observing how a change in the class decision causes a change in the attribute's value It is expected that objects belonging to different classes will tend to have different values for a really significant attribute. . The computation is very similar to the earlier one.

Let V be a subset of attribute values of A_i. As in section 3.1, we introduce two quantities $P_i^j(V)$ and $P_i^{\sim j}(\sim V)$.

- $P_i^j(V)$ denotes the probability that elements belonging to class j, have those attribute values of A_i which are contained in the set V.

- $P_i^{\sim j}(\sim V)$ denotes the probability that elements not belonging to class j, have those attribute values of A_i which are not contained in the set V.

Obviously, if the attribute A_i and class j have a high degree of association then both the above probabilities will be high.

Now, for each class C_j, we find the subset V_i^j comprised of values of A_i, that maximizes the quantity $(P_i^j(V) + P_i^{\sim j}(\sim V))$. V_i^j contains those attribute values which occur predominantly in association to class Cj. As noted earlier, when both $(P_i^j(V_i^j)$ and $P_i^{\sim j}(\sim V_i^j)$ are high, it indicates that the values contained in V_i^j have a high association factor with C_j and the remaining classes have high association with other values of attribute A_i.

Definition 3.2.1: The quantity $(P_i^j(V_i^j) + P_i^{\sim j}(\sim V_i^j))$ is denoted by Λ_i^j and is called the **separability** of the attribute values of A_i with respect to class C_j.

We now define the quantity called Œ(A_i), which denotes the class-to-attribute association for the attribute A_i. We define Œ(A_i) to be the mean of the **separability** of its values. Further, we restrict Œ(A_i) to lie between 0.0 and 1.0 and hence we define it as follows:

$$Œ(A_i) = (1/m * (\sum_{j=1,2---m} \Lambda_i^j)) - 1.0,$$ where the database D has elements of m different classes.

Functions to find the separability and class-to-attribute association are very similar to the ones described above for discriminating power and attribute-to-class associations and hence we skip them.

Definition 3.2.2: The significance of an attribute A_i is computed as the average of ***Æ*(A_i)** and **Œ (A_i)** and is denoted by **σ(A_i).**

3.3. ATTRIBUTES AND THEIR SUPPORT SETS - PHYSICAL SIGNIFICANCE

In this section we will illustrate the practical use of ranking of attributes and also explain the physical significance of the support sets associated with the significant attributes. The above method of significance computation, when applied on Fisher's IRIS data set showed that the attributes when ordered according to their significance are petal length, petal width, sepal length and sepal width respectively. This is in accordance with the results reported in literature [5].

A more practical database that we experimented with is a heart patients data base obtained from the site www.niaad.liacc.up.pt/statlog/datasets.html This set contained samples classified into two categories – patients with and without heart disease. The set contained 13 attributes. Table 3.3.1 shows the ranking of the most significant attributes obtained by our method along with the attribute values in their respective support sets for the class of patients with heart disease. Thus it can be concluded that people with heart disease will have chest pain of type 3 and 4 mostly, indicating high and very high values for it. Similarly, low values for maximum heart rate achieved also indicate likelihood of heart disease.

We now present some interesting insights that we get from image segmentation data also obtained from the same site. This data set contains pixels classified into categories BRICKFACE, SKY, FOLIAGE, CEMENT, WINDOW, PATH and GRASS. Each pixel is described with 19 attributes, of which we find the seven most significant ones are ***intensity, rawred-mean, rawgreen-mean, rawblue-mean, value-mean, saturation-mean and hue-mean*** respectively. The other attributes convey positional

information only and is correctly identified as insignificant by our approach. On analysis of support sets for the significant attributes, we can extract the following classificatory knowledge for image segmentation:

- The class SKY can be always distinguished from all other classes due to high values of all the attributes ***intensity, rawred-mean, rawgreen-mean, rawblue-mean, value-mean.***
- Hue-mean can alone distinguish correctly between classes GRASS, BRICKFACE and WINDOW with high values for GRASS, low for BRICKFACE and medium for WINDOW.
- Classes CEMENT and PATH have substantial overlapping values for all the significant attributes.

Table 3.3.1 – Significant attributes and their support sets for the heart disease data

Rank	Attribute Name	Support set
1.	Tal	{fixed defect, reversible defect}
2.	Number of major blood vessels colored by fluoroscopy	{1,2,3}
3.	Chest pain type	{high, very high}
4.	Exercise induced angina	{Yes}
5.	Slope of peak exercise ST segment	{Medium, High}
6.	oldpeak = ST depression induced by exercise relative to rest	{2.06 – 6.2}
7.	maximum heart rate achieved	{71.0 – 136.0}
8.	Sex	1

3.4. COMPLEXITY OF PROPOSED ALGORITHM

The above algorithms to compute the **attribute-to-class** and **class-to-attribute** associations are linear in terms of the number of elements in data set. Given a database of **n** objects, the computation of conditional probabilities needs one scan of the database and take O(**n**) time. Computation of ϑ_i^r considers for each value of an attribute, the proportion of elements which have that value and belong to different classes. If an attribute A_i has ***k*** different values and there are m different classes, then this of order O(**n**+ **km**). For a total of ***g*** attributes, the total time taken to compute the **attribute-to-class** or **class-to-attribute** associations for all attributes is O(***gn*** + ***gkm***). Normally, in any real data set, ***n*** >> ***km***, so the complexity of the proposed algorithms are effectively linear in terms of ***n*** and the order ≈ O(***gn***).

4. PREDICTION USING SIGNIFICANCE OF ATTRIBUTES VALUES

In this section we propose a prediction methodology which uses the **discriminating** and **separability** powers of the significant attribute values along with their support sets, to predict the class of a new data element.

If the given attribute value A_i^r of the data element belongs to the support set of a class, we compute the likelihood of the class as follows:

Likelihood (t) = $(\Lambda_i^j - 1.0)^* P(A_i^r / t)$, if $A_i^r \in v_i^j$,

Likelihood (t) = 0.0 otherwise.

The total likelihood of each class is then given by the total combined contribution of all the significant attribute values. This computation is also of the order of O(***g´m***), where ***g´*** is the number of significant attributes chosen. The class that receives the maximum total contribution is predicted as the actual class of the data element.

5. PERFORMANCE EVALUATION

We validated the entire approach with several standard databases. The results reported in this section were obtained with a 10-fold cross validation over each data set. All the databases were obtained from the site www.niaad.liacc.up.pt/statlog/datasets.html. Each database was randomly divided into training and testing sets by a 70:30 split of the instances. The training set was used to compute the significance of the attributes. The prediction results obtained with the significant attributes only were validated against the original classes of the test instances and the error percentages are also reported.

Table 5.1 shows that the classification results are comparable with those algorithms which use all the attributes, though our algorithm uses fewer attributes and hence use less time. The reduction in the number of attributes is very significant for the Australian credit card database. The table shows that our algorithm uses only 2 out of all the 16 attributes and gives better classification results. Substantial computational reduction is also obtained for the diabetes patient data set.

The data sets picked up were of varied nature and there was no assumption about the shapes of class volumes in the training data set. It is observed that the classification results for test data are not so good for the Diabetes data set and the Hayes-Roth data set. We observed that for both of the above data sets, none of the features had high class-to-attribute or attribute-to-class correlation. The maximally significant attribute in case of the Diabetes data set was 0.37 and only three attributes had values greater than 0.22. While for the Australian Credit Card data base, the maximum significance value was obtained as 0.71.

One of the implementation issues for this algorithm would be to decide on a threshold value for discarding the non-significant attributes. Our empirical observation is that if the most significant attribute has a significance value less than 0.8 then all attributes which have their significance values within 60% of the most significant value, are to be selected. While, if the most significant value is greater than

0.8, then all attributes which have their significance values within 80% of the most significant value, are to be selected. However, this is just an empirical observation and we are yet to provide a theory for the selection of the appropriate threshold value.

6. CONCLUSIONS

In this paper, we have presented an efficient methodology to compute significance of attributes. We have also shown how the computation of significance of attributes can be linked to the problem of prediction for classification. Results show that the performance of this algorithm is comparable to some of the well-known algorithms though we use fewer attributes. The complexity of computing the significance of attributes is linear in number of training samples. In future, we plan to exploit the support sets for classes to generate classificatory knowledge which can be encoded as classification rules with degrees of accuracy associated to them. On applying the same approach for unsupervized learning or clustering of data sets, we have obtained encouraging results.

REFERENCES

[1] Davis L. [ed.], *Handbook of Genetic Algorithms* ,Van Nostrand Reinhold, 1991

[2] Xindong Wu, D. Urpani, *"Induction By Attribute Elimination", IEEE Transaction on Knowledge and Data Engineering,* vol. 11, No. 5, pp 805-812, Sept./Oct. 1999

[3] J. Hong, "*AE1: An Extension Matrix Approximate Method for the General Covering Problem*", Int'l Journal if Computer and Information Sciences, Vol 14, No. 6, Pg. 421 – 437, 1985.

[4] Narendra P.M. and Fukunaga K., "*A Branch and Bound Algorithm for Feature Subset Selection*", IEEE Transactions on Computers, vol c-26, no. 9, pp 917-922, 1977.

[5] Ian H. Witten and Eibe Frank, "Data Mining – Practical Machine Learning Tools and Techniques with Java Implementations", Morgan Kaufmann Publishers, San Francisco, California, 2000.

[6] Vafaie H. and Ibrahim F.G. Imam, "Feature Election Methods: Genetic Algorithms vs. greedy-like search", Proceedings of the International Conference on Fuzzy and Intelligent Control Systems, 1994. http://citeseer.nj.nec.com/vafaie94feature.html

Table 5.1: Prediction Results – a comparison with other algorithms

Database	**Total Number of attributes**	**Attributes used for prediction after selecting significant ones**	**Total prediction error with our proposed algorithm (In %)**	**Error With C4.5 (In %) using all attributes**	**algorithms (these use all attributes)**	
					Name of algorithm	**Error (In %)**
IRIS	4	2	**4.1**	**6.7**	CBA	**5.3**
Australian-Credit Card	14	2	**14.4**	**15.5**	Cal5	**13.1**
Diabetes	8	3	**20.7**	**27.0**	LogDisc	**22.3**
Hayes-Roth	4	3	**18.1**	**14.4**	HCV	**14.3**
Vote	16	2	**3.9**	**3.0**	HCV	**2.2**
Wine	13	9	**5.8**	**1.9**	HCV	**9.6**