A Parameter Free BBN Discriminant Function for Optimum Model Complexity versus Goodness of Data Fitting

M. Naeem*[1] and S. Asghar[2]

[1] Department of Computer Sciences, Faculty of Computing
Mohammad Ali Jinnah University
Islamabad Pakistan
*naeems.naeem@gmail.com
[2] COMSATS Institute of Information Technology
Islamabad Pakistan

ABSTRACT
Bayesian Belief Network (BBN) is an appealing classification model for learning causal and noncausal dependencies among a set of query variables. It is a challenging task to learning BBN structure from observational data because of pool of large number of candidate network structures. In this study, we have addressed the issue of goodness of data fitting versus model complexity. While doing so, we have proposed discriminant function which is non-parametric, free of implicit assumptions but delivering better classification accuracy in structure learning. The contribution in this study is twofold, first contribution (discriminant function) is in BBN structure learning and second contribution is for Decision Stump classifier. While designing the novel discriminant function, we analyzed the underlying relationship between the characteristics of data and accuracy of decision stump classifier. We introduced a meta characteristic measure AMfDS (herein known as Affinity Metric for Decision Stump) which is quite useful in prediction of classification accuracy of Decision Stump. AMfDS requires a single scan of the dataset.

Keywords: machine learning, Bayesian network, decision stump, K2, data characterization.

1. Introduction

Bayesian Belief Network (BBN) is a notable formalism in structure learning. It is based on joint probability distribution in which every question is submitted to the network in a probabilistic mode and the user can receive the answer with a certain confidence level. A BBN in short is composed of three components. $< G, \Phi, D >$ in which G is a Directed Acyclic Graph (DAG). As from the graph theory we know that each graph is composed of two elements and same is true for G such that $G = (V, E)$.The DAG technically reprents the quality of a model rendered by the structure learning procedure because it is comprised of all of the dependent and independent nodes. Infact the absence of certain arcs realize the conditional independence of the nodes. Moreover G posses a probability ρ which is a quantitative component of structure learning, an indicative of implicit degree of association between the random variables. The vertices in G usually denotes the random variables such that $X = (X_v)_{v \in V}$ and ρ represents the joint probability distribution of X with $\rho(X) = \prod_{v \in V} \rho(X_v \mid X_{pa(v)})$ where $\rho(X_v \mid X_{pa(v)})$ shows a conditional distribution and $pa(v)$ is the set of parents of v.

BBN has proven its usefullness in the diversified domains such as bioinformatics, natural language processing, robotics, forecasting and many more [1]–[3]. The popularity of BBN stems from its generatlity in formalis and amenable visualization because it enables the viewers to render any expert based modification. The process of structure learning is comprised of two major components excluding setting of external parametres. However both of these steps are layered in onion style structure. The first component is a traversing algorithm which takes input variables and formulates a structure in shape of a Directed Acyclic Grapg (DAG).This structure is handed over to the second component. The second component is a discriminant function which evaluates the goodness of the structure under inspection. The discriminant function in BBN captures the assumption during learning phase that every query variable is independent from the rest of the query variables, given the state of the class variable. The goodness of the structure is produced in form of a numeric value. The search algorithm caters for the comparison of last value and current value produced by the discriminant function and marks one of them as the latest posible structure. In a simple brute force searching mechanism, every candidate of BBN structure is passed to evaluate its discriminant function after which the BBN with highest discriminant function is chosen.

Although brute force provides a gold standard, however it is restricted to dataset with only a very small number of features as otherwise generating a score (output of discriminant function) for each possible candidate is NP hard with the increasing count of nodes in structures. A solution to tackle this NP-hard issue is to restrict the number of potential candidates for parent nodes and employing a heuristic searching algorithm such as greedy algorithm [4], [5]. In the greedy search algorithm, the search is started from a specific structure which initially takes the input variables in a predefined order. The obtained structure is analyzed by the discriminant function which results in adding, deleting or reversing the direction of the arc between two nodes. The ordering of the query nodes is characterized by the prior knowledge or by means of sophisticated techniques such as defined by [6].The search continues to the adjacent structure reaching to the maximum value of a score if this value is greater as compared to the current structure. This procedure, which is known as hill-climbing search halts when culminating to a local maxima. One way of escaping local maxima is to employ greedy search. While employing greedy search, random perturbation of the structure is the way through which local maximum can be avoid off. Apart from this approach, there are alternate approaches escaping of local maxima problem. These include simulated annealing introduced by [7], [8] and best-first search [4]. In other words, if one describes the procedure of K2 [9] in simple words then pursuit of an optimal structure is more or less equivalent to selecting the best set of parents for every variable but avoiding any circular dependency. This is the basic concept behind the K2 algorithm.

However, it is an interesting question whether classification accuracy or error rate is sufficient enough for introduction of a new data model learner. There are situations when large number of variables degrades the performance of the learnt network by increasing the model complexity. This issue has motivated us to investigate the behavior of model complexity for various discriminant functions.

This study discusses the issue of formally defining ‘effective number of parameters’ in a BBN learnt structure which is assumed to be provided by a sampling distribution and a prior distribution for the parameters. The problem of identifying the effective number of parameters takes place in the derivation of information criteria for model comparison; where notion of model comparison refers to trade off ‘goodness of data fitting’ towards ‘model complexity’.

We call our algorithm Non Parametric Factorized Likelihood Discriminant Function ($NPFLDF$). The proposed discriminant function is designed to increase the prediction accuracy of the BBN model by integrating mutual information of query variable and class variable. We performed a large-scale comparison with other similar discrattempt functions on 39 standard benchma UCI datasets. The experimental results on a large number of UCI datasets published on the main web site of Weka platform show that $NPFLDF$ significantly outperforms its peer discriminant function in model complexity while showing almost same or better classification accuracy.

2. Model Complexity

The BBN model complexity has been conceptualized in many ways. One simple concept is the frequentist approach of dependent and independent query variables. Another approach is related to the joint distribution of the observed query variables and the random parameters (varies from one discriminant function to another). However we argued for the second approach as in case if the distinct states are few or features are binary in nature then frequentist approach becomes the lonely dominant criteria for explaining the model complexity. Keeping in view of this assumption we shall introduce the mathematical formulation for model complexity along this line.

Lemma 1. The maximum links in a directed acycle graph can be termed as the model complexity. The problem of model complexity is a frequentistic solution.

Proof. Let N be the total number of nodes in a DAG excluding class attribute, then problem of finding the maximum possible nodes is equivalent to the 'How to sum the integers from 1 to N' problem and can be expressed as.

$$\Theta = \frac{N(N+1)}{2} \tag{1}$$

Let P be the constraint of maximum links a node can have in terms of its parent nodes. We can divide all of the links in two sets, one set Θ^{cn} belongs to the arcs realized by the nodes with constraint and the second set Θ^{ot} represents all other arcs. We express it as:

$$\Theta = \Theta^{cn} + \Theta^{ot} \tag{2}$$

Certainly, if the constraint is in its first value (level); that is each node can have maximum one and only one arc then the BBN model will be a simple BBN structure. A higher value of constraint will lead towards more arcs in the second set. Let P denotes the constraint such that each node can have maximum P nodes as its parent nodes with whom the node is independent of, then the second set Θ^{cn} can further be divided into two sub sets $\Theta^{cn}_{\geq P}$ and $\Theta^{cn}_{<P}$ such that:

$$\Theta^{cn} = \Theta^{cn}_{\geq P} + \Theta^{cn}_{<P} \tag{3}$$

The count of the maximum links for the first subset $\Theta^{cn}_{\geq P}$ can be defined as:

$$\Theta^{cn}_{<P} = P(N-P+1) \tag{4}$$

The second subset comprised of the links in which first node can have maximum one parent node, second node can have maximum two parent nodes, third node can have maximum three parent node. If we continue this pattern then last node can have maximum P-1 parent nodes such that.

$$\Theta^{cn}_{>P} = \frac{P(P-1)}{2} \tag{5}$$

Notice that if value of P is one, then the above subset is an empty subset. Now based on the above last three equations, we can re write the equation of set of links with constraint such as:

$$\Theta^{cn} = P(N-P+1) + \frac{P(P-1)}{2} \tag{6}$$

From the above equation, we can derive a simple mathematical formula such as:

$$\Theta^{cn} = 1 + PN - \frac{P^2+P}{2} \tag{7}$$

Where P is the constraint on count of máximum parents, a node can have and N is the total number of non class features or query variables. The equation 5 is a frequentistic representation of model complexity. At this point it is quite easy to calaculate the model complexity in percentage as below.

$$\Psi = 100 \times \frac{\Theta^{cn}}{\Theta} \tag{8}$$

The above equation is quite useful for calaculating the numeric value of the model complexity in this study. The detailed result will be discussed in empirical validation section.

3. Related work

There are two essential properties associated with any discriminant function to optimize the structural learning [10]. The first property is the ability of any discriminant function to balance the accuracy of a structure keeping in view of the structure complexity. The second property is computational tractability of any discriminant function (metric). Bayes [9], BDeu [11], AIC [12], Entropy [13] and MDL [14], [15] and fCLL [16] have been reported to satisfy these characteristics. Among these discriminant functions, AIC, BDeu and MDL are based on Log Likelihood (LL) as given below:

$$LL(G \mid D) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} n_{ijk} \log \left(\frac{n_{ijk}}{n_{ij}} \right) \tag{9}$$

Where G denotes directed acyclic graph given dataset D .Other three counters include n , q_i and r_i indicate number of cases, number of distinct states of a query variable and number of distinct states of parent variable of a ith query variable. The log likelihood tends to promote its value as the number of features increases. The phenomenon occurs because addition of every arc is prone to pay contribution in the resultant log likelihood of final structure. This process can be controlled somewhat by means of introduction of penalty factor or otherwise restricting the number of parents for every node in the graph. The mathematical detail of the discriminant functions in this study are as follow.

3.1 AIC

Akaike Information Criterion (AIC) originally defined by Akaike [12] is defined mathematically:

$$AIC = -2 \times \ln(LL) + 2 \times K \tag{10}$$

In the equation 10, K denotes the number of parameters in the given model. However, [17] decompose AIC into a discriminant function which can be used in BBN. AIC is established on the asymptotic behavior of learnt models and quite suitable for large datasets. Its mathematical equation has been transformed into.

$$AIC(B \mid T) = LL(B \mid T) - \mid B \mid \tag{11}$$

Where |B| is length of network, number of parameters for each variable.

3.2 Bayes

We earlier mentioned that Cooper and Herskovits introduced an algorithm K2 in which greedy search was employed while a discriminant function of Bayes was used [9]. It was described that the structure with highest value of Bayes metric was considered the best representative of the underlying dataset. It motivates us to describe Bayes metric formally expressing in mathematical notations.

Let there is a sequence of n instances such that $\overset{n}{z} = d_1 d_2 d_3 .....d_n$ the Bayes discriminant function of structure $g \in G$ can be formulated in form of the equation.

$$P_b(g, \overset{n}{z}) = P_b(g). \coprod_{j \in J} \left(\prod_{s \in S(j,g)} \frac{(\overset{j}{\alpha} - 1)! \cdot \prod_{\substack{j \\ q \in A}} n[a, s, j, g]!)}{(n[s.j.g] + \overset{j}{\alpha} - 1)!} \right) \tag{12}$$

Where P$_b$ (g) is the prior probability of full network $g \in G$ .The prior probability can be omitted in the computation. The notation $j \in J = \{1,....,N\}$ is the count of the variable of the network g, and $s \in S(j,g)$ is the counting of the set from all sets of values obtained from the parents of the jth node variable. The expansion of the denominator factor can be expressed mathematically as below.

$$n[q, s, j, g] = \sum_{i=1}^{n} I(\underset{i}{z} = q, \overset{j}{\underset{i}{\pi}} = S) \tag{13}$$

$$n[s, j, g] = \sum_{i=1}^{n} I(\overset{j}{\underset{i}{\pi}} = S) \tag{14}$$

Where $\overset{j}{\pi} = \overset{j}{\Pi}$, the function I(E) = 1 given E is true, and I(E)=0 if E is false. The K2 learning algorithm uses Bayes as its core function in its each iteration enumerating all potential candidate graphical structure. The outcome of this enumeration is an optimal learnt structure which is stored in g^* . This optimal structure possess highest value of $P_b(g, \overset{n}{z})$ such that $\forall g \in G - g_0$,

$$if \; P_b(g, \overset{n}{z}) > P_b(g^*, \overset{n}{z}), \; then \; g^* \leftarrow g .$$

The above equation is required to be decomposed simply into a computational model; otherwise this theoretical model requires a very large number of computations involving factorial. It means score value for a network g can be enumerated as the sum of scores for the individual query variables and the score for a variable is calculated based on that variable alone and its parents.

The approach in the “discriminant function inspired learning” performs a search through the space of potential structures. These include Bayes, BDeu, AIC, Entropy and MDL, all of which measures the fitness of each structure. The structure with the highest fitness score is finally chosen at the end of the search. It has been pointed out that Bayes often results in overly simplistic models requiring large populations in order to learn a model which holds the capability to capture all necessary dependencies [18]. On the other hand, BDeu tends to generate an overly complex network due to the existence of noises. Consequently, an additional parameter is added to specify the maximum order of interactions between nodes and to quit structure learning prematurely [19].As noted in another research that the choice of the upper bound given the network complexity strongly affects the performance of BOA. However, the proper bound value is not always available for black box optimization [20].

Nielsen and Jensen in 2009 discussed two important characteristics for discriminant function used in the belief network [10]. The first characteristic is the ability of any score to put the accuracy of a structure in-equilibrium in context of complexity of structure. The second characteristic is its computational tractability. Bayes has been reported to satisfy both of the above mentioned characteristics. Bayes denotes the measurement of how well the data can be fitted in the optimized model. The decomposition of Bayesian Information Criterion [21] can be preceded as below:

$$BIC(S \mid D) = \log_2 P(D \mid \hat{\theta}_S, S) - \frac{size(S)}{2}\log_2(N) \qquad (15)$$

Where $\hat{\theta}$ is an estimation of the maximum likelihood parameters given the underlying structure S. It was pointed out that in case of completion of the database, Bayesian Information Criterion [21] is reducible into problem of determination of frequency counting [10] as given below:

$$BIC(S \mid D) =$$
$$\sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk}\log_2\left(\frac{N_{ijk}}{N_{ij}}\right) - \frac{\log_2 N}{2}\sum_{i=1}^{n} q_i(r_i - 1) \qquad (16)$$

where N_{ijk} indicates the counts of dataset cases with node Xi in its k_{th} configuration and parents Π(Xi) in j_{th} configuration, q_i denotes the number of configurations over the parents for node Xi in space S and r_i indicates the states of node Xi.

3.3 BDeu

Another scoring measure which depends only on equivalent sample size N´ is Bayesian Dirichlet for likelihood-equivalence for uniform joint distribution (BDeu) introduced by [11]. Later on, researchers have provided and discussed its decomposition as below in mathematical form [16]:

$$BDeu(B,T) = \log(P(B)) +$$
$$\sum_{i=1}^{n}\sum_{j=1}^{q_i}\left(\log\left(\frac{\Gamma\left(\frac{N'}{q_i}\right)}{\Gamma\left(N_{ij} + \frac{N'}{q_i}\right)}\right) + \sum_{k=1}^{r_i}\log\left(\frac{\Gamma\left(N_{ijk} + \frac{N'}{r_i q_i}\right)}{\Gamma\left(\frac{N'}{r_i q_i}\right)}\right)\right) \qquad (17)$$

3.4 MDL

Minimum Description Length (MDL) introduced by [14] initially and then refined by [22], and [15]. It is mostly suitable to complex Bayesian network. We shall formally define it as below. Let sequence $\overset{n}{x} = d_1 d_2 d_3 .....d_n$ of n number of instances, the MDL of a network $g \in G$ can be enumerated as below.

$L(g,\overset{n}{x}) = H(g,\overset{n}{x}) + \frac{k(g)}{2}\cdot\log(n)$ Where the function k(g) represents the independent conditional probabilities in the network. $H(g,\overset{n}{x})$ is entropy of structure with respect to the variable $\overset{n}{x}$ which can be expanded into the following notation. $H(g,\overset{n}{z}) = \sum_{j \in J} H(j,g,\overset{n}{z})$ and $K(g) = \sum_{j \in J} K(j,g)$

Given the jth node variable, the value of MDL can be enumerated as below:

$L(j,g,\overset{n}{x}) = H(j,g,\overset{n}{x}) + \frac{k(j,g)}{2}\cdot\log(n)$ where $k(j,g)$ is the count of independent conditional probabilities of j^{th} variable. This value can be expressed in more detail as below.

$K(j,g) = (\overset{j}{a} - 1)\cdot \prod_{k \in \varphi^j} \overset{k}{\alpha}$ while $\varphi(j) \subseteq \{1,....j-1, j+1,..., N\}$

is a set given $\overset{j}{\Pi} = \{\overset{k}{X} : k \in \varphi^j\}$.

Given the jth node variable, the entropy can be expanded into the following expression.

$$H(j,g,\overset{n}{x}) = \sum_{s\in S(j,g)} \sum_{q\in A^j} \left(\begin{array}{l} -n[q,s,j,g]\cdot \\ \log \dfrac{n[q,s,j,g]}{n[s,j,g]} \end{array} \right) \tag{18}$$

$$n(s,j,g) = \sum_{i=1}^{n} \left[I(\overset{j}{\underset{i}{\pi}} = s) \right] \tag{19}$$

$$n(q,s,j,g) = \sum_{i=1}^{n} \left[I(\underset{i}{z} = q, \overset{j}{\underset{i}{\pi}} = s) \right] \tag{20}$$

where $\overset{j}{\pi} = \overset{j}{\Pi}$ indicates that $\overset{j}{X} = \overset{j}{x} \forall k \in \varphi^j$; the function I(E) yields a positive identity number when the predicate E is true and the function I(E) becomes false when I(E)=0.

MDL differs from AIC by the log N term which is a penalty term. As the penalty term is smaller than that of the MDL, so MDL favors relatively simple network as compared to AIC. The mathematical formulation is composed of explanation of Log Likelihood (LL) as given below:

$$LL(B\,|\,T) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log\left(\frac{N_{ijk}}{N_{ij}} \right) \tag{21}$$

The value of LL is used in obtaining the decomposition of MDL as below:

$$MDL(B\,|\,T) = LL(B\,|\,T) - (1/2)\log(N)\,|\,B\,| \tag{22}$$

|B| denotes the length of network which is achieved in terms of frequency calculation of a given feature's possible states and its parent's state combination with feature as following:

$$|\,B\,| = \sum_{i=1}^{n} (r_i - 1)\; q_i \tag{23}$$

3.5 fCLL

In BBN, several algorithms have been introduced to improve classification accuracy (or error rate) by weakening its conditional attribute independence assumption. Tree Augmented Naive Bayes (TAN) in terms of conditional log likelihood (CLL) is a notable example under this category while retaining simplicity and efficiency.

Another Likelihood based discriminant function was introduced known as factorized Conditional Log Likelihood (fCLL) [16] which was optimized particularly for TAN. Its mathematical detail is as below.

$$\hat{f}\,CLL(G\,|\,D) = (\alpha+\beta)\,\hat{L}L(B\,|\,D) - \beta\lambda \sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=0}^{1} n_{ijkc} \left(\log\left(\frac{n_{ijkc}}{n_{ij*c}} \right) - \log\left(\frac{n_{ijc}}{n_{ij*}} \right) \right) \tag{24}$$

Table 1 provides the brief summary of these notable discriminant functions. These scores formulate propositions for well-motivated model selection criteria in structure learning techniques.

The noteworthy issue by employing these well-established scores, however, is that they are prone to intractable optimization problems. It was argued that it is NP-hard to compute the optimal network for the Bayesian scores for all consistent scoring criteria [23]. AIC and BIC are usually applied under the hypothesis that regression orders k and l are identical. This assumption brings extra computation and also come up with an erroneous estimation with theoretical information measure in structured learning. Recently a research was conducted which shows the linear impact of improvement in model quality within the scope of exercising Bayes function score in K2 [9], [24]. However, it was arguable that there must be intelligent heuristics to sharply extrapolate the optimized size of the training data. We are of the view that exploiting various intelligent algorithms for tree and graph, an optimized solution can be achieved.

Discr. Function	Description
AIC [12]	Its penalty term is high. AIC tends to favor more complex networks than MDL. AIC's behavior was erratic when sample size is enlarged. AIC was observed to favors to over-fitting problem in most cases.
Entropy [13]	No penalty factor was involved. The joint entropy distribution always favors for over-fitting. Its behavior was also erratic like AIC.
Bayes [9]	The quantities of interest are governed by probability distribution. These probability distribution and prior information of observed data leads to reason and optimal decision on the goodness of model.
BDeu [11]	The density of the optimal network structure learned with BDeu is correlated with alpha; lower a value typically results in sparser networks and higher value results in dense network. The behavior of BDeu is very sensitive to alpha parameter.
MDL [14], [15]	The large penalty factor is good if gold standard network is thick network Its penalty factor was higher as compared to AIC. It gives better results as compared to AIC and Entropy discriminant function.
fCLL [16]	fCLL involves approximation of the conditional log-likelihood criterion. These approximations were formulated into Alpha, Beta and Gamma. However fCLL is mostly suitable (Restricted) towards Tree Augmented Network (TAN).

Table 1. Summary of discriminant functions.

4. Towards a novel discriminant function

We start with $T = D(F, C)$ as a sampling space of training dataset. The dataset D contains g number of query variables and h number of sample instances for training model. The parameter F denotes the number of query variables or features such that $F = \{f_1, f_2, f_3, ...f_b\}$ while the sample instances in training dataset can be represented as: $D = \{d_1, d_2, d_3, ...d_h\}$. Furthermore, the n number of target class concepts can be described as: $C = \{c_1, c_2, c_3, ...c_n\}$.

Exemplifying the individual data instance as: $d_x \in D$: obviously it can be decomposed into a vector of array V such that $Vx = \{vx_1, vx_2, vx_3, ...vx_b\}$, where vx_k is the value of vx related to the feature set. Given an instances of training dataset $T = D(F, C)$, the objective of learning technique is to induce a hypothesis $h_0 : F \rightarrow C \setminus T$ where F is the value domain of $f \in F$. After this brief mathematical terminology, we shall head towards inscribing the degree of relationship between two variables specifically in context of classification; this relationship must need to be described between a query variable and a class variable. Let the distinct state of the query variable are denoted as $f_1 = \{f_{11}, f_{12}, f_{13}, ...f_{1m}\}$ while the unique states of the class variable can be expressed as $C = \{c_1, c_2, c_3, ...c_n\}$. We already defined the value of h as the count of instances in the dataset. Now we denote a_{ij} as the joint probability between query variable f_1 and class variable C, then Affinity Measure (AM) can be mathematically expressed as below:

$$AM(f_l, C) = \sum_{i=1}^{m}\left[\left(\max_{i=1}^{m} \arg\left[\overset{n}{\underset{j=1}{a_{ij}}}\right]\right)\Big/ h\right] \tag{25}$$

The above equation can be generalized to any of two features where the second argument can be replaced by other query variable. Affinity Measure (AM) is a bounded valued metric, the upper bound and lower bound with specific conditions are expressed as below:

$$AM_{\min} \leftarrow 0 \therefore \left[\underset{i=1,j=1}{\overset{m,n}{[}} \overset{\Uparrow}{a_{ij}}] - \underset{i=1,j=1}{\overset{m,n}{[}} \overset{\Downarrow}{a_{ij}}]\right] \rightarrow 0 \tag{26}$$

$$AM_{\max} \leftarrow 1 \therefore \left[\underset{i=1,j=1}{\overset{m,n}{[}} \overset{\Uparrow}{a_{ij}}] - \underset{i=1,j=1}{\overset{m,n}{[}} \overset{\Downarrow}{a_{ij}}]\right] \rightarrow h \tag{27}$$

Where a_{ij} denotes the joint probability with variable 1 in its ith state and variable 2 in jth state. $\overset{\Downarrow}{a_{ij}}$ and $\overset{\Uparrow}{a_{ij}}$ represents the minimum and maximum joint probability among all of the possible states of two variables. $AM(f_i, c)$ denotes the bounded value of Affinity Measure which is explained by the states of feature variable with respect to the class. However, if we swap the position of feature and class variable then a new meaning is raised where class variables are explaining the value of features. In fact, such a notion also explains the child parent relationship between two features in a given graph or tree based classifier.

At this point we proceed for two different discriminant functions. if we normalize the Affinity Measure (from equation 2) by dividing the count of non class features then we get a discriminant feature which is useful for prediction in a famous weak classifier Decision Stump. Its mathematical equation is as below.

$$AMfDS = \frac{1}{b}\left[\sum_{i}^{g}\left[PM(f_i, c)\right]\right] \tag{28}$$

Decision Stump is one of the tree classifiers which are termed as weak classifiers. It was originally introduced by Iba et al., [25]. This falls under the breed of classifiers in which one level tree is used to classify instances by sorting them, while the sorting procedure is based on futuristic value. Each node in a decision stump dictates a query variable from an instance which is to be classified. Every branch of the tree holds the value of the corresponding node. Although decision stump is widely used classifier; yet it is assumed as a weak classifier. In this threshold oriented classification system, sample instances are classified beginning from the root node variable. The sorting is carried out on their feature values which a node can take on. If the selected feature is specifically informative, this classifier may yield better results, otherwise it may lead generating the most commonsensible baseline in the worst situation. The weak nature of the classifier lies in its inability to tackle the true discriminative information of the node. Although to cope up this limitation, the single node, multi-channel split decision criteria is introduced to accentuate the discriminative capability; nonetheless its results are still not as appealing as compared to its peer classifiers. some empirical results supporting the usefulness of the Affinity Measure for Decision Stump will shown in the result section.

Now we shall move towards the derivation of an optimized discriminant function for the BBN in such a way that it keeps the model complexity at lower level while delivering equal or better results as compared to its peer techniques. The discirminant functions in general are based on Log Likelihood (LL) drawn from the dataset given network structure G as indicated by the equation 29.

$$LL(G \mid D) = \sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk} \log\left(\frac{N_{ijk}}{N_{ij}}\right) \tag{29}$$

Where N_{ijk} indicates that ith feature is instantiated with kth state along with the jth state of qth parent of ith feature. It can be observed from this frequentist approach that addition of an arc to such network always leads to increases the value of LL. Keeping in view of it, several penality terms were introduced to adjust it. However fixing an optimized penality factor has been a research problem for the experts in data mining community. Motivated by this fact, we tailored the Affinity Measure in such a way that it is free of any implicit or explicit penality factor such that

$$\xi_{C,F} = \sum_{c}\left[\max\arg\left(\sum_{f} P(C_c, F_f)\right)\right] \qquad (30)$$

Let $\vartheta(F)$ denote the marginal probability of the feature. The potential shown in the above equation can be converted into conditional probability by placing the marginal probability as the denominator factor in the above equation such that

$$\lambda_{C,F} = \sum_{c}\left[\max\arg\left(\sum_{f} \frac{P(c,f)}{\vartheta(F)}\right)\right] \qquad (31)$$

While generalizing $NPFLDF$, we have n number of non-class feature variables and a single class variable within the dataset D. We can easily reduce this simple point estimation into a generalized maximum a posterior inference notation as below:

$$NPFLDF(D,G) = \sum_{i=1}^{n} \max\arg(X_i, Pa(X_i), C, D) \qquad (32)$$

A discriminant function is decomposable if its expression is convertible to a sum of local scores, where local score refer to a feature (query) variable in pursuit of drawing graph G. The simple calculation between two feature variable is shown in equation 29. An extended version of this equation can be expressed as $\sum_{j=1}^{qi}\sum_{k=1}^{r1} N_{ijk}$ Where i is feature iterator, j is parent iterator, k is feature state iterator and c is class iterator. If we include the factor of class variable, a minor change will be developed into $\sum_{j=1}^{qi}\sum_{k=1}^{r1} N_{ijck}$. Plugging this value into equation 32, we can express as

$$NPFLDF(D,G) = \sum_{c}\left[\frac{1}{|C_c|}\max\arg\left(\sum_{j=1}^{qi}\sum_{k=1}^{r1} N_{ijck}\right)\right] \qquad (33)$$

If the feature set is denoted by $F = \{f_1, f_2, f_3, ... f_n\}$ then ordering weight of any feature will be determined by weight factor shown in equation 34.

$$\omega_F = \lambda_{C,F} - \lambda_{C,F} \qquad (34)$$

The terms $\lambda_{C,F}$ and $\lambda_{C,F}$ play the role of existence restrictions. We shall consider both of them as existence restrictions such that $(F,C) \in \lambda_{C,F}$: the link F→ C explains the discriminant objective with respect to the class and $(F,C) \in \lambda_{F,C}$: the link F← C means the discriminant score with respect to the feature. In our earlier research, we highlighted the correct topological ordering between two features. This was shown by an earlier version of the proposed discriminant function in which we highlight that majority of the discriminant functions can't precisely capture the casual relationship between two variables in pursuit of true topology in numerous situations; this ultimately leads to the selection of potential neighbor and parents becoming unreasonable. However Integration to Segregation (I2S) is capable of rightly identify it in majority of the cases as compared to BIC, MDL, BDeu, Entropy and many more [26], [27]. Moreover, Naeem et al. [26], [27] described that a structure in which class node is placed at the top most may lead to higher predictive accuracies. This type of scheme was termed as ''selective BN augmented NBC'' [26], [27]. It means that the later score value must be eliminated from the first value which will result into a weighted score vector as shown in the equation 35.

$$\lambda_{F,C} = \sum_{f}\left[\max\arg\left(\sum_{c} \frac{P(f,c)}{\vartheta(C)}\right)\right] \qquad (35)$$

A function for simple descending order is applied to the weights achieved from the equation 35 which results into an ordered list of input variables.

$$\overleftarrow{F} = \{\overset{i}{\omega_f} \mid i = 1...n\} \qquad (36)$$

Plugging this ordered set into the equation 33 will give result in

$$NPFLDF(D,G,\overleftarrow{X}) =$$

$$\sum_{c}\left[\frac{1}{|C_c|}\max\arg\left(\sum_{j=1}^{qi}\sum_{k=1}^{r1}N_{ijck}\right)\right] \qquad (37)$$

the equation three gives us the discriminant function to be used in the BBN structure learning. In the next section we shall discuss about its performance comparison.

5. Empirical Validation of Proposed Discriminant Function

We first obtain thirty nine natural dataset from UCI [20]. These datasets were quite diversified in their specifications. The number of attributes, instances, classes were ranging from small to large value so that any possibility of biasness in the dataset in favor of the proposed metric can be avoided off. The detail is shown in the table 1. We in this experimental study select some of basic meta characteristic and then two enhanced meta characteristics and one of our proposed metrics. The simple meta characteristics include number of attributes, class count and size of cases which are also shown in the table 1. The advanced meta characteristics of dataset include Entropy, Mutual Information and our proposed measure (AMfDS) also technically falls in this category. Before we proceed for analysis, it is mandatory to pre process or transform the data, there are many transformations applicable to a variable before it is used as a dependent variable in a regression model.

DB ID	Dataset	Nodes	Max Links
1	Arrhythmia	279	1110
2	Audiology	69	270
3	Autos	25	94
4	balance-scale	4	10
5	breast-cancer	9	30
6	breast-w	9	30
7	bridges_version1	12	42
8	bridges_version2	12	42
9	Car	6	18
10	Colic	22	82
11	credit-a	15	54
12	credit-g	20	74
13	Dermatology	34	130
14	Diabetes	8	26
15	Flags	29	110
16	Glass	9	30
17	Haberman	3	6
18	heart-h	13	46
19	heart-statlog	13	46
20	Iris	4	10
21	kdd_synthetic_control	61	238
22	Labor	16	58
23	Letter	16	58
24	mfeat-fourier	76	298
25	mfeat-karhunen	64	250
26	mfeat-morphological	6	18
27	mfeat-pixel	240	954
28	molecular-biology_promoters	58	226
29	Mushroom	22	82
30	page-blocks	10	34
31	Pendigits	16	58
32	postoperative-patient-data	8	26
33	Segment	19	70
34	Sonar	60	234
35	Spect_test_train	22	82
36	Sponge	45	174
37	Trains	32	122
38	waveform-5000	40	154
39	Zoo	16	58

Table 2. Description of dataset used in this study [Parent count constratint (P)=4].

These transformations can not only restrict towards changing the variance but may incur alteration in the units of variance to be measured. These include deflation, logging, seasonal adjustment, differencing and many more.

However, the nature of data in our study require to adopt the normalization transformation of the accuracy measures and the specific characteristics for which analysis is required. Let x_i denotes the accuracy of ith dataset by any classifier then the normalized accuracy can be obtained by the equation as below:

$$y_i^n \leftarrow \frac{x_i^n}{m} \tag{38}$$

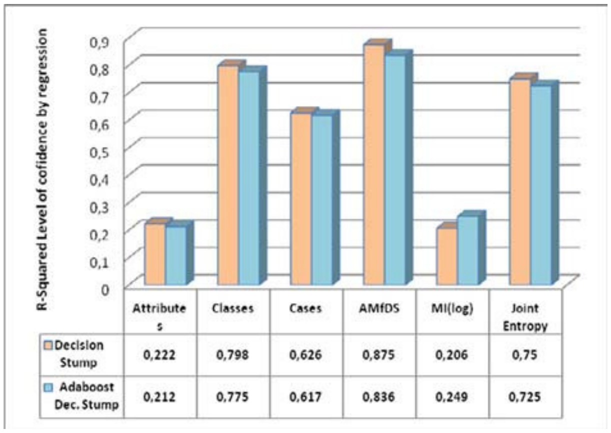
Where

$$m = \max\arg(x_i^n) \tag{39}$$

The next step is to obtain a pair wise list with sorting performed on y_i such that we denote the sorted list as $\overset{\leftarrow}{y_i}$. With the application of this normalization, a set of normalized characteristics was prepared which was used later on to generate a regression model. A linear regression model is quite useful in order to express a robust relationship between two random variables. The linear equation of regression model indicates the relationship between two variables in the model. Y is regressand or simply a response variable whereas X is regressor or simply an explanatory variable. The output regression line is an approximate acceptable estimation of the degree of relationship between variables. One important parameter in linear regression model is co efficient of determination also known as R-squared. The closer this value to 1, the better the fitting of regression line is represented. R-squared dictates the degree of approximation of the line passing through all of the observation.

Wolpert and Macready [28] stated in their 'No Free Lunch Theorem' that no machine learning algorithm is potent enough to be specified outperforming on the set of all natural problems. It clearly points out that every algorithm possesses its own realm of expertise albeit two or more techniques may share their realm in partial. Ali et al., [29] shows that classifiers C4.5, Neural Network and Support Vector Machine were found competitive enough based on the data characteristics measures.

The significance of R-squared is dictated by the fraction of variance explained by a data model but question arises what is the possible relevant variance requiring a suitable explanation. Unfortunately it is not easy to fix a good value of R-squared as in most of the cases, it is far off to get a value of 1.0. In general it is assumed that a value greater than 0.5 indicates the noticeable worthiness of the model. However, still it is a matter of choice as in case of comparison between various models (such as in ours) a more higher value of R-squared counts.



	Attributes	Classes	Cases	AMfDS	MI(log)	Joint Entropy
🟧 Decision Stump	0,222	0,798	0,626	0,875	0,206	0,75
🟦 Adaboost Dec. Stump	0,212	0,775	0,617	0,836	0,249	0,725

Figure 1. Polynomial regression analysis of Decision Support classifier accuracy using simple and information theoretic data characteristics and AMfDS.

Figure 1 shows the R-Squared values of various linear regression model using specific meta classifiers. Two highest values plotted against AMfDS show the substantial model fitting. These curve fitting were tested with many flavors of regression models ranging from 1st degree to 10th degree order polynomial, 1st order logarithm to 5th order logarithm, polynomial inverse and a lot of special cases data fitting model provided in the commercially available tool DataFit [30]. We noticed that the best curve fitting was found for tenth order degree polynomial regression model.

Decision Stump and Adaboost Decesion Stump both can be explained by the number of classes, joint entropy and proposed metric AMfDS. One the other hand, the data characteristics such as number of attributes and Mutual Information (natural logarithm) can't explain it properly. Moreover, the number of cases (instances) can also explain the accuracy of these classifiers for all of the natural dataset used in this study. We calculated the average joint entropy of each attribute with class attribute; hence the final score is indicative of a score of entropy towards the class variable. The root cause lies in the splitting criterion which is characterized by entropy inspired measure.

Mutual Information (MI) which is an information theoretic measure. MI is basically an intersection of entropy of two features. MI strictly defines the mixed relationship of two variables by which both of them are bound to each other. However we noticed that it did not show up better as compared to other meta characteristics.

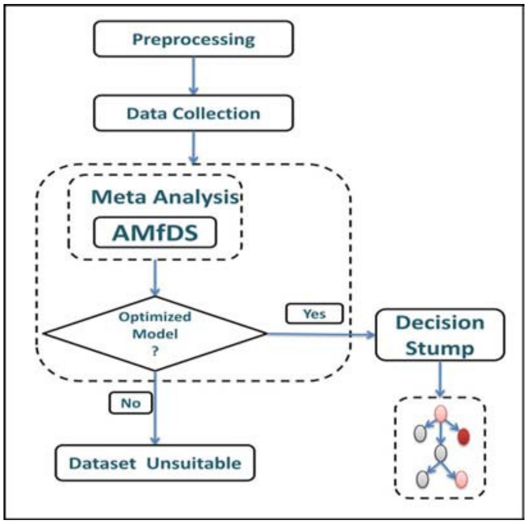


Figure 2. Framework utilizing AMfDS.

Moreover, it is noticeable that AMfDS metric incur significant R-squared value in case of Decision Stump (DS) and its implementation with Ada Boost DS. The R-squared value was 0.875 and 0.836 respectively. It clearly indicates that the classification accuracy of both of these classifiers can be greatly predicted a prior by using AMfDS. It is noticeable that no other meta feature deliver this level of R-squared confidence of determination. The regression model parameters were obtained with 99% confidence interval. The tenth degree regression model defined by AMfDS is shown by the equation six as below:

$$\begin{aligned} \mathrm{Y} &= 96271.29\,X^{10} - 542855.9X^{9} + 1332784.2X^{8} \\ &\text{-}1871297.69\,X^{7} + 1659092.14X^{6} - 967218.97X^{5} \\ &+ 373928.28X^{4} - 94164.7X^{3} + 14683.89X^{2} \\ &\text{-}1268.53X + 45.76 \end{aligned} \tag{40}$$

The proposed metric is useless unless it is utilized in a framework. The figure 2 is a typical framework of machine learning in which AMfDS has been plugged. The first two components are preliminary and essential pre requisite for making any data suitable for a machine learner. Once the data is fully prepared, the meta analysis is an essential and novel component where AMfDS will yield an approximated value for the classification accuracy of decision stump. Once the decision is obtained, the end user can find it easily whether this dataset is suitable for this classifier. The regression model gives the accuracy of 87.5% within the confidence interval of 99%. Table 3 is indicating the result we obtained frm our proposed discriminant function $NPFLDF$. Table 3 indicates that the introduced discriminant function exhibits better in numerous cases. The average accuracy of the proposed function is also better than the other discriminant functions. The last row of the table 3 ('w' stands for win and 'n' stands for neutral) points out that $NPFLDF$ delivers best result for eleven dataset and for five datasets it shares the best result status with other peer functions. The performance of other functions is quite inferior to that of the proposed function. However if we analyse the results in term of the average accuracy of the functions over all of the datasets then a cynical view on these results indicate that the average accuracy for $NPFLDF$, AIC, MDL, Bayes and BDeu is almost close to each other but what is the point of difference? The difference is in the model size. The model size in the table 3 ranges from 1 to 100.

DB	NPFLDF		Bayes		AIC		BDeu		MDL		Entropy		Fcll	
ID	Acc	Den	Acc	Den	Acc	Den	Acc	Den	Acc	Den	Acc	Den	Acc	Den
1	70.80	25.50	70.80	44.41	70.80	28.65	69.91	44.86	71.02	26.40	66.15	58.02	71.68	45.05
2	78.76	20.00	76.11	42.22	76.11	25.56	73.01	45.19	76.11	25.56	75.22	83.70	75.66	51.85
3	80.98	18.09	80.49	61.70	74.63	40.43	83.90	69.15	74.63	32.98	79.02	89.36	76.10	55.32
4	72.64	50.00	70.88	100.00	70.88	100.00	71.84	40.00	72.00	60.00	70.88	100.00	74.24	70.00
5	70.98	40.00	70.28	50.00	68.53	53.33	69.58	43.33	70.63	40.00	62.94	100.00	73.08	56.67
6	96.71	40.00	96.71	53.33	96.85	56.67	96.57	46.67	97.00	46.67	96.42	100.00	96.42	56.67
7	65.71	16.67	65.71	47.62	65.71	30.95	65.71	42.86	65.71	28.57	41.90	52.38	60.95	52.38
8	63.81	16.67	64.76	45.24	62.86	30.95	64.76	40.48	60.95	28.57	41.90	54.76	61.90	52.38
9	91.61	38.89	90.80	44.44	92.65	50.00	90.80	44.44	85.71	33.33	88.25	100.00	88.72	61.11
10	82.34	36.59	81.79	56.10	82.07	48.78	82.07	37.80	81.52	35.37	72.01	85.37	82.07	58.54
11	85.94	37.04	85.07	53.70	85.51	62.96	85.80	46.30	86.23	42.59	81.16	96.30	85.07	53.70
12	74.60	33.78	74.50	45.95	74.70	52.70	75.00	35.14	75.30	35.14	69.60	79.73	74.70	62.16
13	97.54	23.08	98.09	36.92	97.54	26.15	97.54	26.92	97.54	26.15	89.62	100.00	96.45	51.54
14	74.74	30.77	74.48	46.15	74.09	50.00	75.13	38.46	74.87	38.46	72.53	76.92	73.70	61.54
15	61.34	20.91	57.22	53.64	61.34	36.36	57.22	42.73	62.37	27.27	35.57	50.00	64.95	51.82
16	71.03	30.00	72.43	50.00	70.56	36.67	69.16	40.00	70.56	30.00	72.43	80.00	72.43	60.00
17	73.86	50.00	72.55	50.00	72.55	50.00	72.55	50.00	72.55	50.00	73.86	66.67	73.86	83.33
18	84.69	21.74	85.03	36.96	84.35	34.78	85.03	32.61	84.01	30.43	81.97	73.91	84.35	65.22
19	81.48	21.74	81.85	47.83	82.59	52.17	80.74	36.96	80.37	39.13	81.85	73.91	81.11	65.22
20	92.67	40.00	92.67	50.00	92.67	40.00	92.67	60.00	92.67	40.00	90.67	100.00	93.33	70.00
21	98.67	16.81	98.83	44.96	98.00	30.67	97.67	30.25	97.17	25.63	16.67	50.42	96.83	50.42
22	94.74	25.86	92.98	62.07	91.23	51.72	89.47	50.00	91.23	36.21	87.72	89.66	91.23	55.17
23	84.53	29.31	86.48	62.07	83.97	51.72	81.71	46.55	76.62	34.48	87.51	94.83	82.33	53.45
24	79.85	20.13	80.25	72.82	80.15	67.11	77.80	34.90	78.05	37.25	76.75	100.00	77.95	50.67
25	92.75	16.80	92.95	47.20	93.15	53.60	92.10	25.60	92.05	26.40	85.70	100.00	91.75	50.80
26	70.20	33.33	69.85	66.67	67.95	50.00	68.85	66.67	68.20	38.89	68.85	94.44	69.50	61.11
27	94.75	20.96	94.55	60.48	94.00	50.21	93.55	50.42	93.40	26.21	92.00	100.00	92.85	67.61
28	80.19	22.12	82.08	88.94	82.08	40.27	95.28	47.79	89.62	25.66	47.17	50.88	81.13	50.44
29	99.74	51.22	100.00	93.90	100.00	90.24	100.00	96.34	99.99	78.05	100.00	93.90	99.22	52.44
30	95.47	52.94	96.46	88.24	95.30	55.88	96.33	76.47	95.63	50.00	96.62	100.00	94.24	55.88
31	94.78	44.83	96.56	65.52	95.26	53.45	95.14	51.72	93.25	46.55	95.60	100.00	94.78	53.45
32	64.44	30.77	64.44	42.31	65.56	30.77	64.44	30.77	64.44	30.77	62.22	100.00	62.22	57.69
33	95.32	25.71	95.28	52.86	94.85	48.57	94.63	48.57	91.39	28.57	94.85	95.71	94.59	52.86
34	77.88	27.35	78.37	37.18	77.40	40.60	76.92	34.19	79.81	32.48	75.00	50.00	80.77	82.91
35	73.75	36.59	68.98	67.07	67.38	62.20	68.42	60.98	71.66	50.00	62.56	100.00	63.78	79.27
36	94.74	28.74	93.42	64.94	93.42	44.25	94.74	63.22	93.42	36.78	90.79	51.15	92.11	51.15
37	60.00	24.59	50.00	57.38	60.00	33.61	70.00	65.57	60.00	32.79	60.00	54.92	80.00	60.66
38	82.60	25.97	81.72	36.36	81.36	38.96	81.48	35.71	81.54	35.71	72.22	59.09	80.30	76.62
39	97.03	25.86	95.05	48.28	96.04	31.03	100.00	46.55	94.06	27.59	96.04	82.76	96.04	53.45
Avg	**82.14**	**30.03**	**81.55**	**55.78**	**81.39**	**46.97**	**81.99**	**46.82**	**81.11**	**36.32**	**74.67**	**81.76**	**81.60**	**58.99**
w/n	**11/5**		**4/7**		**3/4**		**3/7**		**4/3**		**2/2**		**3/2**	

Table 3. Accuracy and Density of BBN learnt model.

A value of 100 means that the model is composed of all of the posible links. For example the dataset ‘arrhythmia’ contains 279 attributes (query variables or nodes in BBN) and one class variable. If we keep the constraint of four as the maximum number of parent nodes then the DAG will have 1110 links. A value of 44.86% (BDeu) means that the model produced by BDue contains 1110 X 44.86% = 498.

If we proceed for further analysis then we noticed that the model size for MDL is small (Average is 36.2) but this size is even more smaller in case of the proposed function where the average model size is 30.03. The worst performance in this dimension of analysis is exhibited by Entropy (Average size is 81.76) wherein this factor is in the range of 50% for the rest of the discriminant functions. The reason behind it is that whenever a new arc is included then the increase in the disciminant effect is only affected if the contributor query variable can increase the class-variable-explanatory effect significantly. However, the searching algorithm K2 also suffers from feature ordering problem. It is a good practice if a feature ranker can order them in such a way that the explanatory features gets more close to the 1st layer of the dag. Here we asume that the top most layer of the DAG is comprised of only class variable; wherein the second layer is comprised of all features. If the process of additions of layers is stopped here then such a BBN is a simple network and it usually gives reduced classification accuracy because of por goodness of data fitting. We in previous sections demonstrated that addition of new arcs (in further layers) influence the goodness of data fitting abruptly. The discriminant functions such as Entropy and AIC usually prone in this category and produce dense network. The problema with such dense network is two folded. Firstly, it requires more computational resources during parameter learning for the sake of inference from BBN. The second problem is model overfitting problem which sharply reduces the classification accuracy. Table 3 shows the same in case of dataset ‘flags’ and ‘kdd_synthetic_control’ where phenomenon of overfitting has explicitly reduced the classification accuracy of test instances.

The figure 3 gives the explanation from different angle in which we obtained the ratio of classification accuarcy and mode density (both in percentage). The calculation was obtained from the equation eight where the value of the constraint (maximum parent node) was set to four. It is evident from the figure 3 that the proposed discriminant function outperforms the other functions (the top curve). The behaviour of entropy was not much promising wherein MDL also give better result after the proposed function.

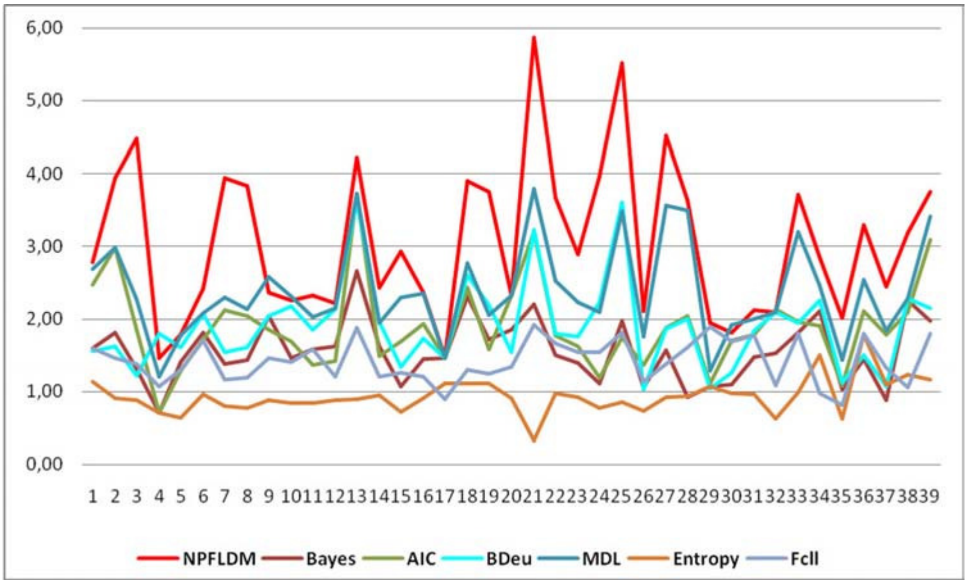


Figure 3. Ratio of Accuracy and Density of the model for 39 dataset.

We have discussed large number of results with various possibilities. However, it is required that we address two simple questions. Why $NPFLDF$ fails in some datasets? What is the justification of results when $NPFLDF$ outperforms? We shall discuss four dataset. These datasets include flags, mfeat-morphological, mfeat-pixel and waveform-5000. All of them vary in their characteristics including attributes, size of the datasets and number of classes. We observed that the datasets with more than two dozen attributes pose computational problems if we set the limit of maximum parent nodes of more than four. The experiment has been performed with setting of maximum node of two, three and four. The noteworthy aspect is that accuracy of $NPFLDF$ was constant in all of the cases. The underlying reason is that the likelihood factor is never getting increased quickly. Usually every segment of the DAG is restricted to two or three nodes while the value of $NPFLDF$ reaches its culmination point. Here the culmination point refers the highest value of $NPFLDF$ for which the goodness of the model is achieved. When we examine the other discriminant functions, this is not the case in most of the situations. We observed that two discriminant functions AIC and Entropy both are drastically accepting nodes under the independence assumption. The performance of entropy in datasets flag (features = 30) and mfeat-pixel (features = 241) is suffering from very large size of conditional probability table. However, the performance of MDL, BDeu and BIC is different. Although these discriminant functions control the unnecessary addition of arcs but usually elimination of wrong orientation is not guaranteed. The behavior of these discriminant functions is implicitly a function of count of parents and unluckily in most of the cases it is erratic. This leaves the problem of "selection of best maximum size of set of parent nodes/features". However in case of $NPFLDF$, its embedded characteristics of ordering features ensure to provide the best features for maximizing the discriminant objective. On the other hand, there are situations when $NPFLDF$ did not give better results in comparison to other discriminant functions. The reason can be explained from the figure 3 in which slope of $NPFLDF$ is drastically declining but up to two or three best features. If we reduce the sharpness of this slope then $NPFLDF$ will start tend to go in favor of more features (in this case more than three). However what is the trade between reducing the degree of slope of $NPFLDF$ versus increasing the links to more features. The answer lies in the experimental evaluation. The experimental results in this section point out that if we chose datasets with varying meta characteristics then sharp slope of $NPFLDF$ is more favorable in most of the cases dealing real datasets.

6. Conclusion

BBN has shown its appealing characterstics in data modeling for causal and noncausal dependencies among a set of data variables. Learning structure out of observational dataset is challenging because of the the model misspecification and nonidentifiability of the underlying structure. In this study, we first tweaked out the affinity relation between two dataset which determines how much one variable can explain the other variable. Keeping in view of it, we fomalized an Affinity Measure which can serve as a meta characteristics for the prediction of classification accuaracy of Decison Stump. the crux of this study was the introduction of a better discriminant function which can learn the BBN structure giving a smart model (reduced complexity in terms of number of arc) while keeping the same or better accuaray of the BBN classifier.

References

[1] W. J. Wang and Q. Xu, "A Bayesian Combination Forecasting Model for Retail Supply Chain Coordination," J. Appl. Res. Technol., vol. 12, no. 2.

[2] H. H. Avilés-Arriaga et al., "A comparison of dynamic naive bayesian classifiers and hidden markov models for gesture recognition," J. Appl. Res. Technol., vol. 9, no. 1, pp. 81–102, 2011.

[3] M. M. Haji et al., "Current Transformer Saturation Detection Using Gaussian Mixture Models," J. Appl. Res. Technol., vol. 11, pp. 79–87, 2013.

[4] D. Heckerman et al., "Learning Bayesian networks: The combination of knowledge and statistical data," Mach. Learn., vol. 20, no. 3, pp. 197–243, 1995.

[5] D. Heckerman, “A tutorial on learning with Bayesian networks,” in Innovations in Bayesian Networks, Springer, 2008, pp. 33–82.

[6] M. Naeem and S. Asghar, “A Novel Feature Selection Technique for Feature Order Sensitive Classifiers,” An. Ser. Inform. Ann. Comput. Sci. Ser., vol. 11, no. 1, pp. 31–38, Dec. 2013.

[7] S. P. Brooks and B. J. T. Morgan, “Optimization using simulated annealing,” The Statistician, pp. 241–257, 1995.

[8] S. Kirkpatrick, “Optimization by simulated annealing: Quantitative studies,” J. Stat. Phys., vol. 34, no. 5–6, pp. 975–986, 1984.

[9] G. F. Cooper and E. Herskovits, “A Bayesian method for the induction of probabilistic networks from data,” Mach. Learn., vol. 9, no. 4, pp. 309–347, 1992.

[10] T. D. Nielsen and F. V. Jensen, Bayesian networks and decision graphs. Springer, 2009.

[11] W. Buntine, “Learning classification trees,” Stat. Comput., vol. 2, no. 2, pp. 63–73, 1992.

[12] H. Akaike, “A new look at the statistical model identification,” Autom. Control IEEE Trans. On, vol. 19, no. 6, pp. 716–723, 1974.

[13] J. Skilling, Maximum entropy and Bayesian methods, vol. 45. Springer, 1989.

[14] W. Lam and F. Bacchus, “Learning Bayesian belief networks: An approach based on the MDL principle,” Comput. Intell., vol. 10, no. 3, pp. 269–293, 1994.

[15] J. Suzuki, “Learning Bayesian belief networks based on the MDL principle: An efficient algorithm using the branch and bound technique,” IEICE Trans. Inf. Syst., vol. 82, no. 2, pp. 356–367, 1999.

[16] A. M. Carvalho et al., “Discriminative learning of Bayesian networks via factorized conditional log-likelihood,” J. Mach. Learn. Res., vol. 12, pp. 2181–2210, 2011.

[17] H. Bozdogan, “Model selection and Akaike’s information criterion (AIC): The general theory and its analytical extensions,” Psychometrika, vol. 52, no. 3, pp. 345–370, 1987.

[18] M. Pelikan et al., Scalable optimization via probabilistic modeling. Springer, 2006.

[19] M. Pelikan, Hierarchical Bayesian optimization algorithm. Springer, 2005.

[20] E. S. Correa and J. L. Shapiro, “Model complexity vs. performance in the Bayesian optimization algorithm,” in Parallel Problem Solving from Nature-PPSN IX, Springer, 2006, pp. 998–1007.

[21] G. Schwarz, “Estimating the dimension of a model,” Ann. Stat., vol. 6, no. 2, pp. 461–464, 1978.

[22] N. Friedman and M. Goldszmidt, “Learning Bayesian networks with local structure,” in Learning in graphical models, Springer, 1998, pp. 421–459.

[23] D. M. Chickering et al., “Large-sample learning of Bayesian networks is NP-hard,” J. Mach. Learn. Res., vol. 5, pp. 1287–1330, 2004.

[24] L. Yang and J. Lee, “Bayesian Belief Network-based approach for diagnostics and prognostics of semiconductor manufacturing systems,” Robot. Comput.-Integr. Manuf., vol. 28, no. 1, pp. 66–74, 2012.

[25] W. Iba and P. Langley, “Induction of One-Level Decision Trees.,” 1992, pp. 233–240.

[26] M. Naeem and S. Asghar, “An Information Theoretic Scoring Function in Belief Network,” Int. Arab J. Inf. Technol., vol. 11, no. 5, pp. 1–10, 2014

[27] M. Naeem and S. Asghar, “A novel mutual dependence measure in structure learning,” J. Natl. Sci. Found. Sri Lanka, vol. 41, no. 3, pp. 203–208, 2013.

[28] D. H. Wolpert, “On bias plus variance,” Neural Comput., vol. 9, no. 6, pp. 1211–1243, 1997.

[29] S. Ali and K. A. Smith, “On learning algorithm selection for classification,” Appl. Soft Comput., vol. 6, no. 2, pp. 119–138, 2006.

[30] Datafit, “DataFit curve fitting (nonlinear regression) and data plotting software, and DataFitX ActiveX curve fitting engine.” 2013.