Data Construction using Genetic Programming Method to Handle Data Scarcity Problem

Abbas M. AL-Bakary ,Samaher Hussein Ali
* Computer Technology Collage, Babylon University, Iraq.
abbasmoh67@yahoo.com
*Computer Science Department, Babylon University, Iraq.
samaher_hussein@yahoo.com

***Abstract*- Genetic Programming Data Construction Method (GPDCM) uses in this work to handle one of the key problems in the supervised learning which is due to the insufficient size of training dataset. The methodology consists of four stages: first, represent each record in small dataset as decision tree(DT) where the collection of these trees represent the population of Genetic Programming algorithm(GPA). Second, attaching the numerical value to each node of those trees (Gain information Ratio). These values represent the fitness of the nodes. Third, expanding the small population by apply parallel method in three different types of crossover which is related to the GPA for each pair of the parents. Fourth, forecasting the classes to new samples generated by GPDCM using back propagation neural network (BPNN) ,then apply ROC graphs as a measures of Robustness Evaluation. The work takes all the important variables in to account, because it is started by collect DTs and it applies on five different datasets (iris dataset, weather dataset, heart dataset, soybean dataset and lamphgraphy dataset). For the theoretical and practical validity, we compare between the proposed method and the other applied methods. As the result, we fined that GPDCM is promising techniques for expanding the extremely small dataset and extracted a useful knowledge .**

Keywords_ *insufficient size of dataset, decision tree, Genetic Programming, Gain Information Ratio, BPN*

I. INTRODUCTION

Data scarcity problem is one of the main problems of machine learning and data mining, because insufficient size of data is very often responsible for poor performances of learning, how to extract the significant information for inferences is a critical issue. It is well known that one of the basic theories in Statistics is the Central Limit Theorem [1,16]. This theorem asserts that when a sample size is large (≥30), the x-bar distribution is approximately normal without considering the population distribution. Therefore, when a given number of samples are less than 30 , it is consider as insufficient size of samples to perform intelligent analysis. There are two possible ways to overcome the data scarcity problem. One is to collect more data while the other is to design techniques that can deal with extremely small data sets.

One major contribution to the above issue has been given by [2] who developed a methodology for integrating different kinds of "hints" (prior knowledge) into usual learning from- example procedure. By this way, the "hints" can be represented by new examples, generated from the existing data set by applying transformations that are known to leave the function to be learned invariant.

Then, [3] modified "hints" into "***virtual samples***" and applied it to improve the learning performances of artificial neural networks such as Back- Propagation and Radial Basis Function Networks. In fact, it is evident that generating more resembling samples from the small training set can make the learning tools perform well. [4] Proposed to use a neural network ensemble to preprocess the training data for a rule learning approach. They used the original training data set to generate an ensemble at first. Then, they randomly generated new instances and passed them to the ensemble for classification. The outputs from the ensemble were regarded as labels of these instances. By combining the predicted labels and the training inputs, new examples were obtained and used to enlarge the training data set. The enlarged training data set was finally used by a rule learning approach. This approach has been applied to gene expression data and obtained interesting results.

[5] Showed that using an ensemble to preprocess the training data set for a succedent learner (e.g., a decision tree) according to the above routine is beneficial so long as: 1) the original training data set is too small to fully capture the target distribution, or the original training set contains noise, and 2) the ensemble is more accurate than the single learner if both of them are directly trained from the original training data set. [6] Combined the concept of ***virtual sample*** generation and the method of intervalized kernel density estimation (IKDE) to overcome the difficulty of learning with insufficient data at the early manufacturing stages. From the results, it can be noted that, when the size of virtual data increases, the average learning accuracy would decrease. [7] proposed machine learning algorithm is modified for mining extremely small data sets. This algorithm works in a *twice-learning* style. In detail, a random forest is trained from the original data set at first. Then, virtual examples are generated from the random forest and used to train a single decision tree.

Recently,[8] proposed to improve the learning ability from a small data set. To demonstrate its theoretical validity, we provided a theorem based on Decomposition Theory. In addition, we proposed an alternative approach to achieving the better learning performance of IKDE called Data Construction Method(DCM).

This work suggests new methodology to expanding the small dataset base on combination between one of the optimization algorithm (Genetic Programming) and decision tree. This methodology called GPDCM then forecasting of the classes of

the new samples by BPNN and to sure of the robustness of this methodology using ROC curve as robustness evaluation measure.

The rest of this paper is organized as follows section 2 discuss the main tool use in this work and how you can avoid the main problems for each it. Section 3 explains the proposed methodology. Section 4 describes five experiments. Section 5 shows the conclusion and compare the suggest methodology with others.

II. METHODOLOGY REQUIREMENTS

A. Decision Tree

Decision trees are one of the fundamental techniques used in data mining. The main properties of it:

- it is tree-like structures used for classification, clustering, feature selection, and prediction.
- It is easily interpretable and intuitive for humans.
- They are well suited for high dimensional applications.
- It is fast and usually produce high-quality solutions.
- Decision tree objectives are consistent with the goals of data mining and knowledge discovery.

A decision tree consists of a root and internal nodes. The root and the internal nodes are labeled with questions in order to find a solution to the problem under consideration. The root node is the first state of a DT. This node is assigned to all of the examples from the training data. If all examples belong to the same group, no further decisions need to be made to split the data set. If the examples in this node belong to two or more groups, a test is made at the node that results in a split. A DT is binary if each node is split into two parts, and it is non binary (multi-branch) if each node is split into three or more parts.

There are two methods to building decision tree:-

- Top-down tree construction

At start, all training examples are at the root. Partition the examples recursively by choosing one attribute each time.

- Bottom-up tree pruning

Remove subtrees or branches, in a bottom-up manner, to improve the estimated accuracy on new cases.

B. Genetic Programming Algorithm

GPA is an extension of Genetic Algorithms (GAs). It solves the representation problem in GAs. The main principles of GP are based on the mechanisms of Evaluation.

- Survival of the fittest and natural selection.
- An offspring's chromosome consists of parts derivate from the chromosome of its parents (i.e., inheritance mechanism).
- A change in the offspring is a characteristic which is not inherited from the parents (i.e, mutation).

This paper proposes a genetic programming system for discovering new patterns to handle data scarcity problem where each individual represents as decision tree in the population and the population is collection of theses decision trees. We believe a use of GP is a promising research area, since GP has the advantage of performing a global search in the space of candidate patterns.

Data construction methods can be roughly divided into two groups, with respect to the construction strategy: hypothesis driven methods and data-driven methods [9].

Hypothesis-driven methods construct new attributes out of previously-generated hypotheses (discovered rules or another kind of knowledge representation). In general they start by constructing a hypothesis, for instance a decision tree, and then examine that hypothesis to construct new attributes [11]. *By contrast*, data-driven methods do not suffer from the problem of depending on the quality of previous hypotheses. They construct new attributes by directly detecting relationships in the data. The process of attribute construction can also be roughly divided into two approaches, namely the interleaving approach and the preprocessing approach.

In the preprocessing approach the process of attribute construction is independent of the inductive learning algorithm that will be used to extract knowledge from the data. In other words, the quality of a candidate new attribute is evaluated by directly accessing the data, without running any inductive learning algorithm. In this approach the attribute construction method performs a preprocessing of the data, and the new constructed attributes can be given to different kinds of inductive learning methods. *By contrast*, in the interleaving approach the process of attribute construction is intertwined with the inductive learning algorithm. The quality of a candidate new attribute is evaluated by running the inductive learning algorithm used to extract knowledge from the data, so that in principle the constructed attributes' usefulness tends to be limited to that inductive learning algorithm. An example of data construction method following the interleaving approach can be found in [12].

In this work we follow the data-driven strategy and the preprocessing approach, mainly for two reasons**.** *First*, using this combination of strategy/approach the constructed data have a more generic usefulness, since they can help to improve the predictive accuracy of any kind of inductive learning algorithm. *Second*, data construction method following the preprocessing approach tends to be more efficient than its interleaving algorithm.

C. Back Propagation Neural Network

Back propagation is a systematic method for training multilayer artificial neural network and its learning rule is generalized from Widrow-Hoff rule for multilayer networks, the Back propagation network is a very popular model in neural network. It does not have feedback connections, but error are Back propagated during training, Least mean squared error is used. Many application can be formulated by using a Back propagation network and the methodology has been a model for most multilayer neural networks. The processing unit or neuron used here is similar in nature to the perceptron cell: it applies in activation function to the weighted sum of the inputs; the activation function is a non-linear function. A sigmoid function is most commonly Used:

$$Out = f(net)=1/1+\exp(-net)$$

After we get the number of new samples generate by GPDCM. These samples send to the bank of samples and beginning train error back propagation neural network on the original small dataset then test if training phase complete successfully then saving the training' weights. After that, using these weights to forecasting the class of the new samples generating by GPDCM.

But, at the first, we need to determine the structure of network (i.e., number of nodes in input layer, number of nodes in hidden layer and number of nodes in output layer) and also the

initial values of weights.

The output class vector cj, (j=1,2,….,j), j is the number of different possible classes. If the output vector belongs to the class k then the element is equal to 1 while all the other elements in the vector are zeros. Therefore, the proposed number of output nodes in the output layer of ANN is j (where j different from dataset to another)

III. THE STRUCTURE OF SUGGESTED METHODOLOGY

There are five open still problems at this time remain in the KDD as explain in figure 1 (data scarcity problem, find the actual values of the missing values of dataset , convert data mining algorithm from black box to white box form , design mathematical model of KDD system and find the optimal method to simplify and reduce the knowledge by KDD system). In this work we try to solve one of these problems which is related to handle the insufficient size of samples and then find intelligent analysis method for these samples.

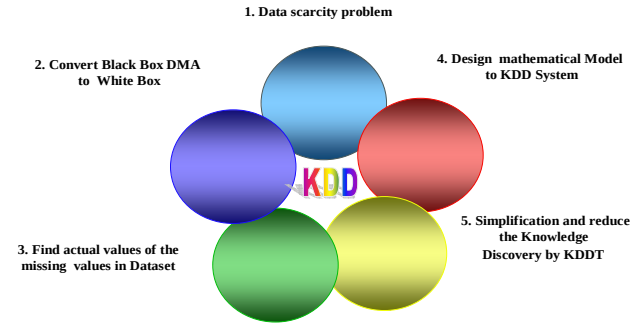


Fig. 1:A: Open Still Problems of KDD

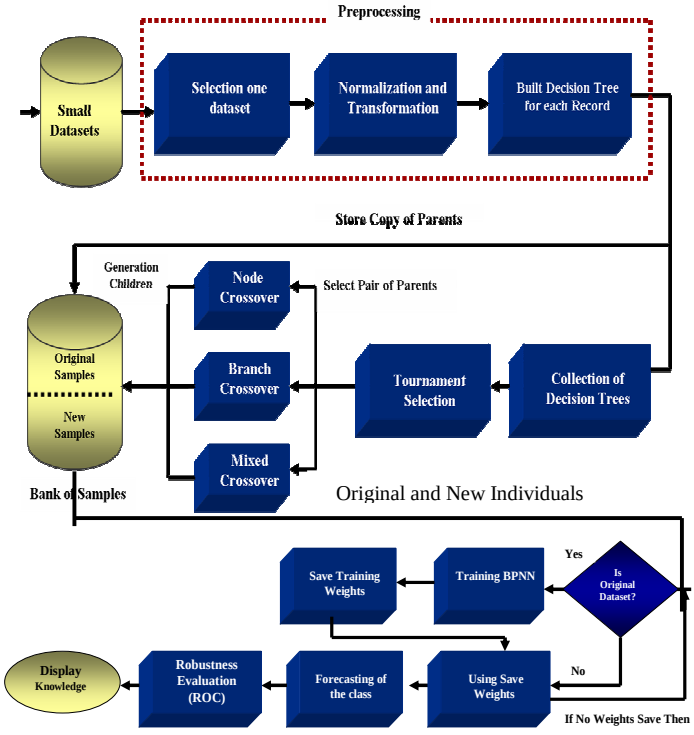


Fig. 1:B: Block Diagram of the Suggested Methodology to

Algorithm Handel Data Scarcity Problem

***Input*:** Small dataset (*insufficient size of dataset)*
Output: Set of New Samples with classes
• ***Step 1:*** *Set Nmin , Nmax to min & max nodes of trees number expected respectively, Set MaxGen to max iteration allowed*
• ***Gen ← 1***
• ***Step 2*** *: Population initialization:*
  o *For each Recode in the datasets*
    • *Perform coding and normalization of features*
    • *Call Create DT Procedure*
    • *Convert General DT to Binary DT*
• ***Step 3****: Fitness calculation:*
  o *For each DT in the population*
    • *If the Node is Root then this node' feature is interesting*
    • *Compute Entropy by Eq(2)*
    • *Compute Expected information by Eq(3)*
    • *Compute Gain information by Eq(4)*
    • *Compute Gain information Ratio by Eq(5)*
• ***Step 4:*** *If Gen> Maxgen GOTO Step 7.*
• ***Step 5****:Genetic Programming Operations*
  o *Tournament selection(size 4,8)*
  o *Call Node crossover Procedure*
  o *Call Branch crossover Procedure*
  o *Call Mixed crossover Procedure*
  o *Save copy of the this population on Bank of samples*
• ***Step 6****:* Validation of New population
  o *Evaluation the New Children(Individuals) Generation by Three Crossover Approaches*
  o *Test if any of the new individuals similarly to any individuals of original population then remove the new individuals* ***Else*** *Add New individuals to Bank of samples*
• ***Step 7:*** *End Generation*
• ***Step 8:*** *Test if sample is Relate to Original small dataset then*
  o *Call Training BPNN Procedures*
    • *If Training Phase Successful Then Saving Weights.*
    •***Else*** *Regeneration New random weights and start new training phase*
  o Else *Call Testing BPNN Procedure.*
  o *Forecasting the Class of New Samples*
  o *Call Robustness Evaluation Procedure*
• ***Step 9:****End GPDCM algorithm*

A. Procedure Normalization and coding

Scale the data value to a range using Min-Max method :(Linear transformation) of the original input range into a newly specified data range.

$$Y' = [(Y-min) / (max-min)]*(max' -min') +min' \qquad (1)$$

Where: min is old minimum value, min' is new minimum, max is old maximum, max' is new maximum.

Example: Consider the old data that ranged from [0-100], we now obtain an equation to migrate it to [5-10] range.

$Y' = [(Y-0)/(100-0)]*(10-5)+5$
$Y' =[Y/100]*5+5$
$Y' =(Y/20)+5$
Let Y=0 Then Y ' = 5
If Y=10 Then Y ' = (1/2)+5=(1+10)/2=5.5

Coding(Convert Linguistic terms to numeric forma)

To encode the attributes of the linguistic variable we can use the following procedure:

- Create the repetition table by determining the repetition times for each linguistic term.
- Rearrange the table by making the large value repetition in the middle and the lesser on right and left of it until minimum repetition becomes at most left and most right.
- Assign the code for each linguistic term depending on its new order in the repetition table.

B. Fitness function of GPDCM

The fitness function used in this work is normalizes information gain ratio, which is a well-known attribute-quality measure in the data mining and machine learning literature. It should be noted that the use of this measure constitutes a data-driven strategy. As mentioned above, an important advantage of this kind of strategy is that it is relatively fast, since it avoids the need for running a data mining algorithm when evaluating an attribute (individual). In particular, the information gain ratio for a given attribute can be computed in a single scan of the training set[13]:

$$\text{entropy}(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i \qquad (2)$$

$$Information(Attribute) = \sum_{i=1}^{m} \frac{Fre.SubAttribute}{TotalNo.Of\ \text{Re}cord} * Entropy \qquad (3)$$

$$Information\ \ Gain = IBS - IAS \qquad (4)$$

Where; IBS: Information before Splitting, IAS: Information after Splitting.

But there are two problems of information Gain [14]:

- Information gain is biased towards choosing attributes with a large number of values and this may result in **overfitting problem** (selection of an attribute that is non-optimal for prediction).
- **fragmentation problem**.

  Therefore to avoid the above two problems. We suggest use ***Gain ratio***: a modification of the information gain that reduces its bias on high-branch attributes and *Gain ratio* takes number and size of branches into account when choosing an attribute.

$$\text{gain\_ratio}(\text{"Attribute"}) = \frac{\text{gain}(\text{"Attribute"})}{\text{intrinsic\_info}(\text{"Attribute"})} \qquad (5)$$

Where Intrinsic information: entropy of distribution of instances into branches (i.e. how much info do we need to tell which branch an instance belongs to)

$$IntrinsicI\ \ nfo(S, A) \equiv -\sum \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}. \qquad (6)$$

C. Procedures Create DT (samples(s), attributes (A), Class(C))

- ***Step1:*** *Create a Root node for the tree*
  - IF *S* = empty, return a single node with value Failure
  - IF *S* = *C*, return a single node with C
  - IF *A* = empty, return a single node with most frequent target attribute (C)

    ELSE
- ***Step2:*** *BEGIN*
  - Let *D* be the attribute with largest Gain Ratio *(D, S)* among attributes in *A*
  - Let *{dj\|j* = 1, 2 , . . . , *n}* be the values of attribute *D*
  - Let *{Sj\|j* = 1, 2 , . . . , *n}* be the subsets of S consisting respectively of records with value *dj* for attribute *D*
  - Return a tree with root labeled *D* arcs *d, d-i,..., dn* going respectively to the trees
- ***Step3:*** *For each branch in the tree*
  - IF *S* = empty, add a new branch with most frequent C
  - ELSE Compute the Gain Ratio for all Nodes of branch
- ***Step4***:- End Create DT

D. Procedure Node Crossover (Pair of Parents, One Child)

- ***Step1:*** Select two Trees as parents from collection of DT
- ***Step2:*** Select random one crossover node from the first tree and search randomly in the second tree for an exchangeable.
- ***Step3:*** Swap the crossover node.
- ***Step4:*** The child is a copy of the modified its first Tree.

E. Procedure Branch Crossover (Pair of Parents, One Child)

- ***Step1:*** Select two Trees as parents from collection of DT
- ***Step2:*** Select random one crossover node from the first tree and search randomly in the second tree for an exchangeable.
- ***Step3:*** Cutoff the branch with the crossover nodes.
- ***Step4:*** Calculate the size of the expected child (remind size of first tree + size of branch cutoff from the second tree).
- ***Step5:*** If the size of child is accepted created the child by appending the branch cutoff from second tree to the ramming of first tree otherwise try again starting from (step 2).

F. Procedure Mixed Crossover (Pair of Parents, One Child)

- ***Step1:*** Select two Trees as parents from collection of DT
- ***Step2:*** Select random one crossover node a terminal node in the second Tree.
- ***Step3:*** Generate the child by replacing the branch with the crossover node in first tree with the terminal node select from second tree.

G. Procedures Training BPNN (Samples, Class)

- ***Step1:*** Input initial values of network parameters: learning rate , momentum rate, number of epochs.
- ***Step2:*** Network error given zero value , learning rate determine, *and generate randomly initial weights in rang[0-1]* .
- ***Step3:*** Pass training samples cross input layer to hidden layer and compute activity for each node on it.
- ***Step4:*** Pass training samples from hidden layer to output layer and compute activity for each node on it.

- ***Step5:*** Compute output nodes error of that sample
- ***Step6:*** Compute hidden nodes error of that sample
- ***Step7:*** Adjust weights between hidden layer and output layer
- ***Step8:*** Adjust weights between input layer and hidden layer
- ***Step9:*** IF complete pass training samples then
  - Determine cost function value(Mean Sequre Error). Else go to Step4
- ***Step10:*** If termination criterion achieve then
  - Stop training phase and saving weight
  - Else go to Step3

H. Procedures Testing BPNN (Input: New Samples, Saving weights, Output: Forecasting Class)

- ***Step1:*** Pass test samples cross input layer to hidden layer and compute activity for each node on it base on saving weights from training phase.
- ***Step2:*** Pass training samples from hidden layer to output layer and compute activity for each node on it.
- ***Step3:*** Compute output nodes error of that sample
- ***Step4:*** Compute hidden nodes error of that sample
- ***Step4:*** IF complete pass testing samples then
  - Determine cost function value(Mean Sequre Error).
- ***Step5:*** If termination criterion achieve then
  - Stop testing phase and giving the actual output as Classes of that samples.

I. Robustness Evaluation

There are many types of robustness evaluation measures such as *accuracy* of a classifier is measured as the percentage of instances that are correctly classified, and the *error* is measured as the percentage of incorrectly classified instances (unseen data) [10]. But when the considered classes are imbalanced or when misclassification costs are not equal both the accuracy and the error are not sufficient. Therefore, in this work we use technique base on confusion matrix Called Receiver Operating Characteristics (ROC) graphs are a useful technique for organizing classifiers and visualizing their performance. ROC graphs are commonly used in medical decision making, and in recent years have been increasingly adopted in the machine learning and data mining research communities. Add that ROC graphs are very useful in assessing the overall behavior and reliability of the classification task under inspection. The ROC graph shows the relation between the True Positive Fraction (TPF) on the y-axis and the False Positive Fraction (FPF) on the x-axis. The ***True Positive Fraction*** is defined as:

$$TPF = \frac{TP}{TP + FN} \quad (7)$$

where TP is the number of true-positive test results, and FN is the number false negative tests. The ***False Positive Fraction*** is defined as:

$$FPF = \frac{FP}{TN + FP} \quad (8)$$

where FP is the total number of false-positive test results, and TN is the number of true negative test results**.**

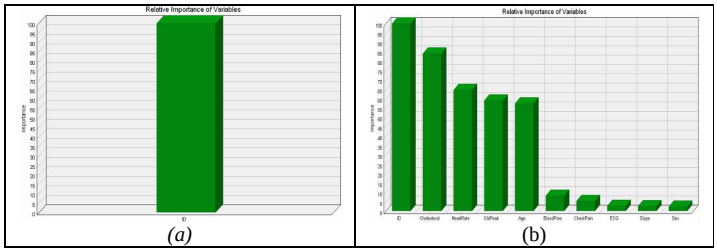
Procedure of Robustness Evaluation

- **Step1***: Input: list of test examples T generated by GPDCM and sorted in Bank of individuals with there class generation by BPNN*
- **Step2:**BEGIN
  - Let K = |T| be the number of test examples
  - Let P be the number of positive examples in T
  - Let N be the number of negative examples in T
  - Set ; TP := 0 and FP := 0
- **Step3***:* for i = 1 to K do
  - if example T[i] is positive then
    - TP := TP + 1
    - Else, FP := FP + 1
- **Step4***:* output point on ROC curve (TP/P),(FP/N)
- **Step5***:* End procedures

IV. EXPERIMENTAL RESULTS

In order to test the performance of the suggested methodology, we can apply it on the five different small dataset (Iris dataset, weather dataset, heart dataset, soybean dataset and lamphgraphy datasets). Table I shows the results of each one of that datasets and table II shows the computed results of one of these datasets (weather dataset), by computing the fitness for each attribute . Table III describe the fitness for each record.

A. Result of Heart Dataset



(a) (b)

Fig 2 . (a):Important variables Using Single DT. (b): Important variables using GPDCM.

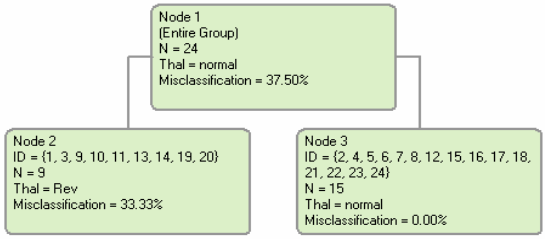
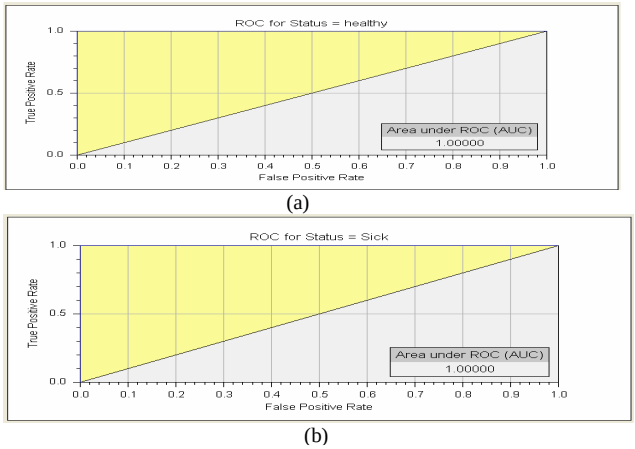


Fig 3. DT of Heart Dataset

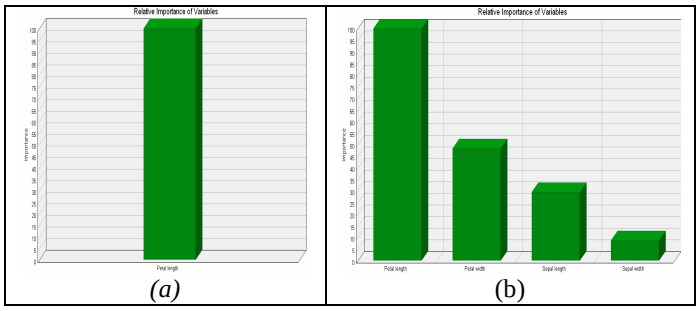


(a)

(b)

Fig 4 . (a): ROC for Status=healthy. (b) :ROC for Status=Sick.

B. Result of Iris Dataset



(a) (b)

Fig 5 . (a):Important variables Using Single DT. (b): Important variables using GPDCM.

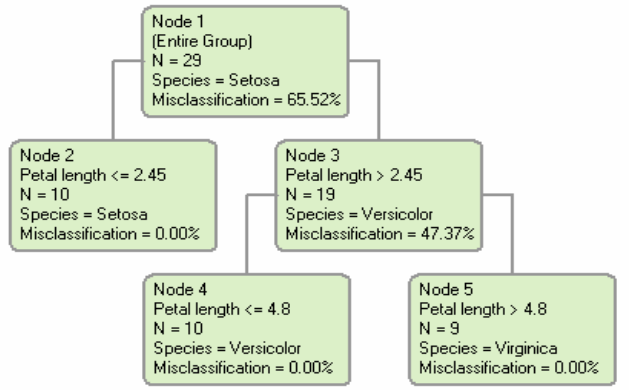
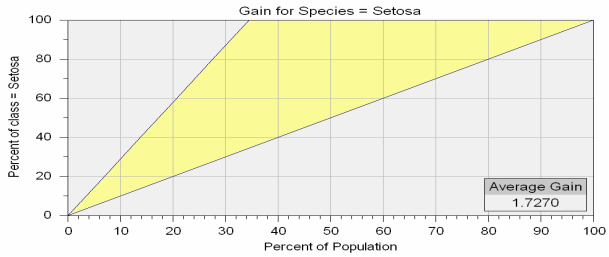
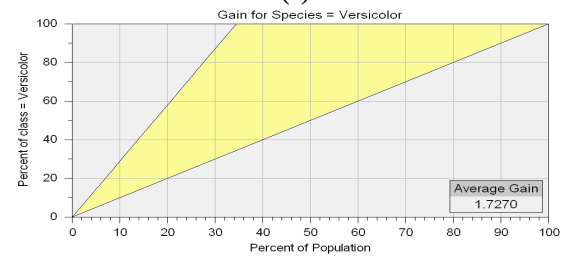


Fig 6. DT of Iris Dataset



(a)



(b)

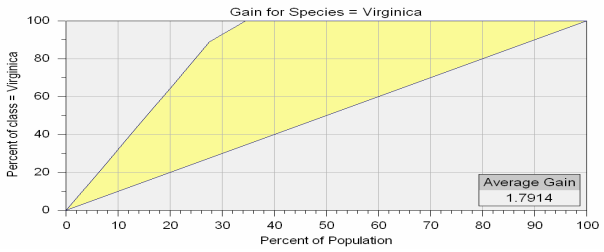
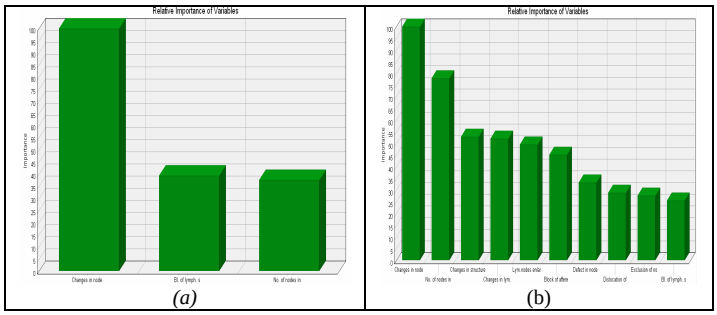


Fig 7 . (a): Gain for Species=setosa. (b) : Gain for Species=Versicolor. (c): Gain for Species=Virginica

C. Result of Lymphography Dataset



(a) (b)

Fig 8 . (a):Important variables Using Single DT. (b): Important variables using GPDCM.

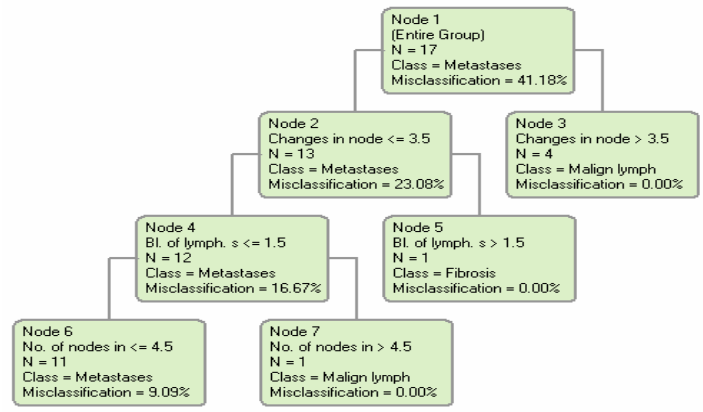
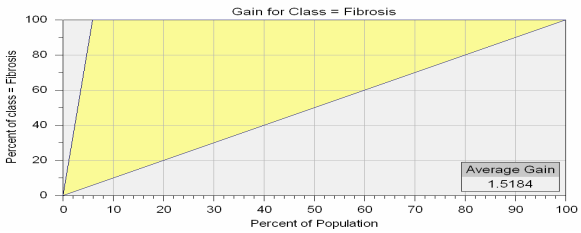
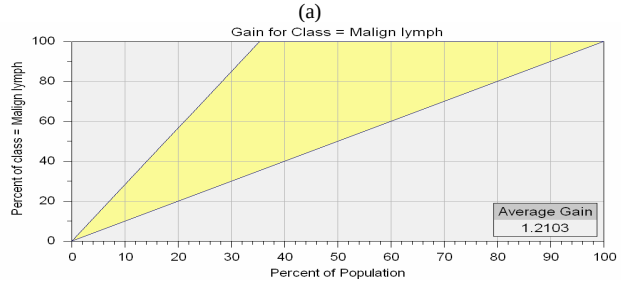


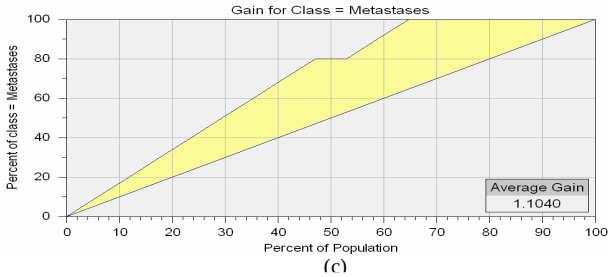
Fig 9. DT of lymphography Dataset



(a)



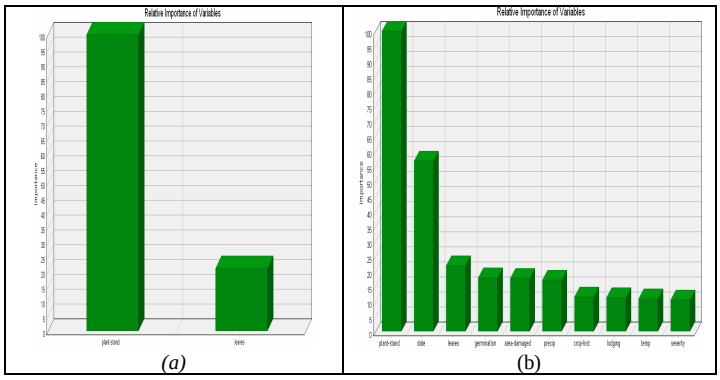
(b)



(c)

Fig 10 . (a): Gain for class =Fibrosis. (b) : Gain for class =Malign Lymph. (c): Gain for class =Metastases

D. Result of Soybean Dataset



(a) (b)

Fig 11 . (a):Important variables Using Single DT. (b): Important variables using GPDCM.

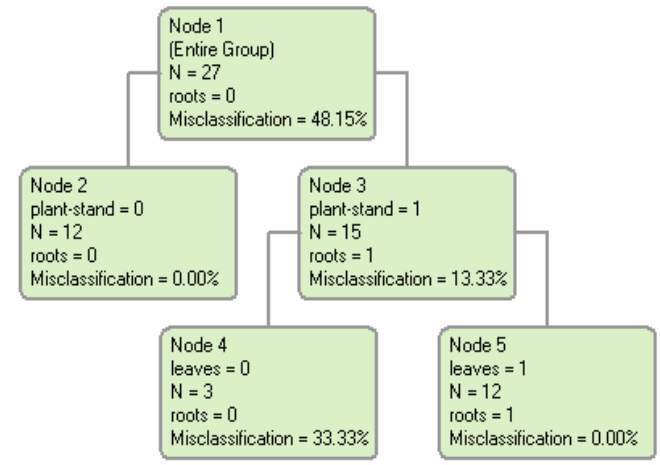
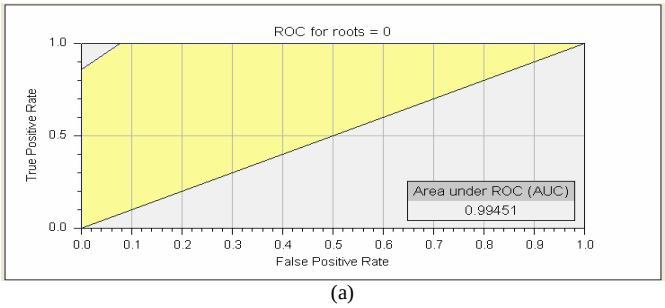
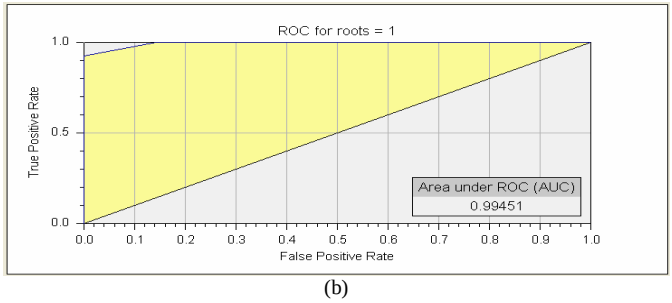


Fig 12. DT of Soybean Dataset



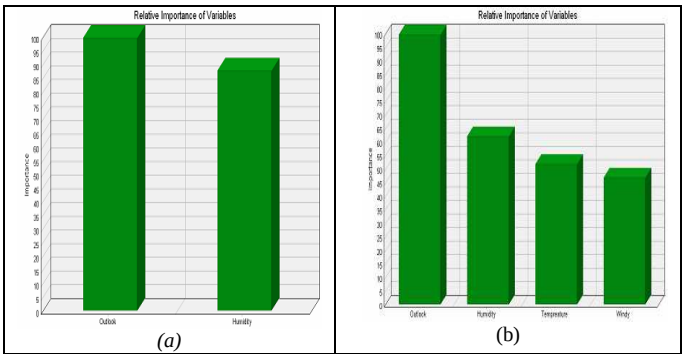
(a)



(b)

Fig 13 . (a): ROC for roots=0. (b) : ROC for roots=1.

E. Result of Weather Dataset



(a) (b)

Fig 14 . (a):Important variables Using Single DT. (b): Important variables using GPDCM.

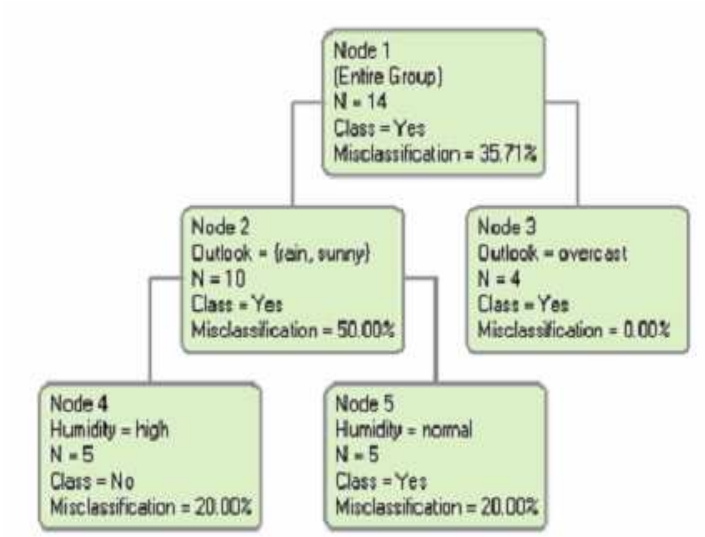
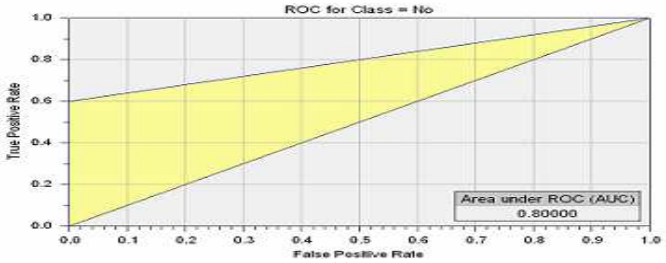
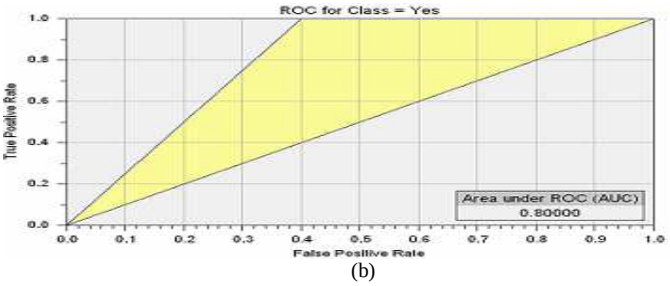


Fig 15. DT of Weather Dataset



(a)



(b)

Fig 16 . (a): ROC for Class=No. (b) : ROC for Class=Yes.

TABLE I. RESULT OF EACH SMALL DATASET

Name of DB	NOS	NOF	NOC	Max NOG.	TNONS	NONTS	NOE BPNN	MSE
Heart	24	14	2	3	375	278	1400	0.711 E-6
Iris	29	5	3	2	180	121	1832	0.333 E-6
Lympho -graphy	17	19	3	3	263	132	3390	0.261 E-5
Soybean	27	35	4	2	168	90	2867	0.498 E-4
Weather	14	5	2	4	542	172	3175	0..973 E-5

Where, NOS: No. of original Samples, NOF: No. of Features, NOC: No. of Classes, Max NOG: Max No. of Generation, TNONS: Total No. of New samples. NONTS: No. of New True Samples, NOE BPNN: No. of Epochs of BPNN, MSE: Mean Square Error.

NONTS=TNONS–NNONS (9)

Where TNONS: total NONS, NNONS: Neglected NONS

(Ex: NONTS of Heart=375-97 =278)

TABLE II. FITNESS FOR EACH ATTRIBUTE

Features	Exp.inf.	Gain inf.	Intr.. inf.	Gain ratio
Outlook	0,689	0,245	1,567	0,156
Temperature	0,905	0,029	1,546	0,019
Humidity	0,784	0,15	0,993	0,151
Windy	0,886	0,048	0,979	0,049

TABLE III. FITNESS FOR EACH SAMPLE

No. of Sample	Exp.inf.	Gain inf.	Intr.. inf.	Gain ratio
1	0,512	0,422	1,511	0,279
2	0,448	0,486	1,361	0,357
3	0,473	0,461	0,964	0,478
4	0,448	0,486	1,361	0,357
5	0,297	0,637	0,717	0,888
6	0,473	0,461	0,964	0,478
7	0,473	0,461	0,964	0,478
8	0,512	0,422	1,511	0,279
9	0,448	0,486	1,361	0,357
10	0,297	0,637	0,717	0,888
11	0,512	0,422	1,511	0,279
12	0,448	0,486	1,361	0,357
13	0,297	0,637	0,717	0,888
14	0,448	0,486	1,361	0,357

TABLE IV : FITNESS FOR EACH ATTRIBUTE AFTER APPLY GPDCM

Features	Exp.inf.	Gain inf.	Int. inf.	Gain ratio
Outlook	0,297	0,537	0,707	0,759
Temperature	0,448	0,486	1,520	0,319
Humidity	0,873	0,621	0,865	0, 717
Windy	0,512	0,356	1,540	0,231

TABLE V : RELATIVE IMPORTAIN OF EACH VARIABLES USING GPDCM

Heart dataset

Variable	Importance
ID	100.000
Cholesteral	84.127
HeartRate	64.682
OldPeak	59.044
Age	57.405
BloodPres	8.365
ChestPain	5.453
ECG	2.899
Slope	2.669
Sex	2.561
Status	1.755
Angina	0.962
VesselCount	0.531
LowBloodSugar	0.266

Iris dataset

Variable	Importance
Petal length	100.000
Petal width	48.553
Sepal length	29.656
Sepal width	8.814

Lymphography dataset

Variable	Importance
Changes in node	100.000
No. of nodes in	78.027
Changes in structure	53.038
Changes in lym.	52.268
Lym.nodes enlar	49.669
Block of affere	45.345
Defect in node	33.349
Dislocation of	29.018
Exclusion of no	27.902
Bl. of lymph. s	25.785
Extravasates	25.716
Bl. of lymph. c	16.735
Early uptake in	15.648
Lym.nodes dimin	14.443
By pass	14.189
Regeneration of	11.737
Lymphatics	0.104

Soybean dataset

Variable	Importance
plant-stand	100.000
date	56.996
leaves	22.082
germination	18.022
area-damaged	17.861
precip	17.219
crop-hist	11.784
lodging	11.587
temp	11.056
severity	10.723
seed-tmt	5.417
hail	5.168

Weather dataset

Variable	Importance
Outlook	100.000
Humidity	62.366
Tempreature	52.080
Windy	47.147

V. CONCLUSION AND FUTURE WORK

The work solve one of the open problem (data scarcity problem) by expanding the small dataset size using GPDCM that depend on collection of DTs as population. Then forecasting the classes of new samples generated by GPDCM using BPNN, and then apply ROC graphs as a measures of Robustness Evaluation.

GPDCM applies three different types of crossover approaches and it takes all the important variables in to account as mentioned in Table V. the starting by collection of DTs while DT take only one variable (target variable) in to account. From the result, we can say GPDCM is more suitable tool to construct samples

and verify them by comparing with other approaches.

As a result, if the population consist of 20 DTs (No. of samples in dataset) and each crossover between two parents yields one sample this mean generation 10 children(samples) when apply node crossover, 10 children(samples) when apply branch crossover, 10 children(samples) when apply mixed crossover at first iteration of GPDCM.

The size of new population =the size of old population + No of children for all the three crossover approaches. The size of new population=20 +30=50 samples. While the size of population in second iteration =50+75=125 samples. At this time must be discuss the following questions: Can you generation infinite number of samples? Of course NO. Is all samples are true? No. where by experiments find when increase no of generation of GPDCM the probability of generation false samples is also increase. Then, what is optimal number of true samples in population can be found and when the algorithm is stop?

From the results, we can say: Optimal Number of True Samples in Population (ONTSP) is:

$$ONTSP = \sum_{G=1}^{G=MAXGEN} [\,Pop.Size(G) + ((Pop.Size(G)/2)*3) - W(G)\,] \quad (10)$$

Where, G : Number of generation in GPDCM (i,e., optimal values[1-8]). W: Const represents number of new samples that similar to original samples(i.e., the samples that neglected).

We can note that BPNN is a classification whose comprehensibility is poor. If we only want to get a high predicative accuracy, BPNN may be sufficient; but if we want to study which class is important (or more suitable of new samples), BPNN could not help since it is "black-box" model, while when combination with GPDCM is become more helpful for this purpose.

The table IV explains how the new proposed technique extreme the problems appear in other methods and it givens the results better than proved by IKDE, DCM and RF2Tree.

In the future work we hope to apply our approach on STR of human DNA to recognize passive personals identity .

REFERENCES

[1] Ross, M. S. (1987). Introduction to probability and statistics for engineers and scientists. John Wiley & Sons, Inc.

[2] Abu-Mostafa, Y. S. (1993). Hints and the VC-dimension. *Neural Computation, 5*, 278-288.

[3] Niyogi, P., Girosi, F., & Tomaso, P. (1998). Incorporating prior information in machine learning by creating virtual examples. *Proceeding of the IEEE, 86*(11), 275-298.

[4] Z.-H. Zhou and Y. Jiang,(2003).,Medical diagnosis with C4.5 rule preceded by artificial neural network ensemble, IEEE Transactions on Information Technology in Biomedicine, vol.7, no.1, pp.37–42.

[5] Z.-H. Zhou and Y. Jiang,(2004)., NeC4.5: Neural ensemble based C4.5 IEEE Transactions on Knowledge and Data Engineering, vol.16, no.6, pp.770–773.

[6] Li, D.-C., & Lin, Y.-S. (2006). Using virtual sample generation to build up management knowledge in the early manufacturing stages. *European Journal of Operational Research, 175*, 413-434.

[7] Yuan J., Ming L. and Zhi H.(2008). Mining Extremely Small Data Sets with Application to Software Reuse. *Elsevier Preprint. China*

[8] Chun-J and Hsiao F. (2009). Virtual Sampling with Data Construction Method. InformatIon scIence reference. Hershey • New York

[9] Hu, Y-J.(1998). A Genetic Programming Approach to Constructive Induction . In Proceeding of 3rd Annual Genetic Programming Conference, pp. 146–151.

[10] Zhao B. and Deng C.(2010), Dataset Quality Evaluation Method Based on Human Visual System" Journal of Electronics Vol.19, No.1,

[11] Pagallo, G. & Haussler, D.(1990). Boolean Feature Discovery in Empirical Learning. In Machine Learning 5, pp.71–99.

[12] Zheng, Z. (.2000).Constructing X-of-N attributes for decision tree learning. Machine Learning 40, pp. 1-43.

[13] C Korhan K, Ahmet A, and Ahmet S. (2008). Long term energy computation forecasting using genetic programming. Mathematical and computational application, vol 13,No 2, pp 71-80.

[14] Leonardo V, Francesco A, Mauro C and Iiaria G I.(2009). Classification of Oncologic data with Genetic Programming Approach. Journal of Artificial Evoulation and applications. Italy. dio:10.115/2009/848532.

[15]. Hammad M. and Conor R.(2009). A re-examination of real world blood flow modeling problem using context aware crossover. Journal of Bioinformatics and Data mining Techniques.

[16] Samaher H. (2010). Main Open Still Problem of KDD . Journal of Babylon , Iraq

TABLE VI : COMPARE AMONG GPDCM AND OTHER APPROACHS

Methodology	Theorem	Function	Fundamental Tools	Solve problems	Remind problem	Size of generated dataset	Robustness Measures
IKDE (2006)	**Decomposition Theory**	**Density Estimation Function**	**Four Rules**	**1. Difficult learning with insufficient dataset**	**1. Predication from insufficient size of training dataset 2. Validations**	**2^G*Popsize**	**Accuracy**
RF2Tree (2008)	**Randomization Theory**	**Probability of error Ratio Function**	**Combination of (random forest, decision tree and A prior algorithm)**	**1. Difficult learning with insufficient dataset 2. Deal with very small of dataset**	**1.Twice- learning style 2. RF is poor comprehensibility 3. Validations**	**1. Unknown Because it base on randomization principle**	**Expert opinions**
DCM (2009)	**Decomposition Theory**	**Multiset Division Function**	**Combination of (Decomposition methods, BPN)**	**1. Difficult learning with insufficient dataset 2. Predication from insufficient size of training dataset**	**1. Complexity 2.Generalization Problem**	**2^G*(Popsize-1**	**Accuracy**
GPDCM (2010)	**Optimization Theory**	**Normalization Gain Information Ratio Function**	**Combination of (DT, GPA, BPN and ROC)**	**1. Difficult learning with insufficient dataset 2. Predication from insufficient size of training dataset 3. complexity problem 4.generalization problem**	**1. Determined initial parameters of BPN 2. Ratio of mapping between new samples and reality**	**Eq. (10)**	**ROC**