T.C.

ISTANBUL ALTINBAS UNIVERSITY

INFORMATION TECHNOLOGY

EVALUATION AND VALIDATION OF THE INTEREST OF THE RULES ASSOCIATION IN DATA-MINING

Ali Yousif Hasan

Master Thesis

Supervisor. Dr. Sefer Kurnaz

Istanbul, (2019)

EVALUATION AND VALIDATION OF THE INTEREST OF THE RULES ASSOCIATION IN DATA-MINING

by

Ali Yousif Hasan Hasan

Information Technology

Submitted to the Graduate School of Science and Engineering

in partial fulfillment of the requirements for the degree of

Master of Science

ALTINBAŞ UNIVERSITY

2019

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of .....

Asst. Prof. Dr. Sefer KURNAZ

Supervisor

Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)

Asst. Prof. Dr. Osman N. UCAN	School of Engineering and Natural Scince, Altinbaş University	_____
Asst. Prof. Dr. Sefer KURNAZ	School of Engineering and Natural Scince, Altinbaş University	_____
Asst. Prof. Name SURNAME	School of Engineering and Natural Scince, Altinbaş University	_____

I certify that this thesis satisfies all the requirements as a thesis for the degree of .....

Asst. Prof. Dr. OĞUZ ATA

Head of Department

Approval Date of Graduate School of Science and Engineering: ____/____/____

Asst. Prof. Dr.OĞUZ BAYAT

DEDICATION

First and foremost, I would like to thank Allah Almighty for giving me the knowledge, ability and opportunity to undertake this research study and to persevere and complete it satisfactorily. Heartfelt thanks goes to my father and my mother. Every success is a direct consequence of their influence in my life and their love. At the end I have to mention my family for their support and love.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Supervisor Dr. Sefer KURNAZ for all the knowledge and support he provided during my study for the Master Degree and throughout the work to complete this thesis and I have to mention the kindness and the support to all my friends particularly Ali Yarg which did not leave me alone the whole time at the courses and while doing this thesis.

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ali Yousif Hasan Hasan

ABSTRACT

EVALUATION AND VALIDATION OF THE INTEREST OF THE RULES ASSOCIATION IN DATA-MINING

Ali Yousif Hasan

M.Sc., Information Technologies, Altınbaş University

Supervisor: Dr. Sefer KURNAZ

Date: March 2019

Pages: 68

The interesting association rules is a special part of knowledge extraction from data. Apriori's support- and rule-based algo-rithms have provided an elegant solution to the problem of rule mining, but they produce too much rules, selecting some rules of no interest and ignoring rules[1] [2]. interesting. Other measures are needed to complete the support and the confidance. In this paper, we review the main measures proposed in the literature and we propose criteria to evaluate them. We then suggest a validation method that uses the tools of statistical learning theory, including VC - dimension. Given the large number of measurements and the multitude of candidate rules, the interest of these tools is to allow the construction of uniform non-asymptotic terminals for all the rules and all the measurements simultaneously.

Keywords: Association, Rules, Data mining, Validation, Evaluation, Dimensions.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

LIST OF ABBREVIATIONS

ARD	:	Association Rule Database
ADA	:	Automating Data Analysis
DRFFSM1	:	Develop Random Forests with Fuzzy Similarity Measure1
DRFFSM2	:	Develop Random Forests with Fuzzy Similarity Measure2
DRFFSM3	:	Develop Random Forests with Fuzzy Similarity Measure3
DRFPC	:	Develop Random Forests with Pearson Correlation
DRFSS	:	Develop Random Forests with Simple Similarity
DRFLLS	:	Developed Random Forest and Local Least Square
FP	:	Frequency Pattern
FP-KC	:	Frequency Pattern-Knowledge Constructions
IDA	:	Intelligent Data Analysis
KDD	:	Knowledge Discovery in Database
LLS	:	Local Least Square

1. INTRODUCTION

The abilities of the two creating and collection data have been augmenting with a high speed. Providing agents combine the computerization of management , scientific, and governmental transactions; the public practice of numeric cameras, broadcasting tools, plus bar codes for highly commercialized items ; and improvements in data collection tools varying from scanned texts and figures programs to spacecraft remote sensing methods. Moreover, social control of the WWW as general "IS" has overwhelmed us with a large volume of data and information.

This significant increase in reserved data has created an urgent demand for new technological methods and computerized devices that can effectively help in changing the huge mass of data into valuable information and knowledge.

As a consequence, Data is the very relevant assets of today's businesses, and critical investigation of accessible data is important for making better decisions and facing in today's rotating real word. Generally, there are three types of data analysis objections

- Analyzing trial: Data holders are mainly field specialists who know the manners that created the data and what the data describes, but they are weak in analyzing skill and supplies.
- Communication trial: that needs the domain expert to describe the data meaning and make a problematic that the analyst can get help from so he can analyze the data
- Application trial: that needs to change analytical find outs into software which can be attached to execution systems.

Consequently, Automating data analysis concerns the duty of identifying a connection between certain characteristics and designing this connection in a figure.

This research reveals the notions and methods of data mining, a assuring and growing limit in data and information systems and their utilization. Data mining, also commonly led to as information results from data (KDD), is the programmed or suitable descent of models describing knowledge completely saved in huge databases, data warehouses, the Web, additional extensive information depositories, or data streams.

1.1 MAJOR PROBLEMS IN TASKS

1.1.1 Intelligent Data Analysis

Intelligent Data Analysis (IDA) is the most important discipline in real life utilization and computer science. IDA is linked to generating different methods within guide detection and rescue gives learning devices of detecting data guides based on artificial intelligence. When carrying any data analysis and preparing to practice the output in an application, it can be pretended to have the following:

- A problem statement: What is the problem to solve? Do we want to divine or group, (segmentation, clustering), commands or dependences, and so on?
- Choices concerning the answer: If the model can be adjusted to new data, if it is obvious to assume its correctness, fulfilling time, and so on

1.1.2 Knowledge Discovery in Database

KDD is the non-trivial process of recognising, novel, probably beneficial and last logical guides in the data KDD method is an interactive and iterative multi-step process which uses six steps to obtain impressive information based on the same criteria, a brief description of the nine-step KDD process, starting with a managerial step

- Generating an meaning of the application domain, the important prior information, and the objectives of the end-user.
- Building a destination data set; choosing a data set, or concentrating on a subset of variables or data representations, on which invention is to be applied
- Data cleaning and preprocessing: primary processes like raising sound, suggesting on approaches for managing lacking data domains, estimating for time progression information and known variations.
- Data reduction: making the data collection, excluding any characteristics to readjust the assortment to the purpose.
- Choosing the data mining task: determining if the purpose for the KDD method is ranking, clustering, etc.
- Selecting the data mining algorithm(s): collection system(s) to use for exploring for models in the data. it can also involves determining which models and sittings may be suitable

- Data mining: explore models of interest in a appropriate organization of commands, regression or clustering.
- Reading mined models, reasonable income to all the levels 1-7 for more repetition.
- Combining realized information

There is a difference between data and information.Data is a combination of processing information, where information is some larger comprehension that shows us something extra about connections.

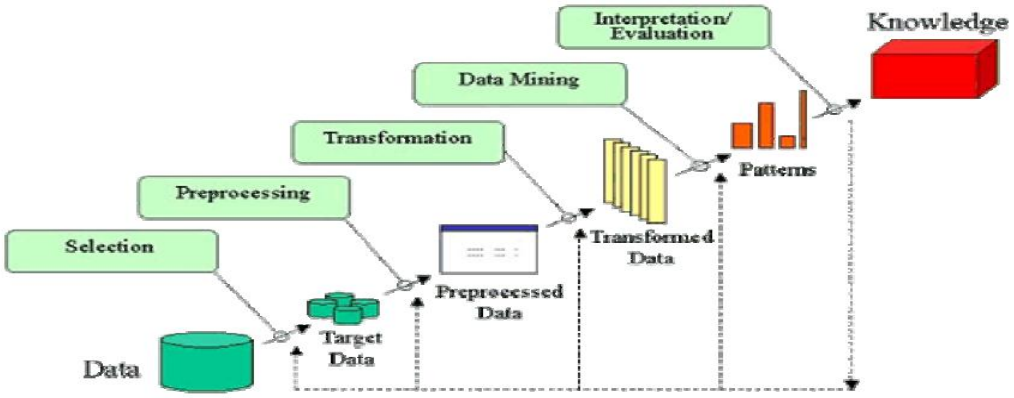


Figure 1.1: KDD Process Main Steps [3]

1.1.3 Data Mining

DM is not just a unique method or system but first a rainbow of different procedures which look for model and connections of data.

Data mining is involved with significant features linked to both database techniques and AI/machine learning mechanisms, and presents an attractive possibility for examining the intriguing correlation between retrieval and inference / reasoning, a significant point regarding the character of data mining.

As a result, DM is a derive of impressive (non-trivial, implicit, before anonymous and possibly serviceable) information from data in extended database. DM IS an interdisciplinar domain, the gathering of a kit of exercises, among them,database system , statistics , machine learning , visualization , and information science. See Fig.1.2.

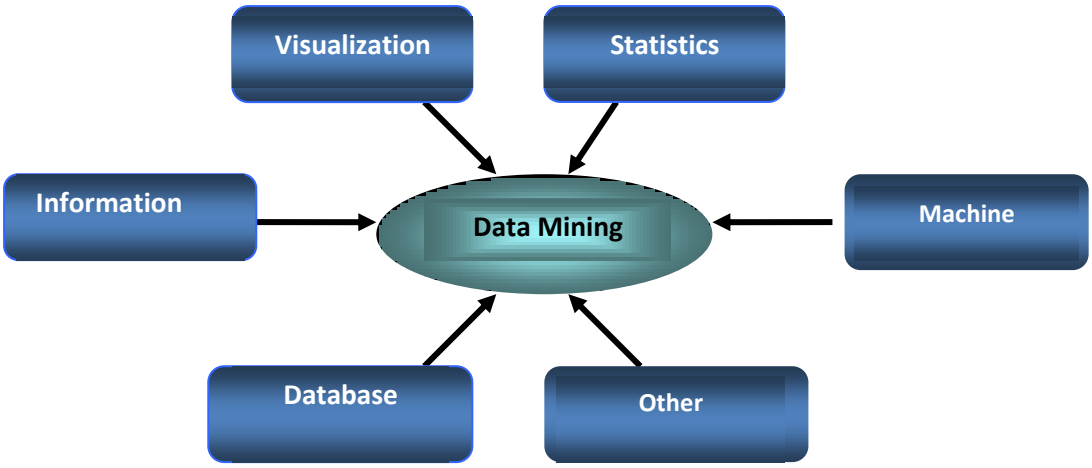


Figure 1.2: Data Mining as a Confluence of Multiple Disciplines

Data mining system is categorized in line with numerous criteria, as follows [4]:

- Division depending on databases mined type.
- Division depending on memorized information
- Division depending on the type of formulas used
- Division depending on the adapted techniques

1.1.4 Principle Component Analysis

Principle Component Analysis (PCA) (also called the Karhunen-Loeve, or K-L, method), searches for k n-dimensional orthogonal vectors that can best be used to represent the data, where $k \leq n$. The original data are thus projected onto a much smaller space, resulting in dimensionality reduction [4].

Add to that, PCA seeks to explain the correlation structures of a set of predicator variables using a smaller set of linear combinations of these variables.

This Linear Combination are called "components", PCA is computationally inexpensive, can be applied to ordered and unordered attributes, and can handle sparse data and skewed data. Multidimensional data of more than two dimensions can be handling by reducing the problem to two dimensions. Principle components can be used as inputs to multiple regression and cluster analysis. In this work, PCA used to solve the dimensions reduction problem.

1.2 THE MAIN STILL OPEN PROBLEMS IN KDD

There are five still uncovered obstacles at this time persist in the KDD as explain in Fig.1.3 (data scarcity problem, find the exact values of the absent values of dataset, convert data mining algorithm from black box to white box form, mathematical scheme model of KDD system and determine the best method to simplify and decrease the knowledge by KDD system). in this work two interrogations will be

discussed. Also as stated in the retell of references [5], [6], [7], [8] , [9].

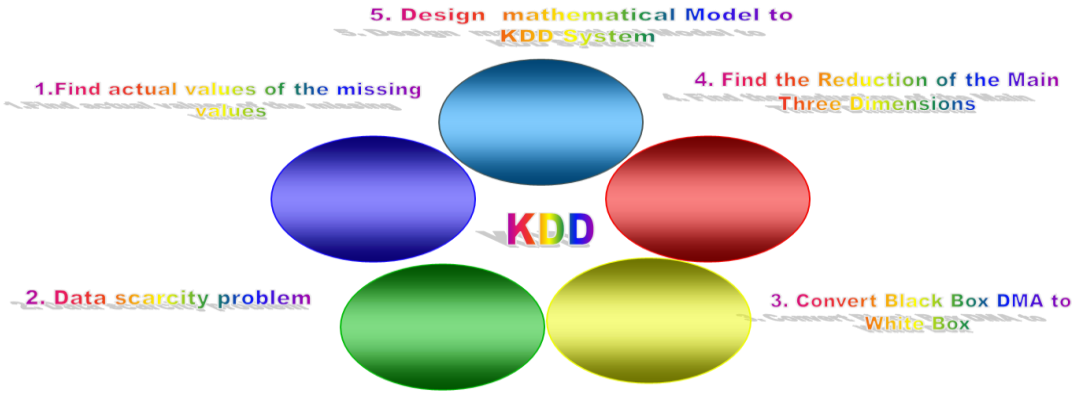


Figure 1.3: Still Open Problems of KDD

1.2.1 Missing Values Problem

Needing values enigma is understood as one of the still uncovered problems. Many answers have been proposed to resolve this problem. First, the easiest resolutions the decrease of the data kit and the exclusion of all units with uncompleted values. This is reasonable when big data kits are ready and needing values happen only in a small percentage of units. Second, a data miner, all with the domain specialist, can manually check units which don't have values and open a logical, presumable value, based on a domain background. This resolution is sincere for few numbers of needing values and almost small data kits. But, if there is no explicit or credible value for every case, the miner is adding noise into the data kit by manually creating a value. Lastly, automated replacement of needed values with few constants [4]. Many resolutions are conceivable here, which are:

- Renew all lacking values with a unique global constant.
- Displace a lacking value with its highlight mean,
- Displace a lacking value with its highlight mean for the distributed class,
- Displace a lacking value with the approaching area from top or bottom.

In intelligent data analysis the research maker is usually involved in learning knowledge which has a imminent power. The central idea is to divine the value that remarkable trait(s) will consider in "the future", based on earlier noted data. But the needed values query forever guide to some of the mastics in found knowledge manner then become very hard to be understandable for the user and in usually the output inadequate intelligent analysis of data kit styles.

1.2.2 Data Scarcity Problem

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 [6]. 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 considered 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.

1.2.3 Black Box Data Mining Problem

Data mining algorithms are "black-box character". as a result of to achieve success, data processing needs accomplished scientific and analytic specialists WHO will make a structure for the analysis and evaluate the created yield. though data treatment will help revealing models and connections, it does not show the user the deserving or significance of those models. In additions, data processing is that whereas it will establish connections between behaviors and/or variables, it doesn't essentially establish a causative relationship [7].

1.2.4 Reduction the Three Main Dimensions

The choice of data representation, and selection, reduction or transformation of features is probably the most important issue that determines the quality of an intelligent data analysis solution. Besides influencing the nature of a KDD algorithm, it can be determined whether the problem is solvable at all, or how powerful the resulting model of KDD. A large number of features can make available samples of data relatively insufficient for mining. In practice, the number of features can be as many as several hundreds. If we have only a few hundred samples for analysis, dimensionality reduction is required in order for any reliable model to be mined or to be of any practical use. On the other hand, data overload, because of high dimensionality, can make some data-mining algorithms non applicable, and the only solution is again a reduction of data dimensions. The three main dimensions of preprocessed data sets, usually represented in the form of flat files, are columns (features), rows (cases or samples), and values of the features [8]. But how one can

combine among these three main dimensions of preprocessed data sets without loss any inform is still open problem as explain in Fig 1.4 [10].

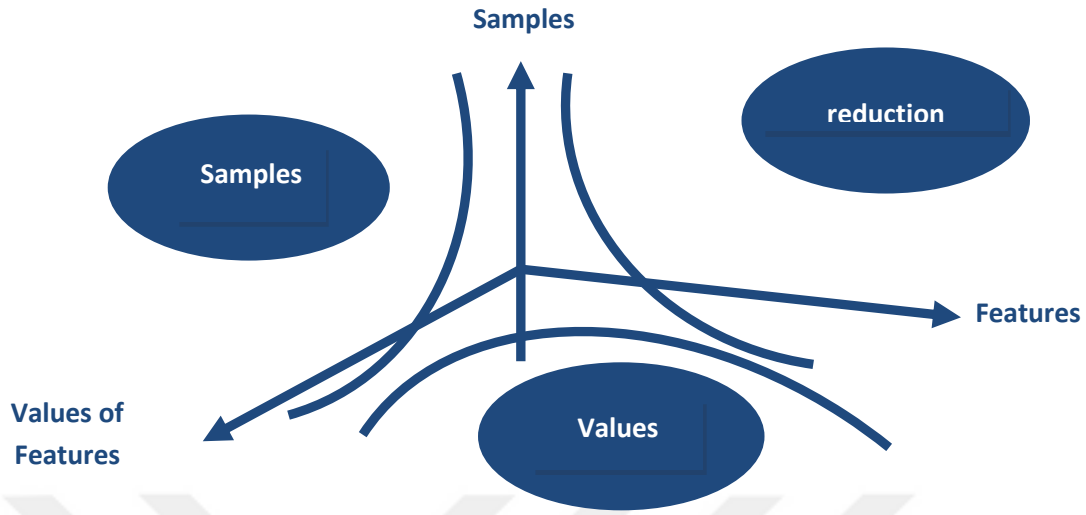


Figure 1.4: Relation between the 3 Dimensions

Dimension on different reduction strategies have the goal of exploitation the correlated schema among the predicted values to achieve subsequent [9]. :

- Cut back the amount of predictor parts.
- Assist make sure that these parts separated.
- Bring a scheme intractability of results.

1.2.5 Design Mathematical Model of KDD System

- A mathematical model is an conceptual, reduced, mathematical construct compared to a piece of truth and built for a special plan. A mathematical model regularly illustrates a system by a kit of variables, a kit of comparisons that organise connections between the variables and a kit of imbalances. The values of the variables can be nearly neutral or integral numbers, Boolean values or strands, for example. But, Why Modelling:
- First device to recognise and interpret complicated systems and phenoms
- As a pursuit to Theory and Experiments, and often Combine them

Mathematical modelling tries to obtain knowledge of science through the application of mathematical model on computers.

- This problem concentrated on mathematical investigation and precise outline plans for arrangements of KDD system. For many reasons:
- Mathematical form more elastic than KDD system.
- The mathematical model can be composed from many mathematical models.

1.3 THESIS OBJECTIVES

The objectives of this work are studying, analyzing and suggesting a solutions for the two problems (still open) in the knowledge discovery in database field, the problems are including decide the best amount of proximal neighbors for needed values and decreases the three main dimensions.

We proposed a system of Knowledge Discovery in Database system to find solutions for these problems. The final result of this system is suggest suitable values for the database having missing values and extracting a classified association rules.

this system represents the integration among preprocessing method, reduction dimension method and data mining to provide best solutions and to establish comprehensive and unified analytical foundations for KDD modeling.

1.4 CONTRIBUTIONS OF THE THESIS

The main contributions of the present work can be synthesized to the following section:

First, the searching capability of building a novel tool called Developed Random Forest and Local Least Square (DRFLLS) to predict the optimal needed values in the databases having missing values. This done by developing random forest algorithm by evolving it seven categories of similarity measures (person similarity coefficient, simple similarity and fuzzy similarity M1, M2, M3, M4, M5)). These measures are sufficient to estimation the optimal number of neighborhoods of missing values in this application. Then, Local Least Square (LLS) has been used to estimate the missing values. We used the Pearson correlation and NRMSE as an accuracy measurement. The optimal number of neighborhoods is associated with the highest value of Pearson correlation and smallest value of NRMSE.

Second, Suggest new algorithm named (FP-KC) "Frequency Pattern-Knowledge C0nstruction "to merge with the options of main element of pre-investigation and other rhythm model extension. This part can be made with a victimization of the 3 criteria (eigen-value, accumulative variability and geological formation plot). It tries to satisfy the goal of reducing the dimensions of dataset explained in Fig. (1.4). The suggest algorithm (FP-KC) generation collection of association rules, before that the generated classified the records using the Adboosting algorithm. The result is a collection of classified association rules which represent the production rule based

knowledge base. The tool that generated classified rules called Miner of obtain accuracy and comprehensible classified association rules (MOACCR).

1.5 LITERATURE SURVEY

We will review and classify the expressive works in this field focusing on the explain similarity and different points between these methods and our work.

1.5.1 Missing Values Processing

Although the matter of estimating the missing values isn't fashionable. Applications of information mining on the ground, even though once there are large amounts of information, the kit of cases with fulfilled knowledge could also be comparatively little. A variable population and additionally next states could be with missing values. a number of method ways settle for needing values and adequately process knowledge to succeed in the ultimate conclusion. different ways need the all values be on the market. The question is whether or not these missing values are often stuffed in throughout knowledge preparation, before the applying of the info mining ways. moreover, variable supervised classification ways like support vector machines, and variable applied math analysis ways like principal element analysis, singular worth decomposition and generalized single value disintegration cannot be practiced with knowledge which have a missing values. Our steered work similar this adds the flexibility to house many varieties of datasets. Jorn, et al., 2009 [11]. Recommend PhyloPars net server give a statistically consistent technique that mixes Associate in Healing unfinished kit of experimental experiences with the sort phylogenesis. This mix to provide an complete kit of views parameter for all classes. It makes upon a growing organic process model, increased with the versatility to handle needed knowledge. The ensuing method makes excellent control of all on the business info to supply measures which will be an adjustment of extent further right than ad-hoc options. this system desires an outsized dataset to figure and it proves triple-crown within the restricted domain of dataset whereas our work makes an attempt to house completely different datasets and realize the optimum estimation of missing values. Anna, et al. use surrogate variables to control needed values at periods the predictor variables. Report the results of Associate in nursing thorough simulation study covering each classification and regression issues below a spread of eventualities, as well as completely. Various missing meriting forming manners besides as complex

relationship arrangements between the variables. Moreover, a great dimensional frame with a high quality of static variables were thought of in every case. The outcomes match the completion of Restricted abstract thought Forests with surrogate variables to it of k-nearest neighbor accusation before meeting. This work plan differs of our prompt add 2 main steps: 1st, this work base on randomization methodology in determines the quantity of nearest neighbors whereas our prompt work base on develop random forest. Second, this work base on k-Nearest neighbor in determines the missing values whereas our prompt work base on native least.

Tyler, et al., 2010 [12]. Present a study which attempts to define the best way to restore missing junction data. When building a consent forecast through mathematical post-processing methods and fix the consequences of displacing the needed data on these post-processing methods. This study differs on our work by focusing on the post-processing method and the effect of missing values on it but not explain how you can handle this problem in details.

We discuss the problem of needed value ascription for E-MAPs, and propose the use of symmetric next neighbor based methods as they give consistently right insinuations beyond various data kits in a flexible manner. This paper is similar to our work by using the local least square as best to determine the needed variables but it differs on our work in the point estimation of the nearest neighbors (because it not handle this problem and left it as still open problem).

The Table 1.1 comparison between various works defined previously based on (Authors, Tools, data set, structure, Method of determining Nearst neighbors)

Table 1.1: Comparison among the Previous Studies for Handle the Missing Values

Authors	Tools	Data set	Structure	Method of determine Nearest neighbors
Jorn, et al., [11]	PhyloPars	Large dataset (web dataset)	State-of-the art evolutionary	Combines an Incomplete set
Tyler, et al. [12]	Statistical Post-Processing	Temperature dataset	Replacement Structure	Not Handle

1.5.2 Dimensional Reduction Processing

The data mining process requires highly computations when evolving large data sets. Dimensionality Reducing (the number of attributes or the number of records) reduce this computations. This step is closely related (often dimensional reduction is used as a step in feature extraction), but the goals can differ. Dimensional reduction has a long

history as a method for data visualization, and for extracting key low dimensional features. It is including wavelet transforms and principal components analysis, clustering, sampling. The main Taxonomies of dimension reduction problem are decreasing learning cost, increasing learning performance, reduction of irrelevant dimension, reduction of redundant dimension.

Mahdi, 2005 [13]. Proposes a way that may find information from any info through simple computing systems that include fuzzy kit, neural network rule by making DBRule Extractor system. This work offers a fuzzy c-mean model for attributes agglomeration.

1.5.3 KDD Tasks Processing

We can summarize the main tasks of KDD algorithms by classifying, clustering, predicating and extracting association rules. Many works suggest in this field.

Sankar et al., 2001 [14] presented a method that defined for growing Rough-Fuzzy Multilayer estimation with the modular notion utilizing a genetic algorithm to reach a good network fitting for both division and rule extraction**.**

1.6 THESIS LAYOUT

The current and remain parts of this thesis organization as following:

1. makes a sense of the general concepts to the main tasks of the problem and main still open problems in KDD. It explains the effect of the computers revolution in increasing the need of intelligent data analysis. Also it introduces the aims of the thesis. The final of that chapter gives an overview of the related work in this field.
2. focuses on the theoretical concepts related to this work: random forest algorithm, classification methods, dimension reduction methods and association rules in data mining.
3. illustrates the advanced method employed in intelligent data investigation for large datasets. Every level in the system has been detailed greatly.
4. represents the implementation of the method and the consequences of the informatics barn.
5. Displays outcomes of this research together with some suggestions for coming researches in this field.

2. KDD AND IDA TOOLS BASED OF THE WORK

The modern world is based on information and therefore the knowledge extracted from it. Therefore, the trendy world considers data-driven machine. We have a tendency to be enclosed by information, numerical and otherwise, that should be analyzed and processed to convert it into data that informs, instructs, answers, or otherwise aids understanding and decision-making. this is often the age of the net, intranets, information warehouses and data marts, and therefore the elementary paradigms of classical information analysis are ripe for amendment.

Knowledge Discovery demonstrates intelligent computing at its best, and is tha the most fascinating and fascinating end-product of knowledge technology. To be able to discover and to extract information from knowledge could be a task that a lot of researchers and practitioners are endeavoring to accomplish. there's lots of hidden information waiting to be discovered, this can be the challenge created by today's abundance of knowledge. KDD is that the method of characteristic valid, novel, useful, and intelligible patterns from giant datasets.

DM is that the step core of the KDD method, involvin the inferring algorithms that explore the information, and find out important patterns (implicit or explicit), that are the essence of helpful data.

2.1 RANDOM FOREST ALGORITHM (RF)

RF algorithm [2] immunes against the over fitting phenomenon and can handle continuous, discrete, and hybrid data. It has many merits such as missing data replacement, clustering analysis, prototype computation and Out-Of-Bag (OOB) estimation of error rate, feature importance score computation, etc. Therefore, RF appears significant in research for classification. It is applied in astronomy, drug discovery, and micro array gene expression data analysis owing to its capabilities to handle large dataset, thousands of features and had fine generalization performance as a combinational classifier.

Non-Parametric Random Forests ensembles of trees grown from bootstrapped training data. For classification, the trees are combined using majority voting with one vote per tree over all the trees in the forest. For regression, forests are created by averaging over trees. Scholars tend to agree that nonparametric ensemble methods, or committee methods', such as Random Forests can offer significant improvements over any single classifier or regression tree[2].

In constructing the ensemble, RFs use two types of randomness. First, in growing any given tree, a random sample of predictors is selected at each node in choosing the best split. A further layer of randomness is added by using a random sample of records for growing each tree in the first place. In theory, using a random sample of records and selecting random predictors at each node should reduce dependence between covariates and thus between the resulting trees as shown in Fig. 2.1. "Random Forests [Brei01] is a classifier consisting of a collection of tree structured classifiers:{h(x, θk), k = 1 .., m}, where the {θk} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x."

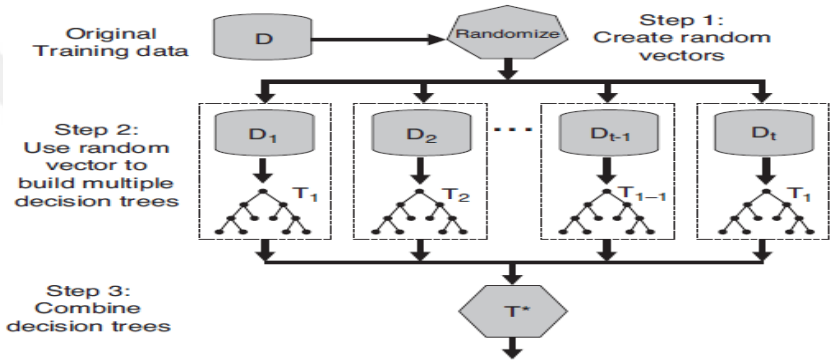


Figure 2.1: Random Forests [15]

There are three main methods of random forests [15]. (which are the RI means randomly select input features, the RC means randomly combination of new features and the BS means best splits at each node)

A. Forest -RI: This model is to randomly choose F input characteristics to split at each link of the resolution tree. As a result, rather of questioning all the possible characteristics, the choice for hole a node is defined from the chosen F features. The power and connection of random forests may depend on the size of F. If F is adequately short, then the trees lead to suit less correlated. On the other side, the strength of the tree classifier leads to grow with a more meaningful number of characteristics F. The number of characteristics is commonly selected as:

$$F = \log_2 d + 1 \quad (2.1)$$

Where d represents the number of input features.

.

A. **Forest -BS:** Generating the random trees is done by chosen one of the F best splits at each node of the decision tree. This type may potentially generate trees that are more correlated than Forest-RI and Forest-RC, unless F is sufficiently large. In this thesis, we will focus on this type of random forest.

There are two traditional methods to handling missing values using random forests:

A. **Use median value:** Random forest can be used to estimate the missing values. For the data missing values at first we assigned the median of all values in the same class .Then, the data vectors are run on the forest, and missing data is re-estimate based on pairs that share the same terminal leaves with this pair. This process can be iterated until estimates converge.
B. **Surrogate Splitters:** compute the association between the primary splitter selected for a node and all other predictors including predicators are not considered as a candidates for the split. If the value of the primary predictor variable is missing for a row. We use the best surrogate splitter whose a value is known for the row.
C. **Random forests:** are an effective tool in prediction. Because of the Law of Large Numbers they do not overfit. Injecting the right kind of randomness makes them accurate classifiers and regressors. Furthermore, the framework in terms of strength of the individual predictors and their correlations gives insight into the ability of the random forest to predict. Using out-of-bag estimation makes concrete the otherwise theoretical values of strength and correlation. In this work, we attempt exploiting this point to estimate the optimal number of neighbors of the missing value through developing RF and using different types of similarity functions inside the random forest.

2.2 DIMENSION REDUCTION METHOD

The use of too many predictor variables to model the relationship with a response variable can unnecessarily complicate the interpretation of the analysis and violates the principle of parsimony. One should consider keeping the number of predictors to a size that could easily be interpreted. Also, retraining too many variables may lead to overfitting, in which the generality of the findings is hindered because the new data do not behave the same as the training data for all the variables.

Further, the analysis solely at the variable level might miss the fundamental underlying relationships among predictors. For example, several predictors might fall naturally

into a single group (a factor or a component) that addresses a single aspect of the data [9].

Dimension reduction methods have the goal of using the correlation structure among the predictor variables to accomplish the following:

- Reduce the number of predictor components
- Help ensure that these components are independent
- Provide a framework for interpretability of the results.

There are several dimension reduction methods such as

- Principal components analysis
- Factor analysis
- User defined composites.

In this work, we deal with Principal components analysis (PCA). Where, PCA seeks to explain the correlation structure of a set of predictor variables using a smaller set of linear combinations of these variables. These linear combinations are called components. For more detail see [9].

The criteria used for deciding how many components to extract are as in the following:

- Eigenvalue criterion
- Proportion of variance explained criterion
- Minimum communality criterion
- Scree plot criterion

The eigenvalue criterion states that each component should explain at least one variable's worth of the variability, and therefore the eigenvalue criterion states that only components with eigenvalues greater than 1 should be retained. For the proportion of variance explained criterion, the analyst simply selects the components one by one until the desired proportion of variability explained is attained. The minimum communality criterion states that enough components should be extracted so that the communalities for each of these variables exceed a certain threshold (for example 50 %). The scree plot criterion is this, the maximum number of components that should be extracted is just prior *to* where the plot begins to straighten out into a horizontal line.

2.2.1 Factor Rotation

Factor loadings are analogous to the component weights in principal components analysis and represent the correlation between the ith variable and the j th factor. To assist in the interpretation of the factors, factor rotation may be performed. Factor rotation corresponds to a transformation (usually, orthogonal) of the coordinate axes,

leading to a different set of factor loadings. Often, the first factor extracted represents a "general factor" and accounts for much of the total variability. The effect of factor rotation is to redistribute the variability explained among the second, third, and subsequent factors.

Three methods for orthogonal rotation are quartmax rotation, varimax rotation, and Equifax rotation [9].

- **a.** Quartmax rotation tends to rotate the axes so that the variables have high loadings for the first factor and low loadings thereafter.
- **b.** Varimax rotation maximizes the variability in the loadings for the factors, with a goal of working toward the ideal column of zeros and ones for each variable.
- **c.** Equimax rotation seeks to compromise between the previous two methods. Oblique rotation methods are also available in which the factors may be correlated with each other.

.2.3 CLASSIFICATION METHODS

This section presents techniques for improving classification accuracy by aggregating the predictions of multiple classifiers. These techniques are known as the ensemble or classifier combination methods. An ensemble method constructs a set of base classifiers from training data and performs classification by taking a vote on the predictions made by each base classifier.

In general, there are two necessary conditions for an ensemble classifier to perform better than a single classifier: (1) the base classifiers should be independent of each other, (2) the base classifiers should do better than a classifier that performs random guessing.

The logical view of the ensemble method is presented in the Figure 2.2. The basic idea is to construct multiple classifiers from the original data and then aggregate their predictions when classifying unknown examples. The ensemble of classifiers can be constructed in many ways

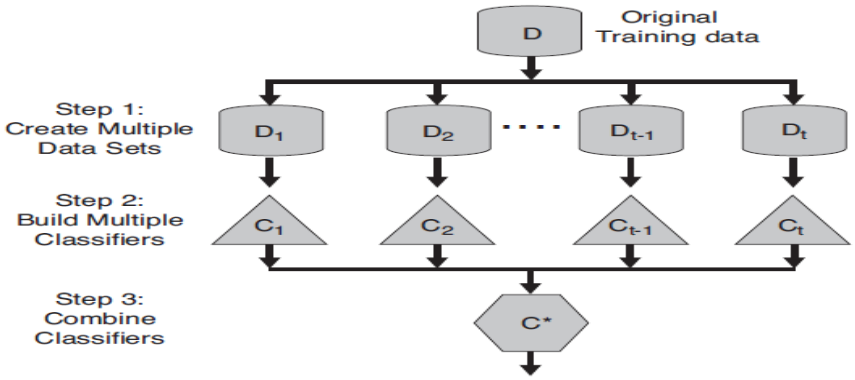


Figure 2.2: A logical view of the ensemble learning method [15], [16]

By manipulatin the training set. In this approach, a multiple training sets are created by resampling the original data according to some sampling distribution. A classifier is then built from each training set using a particular learning algorithm .Bagging and boosting are two examples of ensemble methods that manipulate their training sets.

This work base on using boosting learning algorithm inclassificationthe associatio rules,where use that rules as examples of training set.

We can explain the main classification methods as follow:

A. By manipulating the input features. In this approach, a subset of input features is chosen to form each training set. The subset can be either chosen randomly or based on the recommendation of domain experts.

B. By manipulating the class labels. This method used when the number of classes is sufficiently large. The training data is transformed into a binary class problem by randomly partitioning the class labels into two disjoint subsets A_0 and A_1. Training examples whose class label belongs to the subset A_0 are assigned to class 0, while those that belong to the subset A_1 are assigned to class 1. An example of this approach is the error-correcting output coding method.

C. By manipulating the learning algorithm. Many learning algorithms can be manipulated in such a way that applying the algorithm several times on the same training data may result in different models. For example, an artificial neural network can produce different models by changing its network topology or the initial weights of the links between neurons.

Algorithm 2.1: General Procedure for ensemble Method

Input: dataset(set of records)

Output: classified dataset

Set D the original training data, k the number of base classifiers and T is the test data.

For i = 1 to k do
Create training set, D_i from D.
Build a base classifier C_i from D_i.
End for
For each test record x ∈ T do
$C^*(x) = Vote(C_1(x), C_2(x), \ldots, C_k(x))$
End for

2.3.1 Boosting

Boosting is an iterative procedure used to adaptively change the distribution of training examples so that the base classifiers will focus on examples that are hard to classify. Boosting assigns weights to each training example and may adaptively change the weight at the end of each boosting round. The weights assigned to the training examples can be used in the following ways [Sama07b], [Jiaw06]:

- They can be used as a sampling distribution to draw a set of bootstrap samples from the original data.
- They can be used by the base classifier to learn a model that is biased toward higher-weight examples.

Over the years, several implementations of the boosting algorithm have been developed. These algorithms differ in terms of:

- How the weights of the training examples are updated at the end of each boosting round.
- How the predictions made by each classifier are combined.

In this work, we use Adaboost algorithm that deal with substantial data quality issues, decide how a model should be constructed, and select records and some all other features.

In addition, we using Error Measures to validation from Predictor Model "How can we measure predictor Model accuracy?" Let D^T be a test set of the form (X1, y1), (X2,y2), … , (Xd, yd), where the Xi are the n-dimensional test tuples with associated known values, yi, for a response variable, y, and d is the number of tuples in D^T . Since predictors return a continuous value rather than a categorical label, it is difficult to say

exactly whether the predicted value, y0i, for Xi is correct. Instead of focusing on whether y0i is an "exact" match with yi, we instead look at how far off the predicted value is from the actual known value. Loss functions measure the error between yi and the predicted value by the model, y0i. The most common loss functions are: Based on the above, the test error (rate), or generalization error, is the average loss over the test set. Thus, we get the following error rates.

A. Maximum Error
B. Root Mean Squared Error (RMSE)
C. Mean Squared Error (MSE)
D. Mean Absolute Error (MAE)
E. Mean Absolute Percentage Error (MAPE)

The mean squared error exaggerates the presence of outliers, while the mean absolute error does not. If we were to take the square root of the mean squared error, the resulting error measure is called the root mean squared error. This is useful in that it allows the error measured to be of the same magnitude as the quantity being predicted. Sometimes, we may want the error to be relative to what it would have been if we had just predicted y, the mean value for y from the training data, D. That is, we can normalize the total loss by dividing by the total loss incurred from always predicting the mean.

This work using these measures for both training datasets and testing dataset not that only but add information gain, which is a well-known attribute-quality measure in the data mining and machine learning literature. In particular, the information gain for a given attribute can be computed in a single scan of the training set [8].

$$\text{entropy}(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{2.2}$$

$$Information(Attribute) = \sum_{i=1}^{m} \frac{Fre.SubAttribute}{TotalNo.Of\ \text{Re}cord} * Entropy \tag{2.3}$$

$$Information\ \ Gain = IBS - IAS \tag{2.4}$$

Where; IBS mean the Information before Splitting and IAS mean the Information after Splitting.

2.4 ASSOCIATION RULES IN DATA MINING

Association rules were first introduced by Agrawal et al in 1993 as a means of determining relationships among a set of items in a database [8] . Association rules, like clustering, are a form of unsupervised learning and have been applied to many fields such as retail business, web mining, and text mining. Although numerous applications of association rules involve categorical data, the rule-mining methods available to an analyst can be applied to numerical data as well. The most challenging part of association rule inference involves finding sets of items which satisfy specific criteria, and in turn are used to infer the rules themselves. The reader should note that the literature on association rules often contains algorithms that do not infer the actual rules, but instead creatively find the desired item sets upon which the rules are based

2.4.1. Motivation

Data mining tasks typically involve the analysis of data whose inherent relationships may be obscured by the quantity of data or high dimensionality. Association rules, as a form of unsupervised learning, can be used to extract these relationships so that an analyst can make informed decisions based on the available data given that databases can be quite large, efficient algorithms for mining association rules are required to maximize the quality of inferred information and at the same time minimize the computation time.

2.4.2 Categorize of Frequency Pattern Mining

Frequent pattern mining can be categorized in many different ways according to various criteria, as in the following [4]:

- **A.** Based on the completeness of patterns to be mined, categories of frequent pattern mining include mining the complete set of frequent item sets, the closed frequent item sets, the maximal frequent item sets, and constrained frequent item sets.
- **B.** Based on the levels and dimensions of data involved i the rule, categories can include the mining of single-level association rules, multilevel association rules, single-dimensional association rules, and multi dimensional association rules.
- **C.** Based on the types of values handled in the rule, the categories can include mining Boolean association rules and quantitative association rules.

2.4.3 FP-growth Algorithm

FP-growth is a method of mining frequent item sets without candidate generation. It constructs a highly compact data structure (an FP-tree) to compress the original transaction database. Rather than employing the generate-and-test strategy of Apriority-like methods, it focuses on frequent pattern (fragment) growth, which avoids costly candidate generation, resulting in greater efficiency.

Add to that, FP-growth Algorithm can be define as powerful computational tools in a generation association rules compare with A priori algorithm. It is based on FP-tree.

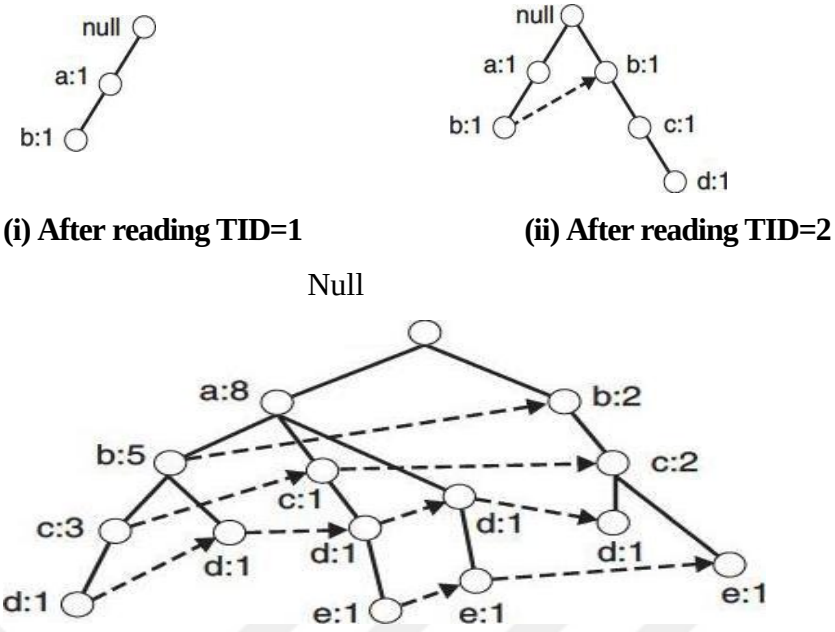
FP-Growth adopts a divide and conquer strategy by (1) compressing the database representing frequent items into a structure called FP-tree (frequent pattern tree) that retains all the essential information and (2) dividing the compressed database into a set of conditional databases, each associated with one frequent item set and mining each one separately. It scans the database only twice. In the first scan, all the frequent items and their support counts (frequencies) are derived and they are sorted in the order of descending support count in each transaction. In the second scan, items in each transaction are merged into an FP-tree and items (nodes) that appear in common in different transactions are counted. Each node is associated with an item and its count.

Nodes with the same label are linked by a pointer called a node-link. Since items are sorted in the descending order of frequency, nodes closer to the root of the FP tree are shared by more transactions, thus resulting in a very compact representation that stores all the necessary information. Pattern growth algorithm works on FP-tree by choosing an item in the order of increasing frequency and extracting frequent item sets that contain the chosen item by recursively calling itself on the conditional FP-tree, that is, FP-tree conditioned to the chosen item.

FP-growth is an order of magnitude faster than the original Apriority algorithm.

FP-growth is a seminal algorithm proposed in this thesis for mining frequent item sets for association rules.

Items	TID
{a,b}	1
{b,c,d}	2
{a,c,d,e}	3
{a,d,e}	4
{a,b,c}	5
{a,b,c,d}	6
{a}	7
{a,b,c}	8
{a,b,d}	9
{b,c,e}	10



(i) After reading TID=1 **(ii) After reading TID=2**

(iii) After reading TID=10

Figure 2.3: Construction of an FP-tree [15]

A data kit that includes 10 activities and 5 things. each link inside the tree carries the name of the associate article concurrently with a bar that confirms the umber of activities planned into the given track. Initially, the FP-tree comprises only the support joint defined by the fantastic image. The FP-tree will be increased later as in [15] :

The data kit is treated previously to run out the post count of every article. rare things are dropped, whereas many things are ordered in reducing support numbers. For the dataset displayed in Fig. 3.3, that the most common article accompanied by b, c, d, and e.

1. The algorithmic habit gets another rise above the knowledge to build the FP-tree. When reading the first dealing, the nodes tagged as a and b are created. A path is then fashioned from null —> a —> b to cipher the dealing. each node on the trail encompasses a frequency count of one.
2. once reading the second dealing, , a replacement set of nodes is created for things b, c, and d. A path is then shaped to represent the dealing by connecting the nodes null —> b —> c —> d. each node on this path conjointly features a frequency count up to one. though the primary 2 transactions have AN item in common, which is b, their methods are disjoint as a result of the transactions don't share a typical prefix.

3. The third dealing, shares a typical prefix item (which is a) with the primary dealing. As a result, the trail for the third dealing, null —> a —> c —> d —> e, overlaps with the trail for the primary dealing, null —>a —> b. thanks to their overlapping path, the frequency count for node a is incremented by 2, whereas the frequency counts for the freshly created nodes, c, d, and e, are adequate to one.
4. This program remains each collection activity has been planned into one in all the programs provided inside the FP-tree. The ensuing FP-tree once expressing all the activities are displayed at the underside of Fig. 2.3.

In this research, we perform to recommend the strengthening of the FP-Growth data processing method to build up a different tool to search out the connection rules known as Frequency Pattern-Knowledge Constructions (FP-KC), principally for various purposes as defined inside the next.

3. DESIGN OF THE SYSTEM

The main idea of this work generated from studying main still challenges in the KDD fields and analyzes it to find solutions of two of these problems as explained in Figure 1.3.

In this system we aim to estimation the optimal number o nearest neighbors of the missing values then estimation the missing values and extracting the knowledge base in form of classified production rules from bioinformatics wearhous.

Our suggested system consists of three process stages, each one processed to handle one of the main challenges in KDD.the system is integration among preprocessing method, Classification method, reduction dimension method and data mining to provide suitable solutions and to establish comprehensive and unified analytical foundations for KDD modeling. Figure 3.1. shows the block diagram of the system.

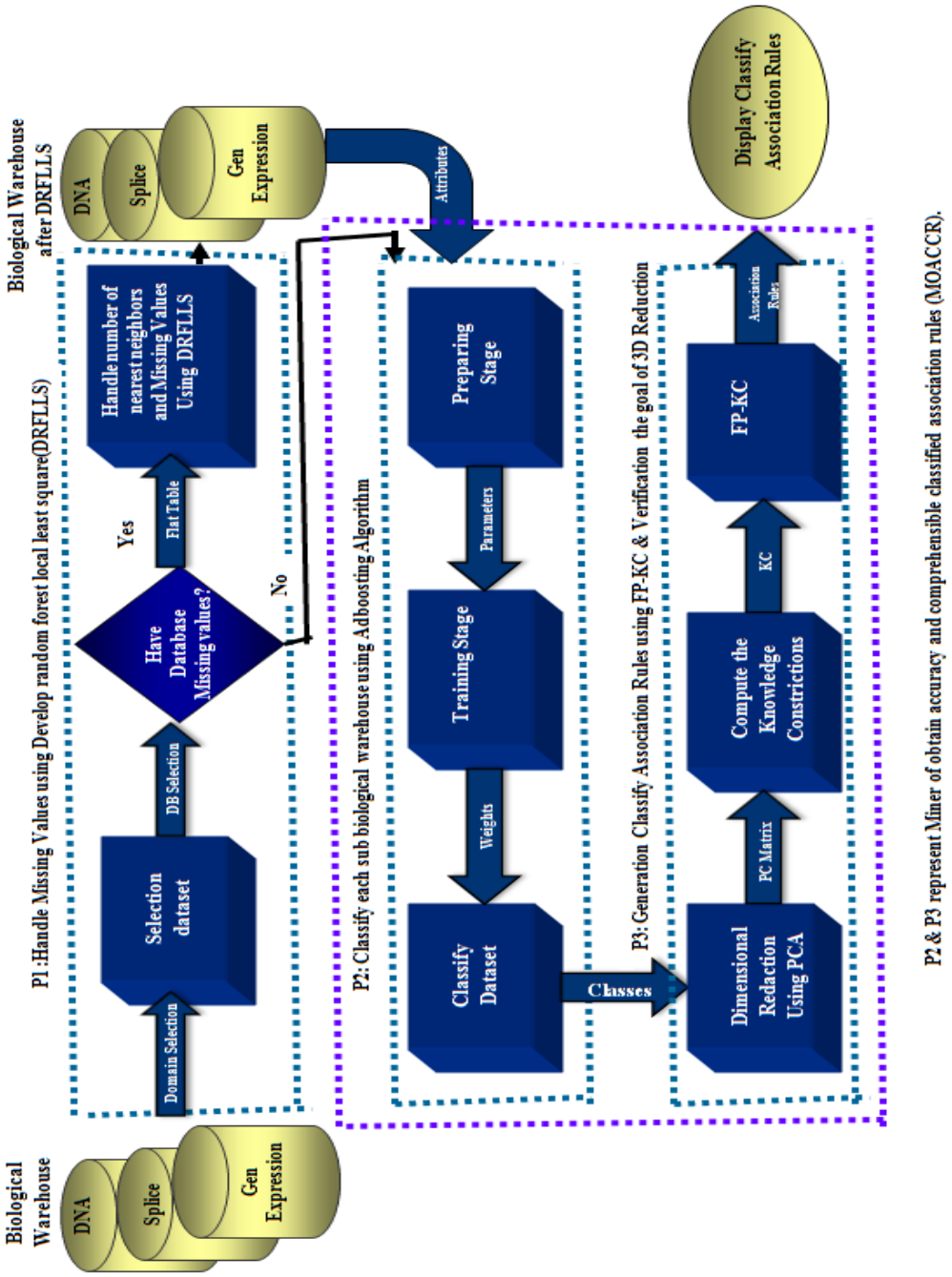


Figure 3.1: Block Diagram of System

3.1 SYSTEM STAGES

3.1.1 Handle Missing Values Problem using DRFLLS

In intelligent data analysis we are often interested in discovering knowledge which has a certain predictive power. The basic idea is to predict the value that some attribute(s) will take on "the future", based on previously observed data. But the missing values problem always lead to some of mastic in discovered

knowledge process then become very difficult to be comprehensible for the user and in many times result incomplete intelligent analysis of dataset behaviors. Therefore, this work presents a novel tool called developed random forests local least square (DRFLLS) to estimate number of neighbors and to find the optimal estimation of missing values. The task of the generated tool divided into three parts:

The first part is to generate k similarity records depending on seven different measures of similarity; each of these measures uses new correlation function of the random forests. As a result, this part generates seven different values of neighbors (i.e., solve the select k nearest neighbor problem).

The second part takes the different values of k results from pervious part and apply LLS to estimation the missing values. This part generates seven values for each missing value based on number of similarity measures.

The third part evaluates the seven estimation values and find the optimal one by applying two types of evaluation measures. The two types are Pearson correlation between the predicted and actual interactions and normalized root mean squared error (NRMSE) measure. Figure 3.2 shows the structure of DRFLLS. Then, we can evaluate which of the similarity measures gives the optimal number of k depending on the important rate of that measure.

In this section, estimating the number of nearest neighbors of the missing value and presenting the similarity measures will be presented in subsection A. Subsection B will focus on how to estimate the missing values. Subsection C evaluates the obtained results.

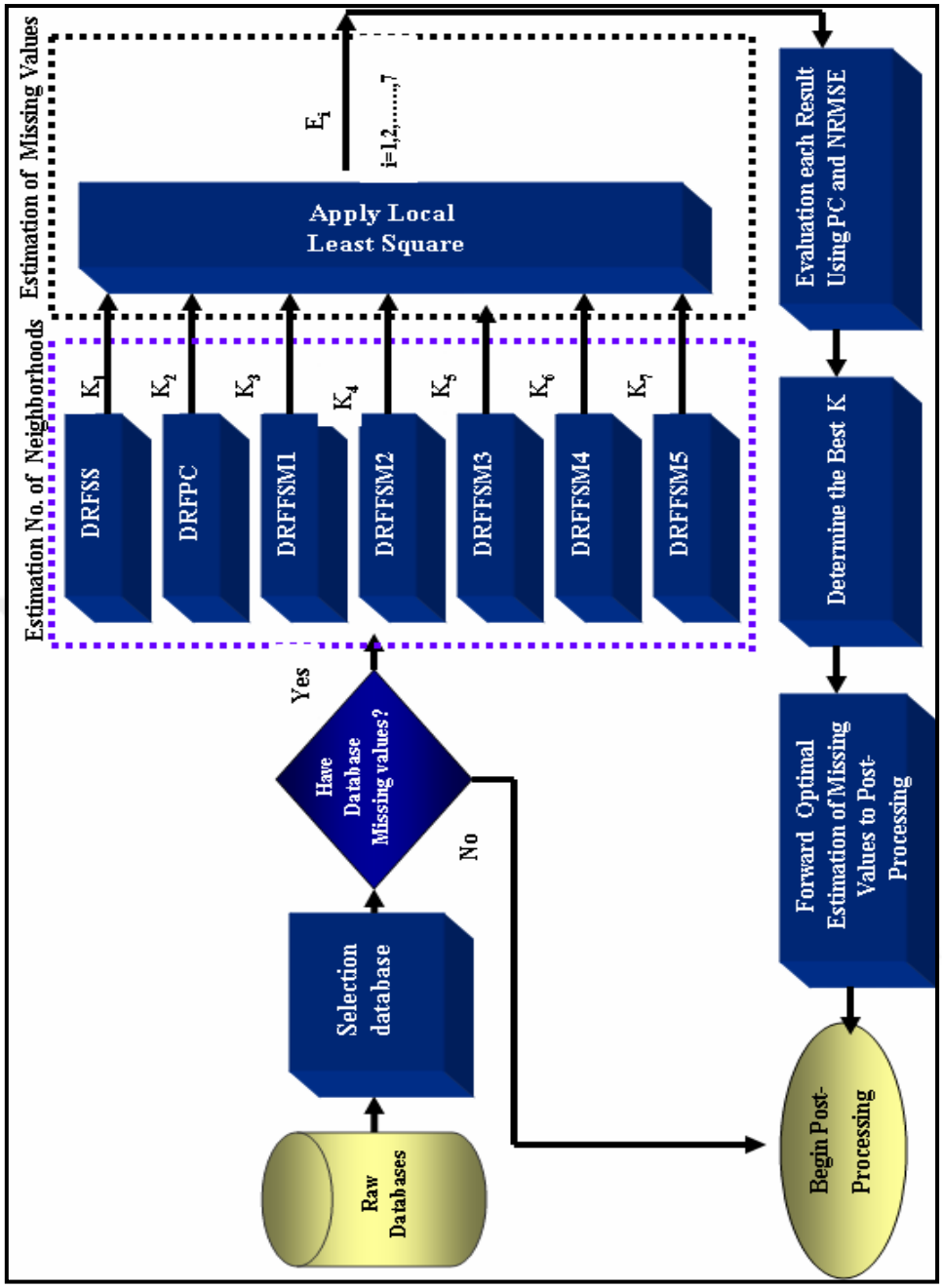


Figure 3.2: Structure of a Novel Tool DRFLLS

Algorithm DRFLLS to Handle Missing Values

Input: Database have a missing values

Output: Database without missing values

- *Step1: Set parameters; number of bootstrap samples, Max no of trees, Max No. of level, Mi no of node, no of terminal node, no of epochs*
- *Step2: Call Build Tree Procedure*
- *Step3: Estimation Number of Nearest Neighbors using (DRFSS, DRFPC, RFFSM1,*

DRFFSM2, DRFFSM3, DRFFSM4, DRFFSM5)

•*Step4: Estimation Missing Values using LLS*

•*Step5: Validation of Results base on Pearson Correlation and NRMSE*

•*Step6: End DRFLLS Algorithm.*

Procedure Build Tree (Grow an un-pruned tree on training records)

While number of records in training set< >Null
Do
read a new record
pass it down the tree
if it not reaches a terminal node
if first record at this node
randomly choose n attributes
find intervals for each of the n attributes update counters
if node has seen n_m in records
if Similarity measure test is satisfied
save node split attribute
save corresponding split value
If no more records in the training set
if node records are highest similarity
Take average response from all individually trained trees
assign the average response to number of nearest neighbor
Else
save best split attribute seen so far
save corresponding split value
End while

A. Estimation The Number of Nearest Neighbors

One of the important problems in estimation missing values methods is how to select the optimal number of nearest neighbors of the missing values (the local nearest base). Here is an overview of the DRF accustomed acquire the best variety of nearest neighbors. Assume the total knowledge set consists of N records.

• *Step 1:* Take a random sample of N records from the info set with replacement (this is termed bagging). Some records are going to be elite over once, et al won't be elite. On average, concerning 2/3 of the rows are going to be elite by the sampling. The

remaining 1/3 of the rows area unit referred to as the out of bag (OOB) rows. a replacement random choice of rows is performed for every tree created.

• *Step 2:* Using the selected rows in step 1, construct a decision tree. Build the tree to the maximum size, and do not prune it. As the tree is built, allow only a subset of the total set of predictor variables to be considered as possible splitters for each node. Select the set of predictors to be considered as a random subset of the total set of available predictors.

• *Step 3:* Repeat steps 1 and 2 a large number of times constructing a forest of trees. Until when reach to total number of trees determined by user.

• *Step 4:* To sign a row runs the row through each tree in the forest and record the predicted value (i.e., terminal node) that the row ends up in (just as you would score using a single-tree model). For a classification analysis, use the predicted categories for each tree as votes for the best category, and use the category with the most votes as the predicted category for the row.

• *Step 5:* End.

There are two methods to compute the error algorithm:

1. To measure the generalization error of the DRF, the out of bag rows for every tree square measure run through the tree and therefore the error rate of the prediction is computed. The error rates for all of the trees within the forest square measure then averaged to relinquish the generalization error rate for the whole forest. There square measure many benefits to the current technique of computing generalization error:(a) all of the rows square measure accustomed construct the model, and none have to be compelled to be command back as a separate check set, (b) the testing is quick as a result of just one forest must be created.
2. The generalization error (E) of a RF of tree classifiers depends on two things

- The correlation among the trees: the smaller the correlation among the trees is that the additional variance canceling takes places because the trees vote and so the smaller the error rate.
- The strength of the every individual tree: the additional correct every tree is, the higher its individual vote, and so the error rate. Where the strength base on the margin function; for more details see [17].

In this work, we use different types of correlation functions or similarity functions among the trees to estimate optimal number of nearest neighbors. For a given forest *f*,

we compute the similarity between two pairs of records (target record and other record) pairs *X1* and *X*j in the following way. For each of the two pairs we first propagate their values down all trees within *f*. Next, the terminal node position for each pair in each of the trees is recorded. Let **Z1** = $(Z_{11}, \ldots, Z_{1L})$ these tree node positions for X_1 and similarly define $\mathbf{Z}_j$. Then the similarity between pair X_1 and X_j is taking different forms apply in parallel as follow:

- *Develop Random Forests with Simple Similarity (DRFSS)*

$$S(X_1, X_j) = \frac{1}{(L-1)} \sum_{i=2}^{L} I(Z_{1i} == Z_{ji}) \tag{3.1}$$

Where I is the indicator function.

- *Develop Random Forests with Pearson Correlation (DRFPC)*

$$S(X_1, X_j) = \frac{1}{(L-1)} \sum_{i=2}^{L} \left(\frac{Z_{1i} - \bar{Z}_1}{\sigma_1}\right)\left(\frac{Z_{ji} - \bar{Z}_j}{\sigma_j}\right) \tag{3.2}$$

Where $\bar{Z}_j$ is the average of values in $\mathbf{X}j_j$ and σ_j is the stander davation of these values. The components of $\mathbf{X}_1$ that correspond to missing values are not considered in computing the coefficients.

- *Develop Random Forests with Fuzzy Similarity Measure 1(DRFFSM1)*

This measure bases on the fuzzy Minkowski distance, and the observation that the smaller the distance between X_1 and X_j, the greater simalarity between them.

$$S(X_1, X_j) = 1 - \left(\frac{1}{(L_1 * L_j)} \left| \sum_{i=2}^{L} Z_{1i} - Z_{ji} \right|^r \right)^{\frac{1}{r}} \tag{3.3}$$

With r ЄN/{0}, for r $\longrightarrow \infty$.

- *Develop Random Forests with Fuzzy Similarity Measure 2 (DRFFSM2)*

$$S(X_1, X_j) = 1 - Max\left|Z_{1i} - Z_{ji}\right| \tag{3.4}$$

Where i=(2,….,L) .

- *Develop Random Forests with Fuzzy Similarity Measure 3(DRFFSM3)*

This measure needs to define the notion of cardinality of fuzzy set. The cardinality of a finite crisp set is given by number of elements in that set. This concept can be

extended to fuzzy sets using the sigma count the sigma count of a fuzzy set X_1 is define as[1] :

$$|X_1| = \sum_{i=2}^{L} Z_{1i}$$

The following similarity measure is based on the notion of cardinality

$$S(X_1, X_j) = \frac{|Z_{1i} \cap Z_{ji}|}{|Z_{1i} \cup Z_{ji}|}$$

$$= \frac{\sum_{i=2}^{L} Min(Z_{1i}, Z_{ji})}{\sum_{i=2}^{L} Max(Z_{1i}, Z_{ji})} \quad (3.5)$$

▪ *Develop Random Forests with Fuzzy Similarity Measure 4 (DRFFSM4)*

$$S(X_1, X_j) = \frac{|Z_{1i} \cap Z_{ji}|}{Max(|Z_{1i}|, |Z_{ji}|)}$$

$$= \frac{\sum_{i=2}^{L} Min(Z_{1i}, Z_{ji})}{Max\left(\sum_{i=2}^{L} Z_{1i}, \sum_{i=2}^{L} Z_{ji}\right)} \quad (3.6)$$

▪ *Develop Random Forests with Fuzzy Similarity Measure 5 (DRFFSM5)*

$$S(X_1, X_j) = \frac{1}{(L_1 * L_j)} \sum_{i=2}^{L} \left[\frac{Min(Z_{1i}, Z_{ji})}{Max(Z_{1i}, Z_{ji})} \right] \quad (3.7)$$

As a result we get seven forests; each one is build basing on one of the above correlation equations and it's providing one predicate value to number of neighbors.

B. Missing Values Estimation

After, we get seven number of nearest neighbors (k_1, k_2, k_3, k_4, k_5, k_6, k_7) by DRF.

Definition 1: Let r1 be a record containing missing values.**r1** contains **n** features and **q** missing values.

We deal with the case in which there is more than one missing value in a record vector by Local Least Square method. In this case, recovering the total number of *q* in any location will be as follow:

• *Step 1:* The k_i-nearest neighbor record vectors for $\mathbf{r}_1$ are founded. In this process of finding the similar records, where the q components of each record have the same location of the missing values in $\mathbf{r}_1$ are ignored.

• *Step 2:* Building a matrix $A \in R^{k*(n-q)}$, where A is 2D matrix in which the number of rows=number of nearest neighbors(k_i) {i=1,…..7}. Number of column=(number of total features (n) - number of columns contain missing values(q)).

• *Step 3:* Building a matrix $B \in R^{k*q}$, Where B is 2D matrix in which the number of rows=number of nearest neighbors (k_i) {i=1,…..7}. Number of columns= number of columns contain missing values (q)).

• *Step 4:* Building a vector $\mathbf{w} \in R^{(n-q)*1}$, Where w is 1D matrix in which the Number of column=(number of total features (n) - number of columns contain missing values(q)).

• *Step 5:* After the matrices *A* and *B* and a vector **w** are formed, the least squares problem is formulated as

$$\min_{x} \| A^{T} x - w \|_2 \tag{3.8}$$

• *Step 6:* The vector $\mathbf{u} = (\alpha_1, \alpha_2, ..., \alpha_q)^T$ of *q* missing values can be estimated as

$$u = \begin{pmatrix} \alpha_1 \\ . \\ . \\ \alpha q \end{pmatrix} = B^T X = B^T (A^T)^t w \tag{3.9}$$

Where $(A^T)^t$ is the pseudo inverse of A^T. The pseudo inverse A^t of *A* can be computed using the following equations

$$\begin{aligned} A^{\dagger} &= [V_1 V_2] \begin{bmatrix} \Sigma_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} [U_1 U_2]^T \\ &= V_1 \Sigma_1^{-1} U_1^T \end{aligned} \tag{3.10}$$

Where $V_1 \in R^{n-1*rank}$, $\sum_{1}^{-1} \in R^{rank*rank}$, $U_1^T \in R^{K*rank}$

The known elements of **w** can be represented by

$$w \simeq x_1 a_1 + x_2 a_2 + \cdots + x_k a_k,$$

Where $\mathbf{x}_i$ are the coefficients of the linear combination, found from the least squares formulation (3.8).

• *Step 7:* As a result, the multiple regressions represent a target record (i.e., record has multi missing value features) as a linear combination of its nearest neighbors from the following: $\text{Target}=x_1b_1+x_2b_2+\ldots\ldots + x_kb_k$ (3.11)

Where b_k represents the *kth* nearest neighbor, and *xk* is the regression coefficient corresponding to that neighbor.

C. Results Evaluation

Imputation accuracy can be measured in several ways. We consider two measures for result evaluations. The first one is the Pearson correlation between the predicted and actual interactions. The second is the NRMSE measured as mentioned in equation bellow [5].

$$NRMSE = \sqrt{\frac{mean\left[\left(ij_{answer} - ij_{guess}\right)^2\right]}{variance\left[ij_{answer}\right]}} \quad (3.12)$$

Where answer denotes the set of far-famed values, and guess denotes the corresponding set of foretold values. a lot of correct imputations can lead to the next correlation score, and a lower NRMSE score.

3.1.2 Classification Datasets using the Adaboost Algorithm

Adaboost data mining algorithm used to find best class for the records, result after handle the missing values using DRFLLS (i.e., find the class base on volt principle) as follow. In addition, we can see the block diagram of that algorithm explain in Figure 3.3.

Input: Database, number of epochs (k), set of Class-labeled training dataset (d)

Output**:** Set of Classes

- *Step1: initialize equal weights for all N records in training database. w = {wj = 1/N | j = 1, 2,. . . , N}*
- *Step2: for i=1 to K do*
- *Step3: Great training set Di by sampling (with replacement) from dataset according to w.*
- *Step4: Train a base classifier Mi on Di.*
- *Step5: Apply Mi to all records in the original training set, D.*
- *Step6: Calculate the weighted error* $error(Mi) = \sum_{j=1}^{d} Wj * error(Xj)$
- *Step7: If error(Mi) > 0.5 then*
- *Step8: Reset the weights for all N records: w = {wj = 1/N | j = 1, 2, . . . ,N}: Go back to Step 4.*
- *Step9: End if*
- *Step10: Compute the weights of classifier Mi ,* $Wi = Log(\frac{1-\text{error(Mi)}}{\text{error(Mi)}})$
- *Step11: Update the weight of each rule and Normalization it.*

$$Wnew = Wj * \frac{\text{error(Mi)}}{1 - \text{error(Mi)}}$$

$$Wnor = \frac{Wnew * \sum_{i=1}^{k} Wold}{\sum_{i=1}^{k} Wnew}$$

- *Step12: End for*
- *Step13: Predication the Class of X*

  C=Mi(X)
- *Step14: End Adaboost Algorithm*

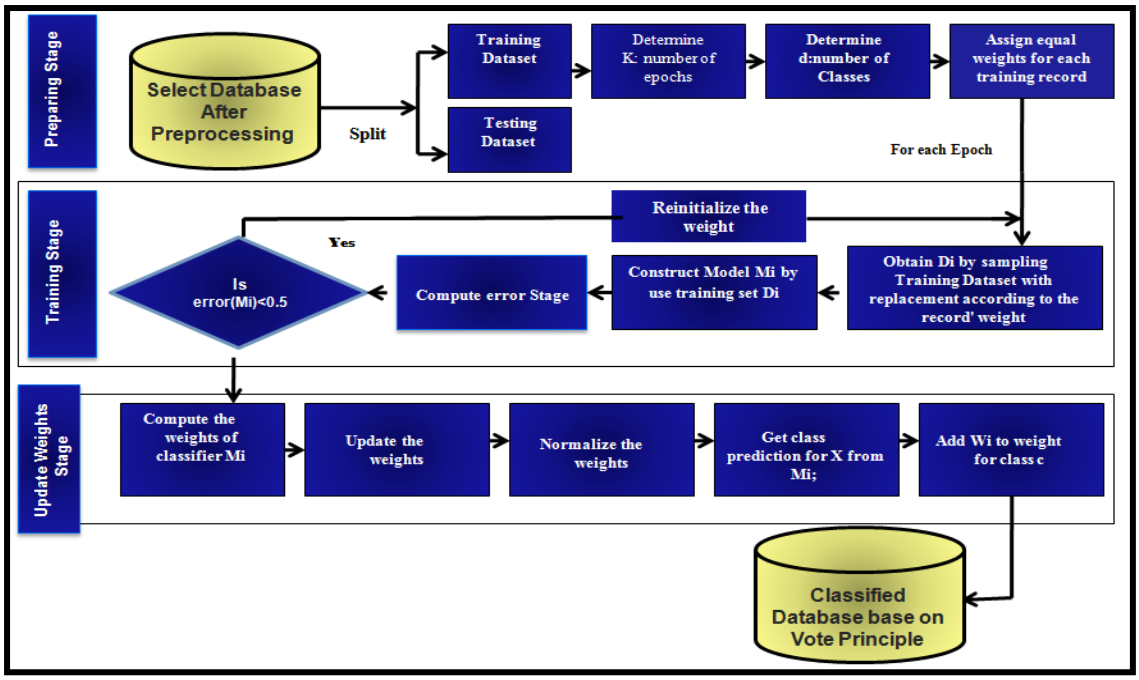


Figure 3.3: The Structure of the Adaboost Algorithm

3.1.3 Handle Dimensions Reduction Problem using PCA and FP-KC

It is not easily to satisfy the relation among the main components of Fig.(1.4) "samples reduction, features reduction and values of feature reduction). But, in this work, we try to establish this relation by using the suitable tools and suggest the FP-KC algorithm to verify this goal. At beginning, we can use analyze these records and find the interest components by using PCA. After that, passing the result to FP-KC algorithm to verify these relations. In order to compute the PCA we can follow the following steps [9]:

Input: Process database

Output: PC-database

• *Step1: Compute the standardized data matrix* $\boldsymbol{Z}$ *=[Z1, Z2, . . . , Zm] base on*

$Zi = (Xi - \mu i) / \sigma ii$ *from the original dataset.*

• *Step2: Compute the Eigen values:*

- *if B be an m × m matrix, and let I be the m × m identity matrix (diagonal matrix with 1's on the diagonal). Then the scalars (numbers of dimension 1 × 1)* $\lambda 1, \lambda 1, \ldots, \lambda m$ *are said to be the Eigenvalues of B if they satisfy* $|B - \lambda I| = 0$.

• *Step3: Compute Eigenvectors:*

- *Let* $\boldsymbol{B}$ *be an m × m matrix, and let* λ *be an Eigenvalues of B. Then nonzero m × 1 vector* $\boldsymbol{e}$ *is said to be an eigenvector of B if* $Be = \lambda e$.

• *Step4: Compute ith principal components:*

- *The ith principal component of the standardized data matrix* $\boldsymbol{Z}$ *= [Z1, Z2, . . . , Zm] is given by* $Yi = \boldsymbol{e}^T i\boldsymbol{Z}$,

• *Step5: End PCA procedure*

where $\boldsymbol{e}i$ *refers to the ith eigenvector (discussed below) and* $\boldsymbol{e}^T i$ *refers to the transpose of* $\boldsymbol{e}i$ *. The principal components are linear combinations Y1, Y2, . . . , Yk of the standardized variables in Z such that (1) the variances of the Yi are as large as possible, and (2) the Yi are uncorrelated. The first principal component is the linear combination* $Y1 = \boldsymbol{e}^T 1Z = e11Z1 + e12Z2 + \cdots + e1m\, Zm$, *which has greater variability than any other possible linear combination of the Z variables. Thus:*

- The first principal component is the linear combination $Y1 = \mathbf{e}^T 1Z$, which maximizes $\mathrm{Var}(Y1) = e^T 1\rho\, e1$.
- The second principal component is the linear combination $Y2 = e^T 2Z$, which is independent of Y1 and maximizes $\mathrm{Var}(Y2) = e^T 2\rho\, e2$.
- The *i*th principal component is the linear combination $Yi = e\_iX$, which is independent of all the other principal components Yj , j < i , and maximizes $\mathrm{var}(Yi) = e^T i\rho\, ei$.

There are several methods to specify a certain association rules. One of these methods depend on finding frequency itemset , such as traditional Apriori, FP-growth are described in rules are given which specifies the attributes that

determine membership of the target. While the goal of this part of thesis is not only determining the association rules but also finding a solution of the three dimension reduction at the same time. Therefore, this part combines between the result of PCA and develop FP-growth called FP-KC. Figure 3.4 explain the block diagram of the FP-KC Algorithm and the main steps of it explain below:

Algorithm FP-KC to Handle Dimensions Reduction

Input**:** Principle Component (PC) Database**,** Set of knowledge construction**,** min-sup

Output: set of association rules

• *Step1:* Construct the FP-tree

- Scan the PCDatabase
- Collect F, the set of frequent items and their support counts
- Sort F in support count descending order as L, the list of frequent items
- Create the root of an FP-tree and label it as "Null"

• *Step2:* Test the Knowledge Constrictions Conditions

For each record in PCdatabase test the KC as follow

- If the record not verification eigenvalue criterion
  Then (remove the record from PCdatabase).
- Else, the record not verify proportion of variance explained criterion
  Then (remove the record from PCdatabase).
- Else, the record not verification The scree plot criterion

Then (remove the record from PCdatabase). Else, Goto Step3.

• *Step3:* Built FP-KC

- For each record verifying Knowledge Constrictions Conditions
- Select and sort the frequent items in records according to the order of L
- Call insert-Tree procedure

• *Step4:* Mining the FP-KC using FP-growth procedure

- If Tree contain single path then

▪ For each combination of nodes CN in path

- Generate pattern CN U α
- Support_count=min_support count of nodes

- Else, if tree contain multi paths

▪ For each $node_i$ in the header of tree

- Generate pattern CN = $node_i$ U α

- Support_count= $node_i$ _support count of nodes
- Construct CN condition pattern then CN conditional $FP\text{-}Tree_{CN}$
- If $Tree_{CN} <> \phi$ Then : go to step 4.

• *Step5:* End FP-KC Algorithm

Procedure of Insert-Tree

• *Step1:* if Tree has child N Then
  - Set i=i+1
  - Else, Create New Node(N)
  - i=1; N.link=Tree
  - End if

• *Step2:* if remind list<> ϕ Then

  - for j=1 to No. of element
  - Call insert-tree(P,N)
  - Next j
  - End If

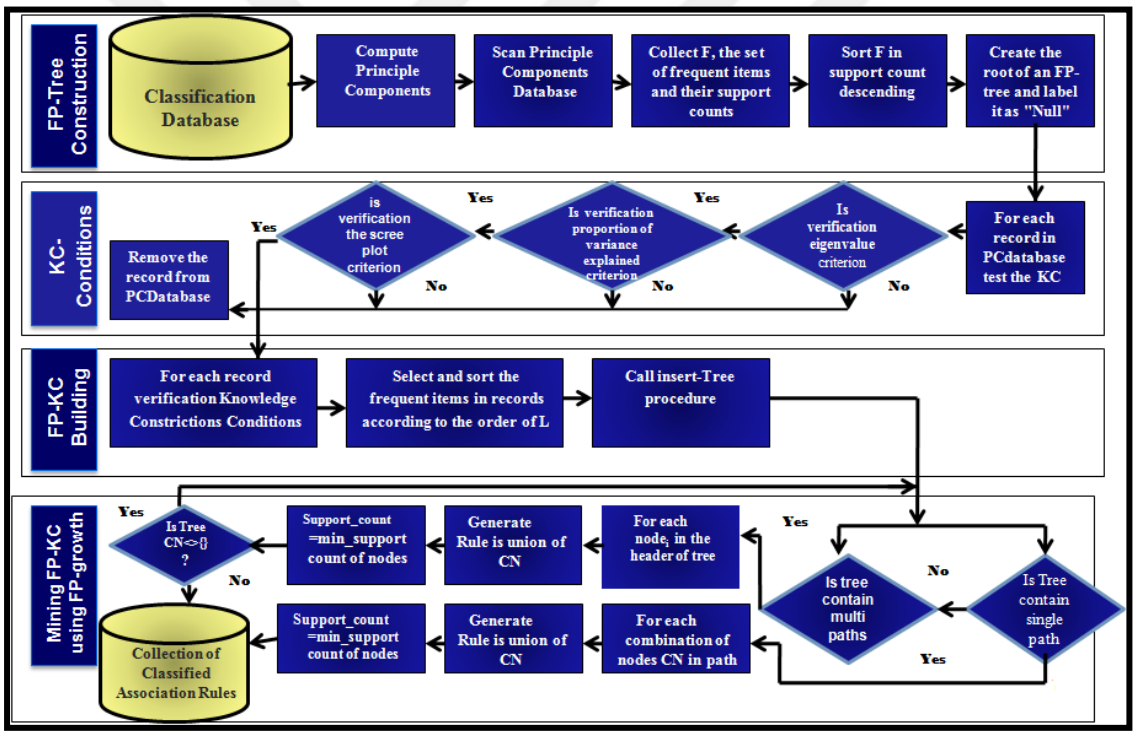


Figure 3.4: The Structure of the FP-KC Algorithm

3.2 SUMMARY

We can summarize the main points in the system as follow:

First: One of the important trends in KDD will be the growing importance of data processing. But this point faces problems similar to those of data mining (High dimensional data, missing values imputation and data integration [18]). As explained in [5], one of the still problems in estimation missing values methods is how to select the optimal number of nearest neighbors of those values. In our

work, we system search the capability to building up a novel tool to estimate missing values called (DRFLLS), which is capable to estimate a given missing values into its various datasets. By developed random forest algorithm through using seven categories of similarity measures were defined. These categories are person similarity coefficient, simple similarity, fuzzy similarity (M1, M2, M3, M4 and M5).

The next cases explain how the modern story device DRFLLS extreme the problems appear in other methods explained in figure 1.1. The next intents can be linked as the current line of that figure:

- Devices : DRFLLS
- Data kit : Biological Warehouse
- Structure: Local and Fuzzy Similarity
- Method of determine Nearest neighbors: Development of Random Forest (Mining Method)
- Method of determine Missing Values: LLS

Second: This work attempts to use one of the ensemble algorithms as technique for improving classification accuracy by aggregating the predictions of multiple classifiers to classification the biological warehouse that it contains from multi databases.

- Tools: Adboosting Algorithm
- Data set: Biological Warehouse
- Structure: Generalization learner
- Method of determine the class: Vote Principle

Third: The purpose of dimension decrease techniques is to practice the relationship construction between the predictor variables to decrease the three principal dimensions (characteristics, units and utility of characteristics). However it is yet as one of the trial in KDD. The next cases reveal how the fresh algorithm FP-KC deals with this dilemma and discover new answers for it. These features relate to figure1.2 . The next features can be joining as a new line of that figure.

- Devices : FP-KC
- Data kit: Biological Warehouse
- Preprocessing: Initial PCA, information structure (eigenvalue criterion, the balance of variety defined pattern and screen area model).
- Result : a kit of Association

4. IMPLEMENTATION OF THE SYSTEM

As we explained previously this work consists of sequence of phases. Each system' phase do one of the require process and then analyzed the results using statically tools.

In this work, the main objectives of the processes is to estimation the number of nearest neighbors of the missing values base on mining method; find the values of missing values base on nearest neighbors ; satisfy the goal of reduction the main 3D and extracting knowledge bases in the form classified association rules .

As we known, a data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data organized in support of management decision making. Several factors distinguish data warehouses from operational databases. Because the two systems provide quite different functionalities and require different kinds of data, it is necessary to maintain data warehouses separately from operational databases. In this chapter, we try to given case study of the Biological warehouse as explained in the following sections. This case study divided into three sub warehouses. The prior information and results of each case are shown.

4.1 EXPERIMENTS ON THE BIOLOGICAL WAREHOUSE

4.1.1 The Results for Discovering the Missing Values using DRFLLS

In this work, we deal with biological warehouse have three types of test databases, two of that suffer from the missing values problem while the third is complete. As explain in Table 4.1.

Table 4.1: Overview of the Biological Dataset Repositorie

DATASET NAME	DESCRIPTION	NO. OF RECORDS	NO. OF FEATURES	NO. OF MISSING VALUES	URL
First	Gene Expression and hybridization array data repository	2000	181	31000	http://archive.ics.uci.edu/ml/machine-learning datasets
Second	Splice junctions are points on a DNA sequence at which superfluous' DNA is removed during the process of protein creation in higher organisms	2979	61	16780	.http://idke.ruc.edu.cn/news/2009/dataset.htm
Third	Extensive collection of microarray data relate of tumor	4339	17	NULL	http://genome-www.stanford.edu/microarry

The number of nearest neighbor estimate by developing random forest is explaining in Table 4.2.

Table 4.2: No. of Nearest Neighbor Estimation by Developed Random Forests

CORRELATION FUNCTION	FIRST	SECOND
DRFSS Eq.(3.1)	30	14
DRFPC Eq.(3.2)	16	**18**
DRFFSM1 Eq.(3.3)	24	15
DRFFSM2 Eq.(3.4)	18	12
DRFFSM3 Eq.(3.5)	23	10
DRFFSM4 Eq.(3.6)	25	26
DRFFSM5 Eq.(3.7)	**28**	22

The best number of nearest neighbor generation by DRF for any dataset is presented in bold font. The above Table(4. 2) explains the dataset have a small rate of missing values given the best estimation to number of nearest neighbors by DRFPC and in second degree by DRFFSM1 when r=4. While if the dataset have high rate of missing values then it gives the best estimation to the number of nearest neighbors by DRFFSM5 and in second degree by DRFFSM3. After estimation the missing observations using LLS, we can compute the accuracy of results by Pearson correlation measure as shown in Table 4.3 and by NRMSE measure as shown in Table 4.4.

Table 4.3: Accuracy, as Measured by Pearson Correlation

CORRELATION FUNCTION	FIRST	SECOND
DRFSS Eq.(3.1)	0.81	0.73
DRFPC Eq(3.2)	0.77	**0.83**
DRFFSM1 Eq.(3.3)	0.68	0.78
DRFFSM2 Eq.(3.4)	0.65	0.61
DRFFSM3 Eq.(3.5)	0.83	0.59
DRFFSM4 Eq.(3.6)	0.74	0.73
DRFFSM5 Eq.(3.7)	**0.88**	0.54

In each column of the Table 4.3, the highest value of the correlation for a listed dataset is shown in clear font. This indicates that corresponding DRF correlation function is the better function for a given dataset.

Table 4.4: Accuracy, as Measured by NRMSE

CORRELATION FUNCTION	FIRST	SECOND
DRFSS Eq.(3.1)	0.72	0.76
DRFPC Eq.(3.2)	0.76	**0.60**
DRFFSM1 Eq.(3.3)	0.80	0.63
DRFFSM2 Eq.(3.4)	0.74	0.73
DRFFSM3 Eq.(3.5)	0.67	0.82
DRFFSM4 Eq.(3.6)	0,69	0.78
DRFFSM5 Eq.(3.7)	**0.60**	0.65

In each column of the Table 4.4. The smallest value of NRMSE for a given dataset is presented in bold font. This means that corresponding DRF correlation function is better function for a given dataset. In addition, the error generated of each

database base on the number of nearest neighbors yield by different types of DRF correlation measures as show in Table 4.5.

Table 4.5: Error Generation base on number of nearest neighbors

DRF Correlation Measures	FIRST		SECOND	
	# Nearest Neighbors	Error Generation	# Nearest Neighbors	Error GENERATION
DRFSS	30	0.063	14	0.174
DRFPC	16	0.073	18	**0.159**
DRFFSM1	24	0.065	15	0.169
DRFFSM2	18	0.071	12	0.194
DRFFSM3	23	0.064	10	0.222
DRFFSM4	25	0.065	26	0.132
DRFFSM5	28	**0.058**	22	0.195

Finally, Table 4.6 explains the values of parameters used in DRF for each dataset and it give the relative importance of each similarity measure.

Table 4.6: Value of each Parameters used in DRF with Relative Important

PARAMETERS	FIRST	SECOND
Total # trees	100	60
Max level of trees	181	61
Max # nodes in tree	362000	194590
RI of DRFSS	1.12	0.96
RI of DRFPC	1.01	**1.38**
RI of DRFFSM1	0.85	1.24
RI of DRFFSM2	0.87	0.84
RI of DRFFSM3	1.23	0.72
RI of DRFFSM4	1.07	0.94
RI of DRFFSM5	**1.46**	0.83

Where RI: mean relative important for each Similarity measures, Max level of trees = No of features of that dataset. Max Number of nodes in tree=No. of Records* No. of Features in each dataset.

In this work, we use different number of tree in the forest lie in the rang [10-120] tree and the error generation from it explain in the Table (4.7) with Figure (4.1).

Table 4.7: Error generation by different number of trees uses in DRF

# Trees	FIRST	Second
10	0.074	0.222
20	0.062	0.174
30	0.063	0.115
40	0.082	0.102
50	0.081	0.091
60	0.086	0.082
70	0.068	0.076
80	0.072	0.074
90	0.068	0.070
100	0.064	0.066
110	0.073	0.065
120	0.075	0.062

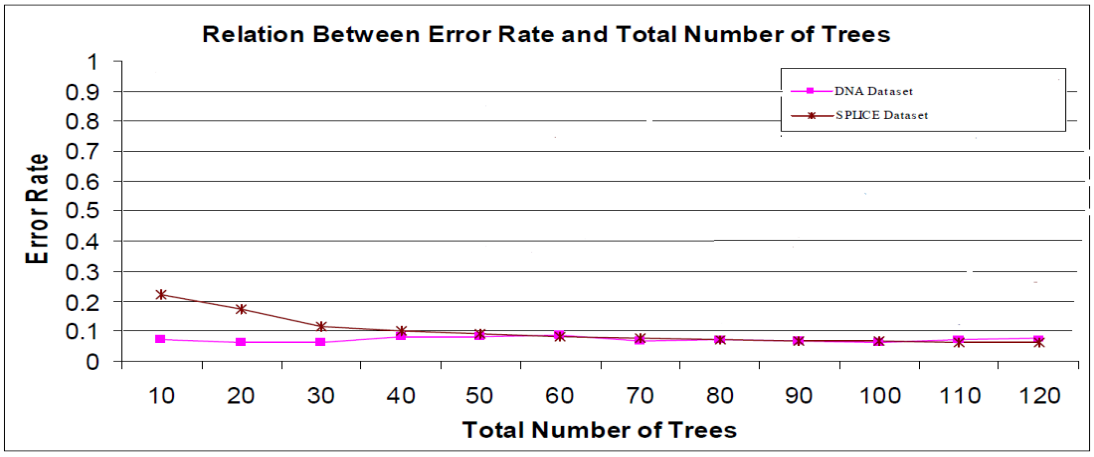


Figure 4.1: Relation between Error Rate and Number of Trees

The main assumption of handle missing vales in that work is processing a original datasets that suffer from many records have missing values in different locations of record but not processing the missing that may be occur in the post processing stages (i.e., in the clustering, association rules and taking decision stages).

In addition, random forest is considered as one of the statistical tools that prove good performance in many fields but by experiments we find the combination between RF and similarity measures to design DRF leads to increase the time complexity and the space complexity. But, on the other hand the results of a novel tool DRFLLS has been proved to be higher estimation accuracy than the traditional estimation methods and gives convenient environment to us.

4.1.2 Classification of Biological Warehouse using Adboosting Algorithm

A. Result of First Sub Warehouse

After, we determine the Main Parameters of Boosting: No of Training samples, No of Testing samples, Max No of Epoch, No of Classes and Assign equal weights for each training samples. We generate multi Classifiers base on the training samples for given classes of all the training samples depend on vote principle. Where the initial values of this stage explaining below:

- The max number of trees=400.
- The minimum error with the training data occurs with 400 trees.
- The minimum error with the test data occurs with 24 trees.
- The minimum point is smoothed by 5 trees.
- The specified minimum number of trees is 10.
- The tree series will be pruned to 24 trees.
- Maximum depth of any tree in the series = 5
- Average number of group splits in each tree = 12.2

For Training Dataset

- Median target value for initial data sample = 3
- Mean target value for initial data sample = 2.2939698
- Mean target value for predicted values = 2.8508714
- Average absolute error for initial data sample = 0.7521527
- Average absolute error after tree fitting = 278.77407
- Variance in initial data sample = 0.6849385
- Residual (unexplained) variance after applying Adaboost model = 0.9684521
- Coefficient of variation = 0.428994
- Normalized mean square error (NMSE) = 1.413926
- Correlation between actual and predicted = 0.486648
- Maximum error = 1.8855464
- RMSE (Root Mean Squared Error) = 0.9840997
- MSE (Mean Squared Error) = 0.9684521
- MAE (Mean Absolute Error) = 278.77407
- MAPE (Mean Absolute Percentage Error) = 55.603193

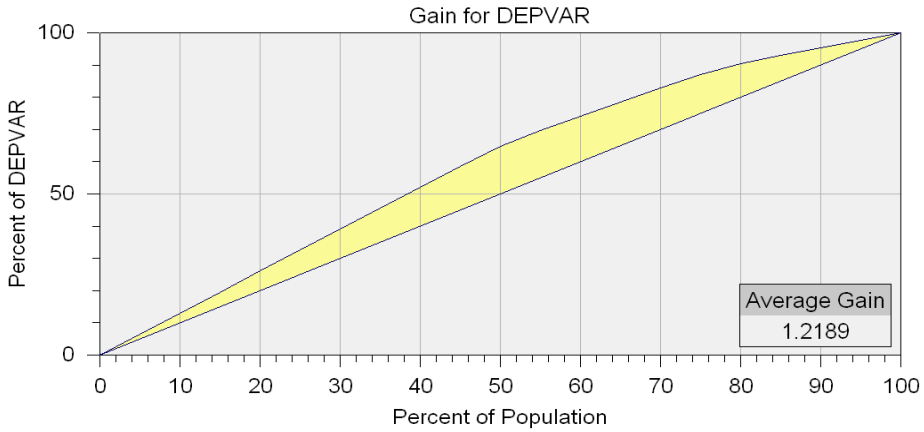


Figure 4.2: Gain Information for DEPVER of the Training Dataset

For Validation Dataset

- Mean target value for initial data sample = 2.29
- Mean target value for predicted values = 2.8458384
- Average absolute error for initial data sample = 0.7526
- Average absolute error after tree fitting = 71.047871
- Variance in initial data sample = 0.6859
- Residual (unexplained) variance after applying Adaboost model = 0.9810701
- Coefficient of variation = 0.432528
- Normalized mean square error (NMSE) = 1.430340
- Correlation between actual and predicted = 0.307521
- Maximum error = 1.8794286
- RMSE (Root Mean Squared Error) = 0.9904898
- MSE (Mean Squared Error) = 0.9810701
- MAE (Mean Absolute Error) = 71.047871
- MAPE (Mean Absolute Percentage Error) = 56.239384

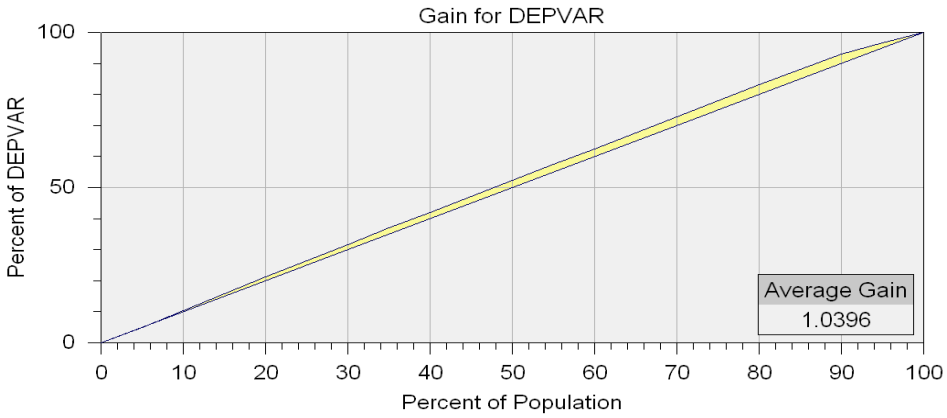


Figure 4.3: Show The Gain Information For DEPVER Of The Testing Dataset

Figure 4.4 show surface of that classes base on the features more important for making the decision

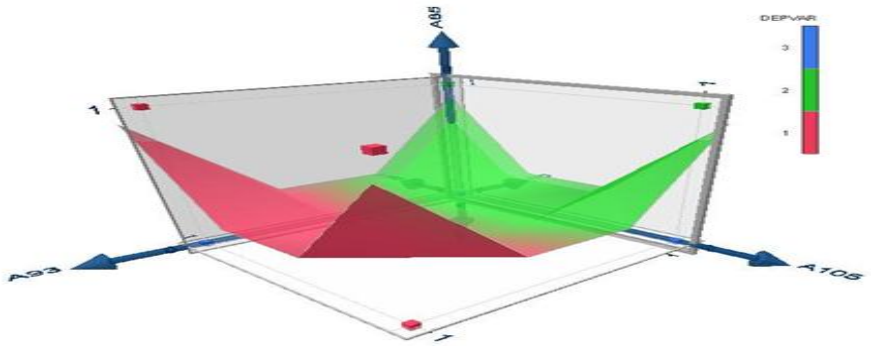


Figure 4.4: Surface of First Sub Warehouse Base on the Main Three Features (A93, A105 and A85)

B. Result of Second Sub Warehouse

After, we determine the Main Parameters of Boosting: No of Training samples= 1583, No of Testing samples= 396, Max No of Epoch, No of Classes and Assign equal weights for each training samples. We generate multi Classifiers base on the training samples for given classes of all the training samples depend on vote principle. Where the initial values of this stage explaining below:

- The max number of trees=400.
- The minimum error with the training data occurs with 398 trees.
- The minimum error with the test data occurs with 366 trees.
- The minimum point is smoothed by 5 trees.
- The specified minimum number of trees is 10.
- The tree series will be pruned to 366 trees.
- Maximum depth of any tree in the series = 5
- Average number of group splits in each tree = 44.4

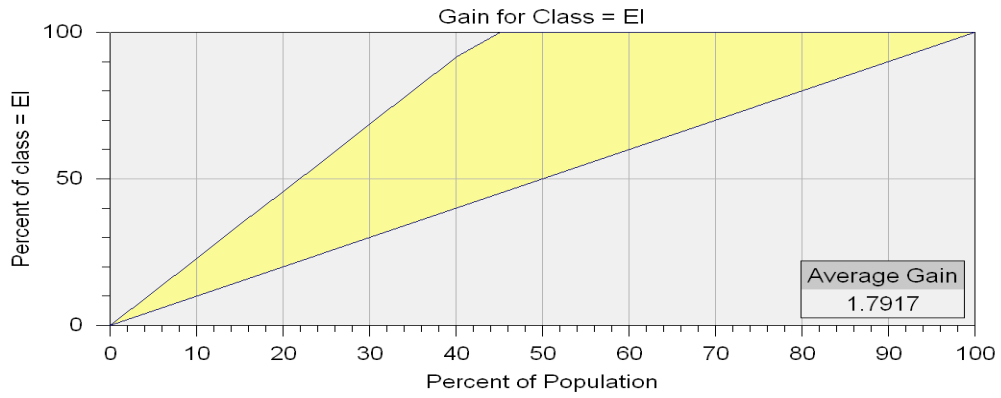
For Training Dataset

- Median target value for initial data sample = 0.16
- Mean target value for initial data sample = 0.2487113
- Mean target value for predicted values = 0.159299
- Average absolute error for initial data sample = 0.1864947
- Average absolute error after tree fitting = 99.892943
- Variance in initial data sample = 0.0579884
- Residual (unexplained) variance after applying Adaboost model = 0.0598907

- Coefficient of variation = 0.983975
- Normalized mean square error (NMSE) = 1.032804
- Correlation between actual and predicted = 0.911039
- Maximum error = 0.8283473
- RMSE (Root Mean Squared Error) = 0.2447257
- MSE (Mean Squared Error) = 0.0598907
- MAE (Mean Absolute Error) = 99.892943
- MAPE (Mean Absolute Percentage Error) = 126.90105

For Training Dataset;

- Mean target value for initial data sample = 0.2505229
- Mean target value for predicted values = 0.1484431
- Average absolute error for initial data sample = 0.1891939
- Average absolute error after tree fitting = 26.71747
- Variance in initial data sample = 0.0598494
- Residual (unexplained) variance after applying Adaboost model = 0.0700859
- Coefficient of variation = 1.056740
- Normalized mean square error (NMSE) = 1.171038
- Correlation between actual and predicted = 0.078412
- Maximum error = 0.852564
- RMSE (Root Mean Squared Error) = 0.2647374
- MSE (Mean Squared Error) = 0.0700859
- MAE (Mean Absolute Error) = 26.71747
- MAPE (Mean Absolute Percentage Error) = 135.81906



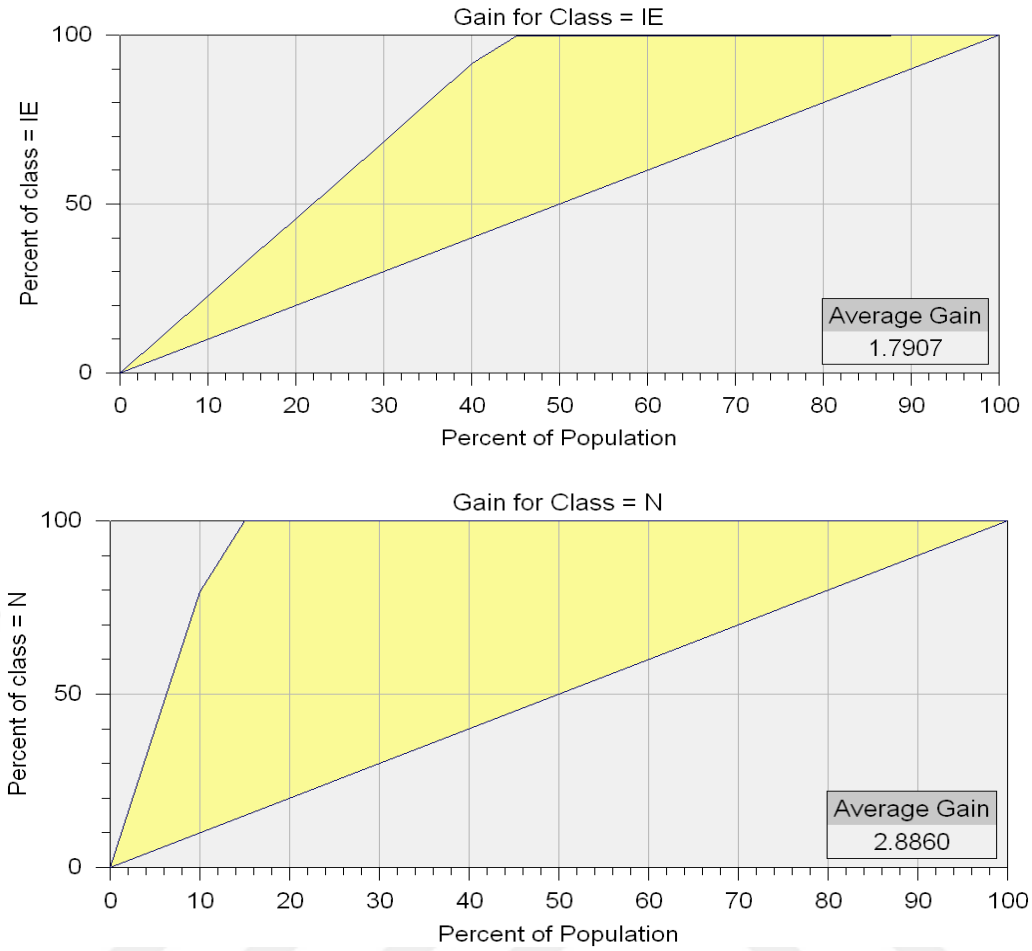


Figure 4.5: Gain Information For Three Classes Of The Training Dataset

For Validation (Testing) Dataset

- Mean target value for initial data sample = 0.2505229
- Mean target value for predicted values = 0.1484431
- Average absolute error for initial data sample = 0.1891939
- Average absolute error after tree fitting = 26.71747
- Variance in initial data sample = 0.0598494
- Residual (unexplained) variance after applying Adaboost model = 0.0700859
- Coefficient of variation = 1.056740
- Normalized mean square error (NMSE) = 1.171038
- Correlation between actual and predicted = 0.078412
- Maximum error = 0.852564
- RMSE (Root Mean Squared Error) = 0.2647374
- MSE (Mean Squared Error) = 0.0700859
- MAE (Mean Absolute Error) = 26.71747
- MAPE (Mean Absolute Percentage Error) = 135.81906

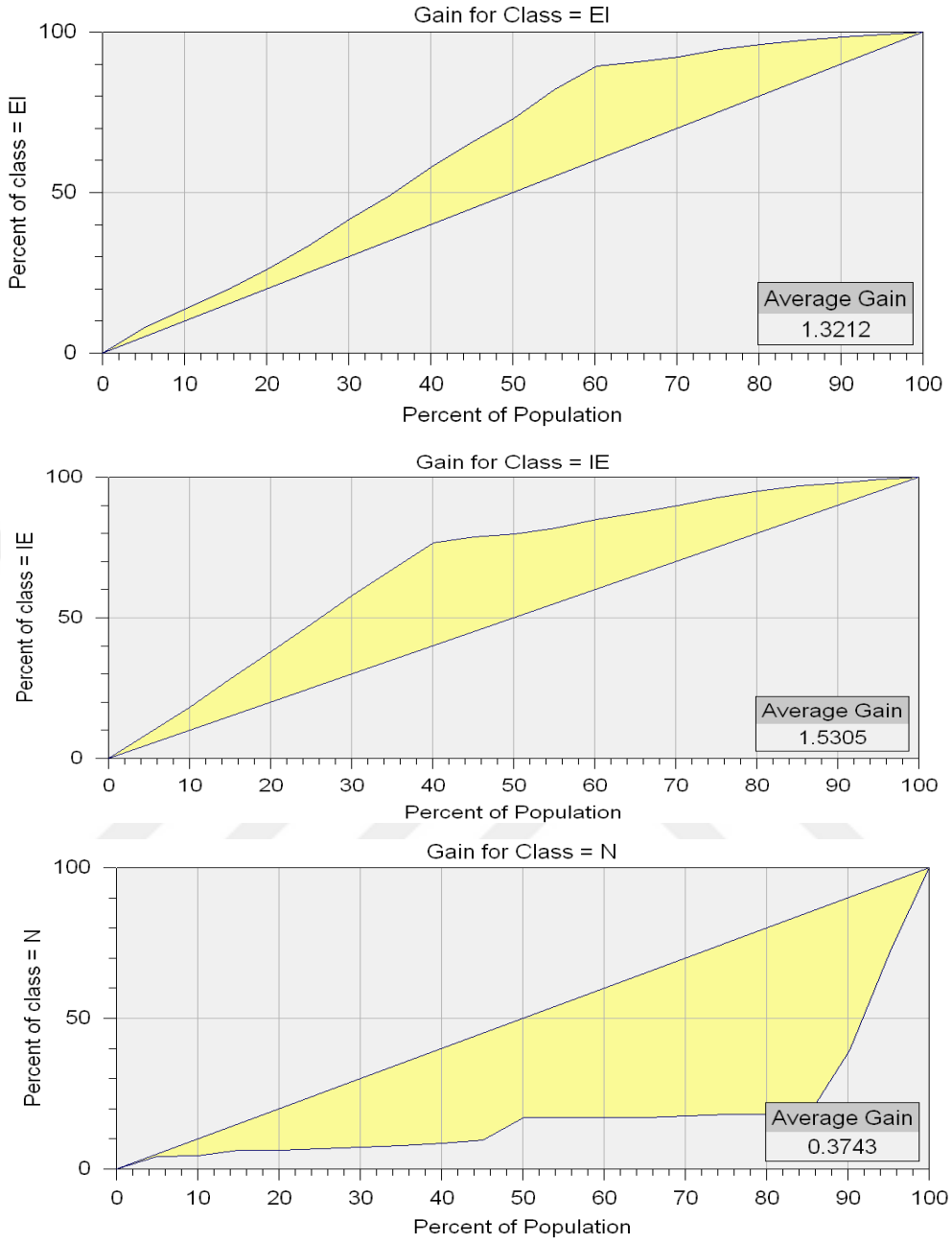


Figure 4.6: Gain Information For Three Classes Of The Testing Dataset

Figure 4.7 show surface of that groups classified on the characteristics more critical in delivering the decision

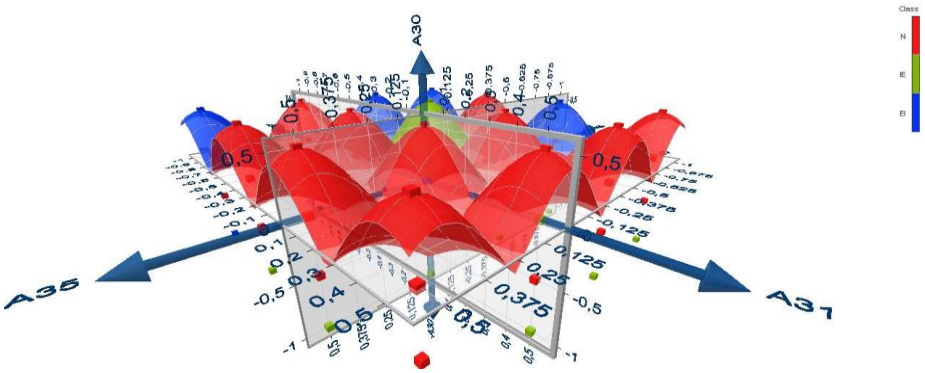


Figure 4.7: Surface of Second Sub Warehouse base on the main three features (A3O , A35 and A31)

D. Result of Third Sub Warehouse

After, we determine the Main Parameters of Boosting: No of Training samples=3139, No of Testing samples=1200, Max No of Epoch=200, No of Classes and Assign equal weights for each training samples. We generate multi Classifiers base on the training samples for given classes of all the training samples depend on vote principle. Where the initial values of this stage explaining below:

- The max number of trees=600.
- The minimum error with the training data occurs with 600 trees.
- The minimum error with the test data occurs with 283trees.
- The minimum point is smoothed by 5 trees.
- The specified minimum number of trees is 10.
- The tree series will be pruned to 283 trees.
- Maximum depth of any tree in the series = 5
- Average number of group splits in each tree = 9.3

  For Training Dataset
- Median target value for initial data sample = 7
- Mean target value for initial data sample = 8.6678967
- Mean target value for predicted values = 8.7402
- Average absolute error for initial data sample = 6.2891845
- Average absolute error after tree fitting = 1004.2439
- Variance in initial data sample = 49.365722
- Residual (unexplained) variance after applying Adaboost model = 19.466799
- Coefficient of variation = 0.509018
- Normalized mean square error (NMSE) = 0.394338
- Correlation between actual and predicted = 0.797354

- Maximum error = 12.023745
- RMSE (Root Mean Squared Error) = 4.4121196
- MSE (Mean Squared Error) = 19.466799
- MAE (Mean Absolute Error) = 1004.2439
- MAPE (Mean Absolute Percentage Error) = 124.63237

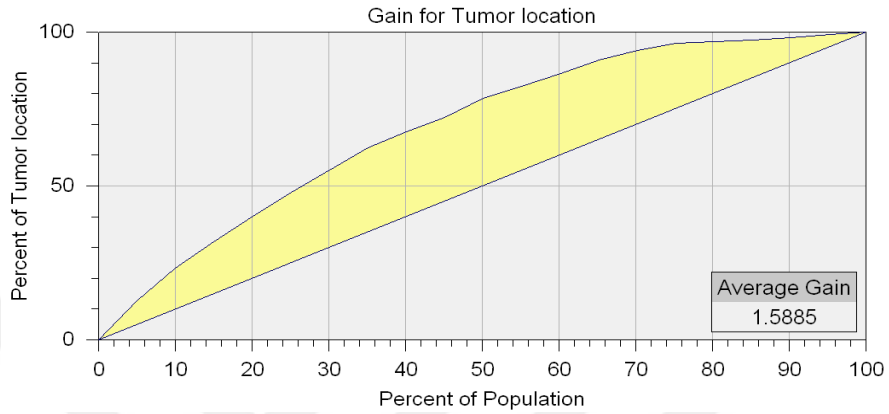


Figure 4.8: Gain Information for Tumor Location of the Training Dataset

For Validation Dataset

- Mean target value for initial data sample = 8.7205882
- Mean target value for predicted values = 9.5986703
- Average absolute error for initial data sample = 6.3724048
- Average absolute error after tree fitting = 310.78183
- Variance in initial data sample = 50.495458
- Residual (unexplained) variance after applying Adaboost model = 32.333587
- Coefficient of variation = 0.652050
- Normalized mean square error (NMSE) = 0.640327
- Correlation between actual and predicted = 0.614834
- Maximum error = 14.14281
- RMSE (Root Mean Squared Error) = 5.686263
- MSE (Mean Squared Error) = 32.333587
- MAE (Mean Absolute Error) = 310.78183
- MAPE (Mean Absolute Percentage Error) = 164.51836

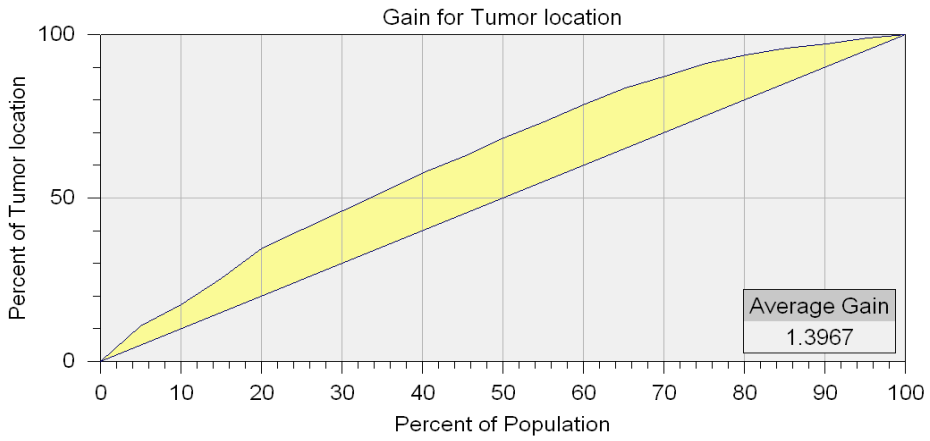


Figure 4.9: Gain Information for Tumor Location of the Testing Dataset

Figure 4.10 show surface of that classes classified on the characteristics more critical in delivering the decision

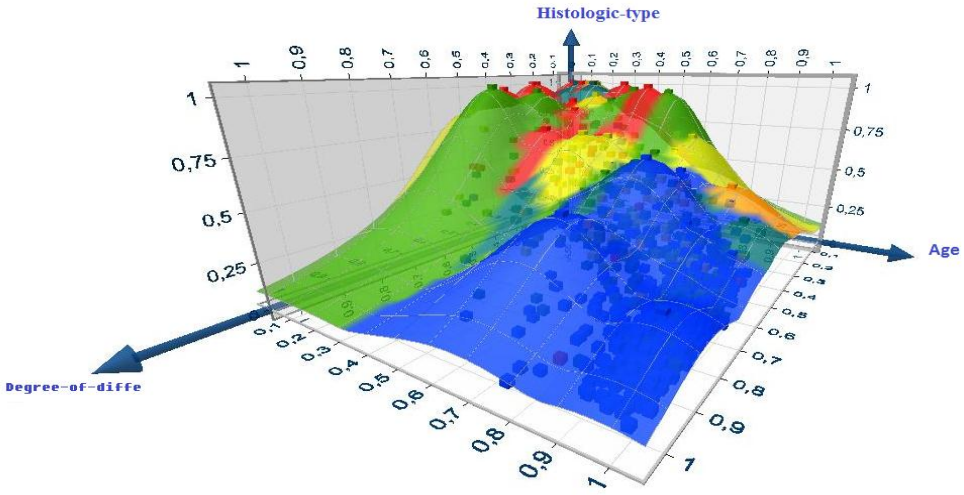


Figure 4.10: Surface of Third Sub Warehouse classified on the principal Three Features (Histologic-type , Degree-of-diffe. and Age)

4.1.3 The Results of Handle Dimension Reduction using FP-KC

A. Result of First Sub Warehouse

Step A.1: Compute the standardized data matrix of classified first sub warehouse

Step A.2: Compute Eigenvalues as in Table (4.8)

Table (4.8): Eigenvalues of the Standardized Dataset

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Eigenvalue	**4.927**	**3.976**	**3.320**	**2.770**	**2.546**	**2.442**	**2.339**	**2.328**	**2.287**	**2.210**
Variability (%	2.737	2.209	1.844	1.539	1.414	1.356	1.299	1.293	1.270	1.228
Cumulative %	2.737	4.946	6.791	8.329	9.744	11.100	12.400	13.693	14.963	16.191

	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
Eigenvalue	**2.128**	**2.107**	**2.067**	**2.052**	**2.034**	**2.009**	**1.965**	**1.931**	**1.920**	**1.892**
Variability (%)	1.182	1.171	1.148	1.140	1.130	1.116	1.092	1.073	1.067	1.051
Cumulative %	17.374	18.544	19.693	20.833	21.963	23.079	24.171	25.244	26.310	27.361

	F21	F22	F23	F24	F25	F26	F27	F28	F29	F30
Eigenvalue	**1.878**	**1.860**	**1.834**	**1.817**	**1.809**	**1.772**	**1.755**	**1.734**	**1.700**	**1.682**
Variability (%)	1.043	1.033	1.019	1.010	1.005	0.985	0.975	0.963	0.945	0.934
Cumulative %	28.404	29.438	30.457	31.466	32.471	33.456	34.431	35.394	36.339	37.273

	F31	F32	F33	F34	F35	F36	F37	F38	F39	F40
Eigenvalue	**1.662**	**1.638**	**1.632**	**1.618**	**1.608**	**1.586**	**1.573**	**1.552**	**1.535**	**1.532**
Variability (%)	0.924	0.910	0.907	0.899	0.893	0.881	0.874	0.862	0.853	0.851
Cumulative %	38.197	39.107	40.013	40.912	41.805	42.687	43.560	44.423	45.275	46.126

	F41	F42	F43	F44	F45	F46	F47	F48	F49	F50
Eigenvalue	**1.512**	**1.510**	**1.488**	**1.473**	**1.455**	**1.440**	**1.432**	**1.407**	**1.400**	**1.386**
Variability (%)	0.840	0.839	0.826	0.818	0.809	0.800	0.796	0.782	0.778	0.770
Cumulative %	46.967	47.805	48.632	49.450	50.259	51.058	51.854	52.636	53.414	54.184

	F51	F52	F53	F54	F55	F56	F57	F58	F59	F60
Eigenvalue	**1.372**	**1.341**	**1.327**	**1.322**	**1.316**	**1.299**	**1.286**	**1.282**	**1.251**	**1.241**
Variability (%)	0.762	0.745	0.737	0.734	0.731	0.722	0.715	0.712	0.695	0.690
Cumulative %	54.946	55.691	56.428	57.163	57.894	58.615	59.330	60.042	60.737	61.427

	F61	F62	F63	F64	F65	F66	F67	F68	F69	F70
Eigenvalue	**1.234**	**1.214**	**1.204**	**1.201**	**1.174**	**1.154**	**1.143**	**.129**	**1.123**	**1.114**
Variability (%)	0.685	0.675	0.669	0.667	0.652	0.641	0.635	0.627	0.624	0.619
Cumulative %	62.112	62.787	63.456	64.123	64.775	65.416	66.051	66.679	67.303	67.922

	F71	F72	F73	F74	F75	F76	F77	F78	F79	F80
Eigenvalue	**1.106**	**1.090**	**1.084**	**1.070**	**1.060**	**1.048**	**1.042**	**1.030**	**1.021**	**1.006**
Variability (%)	0.615	0.606	0.602	0.594	0.589	0.582	0.579	0.572	0.567	0.559
Cumulative %	68.536	69.142	69.744	70.338	70.927	71.510	72.088	72.661	73.228	73.787

	F81	F82	F83	F84	F85	F86	F87	F88	F89	F90
Eigenvalue	**1.000**	0.986	0.985	0.965	0.955	0.945	0.925	0.918	0.912	0.898
Variability (%)	0.555	0.548	0.547	0.536	0.531	0.525	0.514	0.510	0.507	0.499
Cumulative %	74.342	74.890	75.437	75.973	76.504	77.029	77.543	78.053	78.560	79.058

	F91	F92	F93	F94	F95	F96	F97	F98	F99	F100
Eigenvalue	0.892	0.883	0.876	0.866	0.858	0.838	0.836	0.822	0.811	0.801
Variability (%)	0.496	0.490	0.487	0.481	0.477	0.465	0.464	0.457	0.451	0.445
Cumulative %	79.554	80.044	80.531	81.012	81.489	81.954	82.419	82.876	83.326	83.771

	F101	F102	F103	F104	F105	F106	F107	F108	F109	F110
Eigenvalue	0.79	0.785	0.774	0.759	0.745	0.733	0.723	0.717	0.708	0.695
Variability (%)	0.44	0.436	0.430	0.421	0.414	0.407	0.402	0.398	0.394	0.386
Cumulative %	84.21	84.650	85.081	85.502	85.916	86.323	86.725	87.123	87.516	87.902

	F111	F112	F113	F114	F115	F116	F117	F118	F119	F120
Eigenvalue	0.682	0.673	0.666	0.656	0.640	0.630	0.616	0.603	0.579	0.557
Variability (%)	0.379	0.374	0.370	0.364	0.356	0.350	0.342	0.335	0.322	0.309
Cumulative %	88.281	88.655	89.025	89.389	89.745	90.095	90.437	90.772	91.094	91.403

The value of Eigenvalues higher than one for a assigned dataset is bolded. This indicates the characteristics of that state of eigenvalues is practising in the FP-KC algorithm for a assigned dataset. The amount of that characteristics is eighty-one from the total 120 Characteristics. plus, Figure (4.11) explains the connection of Eigenvalues and combined variability**.**

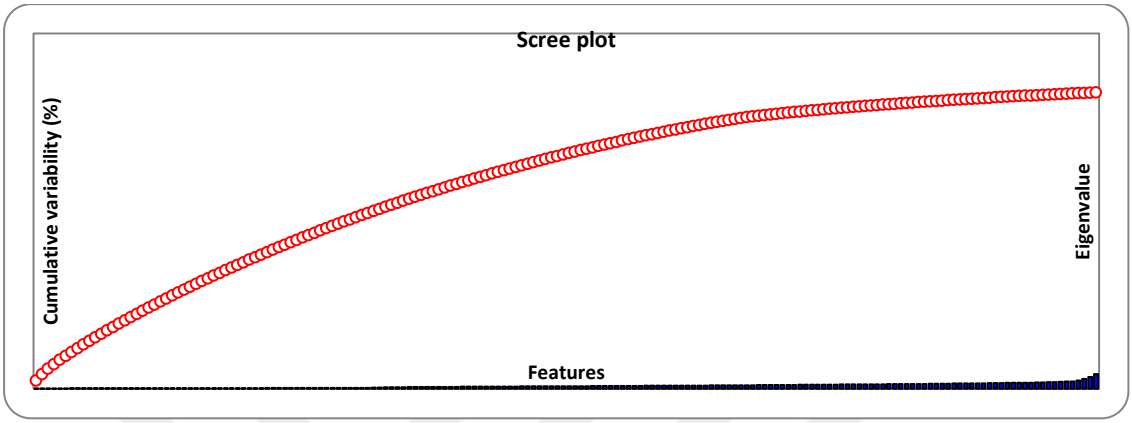


Figure 4.11: correlation between Eigenvalues, Collective and Scree Plot of FIRST Database

Step A.3 ; place the central settings of FP-KC

Step A.4 : generation the Assoaction Rules base on the main split tree

- Rule1: If A85 <= 0.5 and A93 <= 0.5 Then class = 3
- Rule2: If A85 <= 0.5 and A93 > 0.5 and A105 <= 0.5 Then class = 3
- Rule3: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 <= 0.5 and A88 <= 0.5 and A97 <= 0.5 and A72 <= 0.5 Then class = 3
- Rule4: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 <= 0.5 and A88 <= 0.5 and A97 <= 0.5 and A72 > 0.5 Then class = 1
- Rule5: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 <= 0.5 and A88 <= 0.5 and A97 > 0.5 Then class = 1
- Rule6: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 <= 0.5 and A88 > 0.5 Then class = 3
- Rule7: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 > 0.5 and A95 <= 0.5 and A98 <= 0.5 Then class = 1
- Rule8: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 > 0.5 and A95 <= 0.5 and A98 > 0.5 Then class = 3
- Rule9: If A85 <= 0.5 and A93 > 0.5 and A105 > 0.5 and A100 > 0.5 and A95 > 0.5 Then class = 3
- Rule10: If A85 > 0.5 and A105 <= 0.5 and A88 <= 0.5 and A82 <= 0.5 and A89 <= 0.5 and A63 <= 0.5 and A100 <= 0.5 Then class = 2

Rule11: If A85 > 0.5 and A105 <= 0.5 and A88 <= 0.5 and A82 <= 0.5 and A89 <= 0.5 and A63 <= 0.5 and A100 > 0.5 and A97 <= 0.5 Then class = 2

After doing the above steps, the dataset have 2000 record and 180 feature become knowledge base in the form classified association rules have 36 rule and 15 features.

The highest value of membership Function for a given rule is figured in bold font. This indicates the crucial values of that group function is associated with the state for a mentioned rules. While, we found the Importance of Variables is sort from the higher to smaller:

A105 = 100.000, A85 = 98.567, A93 =92.505, A100 =82.220, A88= 77.356, A97= 74.935, A72= 73.755, A98=71.568, A82 =69.449, A89 = 68.105, A63 = 64.857, A83 =63.340, A95 = 63.162, A94 =57.874, A74=51.146.

B. Result of Second Sub Warehouse

Step B.1: Compute the standardized data matrix of classified second sub warehouse

Step B.2: Compute Eigenvalues as in Table (4.9)

Table (4.9): Eigenvalues of the Standardized Dataset

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Eigenvalue	**2,126**	**1,636**	**1,441**	**1,375**	**1,345**	**1,279**	**1,255**	**1,234**	**1,226**	**1,200**
Variability (%)	3,543	2,727	2,401	2,292	2,242	2,131	2,092	2,056	2,044	1,999
Cumulative %	3,543	6,270	8,672	10,964	13,206	15,337	17,429	19,485	21,529	23,528

	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
Eigenvalue	**1,184**	**1,179**	**1,156**	**1,147**	**1,118**	**1,108**	**1,089**	**1,078**	**1,065**	**1,058**
Variability (%)	1,974	1,965	1,927	1,911	1,864	1,846	1,815	1,796	1,775	1,764
Cumulative %	25,502	27,467	29,394	31,306	33,170	35,016	36,831	38,627	40,403	42,167

	F21	F22	F23	F24	F25	F26	F27	F28	F29	F30
Eigenvalue	**1,043**	**1,041**	**1,035**	**1,029**	**1,015**	0,997	0,991	0,984	0,977	0,964
Variability (%)	1,739	1,735	1,725	1,716	1,692	1,662	1,652	1,640	1,628	1,607
Cumulative %	43,905	45,640	47,365	49,081	50,773	52,435	54,087	55,727	57,355	58,962

	F31	F32	F33	F34	F35	F36	F37	F38	F39	F40
Eigenvalue	0,955	0,945	0,932	0,918	0,912	0,905	0,901	0,886	0,882	0,881
Variability (%)	1,591	1,574	1,554	1,530	1,521	1,509	1,501	1,477	1,470	1,469
Cumulative %	60,553	62,127	63,681	65,211	66,732	68,240	69,742	71,219	72,689	74,157

	F41	F42	F43	F44	F45	F46	F47	F48	F49	F50
Eigenvalue	0,864	0,851	0,846	0,838	0,829	0,826	0,814	0,806	0,795	0,790
Variability (%)	1,440	1,418	1,410	1,396	1,382	1,377	1,357	1,343	1,326	1,316
Cumulative %	75,597	77,015	78,425	79,821	81,203	82,580	83,937	85,280	86,606	87,922

	F51	F52	F53	F54	F55	F56	F57	F58	F59	F60
Eigenvalue	0,784	0,771	0,760	0,744	0,735	0,732	0,721	0,695	0,679	0,627
Variability (%)	1,307	1,285	1,266	1,240	1,225	1,221	1,201	1,159	1,131	1,044
Cumulative %	89,229	90,513	91,779	93,019	94,244	95,464	96,666	97,825	98,956	100,000

The value of Eigenvalues more than one for a assigned dataset is bolded. This indicates the characteristics of that state of eigenvalue is using in the FP-KC algorithm for a assigned dataset. The number of that characteristics twenty five from the whole sixty Characteristics. Figure (4.12) shows the connection of eigenvalue and combined variability**.** In this figure the red color explain the boundary between the Eigenvalues more than one and other.

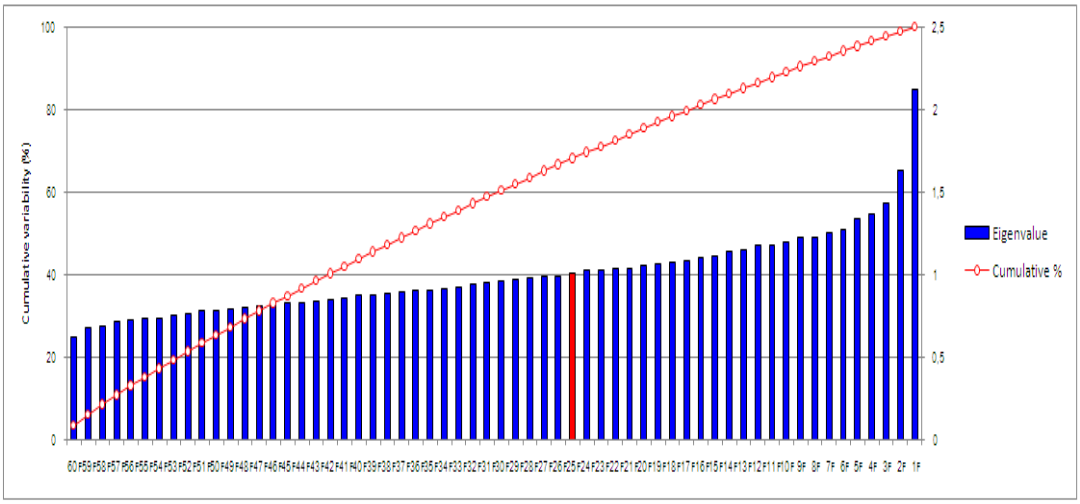


Figure 4.12: Association of Eigenvalues and Combined variability

Step B.3: Compute Eigenvectors as explained in Table (4.10)

Table (4.10): Eigenvectors of Eigenvalues More than One

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13
A 1	0,054	0,000	0,093	0,040	-0,020	-0,055	0,062	-0,223	0,040	-0,184	-0,018	-0,046	0,306
A 2	0,068	-0,079	0,215	0,166	-0,172	-0,040	0,111	-0,071	-0,260	0,049	0,006	0,012	0,131
A 3	0,096	0,066	0,037	0,193	0,264	0,066	-0,020	0,101	-0,130	-0,010	0,039	-0,066	0,037
A 4	0,060	-0,093	0,045	-0,090	0,117	0,184	-0,050	0,020	-0,242	0,101	-0,055	-0,025	0,133
A 5	0,100	-0,090	0,166	0,298	0,050	-0,122	-0,203	0,007	0,143	0,012	0,037	-0,008	-0,011
A 6	0,098	0,083	-0,018	0,054	0,098	-0,062	0,089	-0,010	-0,103	0,207	0,231	-0,190	-0,142
A 7	0,096	-0,162	-0,044	-0,048	-0,247	0,014	-0,034	-0,053	-0,109	-0,063	0,339	-0,082	-0,009
A 8	0,131	-0,212	0,054	0,131	-0,134	-0,229	-0,093	0,174	0,104	0,009	0,024	-0,066	0,005
A 9	0,083	-0,	-0,0	0,044	0,179	-0,1	-0,1	0,111	-0,0	-0,1	-0,1	-0,2	0,047

Step B.5: Set the Main Parameters of FP-KC

Liquidate all the parts not confirmation eigenvalue criterion (i.e., the eigenvalue smaller than one).

- Liquidate all the parts not confirmation balance variation defined criterion (i.e., the value of Collective variability smaller than 50.773).

- Extract all the parts not confirmation The screen plot criterion the highest figure of elements that should be extracted is just previous to where the area starts to order out into a straight position. see Figure 4.13

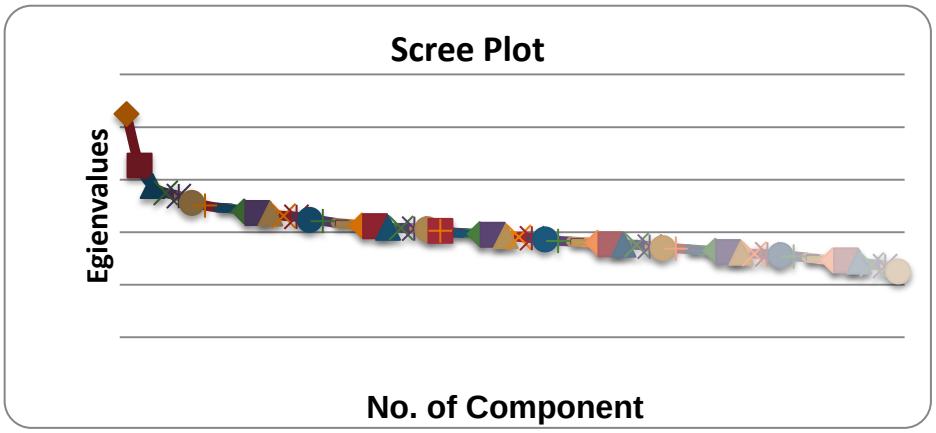


Figure 4.13: Scree Plot of the Component

- Set min_sup as dynamically value base one the distribution of features in database after each alteration and not as constant value for all alterations.

$$confidence(A \Rightarrow B) = P(B|A) = \frac{support\_count(A \cup B)}{support\_count(A)}. \quad (4.1)$$

C. Result of Third Sub Warehouse

Step C.1: Compute the standardized data matrix of classified second sub warehouse

Step C.2: Compute Eigenvalues as in Table (4.11)

Table (4.11): Eigenvalues of the Standardized Dataset

	F1	F2	F3	F4	F5	F6	F7
Eigenvalue	**3.551**	**2.741**	**1.887**	**1.198**	**1.059**	**1.035**	0.885
Variability (%)	19.726	15.230	10.482	6.656	5.883	5.751	4.918
Cumulative(%)	19.726	34.956	45.438	52.094	57.977	63.727	68.646

	F11	F12	F13	F14	F15	F16	F17
Eigenvalue	0.603	0.530	0.482	0.422	0.419	0.335	0.317
Variability (%)	3.348	2.942	2.680	2.345	2.328	1.859	1.761
Cumulative %	84.753	87.695	90.375	92.720	95.048	96.907	98.668

The value of Eigenvalues higher than one for a assigned dataset is bolded. This indicates the characteristics of that state of eigenvalues is practising in the FP-KC algorithm for a assigned dataset.. The number of that characteristics is six from the 17 Features. Addition, Figure (4.14) shows the connection of eigenvalue and collective variability**.**

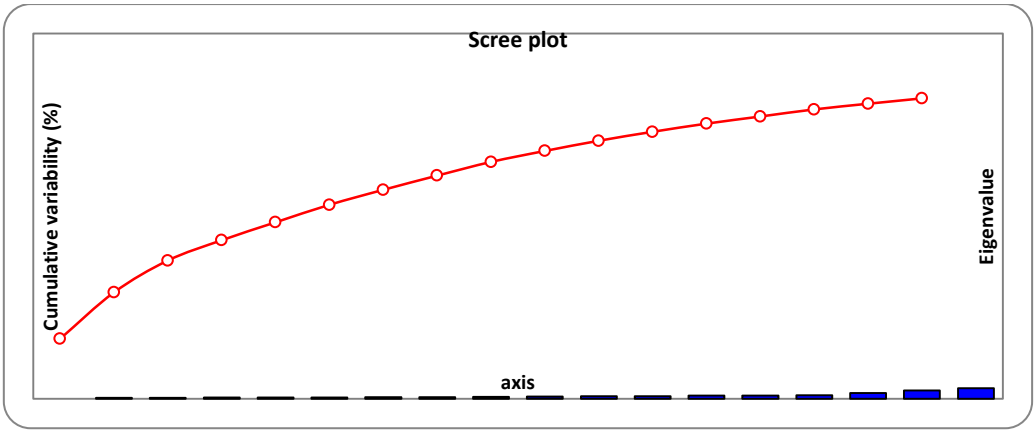


Figure 4.14: Relation of Eigenvalues and Cumulative variability

Step C.3: Compute Eigenvectors as explained in Table (4.12)

Table (4.12): Eigenvectors of Eigenvalues More than One

	F11	F12	F13	F14	F15	F16	F17
A1	-0.005	0.040	-0.014	0.064	-0.279	-0.008	-0.053
A2	0.141	-0.317	-0.442	-0.445	-0.193	0.224	0.099
A3	-0.127	0.440	0.079	0.037	0.077	-0.407	0.475
A4	-0.287	-0.410	0.023	0.079	0.383	0.082	-0.287
A5	0.285	0.399	0.009	0.353	-0.175	0.438	-0.336
A6	-0.084	-0.233	0.526	-0.093	0.193	-0.072	-0.086
A7	0.052	0.076	-0.077	-0.340	-0.209	-0.430	-0.373
A8	0.512	-0.028	-0.059	0.050	0.337	-0.150	-0.122
A9	0.337	-0.276	0.060	0.104	-0.052	-0.056	0.439
A10	-0.158	0.221	0.044	-0.409	-0.171	0.044	0.027
A11	0.075	-0.214	-0.119	0.457	-0.283	-0.380	-0.017
A12	0.264	0.207	-0.215	-0.155	0.463	-0.002	0.041
A13	0.141	-0.192	0.149	0.160	-0.052	0.060	0.168
A14	0.197	0.022	0.270	-0.037	-0.259	0.108	-0.087

Step C.4: Compute principal components that explain the correlations between variables and factors after rotation as shown as shown in Table (4.13)

Table (4.13): Principal Components Matrix

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
A1	-0.073	0.443	0.094	-0.411	0.135	0.372	0.274	0.401	0.311	-0.307
A2	0.397	-0.051	0.688	0.115	-0.080	-0.203	-0.055	0.074	0.121	-0.056
A3	0.614	0.283	0.338	0.077	-0.094	-0.339	0.198	0.075	-0.020	-0.094
A4	0.589	0.404	0.003	0.089	-0.172	-0.026	0.306	0.008	-0.243	-0.239
A5	0.657	0.356	0.275	-0.082	0.072	-0.140	-0.052	-0.131	-0.061	0.079
A6	0.548	0.129	0.296	-0.282	0.183	-0.063	-0.388	0.263	0.118	0.225
A7	0.444	0.616	-0.155	0.093	-0.170	0.270	0.027	-0.246	-0.021	0.172
A8	0.312	-0.381	-0.331	-0.288	-0.403	-0.199	0.028	-0.019	0.355	-0.120

A9	0.118	0.743	-0.205	0.182	-0.003	0.214	-0.247	-0.125	0.100	-0.017
A10	0.511	-0.528	-0.258	-0.282	-0.203	0.180	-0.058	0.113	-0.131	0.044
A11	0.348	-0.569	0.312	-0.108	-0.111	0.208	-0.236	0.014	-0.328	-0.149
A12	0.482	-0.239	-0.029	0.184	0.534	0.397	-0.037	0.095	-0.144	-0.108
A13	0.359	-0.414	0.170	0.134	-0.034	0.272	0.561	0.012	0.105	0.434
A14	0.204	-0.088	-0.411	0.621	-0.164	-0.111	-0.034	0.482	-0.100	-0.117

Step C.5: Place the Principal settings of FP-KC

- Extract all the constituents non confirmation eigenvalue criterion (i.e., the eigenvalue less than one).
- Extract all the parts without confirmation proportion of variance explained criterion (i.e., the value of Cumulative variability less than 63.727).
- Remove all the factors not confirmation. The scree area criterion (i.e., the highest figure elements that should be removed is just before where the plot starts to unravel out into a straight line.
  - Set min_sup as actively value base on the arrangement of characteristics in the database after each adjustment and not a fixed mark for all modifications

$$confidence(A \Rightarrow B) = P(B|A) = \frac{support\_count(A \cup B)}{support\_count(A)}. \quad (4.2)$$

After doing the above mentioned levels, the dataset have 4339 release and 17 characteristics become information base in the form classified association rules have 76 rule and 4 features

4.2 DISCUSSION

The system is integration among (preprocessing method, reduction dimension method and data mining) to provide suitable solutions of (number of nearest neighbors, missing values, reduction the main 3D, generation rules) and to establish comprehensive classified association rules base one the Miner of obtain accuracy and comprehensive classify association rules.

We can summarize the main problems in the IDA and KDD as the following:

First: One of the important trends in KDD will be the growing importance of data processing. But this point faces problems similar to those of data mining (High dimensional data, missing values imputation and data integration). As explained before, one of the open still problems in estimation missing values methods is how to select the optimal number of nearest neighbors of those values.

Second: The goal of dimension reduction methods is using the correlation structure among the predicator variables to reduction the three main dimensions (features, samples and value of features). But, how we can combination among these three dimension without lose any important inform is still as one of the challenge in KDD system.

Third: As we know data mining algorithm are “black box character” for many reasons explained in chapter one. But, how we can convert it to system will behaviors and conclusion that can be explained and analyzed in a comprehensible manner (i.e., white box) is remained one of the main challenges of KDD system.

Therefore, this work attempts to solve the missing values problem and the problem of estimation the best number of nearest neighborhoods for the Biological warehouse used to test the System. This solution depend on develops RF through replacing the correlation function of original RF by seven different types of similarity functions. We get the optimal estimation of missing values by LLS that depend on the values of nearest neighbors’ generation by DRF. Finally, the results evaluated by two measures Pearson correlation and NRMSE.

The second goal of this work is classify the biological warehouse base on their 'attributers by using adboosting algorithm, in this method determine the class for each record by taking the most vote from multi classifiers.

The third goal of this work is the satisfactions of the dimension reduction methods, using the correlation structure among the predicator variables. The main techniques suggested here is the cooperation between PCA and FP-growth, the new algorithm called FP-KC. By suggesting algorithm FP-KC, we can get accurate association rule base. In addition, it can be used to compress dataset in three dimensions (number of features" by PCA", number of records and features " by FP- Growth" and value of features through standardization matrix.(

Each sub warehouse (i.e., database) generates different number of association rules base on PCA result and FP-KC algorithm.

4.3 SUMMARY

We attempt to design system solve many challenges in KDD field, by integration among the preprocessing, data reduction and data mining concepts.

In general, we can summarize the main benefits points of this work as followings:

First**:** In missing values problem, this system can solve the problem by exploring novel tool DRFLLS. DRFLLS prove the database which have small rate of missing values given the best estimation to number of nearest neighbors by DRFPC and in second degree by DRFFSM1 when r=4. While, if the dataset have high rate of missing values then, it's given the best estimation to number of nearest neighbors by DRFFSM5 and in second degree by DRFFSM3. After that, the missing value estimation by LLS and the results accuracy measure by (NRMSE and Pearson correlation).

Second**:** we can classify the biological warehouse base on their 'attributers by using adboosting algorithm, in this method determine the class for each record by taking the most vote from multi classifiers.

Third**:** In three dimensions reduction problem, this work proposes cooperation algorithm between PCA&FP-growth, these algorithm called FP-KC. The generated FP-KC can get accurate association rules and compress the dataset in three dimensions (number of features" by PCA", number of records and features "by FP-Growth" and value of features through standardization matrix). The confidence degree of all the association rules yield by FP-KC is equal to 95%.

5. CONCLUSIONS

- DRFLLS tool has been proving ability to higher estimation accuracy of the missing values than the traditional estimation methods and gives convenient environment to us. By experiments, we find dataset have small rate of the missing values then the best estimation of number of nearest neighbors getting by DRFPC and in second degree by DRFFSM1 when r=4. while if the dataset have high rate of missing values then the best estimation of number of nearest neighbors getting by DRFFSM5 and in second degree by DRFFSM3.
- In this work, not use the traditional Adboosting algorithm to perform the classification but we used it with different types of satisfy measures including Normalized mean square error, Correlation between actual and predicted , Maximum error, Root Mean Squared Error, Mean Squared Error, Mean Absolute Error and Mean Absolute Percentage Error pulse the Gain information. All these measures increase the accuracy of decisions.
- Using dynamical values of Min _support in the FP-KC increase the activity of the proposed algorithm. These values change in each level from the FP-KC base on the types of dataset used.
- Compute the Gaussian as membership functions of the classified assoaction rules after applied the FP-KC prove the correct and accuracy of the suggested tool that called MOACCR.
- This work highlights the main challenges in the Knowledge discovery in database and intelligent data analysis explained it in the figure 1.3 . Then it can conclude a strong relationship between two concepts.
- There are many reasons for developing the FP-Growth data mining algorithm in build up a novel algorithm FP-KC to find the association rules
    - **A.** The size of an FP-tree is typically smaller than the size of the uncompressed data because many records in dataset often share a few items in common.
    - **B.** Given the best result, if all the records have the same set of items, and this point always satisfy in the scientific dataset.
    - **C.** FP-growth is an efficient algorithm because it illustrates how a compact representation of the transaction data set helps to efficiently

generate frequent item sets**.**

D. The run-time performance of FP-growth depends on the compaction factor of the data set.

- This policy resolution to accomplish the purpose of dimension decrease ways is utilising the relationship construction among the predictor variables to reduce the three main dimensions (characteristics, units and value of characteristics). FP-KC is joining connecting the features of principal element investigation and rhythm model increase by doing three measures including eigenvalue, additive variability and Screeplot linked to PCA as information decreases of FP-growth algorithm.

5.1 SUGGESTIONS FOR FUTURE WORK

- We show the many advantages of DRFLLS can be employing in the data integration stage of knowledge discovery in data base system.
- By analyses, we found FP-KC algorithm provided good concentration effects; hence, we propose utilizing this limit in the concentration the digital video clip tracks and pictures because FP-KC gives with valuable knowledge as registered union habits.
- 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, therefore, we suggest to using one of the optimization algorithm as a tool of Constriction new dataset from the original dataset and verification from the verification ratio between the original and generation datasets.
- Data mining algorithms are "black-box character" as neural network because many reasons explained in this thesis; therefore, we suggest studying this problem and analysis it base on the mathematical principles.
- Design mathematical model of KDD System is not easy task for many reasons; one of it, the KDD system is consist from multi stage therefore, the researchers must design sub model for each stage then the result model is linear combination among these sub models. We believe this problem represent the main three types of data analysis challenges: Analytics challenge, Communication challenge and Application challenge. Therefore, to solve this

problem and satisfy the goal of automating data analysis is discovering a relationship between anumbers of attributes and representing this relationship in form of a model. We need the cooperating among three individuals data owners, programmers and Mathematician.

REFERENCESS

[1] S.M.Chen, M.S.Yeh and P.Y.Hsiao."A Comparison of Similarity Measures of Fuzzy Values". Fuzzy sets and system s. vol 72, pp. 79- 89, 1995.

[2] David S. Siroky., "Navigating Random Forests and related advances in algorithmic modeling", Statistics Surveys , Vol. 3 , PP 147–163 , ISSN: 1935-7516, DOI: 10.1214/07-SS033, 2009.

[3] Oded Maimon and Lior Rokach," Introduction to Knowledge Discovery and Data Mining", Data Mining and Knowledge Discovery Handbook, 2nd ed., Springer Science and Business Media, LLC 2010. DOI 10.1007/978-0-387-09823-4_1.

[4] Jiawei Han and Micheline Kamber.," Data Mining: Concepts and Techniques" Second Edition. University of Illinois at Urbana-Champaign. 500 Sansome Street, Suite 400, San Francisco. Elsevier 2006. http:// www.books.elsevier.com.

[5] Colm R, Derek G., Gerard C. and Padraig C.," Missing Value Imputation for Epistatic MAPs" School of Computer Science and Informatics, University College Dublin, Dublin, Ireland. *Bioinformatics* 2010.

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

[7] Jeffrey W. Seifert, "Data Mining: An Overview," CRS Report for Congress, 2004.

[8] Mehmed Kantardzic ," Data Mining: Concepts, Models, Methods, and Algorithms" IEEE Computer society, *Sponser* ,2003.

[9] Daniel T. Larose,"Data Mining Methods and Models" Department of Mathematical Sciences Central Connecticut State University 2006.

[10] Barak Chizi and Oded Maimon," Dimension Reduction and Feature Selection", *Data Mining and Knowledge Discovery Handbook*, 2nd ed., Springer Science and Business Media, LLC 2010.

[11] Jorn B., Jaap H. and Bernd W. **"**PhyloPars: estimation of missing parameter values using phylogeny". University Amsterdam, Nucleic Acids Research, Vol. 37, No. suppl_2 W179-W184, 2009.

[12] Tyler M., Sue E. and George Y. “Replacing Missing Data for Ensemble Systems”. The Pennsylvania State University. PA 16804-0030, 2010.

[13] Mahdi A.," Extracting Rules from Databases Using Soft Computing", M.Sc Thesis, University of Babylon, 2005.**Patents**

[14] Pal S., Mitra S., and Mitra P., “Rough fuzzy MLP: Modular evolution, rule generation and evaluation,” IEEE Trans. Knowledge Data Eng., 2001.

[15] Pang-Ning Tan, Michael Steinbach and Vipin Kumar, " Introduction to Data Mining ", University of Minnesota,USA,2006..

[16] M.M. Gaber, "Scientific Data Mining and Knowledge Discovery Principles and Foundations", Springer, 2010. ISBN: 978-3-643-02787-1. http:// www.springer.com/978-3-643-02787-1.

[17] Breiman, L. “Random Forests.” Machine Learning, vol 45: pp 5–32. 2001.

[18] Hans-Peter Kriegel , Karsten M. Borgwardt · Peer Kröge. Alexey Pryakhin · Matthias Schubert and Arthur Zimek.,“Future Trends in Data Mining”. Data Mining and Knowledge Discovery, Springer. 15:87–97. 2007.