# Development of Novel Soft Computing Architectures and Applications to Banking and Finance

---

A Dissertation submitted to the University of Hyderabad
in partial fulfillment of the degree of

## MASTER OF TECHNOLOGY

in

## Information Technology

by

**ANKAIAH NARRAVULA**


Department of Computer and Information Sciences

School of Mathematics, Computer and Information Sciences

University of Hyderabad
(P.O.) Central University, Gachibowli
Hyderabad – 500 046
Andhra Pradesh
India


# CERTIFICATE

This is to certify that the dissertation entitled "**Development of Novel Soft Computing Architectures and Applications to Banking and Finance"** submitted by **Ankaiah Narravula** bearing Reg. No **09MCMB29** in partial fulfillment of the requirements for the award of Master of Technology in Information Technology is a bonafide work carried out by him under my supervision and guidance.

The dissertation has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Signature of the Supervisor

Head of the Department

Dean of the School

# DECLARATION

I **Ankaiah Narravula** hereby declare that this Dissertation entitled "**Development of Novel Soft Computing Architectures and Applications to Banking and Finance** ", submitted by me under the guidance and supervision of **Dr. V. Ravi, Associate Professor, IDRBT**, is a bonafide work. I also declare that it has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma.

Date: Name: **Ankaiah Narravula**

Signature of the Student:

Regd. No. **09MCMB29**

# ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to Dr. V. Ravi, Associate Professor, IDRBT, Hyderabad, who generously supported and guided me throughout my project, IDRBT for providing me with the infrastructure and technical support that I needed for this project. The project would not have been possible without his assistance.

I would like to thank Mr. B. Sambamurthy, Director, IDRBT, Prof. Arun Agarwal, Dean, School of MCIS, Prof. C. Raghavendra Rao, Head of the Department (DCIS), University of Hyderabad for extending their cooperation.

The cooperation of all the faculty members of IDRBT and the Department of Computer and Information Sciences, University of Hyderabad has been precious and timely. I wish to thank them for being very patient, understanding and helpful.

I thank my friend Mahesh, my classmate. I also thank Abdul and Naveen for their valuable advices. Finally, I thank all of my classmates for providing me the necessary support and encouragement throughout the M.Tech. program.

**Ankaiah Narravula**
Reg.No.09MCMB29
Email Id: ankireddy.cse@gmail.com
University of Hyderabad &IDRBT,
Hyderabad.

# LIST OF PAPERS COMMUNICATED

1. N. Ankaiah and V. Ravi, "A Novel Soft-computing Hybrid for Data Imputation", *DMIN-11, 7th International Conference on Data mining,* Las Vegas, USA. (*Accepted*).

2. N. Ankaiah and V. Ravi, "Multi objective formulation of Data Envelopment Analysis – Solution by Differential Evolution Algorithm", *IEEE Transactions on Evolutionary Computations. (Under Review).*

3. N. Ankaiah, V. Ravi and I. Bose "Data Imputation via Soft Computing: Application to Anti Phishing in Financial Institutions". (*To be communicated*).

# Abstract

Computing which has tolerance of imprecision, uncertainty, partial truth, and approximation is called soft computing. Despite various soft computing architectures proposed towards solving various banking and finance problems, still some of the problems are incomplete. In this thesis, we try to develop several novel soft computing architectures for solving some of the problems that are facing in banking and finance.

First, we developed a new methodology called Multi Objective DEA (MODEA) solved by a meta-heuristic, viz. Differential Evolution. The aim of MODEA is to solve several shortcomings of Data Envelopment Analysis (DEA). That is in DEA, linear programming should be run as many times as the number of DMUs. Consequently, there is no single set of weights for all the DMUs. Further, even though the efficiency is a fraction, the non linear optimization problem, is approximated as an equivalent linear programming problem (LPP). In our new methodology, we maximize the efficiencies of all the DMUs simultaneously in a multi objective framework and also maximize the efficiencies which are fractions as they are without approximating and reformulating the original optimization problem. We developed two variants of the MODEA; viz., MODEA1 and MODEA2, wherein MODEA1 takes recourse to scalar optimization and MODEA2 follows Max-Min approach. In both cases, we adopt differential evolution (DE) to solve the resulting non-linear optimization problem. Our formulation ensures that we get a single set of weights for all DMUs. Further, we obtain absolute efficiencies albeit in a compromise solution unlike the traditional DEA. The effectiveness of the two variants of MODEA is demonstrated on a set of datasets taken from literature. We also applied NSGA-II to solve the non-linear optimization problem in the strict multi objective sense. It was found that the empirical results yielded by MODEA1, MODEA2 and NSGA-II are comparable, as evidenced by Spearman's rank correlation coefficient test. The correlation between the efficiency scores obtained by proposed models and DEA-CCR is greater than 99%. However, in the proposed method, we are solving fractional objective as it is without converting into LPP in a multi-objective framework and also we are obtaining common set of weights. We are solving multiple objectives with single framework by using Differential Evolution algorithm.

Later, we proposed a novel 2-stage soft computing approach for data imputation, involving local learning and global approximation in tandem, whereas in the literature only one of them is used. In stage 1, K-means algorithm is used to replace the missing values with cluster centers. Stage 2 refines the resultant approximate values using multilayer perceptron (MLP). MLP is trained on the complete data with the attribute having missing values as the target variable one at a time. The hybrid is tested on 2 benchmark problems each in classification and regression using 10-fold cross validation. In all datasets, some values, which are randomly removed, are treated as missing values. The actual and the predicted values obtained are compared by using Mean Absolute Percentage Error (MAPE). We observe that, the MAPE value is reduced from stage 1 to stage 2, indicating the hybrid approach resulted in better imputation compared to stage 1 alone.

Lastly, we classified the severity of phishing attacks in terms of risk levels i.e low, medium and high by using the financial data of the organizations which are targeted by phishing attacks. The main problem in classifying is availability of complete data. From the available 1028 instances of financial data, 777 instances are incomplete, and only 251 instances are complete. So we solved the problem of missing data imputation by using a 2- stage approach which uses local learning and global approximation. In the stage 1 we used K-means algorithm and in the stage 2, we used MLP. We also used General Regression Neural Network in the stage 2 of imputation instead of MLP and compared the classification accuracy. We used Decision Tree (DT), MLP, and Support Vector Machine (SVM) for classification. The classification accuracy by replacing the missing data using proposed approach is 89% where as it is only 70% by using other imputation techniques.

# Table of Contents

# 1

# Introduction

Objective of this study is to develop novel soft computing architectures and we tries to solve some of problems like ranking the organizations based on the utilization of resources that are available and generation of output, problem of missing data imputation, assessing severity of phishing attacks etc, in banking and finance. This chapter tries to give the overview about soft computing, and also explains about Data Envelopment Analysis (DEA) and missing data which are major problems we solved in this thesis.

## 1.1 Soft computing

The paradigm of soft computing or computational intelligence refers to the seamless integration of different, seemingly unrelated, intelligent technologies such as fuzzy logic, neural networks, genetic algorithms, machine learning (case-based reasoning and decision trees subsumed), rough set theory and probabilistic reasoning in various permutations and combinations to exploit their strengths. This term was coined by Zadeh (1994) in the early 1990s to distinguish these technologies from the conventional ‘‘hard computing’’ that is inspired by the mathematical methodologies of the physical sciences and focused upon precision, certainty and rigor, leaving little room for modeling error, judgment, ambiguity, or compromise. In contrast, soft computing is driven by the idea that the gains achieved by precision and certainty are frequently not justified by their costs, whereas the inexact computation, heuristic reasoning and

subjective decision making performed by human minds are adequate and sometimes superior for practical purposes in many contexts. Soft computing views the human mind as a role model and builds upon a mathematical formalization of the cognitive processes those humans take for granted (Zadeh, 1994). Within the soft computing paradigm, the predominant reason for the hybridization of intelligent technologies is that they are found to be complementary rather than competitive in several aspects such as efficiency, fault and imprecision tolerance and learning from example (Zadeh, 1994). Further, the resulting hybrid architectures tend to minimize the disadvantages of the individual technologies while maximizing their advantages. Some of the soft computing architectures employed are neuro–fuzzy, fuzzy–neural, neuro–genetic, genetic–fuzzy, neuro–fuzzy–genetic, rough–neuro, etc. Multi-classification systems or ensemble classifiers are also treated as soft computing systems.

## 1.2 Data Envelopment Analysis (DEA)

Data Envelopment Analysis (DEA) (Charnes, Cooper and Rhodes, 1978, 1981) is a well established technique which uses linear programming methodology to measure the relative efficiency of homogeneous entities called Decision Making Units (DMUs). DMUs are non-profit organizations, where the measurement of performance efficiency is difficult. The efficiency of commercial organization can be assessed by their yearly profits or stock market indices. The technique aims to measure, how efficiently a DMU uses the resources available to generate a set of outputs.

In DEA, the performance of DMUs is assessed by using the concept of efficiency, which is the ratio of total output to total inputs. DEA gives relative efficiencies of DMUs by comparing with the best performing DMUs. The best performance of other DMUs varies, between 0 and 100 percent relative to their performance.

The applications of DEA in various fields have proliferated in the last two decades. It is most useful when a comparison is sought against "best practices" where the analyst doesn't want the frequency of poorly run operations to affect the analysis. DEA has been applied in many situations such as: Health care (hospitals, doctors), Education (schools, universities), Banks, Manufacturing, Benchmarking, Management evaluation, Fast food restaurants, Retail stores.

**Strengths of DEA**

As the earlier list of applications suggests, DEA can be a powerful tool when used wisely. A few of the characteristics that make it powerful are:

- DEA can handle multiple input and multiple output models.
- It doesn't require an assumption of a functional form relating inputs to outputs.
- DMUs are directly compared against a peer or combination of peers.
- Inputs and outputs can have very different units.

**Limitations of DEA**

The same characteristics that make DEA a powerful tool can also create problems. An analyst should keep these limitations in mind when choosing whether or not to use DEA.

- Since DEA is an extreme point technique, noise (even symmetrical noise with zero mean) such as measurement error can cause significant problems.
- DEA is good at estimating "relative" efficiency of a DMU but it converges very slowly to "absolute" efficiency. In other words, it can tell you how well you are doing compared to your peers but not compared to a "theoretical maximum."
- Since DEA is a nonparametric technique, statistical hypothesis tests are difficult and are the focus of ongoing research.
- Since a standard formulation of DEA creates a separate linear program for each DMU, large problems can be computationally intensive.

## 1.3 Missing Data

Missing data in real life data sets is an unavoidable problem in many disciplines. For analyzing the available data completeness and quality of the data plays a major role, because the inferences made from a complete data are more accurate than those made from an incomplete data (Abdella and Marwala, 2005). Data in the databases may be missed because of data entry errors, system failures at the time of data retrieval or several other reasons like sensor failures, noisy channels cultural issues in updating the databases etc. According to Little and Rubin (1987), missing data is categorized into 3 categories: (i) missing completely at random (MCAR), (ii) missing at random (MAR), (iii) missing not at random (MNAR). MCAR occurs if the probability of missing variable X does not depends on the values of any other variable in the dataset. This

means that the value of missing variable is unrelated to any other variable. For example, if the age of the husband is missed in customers database then it does not depend on the any other variable of database which is meant for wife. MAR occurs if the probability of a missing variable X depends on the other remaining variables in that dataset but not on the variable X. For example, income of a person is missed because of missing in profession and age. MNAR occurs when the probability of a missing variable X depends on the variable X itself. For example, if citizens did not participate in a survey, then MNAR occurs. MCAR and MAR data are recoverable, whereas MNAR is irrecoverable.

Missing data creates various problems in many research fields like data mining, mathematics, statistics and various other fields (Abdella and Marwala, 2005). The process of replacing or estimating missing data is called data imputation. Data imputation is very useful for data mining applications for getting completeness in the data. For analyzing the data through any technique completeness and quality of data are very important things. For example researchers rarely find the survey data set that contains complete entries (Hai and Shouhong, 2010). The respondents may not give complete information because of negligence, privacy reasons or ambiguity of the survey questions. But the missing parts of variables may be important things for analyzing the data. So in this situation data imputation plays a major role. Data imputation is also very useful in the control based applications like traffic monitoring, industrial process, telecommunications and computer networks, automatic speech recognition, financial and business applications, and medical diagnosis etc.

To impute with incomplete or missing data, several techniques are reported based on statistical analysis (Garcia-Laencia, Sancho-Gomez and Fiqueiras-Vidal, 2010). These methods include like mean substitution methods, hot deck imputation, regression methods, expectation maximization, multiple imputation methods. Some other techniques proposed based on machine learning methods include SOM, K-Nearest Neighbor, multi layer perceptron, recurrent neural network, auto-associative neural network imputation with genetic algorithms, and multi-task learning approaches.

## 1.4 Organization of the Thesis

The rest of the thesis is organized as follows.

Chapter 2 provides the literature review on data envelopment analysis and its application areas and also about the models developed for developing common set of weights for DEA. It also provides missing data imputation techniques in the literature and the methods developed against phishing attacks.

Chapter 3 provides the details about Multi objective optimization of Data Envelopment Analysis by Differential Evolution. Section 3.1 presents the overview about DEA and the shortcomings about DEA. Section 3.2 and 3.3 describes about Differential Evolution algorithm NSGA-II. Detailed overview of proposed method and its implementation using DE presented in section 3.4. Section 3.5 provides the description of datasets used in this study. Results are discussed in section 3.6. Finally conclusions are given in section 3.7.

Chapter 4 presents the details about Novel Soft Computing Hybrid for Data Imputation. Section 4.1 and 4.2 presents the overview of K-Means algorithm and MLP respectively. Proposed methodology is presented in section 4.3. Section 4.4 describes the dataset analyzed in this study. Experimental procedure presented in section 4.5. Results are discussed in section 4.6. Conclusions are given in section 4.7.

Chapter 5 presents the details about assessing severity of phishing attacks. Section 5.1 presents the overview phishing attacks. Details about dataset used in this study described in section 5.2. Section 5.3 describes the methodology used in this study. Results are discussed in section 5.4. Conclusions are given in section 5.5.

Chapter 6 provides the overall conclusion of the thesis.

# 2

# Literature Review

## 2.1 Data Envelopment Analysis

In literature, different techniques are proposed for performance analysis. Among all, the technique Data Envelopment Analysis (DEA) (Charnes, Cooper and Rhodes, 1978, 1981) is widely used in measuring technical efficiency of an entity (it can be a University, Hospital, Company etc- usually referred to as Decision Making Unit (DMU) in the parlance of DEA) relative to other similar DMUs. The applications of DEA in various fields have proliferated in the last two decades. In the banking industry DEA is considered as a viable technique to measure the efficiency of banks for last two decades. Several researchers (Sherman and Gold, 1985; Oral and Yolalan, 1990; Bhattacharyya, Lovell and Sahay, 1997) examined the productive efficiency of Indian commercial banks during 1986–1991. They reported that after 1987, a marginal increase in overall average performance and the average efficiency of publicly owned banks is much higher than that of the privately owned or foreign owned banks. Saha and Ravi Sankar (2000) used DEA to rate the Indian banks with respect to their relative performance. Shanmugam and Das (2004) on the other hand investigated the efficiency of Indian commercial banks during the reform period, 1992–1999 using a parametric methodology. Zhou et al. (2008) present a literature survey on the application of DEA to energy and environmental studies. Dula (2008) studied the aspects of computational performance and scale limits of the standard LP-based procedures currently used in DEA. They proposed ideas for extending DEA into a data mining tool. Johnson and

McGinnis (2008) propose using both the efficient frontier and the inefficient frontier to identify outliers and thereby improving the accuracy of the second stage results in a two-stage nonparametric analysis of DEA.

Later researchers introduced neural networks for solving DEA problem. To mention a few; Athanasopoulos and Curram (1996) pioneered this hybridization and treated DEA as preprocessing methodology to train a Neural Network to forecast the number of employees in a healthcare industry. Then, Costa and Markellos (1997) used this hybrid technique in application to London underground efficiency analysis and compared with DEA. Then, Wang (2003) proposed a neural network model in estimating efficiencies by creating frontier functions for efficiencies. Then, Santin et al. (2004) reported that MLP can be applied, even though DEA with variable return to scale is giving slightly better results than MLP. Wu et al. (2006) implemented DEA with Neural Networks for measuring efficiency for the branches of large Canadian bank. Hu, Chung and Chen (2008) employed Hopfield Neural Network in predicting the efficiencies. Most recently Emrouznejad and Shale (2009) proposed a combined neural network and DEA method and found to be helpful in the case of large number of DMUs. This approach is helpful in reducing computer resources, processing time and complexity in case of large datasets.

Common set of weights are most preferable for comparing homogeneous units. The idea of common set of weights and the ranking of DMUs by using common set of weights has developing gradually. Kao and Hung (2005), proposed a method to generate a common set of weights for all DMUs. The model is based on multiple objective nonlinear programming. The weights are able to produce a vector of efficiency scores closest to the efficiency scores of classical DEA model. Liu et al. (2006) proposed a method for ranking the efficient DMUs by identifying the most compromising common weights. Makui et al. (2008) proposed a goal programming approach for finding common weights in DEA and they compared the ranks with the ranks obtained through Kao et al. proposed methods. Liu and Peng (2008) proposed a model for determining common set weights to create the best efficiency score among the group of efficient DMUs. Then they use this common set of weights to evaluate the absolute efficiency of each efficient DMUs in order to rank them. Jahanshahloo et al. (2010) determine the common set of weights by defining an ideal line for efficient DMUs then a new efficiency score will be obtained and ranked the DMUs.

Hosseinzadesh et al. (2010), gave the relationship between Multi objective linear programming (MOLP) and DEA, and solved the DEA problem by transforming into MOLP.

## 2.2 Missing Data Imputation Methods

Missing data handling methods can be broadly classified into two categories: deletion and imputation (Gheyas and Smith, 2010). The missing data ignoring techniques or deletion techniques simply delete the cases that contain missing data. Because of their simplicity, they are widely used (Roth, 1994) and tend to be the default for most statistics packages, but this solution is not an effective solution. This approach has two forms: $(i)$ Listwise deletion omits the entire cases or records containing missing values. The main drawback of this method is that the application may lead to large loss of observations, which may result in high inaccuracy in particular if the original dataset is itself too small (Song and Shepperd, 2007). $(ii)$ Pairwise deletion method considers each feature separately. For each feature, all recorded values in each observation are considered and missing data ignored (Strike et al., 2001). Unlike list wise deletion which removes cases (subjects) that have missing values on any of the variables under analysis, pair wise deletion only removes the specific missing values from the analysis (not the entire case). It is good when the overall sample size is small or missing data cases are large (Song and Shepperd, 2007).

On the other hand, imputation method uses the available data to estimate the missing values. The earliest method of imputation is mean imputation, in which the missing values of a variable are replaced with the average value of all remaining cases of that particular variable (Little and Rubin , 1987). The disadvantage of this method is that it ignores the correlations between various components (Schafer, 1997). When the variables are correlated data imputation can be done with regression imputation. In the regression imputation regression equations are fitted each time by making the variable with incomplete values as the target variable. This method preserves the variance and covariance of missing data with other variable. Hot and cold deck imputation replaces the missing values with the closest complete components. Closest is in terms of components that are present in both vectors for each case with a missing value (Schafer, 1997). The drawback with hot deck imputation is that the estimation of missing data is based on single complete vector. It ignores the global properties of the

dataset. The drawback of cold deck imputation is that missing values are replaced with the different dataset values (Little and Rubin , 1987). In multiple imputation procedure, each missing value is replaced with a set of reasonable and valid values, so that we will get M complete datasets by replacing each value M times and by analyzing all datasets we can make combined inferences. According to (Little and Rubin , 1987), multiple imputation is better than case wise and mean substitution. Regression methods are not as effective as multiple imputation. Expectation maximization is an iterative process that continues until there is convergence in the parameter estimates.

In K-nearest neighbor (K-NN) approach the missing values are replaced with nearest neighbors. The nearest neighbors are the complete components which minimize the distance. In this Method, K nearest neighbors are selected from the complete cases or components. Jerez, Molina, Subirates, and Franco (2006) used K-NN for breast cancer prognosis. Batista and Monard (2002, 2003) also used K-NN for missing data imputation. Samad and Harp (1992) implemented SOM approach for handling the missing data. In this imputation once the training of SOM is over with complete records, then incomplete pattern is presented to SOM, its image node is chosen ignoring the distances in the missing variables. An activation group composed of image nodes neighbor is selected. Based on the weights of activation group of the nodes in the missing dimensions the missing values are imputed.

In Multi layer perceptron approach, by using only the complete cases MLP should be trained as nonlinear regression model by making each time one variable as target. By using appropriate MLP model, each incomplete pattern values are predicted. Several researchers (Sharpe and Solly, 1995; Nordbotten, 1996; Gupta and Lam, 1996; Yoon and Lee , 1999, Kallin, 2002) used MLP scheme for missing data imputation. Imputation using auto-associative neural network (AANN) is another machine learning technique. In AANN the network is trained for predicting the some inputs by taking same input variable as target variable. Researchers (Marseguerra and Zoia, 2005 and Marwala and Chakraverty, 2006) developed imputation models based on AANN.

## 2.3 Assessing Severity of Phishing attacks

Phishing has aroused great interest among information security researchers. Understanding the critical success factors of phishing and determining methods that can

prevent or detect such a crime has been a popular area of research. We can roughly split current research on phishing into three streams, namely, phenomenal studies, economic analysis, and technical research.

As an example of a phenomenal study related to phishing, Jagatic et al. (2006) found that the social engineering skill of the adversary was a critical success factor for phishing. Dhamija and Tygar (2005) discovered that lack of knowledge, inability to control visual deception, and lack of attentiveness to detail is the major weaknesses of people who fall prey to phishing attacks. Interestingly, Workman found that the critical success factors for some marketing strategies were applicable to phishing attacks as well (Workman & Wisecrackers, 2008). Researchers also found that education of customers, standardization of technology, and sharing of phishing information were among the most important policies that could deter phishing attacks. Some researchers conducted experimental studies and confirmed that if a user was trained to identify phishing attacks, the chance of being cheated in future was significantly lowered.

Among economic studies related to phishing, Jakobsson (2007) classified the costs of phishing into three categories, namely, direct cost, indirect cost, and opportunity cost. Singh (2007) studied a number of international phishing incidents and found that the direct financial loss per incident ranged from US $900 to 6.5 million pounds. However, it is widely believed that as companies are quite reluctant to disclose information related to direct financial loss caused by phishing, the actual financial loss might be ten times more than the estimated numbers that appeared in research reports. In their attempt to estimate the indirect financial loss caused by phishing, Leung and Bose (2008) found that phishing related announcements caused a significant negative reaction among investors of targeted companies. It is interesting to note that a significant negative investor reaction of 2.1% loss in market value within two days of the announcement was reported in the broader context of analyzing the economic impact of information security breaches (Kannan et al., 2007).

In the area of technical research, information security researchers have toiled to discover better countermeasures of phishing. A number of anti-phishing toolbars and phishing filters have been developed. Data mining based approaches have been frequently adopted in the development of such countermeasures. Data mining techniques have been used to filter out phishing emails that contained fraudulent

messages (Airoldi & Malin, 2004). By analyzing the headers of emails, researchers were able to prevent the spread of malicious emails containing virus/worms/Trojans, and stop crimes such as phishing and distributed denial of service attacks with an accuracy of 99% (Zhang et al., 2007). Among the various data mining techniques that have been adopted for determination of phishing emails are support vector machines (Chandrasekaran et al., 2006), random forest (Fette et al., 2007) one-step ternary and repeated binary classification techniques (Gansterer & Polz, 2009) and ensemble methods (Saberi et al., 2007). To authenticate the URL embedded in the emails, logistic regression (Garera et al., 2007) and decision trees have also been used (Ludl et al., 2007). The focus of the extant research was on analysis of the characteristics of the emails and determination of the malicious nature of the emails. However, the focus of the current research is on the assessment of the influence of such phishing emails. The use of data mining techniques in research related to information security is not new. Zhu et al. (2001) had used rough sets, neural networks, and decision trees for detection of network intrusion. They tried different combinations of classification tools and data representation format, and experimented with variation in the proportion of training and testing data. They showed that rough sets performed best when the data was presented in binary format, and the proportion of training and testing data was balanced (Zhu et al., 2001). Zhao et al. (2008) proposed a hybrid system for network intrusion detection. The system consisted of three main components: service pattern databases that were used to detect the network traffic patterns for different services, anomaly detection module that was based on unsupervised clustering and detected anomalous network traffic patterns, and a random forest module that was used to differentiate between intrusion cases and normal cases of service usage. Zhao and Huang (2002) proposed a data mining approach that mimicked the human immune system and detected network intrusion. Ansari et al. (2007) detected misuse and anomalies using a soft computing method like fuzzy logic. From the various examples cited in this paragraph we can see that data mining techniques have been favored by researchers in the area of information security. For a comprehensive review on this topic the interested reader may refer to Tsai et al. (Tsai et al., 2009). However, past research mainly focused on the detection of security events such as misuse, anomalies, intrusion, and other types of security breaches. Research on the use of data mining techniques to assess the influence of security events such as phishing attacks was limited.

# 3

# Multi objective optimization of Data Envelopment Analysis by Differential Evolution

## 3.1 Introduction

With the increasing competition among the organizations such as banks, airlines, hospitals, universities etc, measuring their efficiency and rating them has become extremely important.

In general, the efficiency is measured can be done by using the following equation:

$$Efficiency = \frac{Output}{Input} \tag{1}$$

Let $X$ and $Y$ denote inputs and outputs for any $DMU$, $I$ and $J$ be the total number of inputs and outputs of a $DMU$ where $I, J > 0$. Lets $D$ be the total number of $DMUs$. Let $X_{1p}, X_{2p}, X_{3p} \ldots\ldots X_{Ip}$ represents the inputs of $p^{th}$ DMU and $Y_{1p}, Y_{2p}, Y_{3p} \ldots\ldots Y_{Jp}$ represent its corresponding outputs. For $p^{th}$ DMU, let $U_{ip}$ represent the weight assigned to $i^{th}$ input $U_i$ $(i = 1\ to\ J)$ and $V_j$ denote the weight assigned ti the $j^{th}$ output $V_j$ $(j = 1\ to\ J)$. Let $E_1, E_2 \ldots\ldots E_D$ are the Efficiencies of '$D$' DMUs.

The virtual input of a firm is obtained as the linear weighted sum of all its inputs

$$Virtual\ Input = \sum_{i=1}^{I} U_i * X_i \tag{2}$$

Where $U_i$ is the weight assigned to input $I_i$ during the aggregation

Virtual output of a firm is obtained as the linear weighted sum of all its outputs.

$$Virtual\ output = \sum_{i=1}^{J} V_i * Y_i \quad (3)$$

Where $V_j$ is the weight assigned to input $O_j$ during the aggregation

$$Efficiency = \frac{Virtual\ Output}{Virtual\ Input} = \frac{\sum_{j=1}^{J} V_{jp} * Y_{jp}}{\sum_{i=1}^{I} U_{ip} * X_{ip}} \quad (4)$$

The most important issue at this stage is the assessment of weights. The set of weights for a given DMU is determined by using Linear Programming.

### 3.1.1 Fractional DEA Programs

Let there be $N\ \ DMUs$ whose efficiencies have to be maximized. Let us take one of the DMUs, say the $p^{th}$ DMU, and maximize its efficiency according to the formula given (Charnes, Cooper and Rhodes, 1978, 1981; Ramanathan, 2003).

$$Max \left\{ \ E_p = \frac{\sum_{j=1}^{J} V_{jp} * Y_{jp}}{\sum_{i=1}^{I} U_{ip} * X_{ip}} \right. \quad (5)$$

Subject to

$$0 \leq \frac{\sum_{j=1}^{J} V_{jp} * Y_{jt}}{\sum_{i=1}^{I} U_{ip} * X_{it}} \leq 1\ ;\ t = 1,2, \dots D$$

$$U_{ip}, V_{jp} \geq 0\ ;\ i = 1,2,3, \dots . k, I;\ j = 1,2,3, \dots . k, J$$

### 3.1.2 General Form of CCR DEA models

***3.1.2.1 Output Orientation Model:*** General output maximization CCR model can be represented as follows (Charnes, Cooper and Rhodes, 1978, 1981; Ramanathan, 2003).

$$\text{Max } Z = \sum_{j=1}^{J} V_{jp} * Y_{jp} \quad (6)$$

Subject to

$$\sum_{i=1}^{I} U_{ip} * X_{ip} = 1.$$

$$\sum_{j=1}^{J} V_{jp} * Y_{jp} - \sum_{i=1}^{I} U_{ip} * X_{ip} \leq 0 \; ; p = 1,2, \ldots D.$$

$$U_{ip}, V_{jp} \geq 0 \; ; \; i = 1,2,3, \ldots . k, I; \; j = 1,2,3, \ldots . k, J.$$

***3.1.2.2 Input Orientation Model***: Similarly, a general input minimization CCR DEA model (Charnes, Cooper and Rhodes, 1978, 1981; Ramanathan, 2003). can be represented as follows

$$\text{Min Z} = \sum_{i=1}^{I} U_{ip} * X_{ip} \qquad (7)$$

Subject to

$$\sum_{j=1}^{J} V_{jp} * Y_{jp} = 1.$$

$$\sum_{j=1}^{J} V_{jp} * Y_{jp} - \sum_{i=1}^{I} U_{ip} * X_{ip} \leq 0 \; ; p = 1,2, \ldots D.$$

$$U_{ip}, V_{jp} \geq 0 \; ; \; i = 1,2,3, \ldots . k, I; \; j = 1,2,3, \ldots . k, J.$$

Both these models are solved by applying Simplex Method.

DEA uses linear programming technique in assessing efficiency. In ordered to accomplish this, each time by taking single objective function we need to run the linear programming as many times as the number of DMUs. The traditional DEA model has two important drawbacks. (i) We cannot get a single set of weights for all DMUs. (ii) Another problem with DEA is the fractional objective function viz efficiency which is converted into an equivalent linear objective function with constraints on the denominator. Therefore it gives an approximate solution to the original DEA.

In order to alleviate some of these problems researchers employed Artificial Neural Networks (ANNs) in conjunction with traditional DEA model. Many of the traditional applications of DEA such as bank branches, hospitals, university departments, etc., a centralized decision maker used to control and organize. In this situation, the decision maker is interest to rank efficient DMUs, by using the common set of most compromising weights.

In the case of combined neural network and DEA approach first we need to run conventional DEA and by using those scores as output variable, neural network is trained. However, the quality of training data affects the results of ANN. To train the network conventional DEA should be run as many times as number of DMUs. Thus, even though DEA-ANN model yields single set of weights, it does not remove the complexity of the traditional DEA. Secondly the combined DEA-ANN model yields relative efficiencies but not absolute efficiencies. To overcome the disadvantages of traditional DEA and DEA-ANN hybrids, we propose multi-objective formulation of traditional DEA, for all DMUs which gives single set of weights. And also we are not approximating the original fractional optimization problem and instead, the problem is solved as Non-Linear Programming Problem (NLPP). We used the meta-heuristic Differential Evolution (DE) algorithm to solve the problem.

## 3.2 Differential Evolution (DE) Algorithm

DE proposed by Storn and Price (1997), is a novel approach in the class of evolutionary algorithms. DE is a stochastic, population based optimization method. The flowchart of the algorithm is shown in the Figure 1. DE algorithm mainly consists of four steps: $(i)$ Initialization, $(ii)$ Mutation, $(iii)$ Recombination and $(iv)$ Selection.

Let $\boldsymbol{S} = (y_1, y_2, \ldots . y_n)$ is a vector of $n$ decision variables and $f(S)$ be the objective function which is to be optimized. The aim of DE algorithm is to find a solution vector $S$ in the given search space. The solution should be an optimal solution for the objective function. The search space of each $n$ dimensional variable of $S$ is initialized by providing the lower and upper bounds for every variable, that is

$$s_{i\min} \leq s_i \leq s_{i\max} \quad where\ i = 1\ to\ n$$

In the initialization step, we define the population of $N_p$ vectors, each of $n$ dimensions. All $N_p$ vectors are randomly initialized by using the equation given below.

For each of $N_p$ vectors,

$$s_i = s_{i\,min} + rand(0,1) * (s_{i\,max} - s_{i\,min}) \tag{8}$$

Where $i$ is from 1 to $n$ and $rand(0,1)$ denotes a random number generated between 0 and 1 with uniform distribution.

To direct the search towards potential areas of optimal solution, DE uses a search mechanism in the Mutation phase. In this step, three distinct target vectors$S_a$, $S_b$ and $S_c$ are randomly chosen from the $N_p$ parent population on the basis of three random numbers $a, b$ and $c$, chosen between 1 and $N_p$. One of the vectors $S_c$ is the base of the mutated vector. To this the weighted difference of the remaining two vectors, viz., $(S_a - S_b)$ is added to generate a noisy random vector, $N_i$.

$$N_i = S_c + F * (S_a - S_b) \tag{9}$$

Where $i = 1\ to\ N_p$

$F$ $(0 < F \leq 1.2)$ is user supplied parameter and termed as scaling factor. This mutation process is repeated to create a mate for each member of the parent population.

In the recombination operation, we will get a trial vector $S_i$ , a child of two parent vectors: noisy random vector, $N_i$ and the target vector $S_i$. The vector $S_i$ is recombined with the noisy random vector $N_i$ to generate a trial vector $t_i$. Each element of every trial vector $(t_m, where\ m = 1, \ldots, n)$, is determined by a binomial experiment whose success or failure is determined by the user-supplied parameter called crossover factor, $CR$. The parameter $CR$ is used to control the rate at which the crossover takes place. Based on the problem it is chosen in the range 0 to 1.

For each of $N_p$ trial vectors,

$$t_m = \begin{cases} n_m, \ if\ \ rand\ (0,1) < CR\ \ or\ \ m = rand\ (1,n) \\ s_m, \ Otherwise \end{cases} \tag{10}$$

Where $m$ = 1 to $n$.

If the trial vector is found to be violating the upper bound, error is calculated by subtracting the upper bound from the trial vector. Then the difference between the upper bound and the found error is taken as the new trial vector. If this newly generated trial vector still violates the lower bound, trial vector is regenerated using equation (8).

If the trial vector is found to be violating the lower bound and upper bound, it is brought within bounds using the equations as follows.

$$t_i = s_{i\min} + 2.0 * (p/q) * (s_{i\max} - s_{i\min}), \quad (11)$$
$$If \quad t_i > s_{i\max}; \qquad with \quad p = t_i - s_{i\max}, q = t_i - s_{i\min}$$

$$t_i = s_{i\min} + 2.0 * (p/q) * (s_{i\max} - s_{i\min}) \quad (12)$$
$$If \quad t_i < s_{i\min} \; ; \; with \quad p = s_{i\min} - t_i, q = s_{i\max} - t_i$$

It is in the last stage of 'selection' that the fitter of the two vectors (trial vector and target vector) survives and proceeds to the next generation. The vector having minimum value of objective function goes to next generation. This procedure is similar to the 'tournament selection' (Deb, 2000).

## 3.3 Overview of NSGA-II

Non-dominated sorting genetic algorithm II (NSGA-II) proposed by Deb et al (2002) to eliminate the main criticisms (like high computational complexity of non-dominated sorting, lack of elitism, need for specifying the sharing parameter) of NSGA (Srinivas and Deb, 1995), which was one of the first such Evolutionary Algorithms. The currently used non-dominated sorting algorithm has a computational complexity of $O(MN^3)$ (where $M$ is the number of objectives and $N$ is the population size). This makes NSGA computationally expensive for large population sizes. In NSGA-II the computational; complexity reduced to $O(MN^2)$. NSGA-II does not require any user-defined parameter for maintaining diversity among population members. Also, this approach has a better computational complexity.

The procedure of NSGA-II is described as follows. Initially, parent population $P_0$ is created randomly. For each solution assigned the ranks based on the non-domination. The offspring population $Q_0$ of size $N$ is created by binary tournament selection, recombination, and mutation operator. We form a population $R_t$ of size $2N$ is formed by combining $P_0$ and $Q_0$ ($R_t = P_0 \, U \, Q_0$). Then, the population $R_t$ is sorted according to non-domination. Now, solutions belonging to the best non-dominated set $f_1$ are of best solutions in the combined population and must be emphasized more than any other solution in the combined population. If the size of $f_1$ is smaller than $N$, we definitely

choose all members of the set $f_1$ for the new population $P_{t+1}$. The remaining members of the population $P_{t+1}$ are chosen from subsequent non-dominated fronts in the order of their ranking. Thus, solutions from the set $f_2$ are chosen next, followed by solutions from the set $f_3$, and so on. This procedure is continued until no more sets can be accommodated. Say that the set $f_l$ is the last non-dominated set beyond which no other set can be accommodated. Say that the set $f_1$ is the last nondominated set beyond which no other set can be accommodated. In general, the count of solutions in all sets from $f_1$ to $f_l$ would be larger than the population size. To choose exactly $N$ population members, we sort the solutions of the *last* front $f_l$ using the crowded-comparison operator $< n$ in descending order and choose the best solutions needed to fill all population slots. The new population $P_{t+1}$ of size $N$ is now used for selection, crossover, and mutation to create a new population $Q_{t+1}$ of size $N$. It is important to note that we use a binary tournament selection operator but the selection criterion is now based on the crowded-comparison operator $< n$. Since this operator requires both the rank and crowded distance of each solution in the population, we calculate these quantities while forming the population $P_{t+1}$, as shown in the above algorithm.

Consider the complexity of one iteration of the entire algorithm. The basic operations and their worst-case complexities are as follows:

1) Non dominated sorting is $O(M(MN)^2)$;
2) Crowding-distance $< n$ assignment is $O(M(2N)\log(2N))$;
3) Sorting on is $O(2Nlog(2N))$.

The overall complexity of the algorithm is $O(MN^2)$, which is governed by the non dominated sorting part of the algorithm. If performed carefully, the complete population of size $2N$ need not be sorted according to non domination. As soon as the sorting procedure has found enough number of fronts to have $N$ members in $P_{t+1}$, there is no reason to continue with the sorting procedure.

The diversity among non dominated solutions is introduced by using the crowding comparison procedure, which is used in the tournament selection and during the population reduction phase. Since solutions compete with their crowding-distance (a measure of density of solutions in the neighborhood), no extra niching parameter is required. Although the crowding distance is calculated in the objective function space, it can also be implemented in the parameter space, if so desired.

## 3.4 Multi Objective Data Envelopment Analysis (MODEA) via Differential Evolution

In this method, we formulated the original DEA problem as a Multi objective non-linear optimization problem in two distinct ways: (i) Maximizing the sum of the efficiencies of all DMUs (ii) Maximizing the minimum of the efficiencies of all DMUs. Accordingly, when we maximize the minimum efficiency, all the remaining efficiencies will also reach their maximum. In both cases we employed DE algorithm to solve the resulting single objective non-linear optimization problem. These two variants are named as MODEA1-DE and MODEA2-DE respectively. In both the models we get the results that are close to absolute efficiencies. Now we describe both the variants as follows:

### 3.4.1 MODEA1

$$\begin{aligned} &\text{Max } E_1 \\ &\text{Max } E_2 \\ &\cdot \\ &\cdot \\ &\cdot \\ &\text{Max } E_D \end{aligned} \tag{13}$$

Subject to

$$E_p = \frac{\sum_{j=1}^{J} V_{j*} Y_{jp}}{\sum_{i=1}^{I} U_{i*} X_{ip}}$$

$$0 \leq E_p \leq 1 \; ; \; p = 1,2,3, \ldots . D.$$

$$U_{ip}, V_{jp} \geq 0 \; ; \; i = 1,2,3, \ldots . k, I; \; j = 1,2,3, \ldots . k, J.$$

Now, we follow the scalar optimization approach and formulate the Multi objective problem (Equation 13) as a single objective optimization problem. Since, the efficiencies are non-linear the optimization problem is also non-linear.

Thus, the scalar optimization problem with equal weights for all objective functions.

$$Max \ \sum_{p=1}^{D} E_p \qquad (14)$$

Subject to

$$E_p = \frac{\sum_{j=1}^{J} V_j * Y_{jp}}{\sum_{i=1}^{I} U_i * X_{ip}}.$$

$$0 \leq E_p \leq 1 \ ; \ p = 1,2,3, \ldots . D.$$

$$U_{ip}, V_{jp} \geq 0 \ ; \ i = 1,2,3, \ldots . k, I; \ j = 1,2,3, \ldots . k, J$$

### 3.4.2 MODEA2

Similarly by using the Max-Min approach we formulated the multi objective optimization problem as a single objective optimization problem as follows:

$$Max \ \{ \ Min \ ( E_1, E_2, E_3, \ \ldots \ , E_D) \ \} \qquad (15)$$

Subject to

$$E_p = \frac{\sum_{j=1}^{J} V_j * Y_{jp}}{\sum_{i=1}^{I} U_i * X_{ip}}.$$

$$0 \leq E_p \leq 1 \ ; \ p = 1,2,3, \ldots . D.$$

$$U_{ip}, V_{jp} \geq 0 \ ; \ i = 1,2,3, \ldots . k, I; \ j = 1,2,3, \ldots . k, J$$

### 3.4.3 MODEA1 implementation with DE

1 The first step is the random initialization of the parent population of solutions. Generate randomly $N_p$ vectors, each of $n$ dimensions using equation given below

$$s_i = s_{i\,min} + rand(0,1) * (s_{i\,max} - s_{i\,min}) \qquad (16)$$

2 Calculate the objective function $\boldsymbol{maximize - sum}( )$ values, $f(S_i)$, for all $S_i$,

$i = 1 \ to \ N_p$

Do the following until $maximum$ number of iterations are completed.

3 Select three random numbers $a$, $b$ and $c$ within the range 1 to $N_p$. The

weighted difference ($S_a$ –$S_b$) is used to perturb $S_c$ to generate a noisy vector $N_i$ as follows:

$$N_i = S_c + F * (S_a - S_b) \quad (17)$$

4 Recombine each target vector $S_i$ with the noisy random vector $N_i$ to generate a trial vector $t_i$ as follows:

$$t_m = \begin{cases} n_m, \ if \ \ rand(0,1) < CR \ \ or \ \ m = rand(1,n) \\ s_m, \ Otherwise \end{cases} \quad (18)$$

5 Check whether each decision variable of every trial vector is within the bounds. Otherwise force it within the bounds using:

$t_i = \ \ t_{i\,min} + 2.0 * (p/q) * (s_{i\,max} - s_{i\,min})$, (19)

If $t_i > \ s_{i\,max}$ , $with \ \ p = t_i - s_{i\,max}$ , $q = t_i - s_{i\,min}$

$t_i = t_{i\,min} + 2.0 * (p/q) * (s_{i\,max} - s_{i\,min})$, (20)

If $t_i < \ s_{i\,min}$ , $with \ \ p = s_{i\,min} - t_i$ , $q = s_{i\,max} - t_i$

6 Calculate the objective function $\boldsymbol{maximize - sum}(\,)$ values, $f(t_i)$, for all $t_i$,

$i = 1 \ to \ N_p$

7. If $(f(t_i) < \ f(y_i)$ then $\ s_i = t_i$

8. Take the best solution of the $i^{th}$ iteration from $s$ into $b_i$

9. If $(b_i - b_{i-1} \leq convergencecriteria)$ then $goto \ step$ 12

10. Else if $\ number \ of \ iterations$ less than $maximum \ iterations$ then $goto \ step$ 3

11. Else stop the iteration loop

12. Take the best solution from $s$ into $best$

### 3.4.4 MODEA2 implementation with DE

In this model everything is similar to the MODEA1 except that the object function is changed here to implement Max-Min problem. Hence the algorithm is not presented in detail.

## 3.5 Description of the Data sets

We tested the proposed models on 8 datasets taken from literature. Those are $(i)$ Data of Physics Departments across several universities, $(ii)$ Data of Chemistry Department across several universities, $(iii)$ Data of State Transport Undertakings across all the states in India, $(iv)$ Data of Insurance Companies in Iran, $(v)$ Data of Bank Branches in Iran, $(vi)$ Data of public libraries in Tokyo, $(vii)$ Data of electric power generation companies in Japan, and $(viii)$ Data of Information Technology firms. The Inputs and Outputs of each dataset are shown in Table 1.

Both the Datasets of Physics Departments and Chemistry Departments, taken from several universities in UK (Beasley, 1990), consist of 3 inputs and 3 outputs each. Inputs include general expenditure, equipment expenditure and research income. The measures of output are the number of undergraduate students, the number of postgraduates on taught courses and the number of postgraduates doing research. The number of DMUs in the datasets of Physics Department and Chemistry Department are 50 and 52 respectively.

Dataset of State Transport Undertakings (STUs) in India (Ramanathan, 2003) consists of 29 DMUs with 3 inputs and 1 output. Inputs are Fleet Size, total staff and diesel consumption. The output attribute is passenger travelled in kilometers. The dataset of Information Technology Investment in banks (Chen, Liang, Yang, Zhu, 2006) consists of 27 DMUs with 4 inputs and 2 outputs. The inputs are fixed assets, IT budget, number of employees, deposits, and the outputs are profit, fraction of loans recovered. The dataset Carbon-dioxide emissions (Ramanathan, 2003) consists of 64 DMUs with 2 input attributes, viz, per capita carbon-dioxide Emissions and fossil fuel energy consumption per capita and outputs are non-fossil fuel energy consumption per capita and gross domestic product per capita. The data of public libraries in Tokyo (Cooper, Seiford, Tone, 2003) consists of 23 DMUs with 4 inputs and 2 outputs. The inputs

taken are occupying area, number of books, number of staff and population that using. The outputs are number of registered users and number of books borrowed. Japanese Electric power Generation Companies data (Cooper, Seiford, Tone, 2003) consists of 25 DMUs. Each DMU has 3 inputs viz. Generation Capacity, Fuel Consumption and the number of Employees. The attribute Total Generation is the output. The data on Information Technology firms (Ramanathan, 2003) have 36 DMUs. IS Budget as percentage of Revenues, Processer Value as percentage of Revenues, Training Budget as percentage if IS Budget are the 3 inputs and 5 Yr Compound Annual Revenue Growth, 5 Yr Compound Annual Income Growth are the 2 outputs.

## 3.6 Results and Discussions

The proposed MODEA1 and MODEA2 with DE are implemented in ANSI C, using Microsoft Visual C++ compiler. The numerical simulations were performed on a Pentium 4 PC with 512 MB RAM and 256 MHz clock speed on Microsoft Windows XP environment.

In addition to the proposed models, we also solved the Milti-objective optimization problem using NSGA-II developed by Deb et al (2002).

Both MODEA1 and MODEA2 were run on 8 datasets taken from Literature. We compare the efficiencies obtained by our proposed models, and NSGA-II with that of conventional DEA-CCR. Here, we used the approach proposed by Wu et al (2006). Thus, the results are grouped into four categories based on the efficiency scores. The *strong efficient* DMUs are those whose efficiencies lie in the interval (0.98, 1]. If the efficiency scores lies in the interval (0.80, 0.98] then those DMUs are considered to be *efficient*. The efficiency score interval of (0.50, 0.80] is referred to as *inefficient* interval. The efficiency score interval of (0, 0.50] is referred to as *very inefficient* interval. The comparison of conventional DEA-CCR, with MODEA1, MODEA2 and NSGA-II models is presented in Table 2. From the results we can observe that the number of DMUs that fall in a particular interval differed in case of MODEA models DEA-CCR, and in most of the datasets, the number of DMUs in *strong efficient* and *efficient* intervals is more in case of conventional DEA-CCR than in MODEA1 and MODEA2. This is because in conventional DEA-CCR model we maximize the efficiency of one DMU by keeping constraints on the efficiencies of other DMUs. This

is a single objective optimization setup. Hence, we tend to overestimate the efficiencies of DMUs. But in our models the original non-linear optimization is solved as it is without approximating it to a linear programming problem. Further, we solve the Non-linear programming problem in a multi objective optimization framework in a single shot by using DE. This is in contrast with the DEA-CCR model, where linear programming problem is to be solved as many number of times as the number of DMUs. In the process, we maximize the efficiencies of all DMUs simultaneously. Consequently we will obtain a single set of weights for all DMUs. The results indicate that in the case of NSGA-II and MODEA models, the number of DMUs falling in an interval is almost same. Further, the number of DMUs falling in a particular range is almost same in case of MODEA1, MODEA2 and NSGA-II. From the results we can infer that MODEA1, MODEA2 and NSGA-II models yielded different pareto-optimal solutions. In case of NSGA-II at the end all the populations are yielding to unique solution.

From the results, MODEA models are giving better efficiencies for 83 % of DMUs than NSGA-II in case of Chemistry department, IT investigation, CO2 emissions, and IT Firms. 73% of DMUs obtained more efficiency in case of MODEA than NSGA-II in case of all other datasets. MODEA1 model is giving better efficiencies for 63% of DMUs in case of Physics Department dataset, RTC dataset. 54% of DMUs obtained better efficiencies in case ofMODEA1 than MODEA2 for the datasets IT Firms, Public libraries in Tokyo, IT investigation. For all the remaining datasets (Chemistry Departments in UK, Japanese electric power generation, CO2 Emissions) 76% of DMUs obtained more efficiency with MODEA1 model than MODEA2.

To measure the similarities between the ranks of DMUs MODEA1, MODEA2, NSGA-II models and conventional DEA-CCR we used the Spearman's rank correlation co-efficient. Spearman's Rho values obtained for all the datasets are presented in Table 3. All the values are statistically insignificant at 1% level of significance. The correlation between the efficiency scores obtained by proposed models and DEA-CCR is greater than 99%. Thus we observe that if we compare the models by taking two at a time, the rank ordering is found to be very similar for all the models. However, in the proposed method we are solving fractional objective as it is without converting into LPP in a multi-objective framework and also we are obtaining common set of weights. The

comparison of the ranks of DMUs using MODEA1, MODEA2, NSGA-II and DEA-CCR models are shown in Tables 4-11.

## 3.7 Conclusions

In this chapter,, we proposed Multi Objective optimization of DEA (MODEA) solved by a meta-heuristic, viz. Differential Evolution. With this model we maximized the efficiencies of all the DMUs simultaneously in a multi objective framework in a single numerical experiment by using DE. Our model solves original DEA, which is a fractional programming problem (FPP) without approximation because we maximize the efficiencies, which are fractions, as they are. We developed two variants of the MODEA, viz., MODEA1 and MODEA2, wherein MODEA1 takes recourse to scalar optimization and MODEA2 follows Max-Min approach. Our formulation ensures that we get a single set of weights for all DMUs, thus making the comparison of the efficiencies easier and sensible. We also solved the multi-objective optimization problem using NSGA-II developed by Deb et al (2000) in order to compare the performance of the models that are developed in this thesis. We demonstrated the effectiveness of the MODEA1, MODEA2, and NSGA-II on eight datasets taken from literature and compare the results with those of the conventional DEA-CCR models. The correlation between MODEA and NSGA-II, MODEA and DEA-CCR is greater than 99%. The original proposed DEA-CCR objective is fractional and solved by converting into LPP. However, we are solving fractional objective as it is without converting into LPP in a multi-objective framework and also we are obtaining common set of weights which the decision maker is interest to rank efficient DMUs.

# 4

# Novel Soft Computing Hybrid for Data Imputation

## 4.1 K-Means Algorithm

K-means (MacQueen, 1967) is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.
Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function is

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \qquad (21)$$

Where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre $c_j$, is an indicator of the distance of the $n$ data points from their respective cluster centres.

The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

## 4.2 Multilayer Perceptron

Artificial Neural Networks (ANN) have been applied in applications involving classification, function approximation, optimization and control. In a popular form of ANN called the Multi-Layer Perceptron (MLP), all nodes and layers are arranged in a feed forward manner. The first or the lowest layer is called the input layer where external information is received. The last or the highest layer is called the output layer where the network produces the model solution. In between, there are one or more hidden layers which are critical for ANNs to identify the complex patterns in the data. Acyclic arcs from a lower layer to a higher layer connect all nodes in adjacent layers. Three-layer MLP is a commonly used ANN structure for two-group classification problems like the bankruptcy prediction, fraud detection and credit scoring etc. The parameters (arc weights) of a neural network model need to be estimated before the network can be used for prediction purposes. The process of determining these weights is called training. The training phase is a critical part in the use of neural networks. For classification problems, the network training is a supervised one in that the desired or target response of the network for each input pattern is always known a priori. During the training process, patterns or examples are presented to the input layer of a network. The activation values of the input nodes are weighted and accumulated at each node in the hidden layer. The weighted sum is transferred by an appropriate transfer function

into the node's activation value. It then becomes an input into the nodes in the output layer. Finally an output value is obtained to match the desired value. The aim of training is to minimize the differences between the ANN output values and the known target values for all training patterns. From this perspective, network training is an unconstrained nonlinear minimization problem. The most popular algorithm for training is the well known back propagation (Williams et al., 1986), which is basically a gradient steepest descent method with a constant step size. Due to problems of slow convergence and inefficiency with the steepest descent method, many variations of back propagation have been introduced for training neural networks. For a prediction problem, only one output node is needed. The output values from the neural network (the predicted outputs) are used for prediction.

## 4.3 Proposed hybrid Soft Computing Architecture

The proposed missing data imputation approach is a 2 stage approach. The block diagram (Fig 2) depicts the schema of the proposed imputation method. In this novel hybrid we used K-means clustering for stage 1. K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure for stage 1 imputation as follows:

1. Identify K cluster centers by using K-means clustering algorithm with complete records.

2. Replace the incomplete records with the corresponding features of the nearest cluster center by measuring the Euclidean distance of complete components of an incomplete record and cluster centers.

The distance is measured by using the following formula:

$$d_j = \sum_{i=1}^{m} \left| x^{(j)}{}_i - c_j \right|^2 \qquad (22)$$

Where $j$ is the number of cluster centers. $m$ is the number of complete components in each record (The value of $m$ may change from one incomplete record to other).

In the second stage, we used multilayer perceptron (MLP) for imputation. MLP is trained by using only complete cases. We have to train as a regression model by taking

one incomplete variable as target and remaining variables as inputs. So that we have to form different regression models that are equal to the number of incomplete variables in a given dataset. The steps for MLP imputation (Stage 2) scheme as follows:

1. For a given incomplete dataset $X$, separate the records that contain missing values from the set of those without missing values (or with complete values). Let us take the set of complete records as known values $X_k$ and incomplete records as unknown records $X_u$

2. For each incomplete variable, construct an MLP by considering the remaining variables in $X_k$ as inputs for training.

3. Predict the missing values in the variable, which is the target variable in MLP. While predicting we use the initial approximate which are given by K-means clustering from stage 1 as part of test set for predicting the target variable if we have more than one missing value in a record.

4. Repeat step 2 and step 3 for all incomplete variables.

## 4.4 Experimental Design

The effectiveness of the proposed method is tested on 2 classification and 2 regression datasets. Since none of these datasets has missing values, we conducted the experiments by deleting some values from the original datasets randomly. Every dataset is divided into 10 folds and 9 folds are used for training and the tenth one is left out for testing. From th test fold, every time, we deleted nearly 10% of the values (cells) randomly. We ensured that at least one cell from every record is deleted. In the stage 1 of data imputation, K-means clustering is performed by using only complete set of records (training data comprising 9 folds). The value of K in K-means is set equal to the number of classes in case of classification datasets. In the case of Wine data the number of classes is 3, so we have chosen K-value as 3. Similarly, in the case of UK banks dataset the number of clusters are chosen as 2. However, in the case of regression datasets, the number of clusters, K, is chosen by visualizing the data using principle component analysis (PCA). By visualizing the plot of PC1 vs PC2, we can set the approximate number of clusters. Thus, the number of clusters is taken as 2 for Boston

housing dataset and 3 for forest fires dataset. We can see the plots of PCA visualization for Boston housing and forest fires dataset in Figures 3 and 4 respectively.

In stage 1, the missing values of incomplete records are replaced by closest cluster center components by measuring the distance as explained in section 3. So in stage 1 missing values are replaced by local approximate values. In stage 2, we use MLP for approximating the values closest to actual values by using stage 1 values. We predict the missing values in one attribute, which is the target variable in MLP. While predicting we use the initial approximate which are given by K-means clustering from stage 1 as part of test set for predicting the target variable if we have more than one missing value in a record. The estimation is using 10 fold cross validation of all datasets.

## 4.5 Datasets Description

In this chapter we analyzed 4 datasets. Those include two regression datasets viz., Forest fires, Boston housing and two classification datasets viz., Wine and UK banks. The benchmark datasets, Wine, Boston housing, and Forest fires are taken from UCI machine learning repository. Forest fires dataset contains 11 predictor variables and 517 records, whereas Boston housing dataset contains 13 predictor variables. The description of variables used in Forest fires dataset and Boston housing dataset shown in Table 12 and Table 13. Another two datasets we used are Wine and UK bank bankruptcy datasets. Both these datasets are classification datasets. Wine dataset contains 13 predictor variables and 248 records. UK banks dataset contains 10 predictor variables and 60 records. The predictor variables of UK banks dataset are *(i)* Sales *(ii)* Profit Before Tax / Capital Employed *(%) (iii)* Funds Flow/Total Liabilities *(iv)* (Current Liabilities + Long Term Debits)/Total Assets *(v)* Current Liabilities/Total Assets, (*vi)* Current Assets/Current Liabilities *(vii)* Current Assets-Stock / Current Liabilities *(viii)* Current Assets-Current Liabilities/Total Assets *(ix)* LAG (Number of days between account year end and the date of annual report *and (x)* Age. The description of variables used in UK bankruptcy datasets shown in Table 14.

## 4.6 Results and Discussion

We measured the performance of the proposed approach by using Mean Absolute Percentage Error (MAPE) as the measure of accuracy. MAPE is defined as

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} \frac{|x_i - \hat{x}_i|}{x_i} \qquad (23)$$

Where $n$ is the number of missing values in a given dataset. $\hat{x}_i$ is the predicted value by the hybrid model for the missing value and $x_i$ is the actual value.

All the estimations of MAPE value using 10 fold cross validations on all datasets. The MAPE values of stage 1 and stage 2 for four datasets are shown in Table 15. In the case of wine dataset the MAPE value after K-means clustering i.e stage 1 is 28.84% and the MAPE value is reduced to 21.58% after performing stage 2 on the results of stage 1. Similarly in case of another classification dataset the MAPE value is reduced from 46.45% in stage 1 to 32.17% in stage 2.

In case of Boston housing dataset also the value of MAPE is reduced from 26.55% in stage 1 to 15.64% in stage 2. In forest fires dataset also the value of MAPE in stage1 is 37.58% and it is reduced to 26.61% after performing stage 2. T- test is applied between the MAPE values that are obtained after stage 1 and after stage2 for all folds of a given dataset. The purpose of T-test is to know that the reduction in MAPE from stage 1 to stage 2 is a chance happening or not. The T-test values are shown in Table 16. All the values obtained from T-test are significant at 95% level of confidence. The values in all datasets proves that the reduction is not a chance happening.

We also experimented by changing the K-value in stage 1 to investigate the impact of K value on the results. The resultant variation in MAPE values are presented in Table 17. For Wine dataset the MAPE value increasing from 21.58 to 28.32 by changing the K value from 2 to 1.

In case of UK banks dataset also MAPE value increasing from 32.17 to 40.25 with the change of K-value from 2 to 1. Thus, in case of wine and UK banks dataset, if the K-value is not equal to the number of classes, then the MAPE value after stage 2 is more compared to the case where K-value equals the number of classes. So for the classification datasets MAPE value is less if K value equals to the number of classes.

We also conducted similar experiment by changing the K-value in case of regression datasets. We have taken the K-value as 1 for Boston housing dataset and as 2 for forest fires dataset. In Boston housing dataset, MAPE is more when K=1 than when K=2. In case of forest fires dataset the MAPE value is 32.15 for K=2, and for K=3 MAPE value is 26.61. For Boston housing dataset the MAPE value is changing from 15.64 to 25.84 by changing the K value from 2 to 1.

## 4.7 Conclusions

In this chapter, we proposed a 2-stage novel soft computing hybrid based on K-means clustering and MLP f or solving the missing data imputation problem. The techniques proposed for missing data imputation in the literature used either local learning or global approximation only. In this method, we replaced the missing values by using both local learning and global approximation. The proposed hybrid is tested on four datasets in the framework of 10 fold cross validation. In all the data sets some values are randomly removed and we treated those values as missing values. In stage 1, by using K-means clustering we replaced missing values by local approximate values. In stage 2 by using the local approximate values which are resulting from stage 1 and trained MLP from complete records, we further approximate the missing value to the actual value. The missing values are replaced by using proposed novel hybrid approach, and then we compared predicted values with actual values by using MAPE. We observed that MAPE value decreased from stage 1 to stage 2. t- test is performed on four datasets, and from the values of t-test we can say that the reduction in MAPE from stage 1 to stage -2 is statistically significant. We conclude that, we can use the proposed approach as a viable alternative to the extant methods for data imputation. In particular, this method is useful for a dataset with a records having more than one missing values.

# 5

# Data Imputation by Soft Computing Hybrid: Predicting the Severity of Phishing Attacks in Financial Institutions

## 5.1 Introduction

Phishing is the new 21st century crime. It is a major security threat to the online community. Phishing scams have been escalating in number and sophistication with every month that goes by. A phishing attack today now targets audience sizes that range from mass-mailings to millions of email addresses around the world, through to highly targeted groups of customers that have been enumerated through security faults in small clicks-and-mortar retail websites. A typical phishing attack consists of four phases, namely, preparation, mass broadcast, mature, and account hijack (Bose & Leung, 2007).

Phishing attacks in the United States soared in 2007 as $3.2 billion was lost to these attacks, according to a survey by Gartner, Inc. The survey found that 3.6 million adults lost money in phishing attacks in the 12 months ending in August 2007, as compared with the 2.3 million who did so the year before. According to Anti-phishing Working Group (APWG) 2010 reports, the concentration of phishing attacks is more on payment services industry (37.9%) and financial industry (33.1%). Not only do phishing attacks cause financial loss, but they also shatter the confidence of customers in conducting e-commerce. Managers of some of the US super regional banks have indicated that the

deteriorating customer trust is a major concern with respect to phishing (Smith & Jordan, 2007).

A recent survey found that most customers of European banks only use online banking to check their account balances instead of conducting online transactions due to the fear of getting phished (Ensor et al., 2007). Another study also reported that the customer fear psychosis has resulted in a 20% decrease in the rate of opening of genuine emails (Brandt, 2005). To make customers aware of latest phishing attacks, some international organizations and government statutory bodies, such as APWG, have published phishing alerts on their websites. To assess the risk level of each phishing attack, some firms have sought help from information security experts who evaluated reported phishing incidents based on the contents of the phishing email and the phishing websites. However, as phishing incidents continue to increase at a tremendous rate, the manual risk assessment method involving experts may be too slow.

Data mining techniques can improve the assessment of phishing attacks. They can discover the knowledge embedded in the traits of prior phishing attacks and identify the inherent characteristics that contribute to the different risk levels of a phishing attack. This can help predict the associated risk level of a new phishing incident in a short period of time with a reasonable accuracy. Furthermore, the risk level, which is based on the technical sophistication of phishing attacks, may not be directly related to financial loss caused by an attack. Past research has shown that the impact of sophisticated phishing alerts on stock markets is not as significant as phishing alerts whose risk level is considered to be moderate (Leung & Bose, 2008). However, the financial loss resulting from a phishing attack is always of great concern to security administrators as well as consumers of an organization. Therefore, a warning mechanism that can identify the phishing incidents that are either very risky or likely to cause a large financial loss will be of great interest to shareholders and senior managers of the targeted companies. In this chapter we use supervised classification techniques, which is a major stream of data mining, to assess the severity of phishing attacks. We used the financial data of the targeted company to assess the severity of a phishing attack according to its risk level or financial loss generating potential. The three classifiers used for this purpose result in a classification accuracy of up to 89%. Our results also show that the key identifying variables for risk level and potential financial loss of phishing attacks are different from each other. High risk level is associated with

phishing emails that ask customers of large firms to update their accounts whereas high financial loss is characterized by phishing attacks targeted to customers of large firms that have high total liabilities.

## 5.2 About the dataset

To determine the severity of phishing attacks, we utilized financial data available from the financial statements of the firms. The phishing alerts data used in this chapter is the largest available phishing alerts data set at the time of research, and was collected from mid-2005 to mid-2008. As phishing is primarily motivated by financial gains, corporate financial data may be relevant for the assessment of severity of phishing. Relevant financial data, reported in the last month of the year prior to the release of the phishing alert, was retrieved from The Center for Research in Security Prices (CRSP). In the raw data set, there were 168 financial variables. According to Chen et al. (2010) only 25 variables are important and relevant to the problem. So we used only 25 financial variables in assessing the severity of phishing attacks. We have 885 missing values from the selected 25 attributes. From the total 1028 instances, the complete records are 251, and the instances with missing values are 777. The list of 25 financial variables is shown in Table 18.

## 5.3 Methodology

We have chosen 25 variables for the classification purpose. The total number of records that available is 1028. It is a three class classification problem. All the missing values imputed by using proposed 2 stage approach. For the stage 1 data imputation using K-means clustering is performed by using only complete records of dataset. The value of K in K-means is equal to the number of classes in the dataset i.e 3. In stage 1, the missing values of incomplete records are replaced with closest cluster centers by measuring the distance as explained in chapter 4. So in stage 1 missing values are replacing by local approximate values. Stage 2 refines the resultant approximate values using multilayer perceptron (MLP). MLP is trained on the complete data with the attribute having missing values as the target variable one at a time. So after the completion of this 2 stage imputation process, we have complete dataset with 1028 instances. We also experimented by using GRNN instead of MLP in missing data imputation procedure without changing the methodology.

The dependent variable in our method to measure the severity of phishing alerts is risk level. We first categorized phishing alerts according to the five risk levels assigned by Millersmiles. These risk levels were: Low, Low-Medium, Medium, Medium-High, and High. The majority of the alerts belonged to the category of Medium. For the sake of simplicity, we grouped risk levels Low and Low-Medium to form a new group Low, and Medium-High and High to form a new group High. NN, SVM, and DT were used in this chapter due to their history of superior performance in other applications related to information security. for classification of risk level, we oversampled the high risk and low risk instances of data but kept the medium risk instances the same so that the distribution of the three groups became 1:1:1 in the training and testing data sets. However, in the validation data set, we retained the original data. We also used 10-fold cross validation, and calculated the average accuracy of the model from the cross-validation models.

## 5.4 Results and discussion

We used DT, NN and SVM as classification models. We compared the results with Chen et.al results in case of financial data alone. Chen et.al performed the classification by imputing the missing value using mean value imputation. In this chapter, we imputed the missing value with the proposed 2 stage model. We compared the classification accuracy when we used mean value imputation method for replacing missing values presented in the dataset with the classification accuracy after replacing missing values using proposed 2-stage imputation procedure. The classification accuracy is superior if we use the proposed 2-stage methodology for missing data imputation.

The classification results in terms of accuracy by using DT, SVM, and NN are shown in Table 19. The accuracy in classification by using classifier DT is 63.88% with Chen et al. model and it is 70.87% with our experiment when we use MLP in stage 2. If we use GRNN in stage 2 the accuracy is 89.1%. By using NN classifier the classification accuracy is 54.66% with Chen et al. model. By replacing missing values with our proposed method, the accuracy is 70.67% if we use MLP in stage 2. If we use GRNN in stage 2 the accuracy using NN is 52.64%. SVM classifier is giving good accuracy 89.2% if we use our imputation procedures for replacing missing values; whereas Chen

et al. model is giving 56.21% only. So we can say that our proposed imputation procedures for replacing missing values are much superior.

## 5.5 Conclusions

In this chapter, we classified severity of phishing attacks. Phishing has become one of the biggest threats to the online community. Many researchers have explored ways to deter such crime. Information security specialists and anti-phishing organizations have set up phishing alerts databases that assess each reported phishing incident in terms of its risk level. In the view of increasing number of reported phishing incidents, we believe that such a manual assessment approach is not efficient enough to provide a timely report, and is also not complete as it ignores the possible financial impact of phishing incidents. In this method, we used financial data of organizations which are targeted by phishing attacks and classified phishing attacks in terms of risk levels (high, medium and low). We used novel soft computing hybrid for missing data imputation in the dataset. We used DT, SVM and NN for classification and classified with 89% accuracy using SVM and DT, where as the classification accuracy is only 63.88% if we use average value imputation technique for replacing missing values in dataset.

# 6

# Overall Conclusions

In the first part of study, we proposed Multi Objective optimization of DEA (MODEA) solved by a meta-heuristic, viz. Differential Evolution. With this model we maximized the efficiencies of all the DMUs simultaneously in a multi objective framework in a single numerical experiment by using DE. Our model solves original DEA, which is a fractional programming problem (FPP) without approximation because we maximize the efficiencies, which are fractions, as they are. We developed two variants of the MODEA, viz., MODEA1 and MODEA2, wherein MODEA1 takes recourse to scalar optimization and MODEA2 follows Max-Min approach. Our formulation ensures that we get a single set of weights for all DMUs, thus making the comparison of the efficiencies easier and sensible. We also solved the multi-objective optimization problem using NSGA-II developed by Deb et al (2000) in order to compare the performance of the models that are developed in this thesis. We demonstrated the effectiveness of the MODEA1, MODEA2, and NSGA-II on eight datasets taken from literature and compare the results with those of the conventional DEA-CCR models. The correlation between MODEA and NSGA-II, MODEA and DEA-CCR is greater than 99%. The original proposed DEA-CCR objective is fractional and solved by converting into LPP. However, we are solving fractional objective as it is without converting into LPP in a multi-objective framework and also we are obtaining common set of weights which the decision maker is interest to rank efficient DMUs.

In the second part of study, for solving the missing data imputation problem, we proposed a 2-stage novel soft computing hybrid based on K-means clustering and MLP.

The techniques proposed for missing data imputation in the literature used either local learning or global approximation only. In this method, we replaced the missing values by using both local learning and global approximation. The proposed hybrid is tested on four datasets in the framework of 10 fold cross validation. In all the data sets some values are randomly removed and we treated those values as missing values. In stage 1, by using K-means clustering we replaced missing values by local approximate values. In stage 2 by using the local approximate values which are resulting from stage 1 and trained MLP from complete records, we further approximate the missing value to the actual value. The missing values are replaced by using proposed novel hybrid approach, and then we compared predicted values with actual values by using MAPE. We observed that MAPE value decreased from stage 1 to stage 2. t- test is performed on four datasets, and from the values of t-test we can say that the reduction in MAPE from stage 1 to stage -2 is statistically significant. We conclude that, we can use the proposed approach as a viable alternative to the extant methods for data imputation. In particular, this method is useful for a dataset with a records having more than one missing values.

In the last part of the study, we classified severity of phishing attacks. Phishing has become one of the biggest threats to the online community. Many researchers have explored ways to deter such crime. Information security specialists and anti-phishing organizations have set up phishing alerts databases that assess each reported phishing incident in terms of its risk level. In the view of increasing number of reported phishing incidents, we believe that such a manual assessment approach is not efficient enough to provide a timely report, and is also not complete as it ignores the possible financial impact of phishing incidents. In this method, we used financial data of organizations which are targeted by phishing attacks and classified phishing attacks in terms of risk levels (high, medium and low). We used novel soft computing hybrid for missing data imputation in the dataset. We used DT, SVM and NN for classification and classified with 89% accuracy using SVM and DT, where as the classification accuracy is only 63.88% if we use average value imputation technique for replacing missing values in dataset.

# References


Abdella, M., & Marwala, T. (2005). The use of genetic algorithms and neural networks to approximate missing data in database. *Computational Cybernetics, ICCC 2005. IEEE 3rd International Conference*, 207-212.

Airoldi, E., & Malin, B. (2004). Data Mining Challenges for Electronic Safety: The Case of Fraudulent Intent Detection in E-Mails. *Proceedings of the Workshop on Privacy and Security Aspects of Data Mining 2004*, Brighton, UK, 57-66.

Ansari, A. Q., Patki, T., Patki, A. B. & Kumar, V. (2007). Integrated Fuzzy Logic and Data Mining: Impact on Cyber Security. *Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, Haikou, China.

Athanassopoulos, A. D., & Curram, S. P. (1996). A comparison of data envelopment analysis and artificial neural networks as tool for assessing the efficiency of decision making units. *Journal of the Operational Research Society,* 47(8), 1000–1016.

Banker, R. D., Morey, R. C. (1986). The use of categorical variables in Data Envelopment Analysis. *Management Science*, 32 (12), 1613-1627.

Batista, G., & Monard, M. C. (2002). A study of K-nearest neighbour as an imputation method. *Abraham A et al (eds) Hybrid Intell Syst*, Ser Front Artif Intell Appl 87, IOS Press, 251–260.

Batista, G., & Monard, M. C. (2003). Experimental comparison of K-nearest neighbour and mean or mode imputation methods with the internal strategies used by C4.5 and CN2 to treat missing data. *Tech. Rep.*, University of Sao Paulo.

Beasley, J. *E. (1990). Comparing University Departments. OMEGA International Journal of Management Science, 18 (2), 171-183.*

*Bhattacharyya, A., Lovell, C. A.* K., & Sahay, P. (1997). The impact of liberalization on the productive efficiency of Indian commercial banks. *European Journal of Operational Research,* 98, 332–345.

Brandt, A. (2005). Phishing Anxiety May Make You Miss Messages, *PC World,* 23(10), 34.

Bose, I. ., & Leung, A. C. M. (2007). Unveiling the Mask of Phishing: Threats, Preventive Measures, and Responsibilities. *Communications of the Association for Information Systems,* 19(24), 544-566.

Chandrasekaran, M., Narayanan, K., & Upadhyaya, S. (2006). Phishing E-mail Detection Based on Structural Properties. *Proceedings of NYS Cyber Security Conference, Albany*, NY, USA.

Charnes, A., Cooper, W.W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2, 429–444.

Charnes, A., Cooper, W.W., & Rhodes, E. (1981). Evaluating program and managerial efficiency: An application of data envelopment analysis to program follow through. *Management Science*, 27, 668–697.

Chen, Y., Liang, L., Yang, F., & Zhu, J. (2006). Evaluation of information technology investment: a data envelopment analysis approach. *Computers & Operations Research*, 33, 1368–1379.

Cooper, W. W., Seiford, L.M., & Tone, K. (2007). Data envelopment analysis: a comprehensive text with models, applications, References and DEA-Solver Software, (2nd Edition), Springer publications.

Costa, A., & Markellos, R. N. (1997). Evaluating public transport efficiency with neural network models. *Transpn Res.*, 5(5), 301-312.

Deb, K. (2000). An efficient constraint handling method for Genetic Algorithms. *Computer Methods in Applied Mechanics and Engineering*, 186, 311-338.

Deb, K., Agrawal, S., Pratap. A., & Meyarivan, T. (2002). A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6 (2), 182 – 197.

Dhamija, R., & Tygar, J. D. (2005). Phish, and HIPs: Human Interactive Proofs to Detect Phishing Attacks, In: H.S. Baird and D.P. Lopresti (eds.). *Proceedings of the Second International Workshop on Human Interactive Proofs*, Bethlehem, PA, USA, 127-141.

Dula, J. H. (2008). A computational study of DEA with massive data sets. *Computers and Operations Research*, 35, 1191–1203.

Emrouznejad, A., & Shale, E. (2009). A combined neural network and DEA for measuring efficiency of large scale datasets. *Computers & Industrial Engineering*, 56, 249-254.

Ensor, B., Giordanelli, A., Lussanet, M.D., & Tongeren, T.V. (2007). Many Online Banking Users Use Few Features, Forrester Research, 1-6.

Fette, I., Sadeh, N., & Tomasic, A. (2007). Learning to Detect Phishing Emails. *Proceedings of the Sixteenth International Conference on World Wide Web*, Banff, Alberta, Canada, 649-656.

http://geographyfieldwork.com/SpearmansRank.htm retrieved on 20 October 2010.

Garcı´a-Laencina, P. J., Sancho-Go´mez, J. L., & Figueiras-Vidal, A. R. (2010). Pattern classification with missing data: a review. *Neural Comput & Applic*, 19, 263–282.

Garera, S., Provos, N., Chew, M., & Rubin, A. D. (2007). A Framework for Detection and Measurement of Phishing Attacks. *Proceedings of the 2007 ACM Workshop on Recurring Malcode, Alexandria,* VA, USA, 1-8.

Gheyas, I. A., Smith, L. S. (2010). A neural network-based framework for the reconstruction of incomplete data sets. *Neurocomputing*, 73(16), 3039-3065.

Gupta, A., & Lam, M. S. (1996). Estimating missing values using neural networks. *Journal of Operational Research Society*, 47(2), 229–238.

Hai, W., & Shouhong, W. (2010). The Use of Ontology for Data Mining with Incomplete Data. *Principle Advancements in Database Management Technologies*, 375-388, 2010.

Hosseinzadesh, L. F., Jahanshahloo, G. R., Soltanifar, M., Ebrahimnejad, A., & Mansourzadeh, S. M. (2010). Relationship between MOLP and DEA based on output-orientated CCR dual model. *Expert Systems with Applications*, 37, 4331-4336.

Hu, S. C., Chung, Y. K., & Chen, Y. S. (2008). Using Hopfield neural networks to solve DEA problems. *Cybernetics and Intelligent Systems, IEEE conference*, 606 – 611.

Jagatic, T., Johnson, N., Jakobsson, M., & Menczer, F. (2006). Social Phishing. *Communications of the ACM*, 50(10), 1-10.

Jahanshahloo, G. R., Hosseinzadeh, L. F., Khanmohammadi, M., Kazemimanesh, M., & Razaie, V. (2010). Ranking of units by positive ideal DMU woth common weights. *Expert System with Applications,* 37, 331–337.

Jakobsson, M., & Myers, S. (2007). Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft. *Wiley-Interscience*, Hoboken, NJ, USA.

Jerez, J., Molina, I., Subirates, J., & Franco, L. (2006). Missing data imputation in breast cancer prognosis. *BioMed'06 Proceedings of the 24th IASTED international conference on Biomedical engineering*.

Johnson, A. L., & McGinnis, L. F. (2008). Outlier detection in two-stage semi parametric DEA models. *European Journal of Operational Research*, 187, 629–635.

Kallin, L. (2002). Missing data and the preprocessing perceptron", *Tech. Rep.*, Umeaa University.

Kao, C., & Hung, H. T. (2005). Data envelopment analysis with common weights: the compromise solution approach. *Journal of the Operational Research Society,* 56, 1196-1203.

Leung, A. C. M., & Bose, I. (2008). Indirect Financial Loss of Phishing to Global Market. *Proceedings of the Twenty-Ninth International Conference on Information Systems*, Paris, France, 1-15.

Liao, H., Wang, B., Weyman-Jones, T. (2007). Neural network based models for efficiency frontier analysis: an application to East Asian economies Growth Decomposition. *Global Economic Review,* 36 (4), 361-384.

Little, R. J. A., & Rubin, D. B. (2002). Statistical analysis with missing data. *Wiley* (2nd ed.), New Jersey.

Liu, F. F., & Peng, H. H. (2008). Ranking of units on the DEA frontier with common weights. *Computers and Operations Research*, 35, 1624-1637.

Liu, F. F., Peng, H., & Chang, H. (2006). Ranking DEA efficient units with the most compromising common Weights. *The Sixth International Symposium on Operations Research and Its Applications (ISORA'06)*, Xinjiang, China, 219-234.

Ludl, C., McAllister, S., Kirda, E., & Kruegel, C. (2007). On the Effectiveness of Techniques to Detect Phishing Sites. *In: B.M. Hammerli and R. Sommer (eds.), Proceedings of the Fourth International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, Lucerne, Switzerland, 20-39.

MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability"*, Berkeley, University of California Press,1, 281-297.

Makui, A., Alinezhad, A., Mavi, R. K., & Zohrehbandian, M. (2008). A goal programming method for finding common weights in DEA with an improved discriminating power for efficiency. *Journal of Industrial and Systems Engineering*, 1(4), 293-303.

Marseguerra, M., & Zoia, A. (2002). The autoassociative neural network in signal analysis. II. Application to on-line monitoring of a simulated BWR component. *Ann Nuclear Energy*, 32(11), 1207– 1223.

Marwala, T., & Chakraverty, S. (2006). Fault classification in structures with incomplete measured data using autoassociative neural networks and genetic algorithm. *Current Science, India*, 90(4), 542–548.

Nordbotten, S. (1996). Neural network imputation applied to the Norwegian 1990 population census data. *Journal of statistics*, 12, 385–401.

Oral, M., & Yolalan, R. (1990). An empirical study on measuring operating efficiency and profitability of bank branches. *European Journal of Operational Research*, 46, 282–294.

Pendharkar, P. C., & Rodger, J. A. (2006). Technical efficiency-based selection of learning cases to improve forecasting accuracy of neural networks under monotonicity assumption. *Decision Support Systems*, 36, 117-136.

Punnee, S. (2002). Estimating missing data of wind speeds using neural networks", *IEEE proceedings SoutheastCon*.

Ramanathan, R. (2003). An Introduction to Data envelopment Analysis. Saga Publication Pvt. Ltd, New Delhi.

Saberi, A., Vahidi, M., & Bidgoli, B. M. (2007). Learn to Detect Phishing Scams Using Learning and Ensemble Methods. *Proceedings of the International Conferences on Web Intelligence and Intelligent Agent Technology Workshop*, Silicon Valley, CA, USA, 311-314.

Saha, A., & Ravi Sankar, T. S. (2000). Rating of Indian commercial banks: a DEA approach. *European Journal of Operational Research*, 124, 187–203.

Santin, D., Delgado, F. J., & Valino, A. (2004). The measurement of technical efficiency: a neural network approach. *Applied Economics*, 36, 627-635.

Schafer, J. L. (1997). Analysis of incomplete multivariate data. *Chapman & Hall*, Florida.

Shanmugam, K. R., & Das, A. (2004). Efficiency of Indian commercial banks during the reform period. *Applied Financial Economics*, 14, 681–686.

Sharpe, P. K., & Solly, R. J. (1995). Dealing with missing values in neural network-based diagnostic systems. *Neural Comput Appl*, 3(2), 73–77.

Sherman, H. D., & Gold, F. (1985). Bank branch operating efficiency: evaluation with data envelopment analysis. *Journal of Banking and Finance*, 9, 297–315.

Singh, N. P. (2007). Online Frauds in Banks with Phishing, *Journal of Internet Banking and Commerce*, 12 (2), 1-27.

Smith, T., & Jordan, J. (2007). Banks' Top Phishing Fears – Financial Loss and Lost Customer Trust, *MarkMonitor*, San Francisco, CA, USA.

Song, Q., & Shepperd, M. (2007). A new imputation method for small software project datasets. *Journal of Systems and Software*, 80, 51-62.

Srinivas, N., & Deb, K. (1995). Multiobjective function optimization using nondominated sorting genetic algorithms. *Evolutionary Computation*, 2(3), 221–248.

Storn, R., & Price, K. (1997). Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11, 341–359.

Tsai, C.-F., Hsu, Y.-F., Lin, C.-Y., & Lin, W.-Y. (2009). Intrusion Detection by Machine Learning: A Review. *Expert Systems with Applications*, 36(10), 1194-1200.

Wang, S. (2003). Adaptive non-parametric efficiency frontier analysis: a neural-network-based model. *Computers and Operations Research*, 30, 279-295.

Workman, M., & Wisecrackers. (2008). A Theory-grounded Investigation of Phishing and Pretext Social Engineering Threats to Information Security. *Journal of the American Society for Information Science and Technology*, 59(4), 662-674.

Wu, C., Chen, X., & Yang, Y. (2004). Decision-making modeling method based on artificial neural network and data envelopment analysis. *International Geoscience and Remote Sensing Symposium Proceedings, Science for Society: Exploring and Managing a Changing Planet, IGARS*.

Wu, D. Yang, Z., & Liang, L. (2006). Using DEA-neural network approach to evaluate branch efficiency of a large Canadian bank. *Expert System with Applications*, 31, 108-115.

Yoon, S. Y., & Lee, S. Y. (1999). Training algorithm with incomplete data for feed-forward neural network. *Neural Process Lett*, 10, 171–179.

Zadeh, L.A. (1994). Soft computing and fuzzy logic. *IEEE Software*, 11 (6), 48–56.

Zhang, J., Du, Z-H., & Liu, W. (2007). A Behavior-Based Detection Approach to Mass-Mailing Host. *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, Hong Kong, China, 2140-2144.

Zhao, J–Z., & Huang, H–K. (2002). An Intrusion Detection System Based on Data Mining and Immune Principles. *Proceedings of the First International Conference on Machine Learning and Cybernetics*, Beijing, China.

Zhao, J., Zulkernine, M., & Haque, A. (2008). Random-forests-based Network Intrusion Detection Systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, 38(5), 649-659.

Zhou, P., Ang, B. W., & Poh, K. L. (2008). A survey of data envelopment analysis in energy and environmental studies. *European Journal of Operational Research*, 189, 1–18.

Zhu, D., Premkumar, G., Zhang, X., & Chu, C–H. (2001). Data Mining for Network Intrusion Detection: A Comparison of Alternative Methods. *Decision Sciences*, 32(4), 635-660.

# Appendix A: Figures

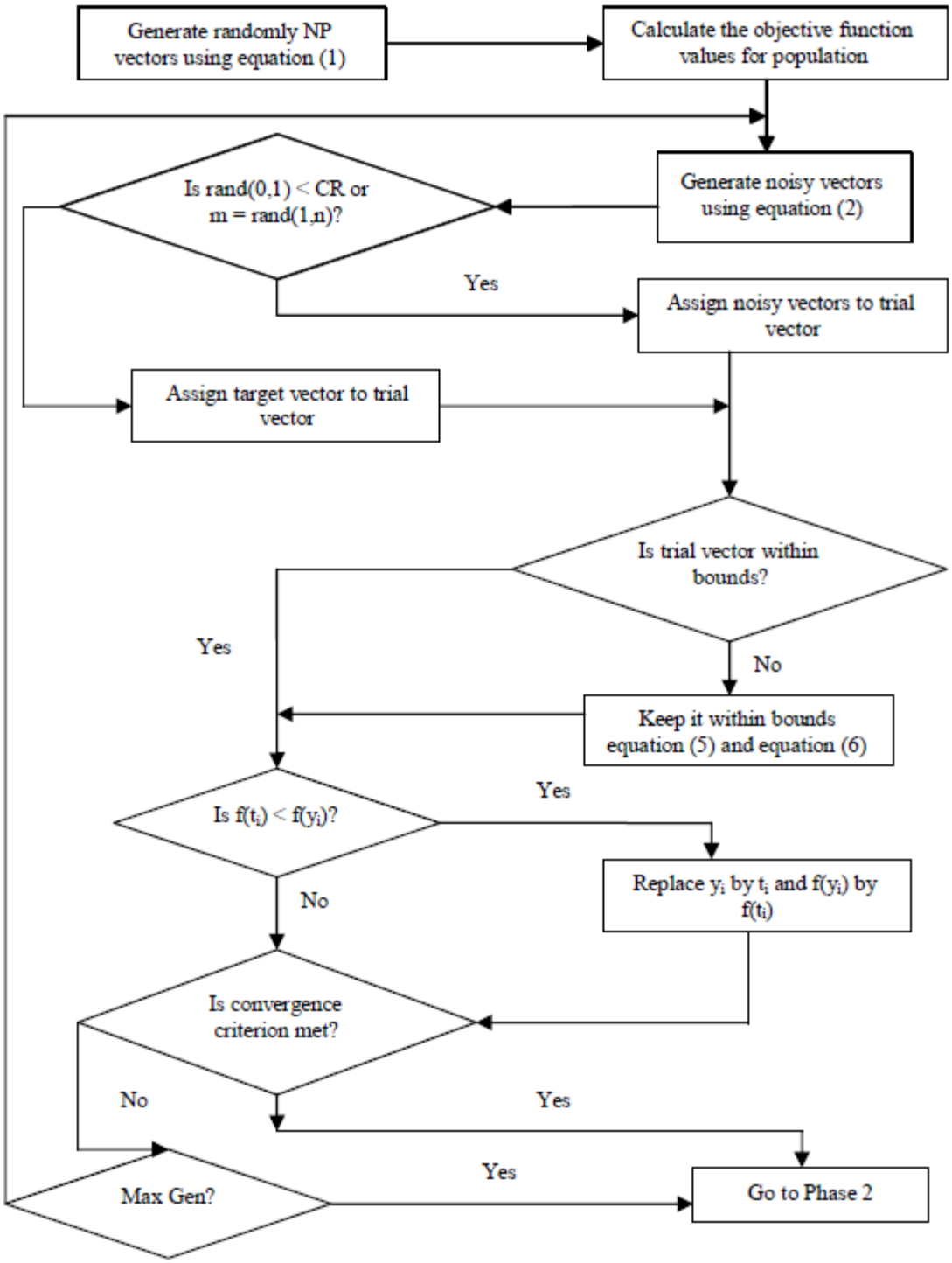


**Fig 1: Flowchart of DE algorithm**

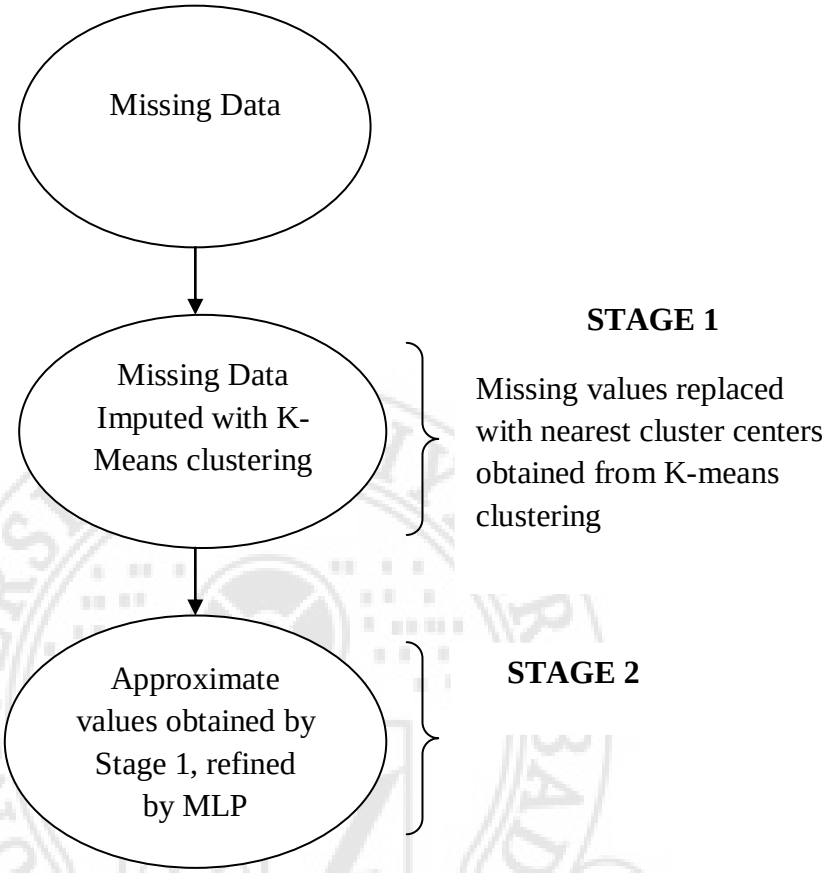


**Fig. 2 Data Flow diagram for proposed 2-stage imputation**

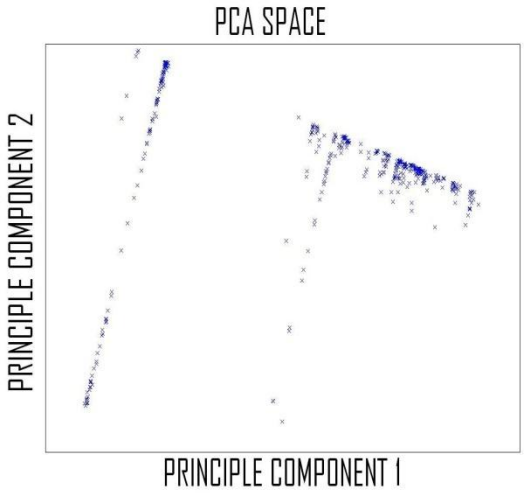


**Fig 3: Data visualization by using PCA for Boston housing dataset**

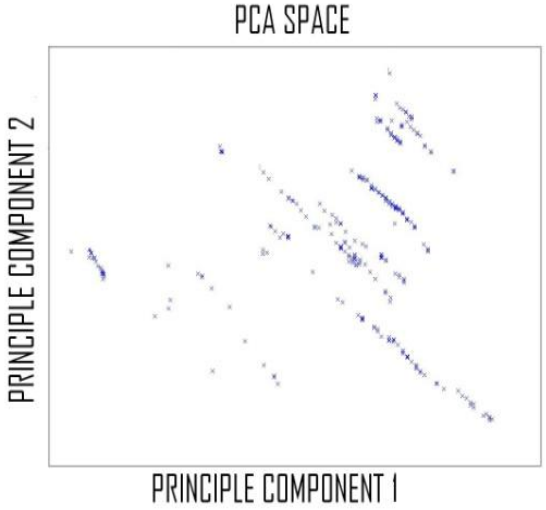


**Fig 4: Data visualization by using PCA for Forest fires dataset**

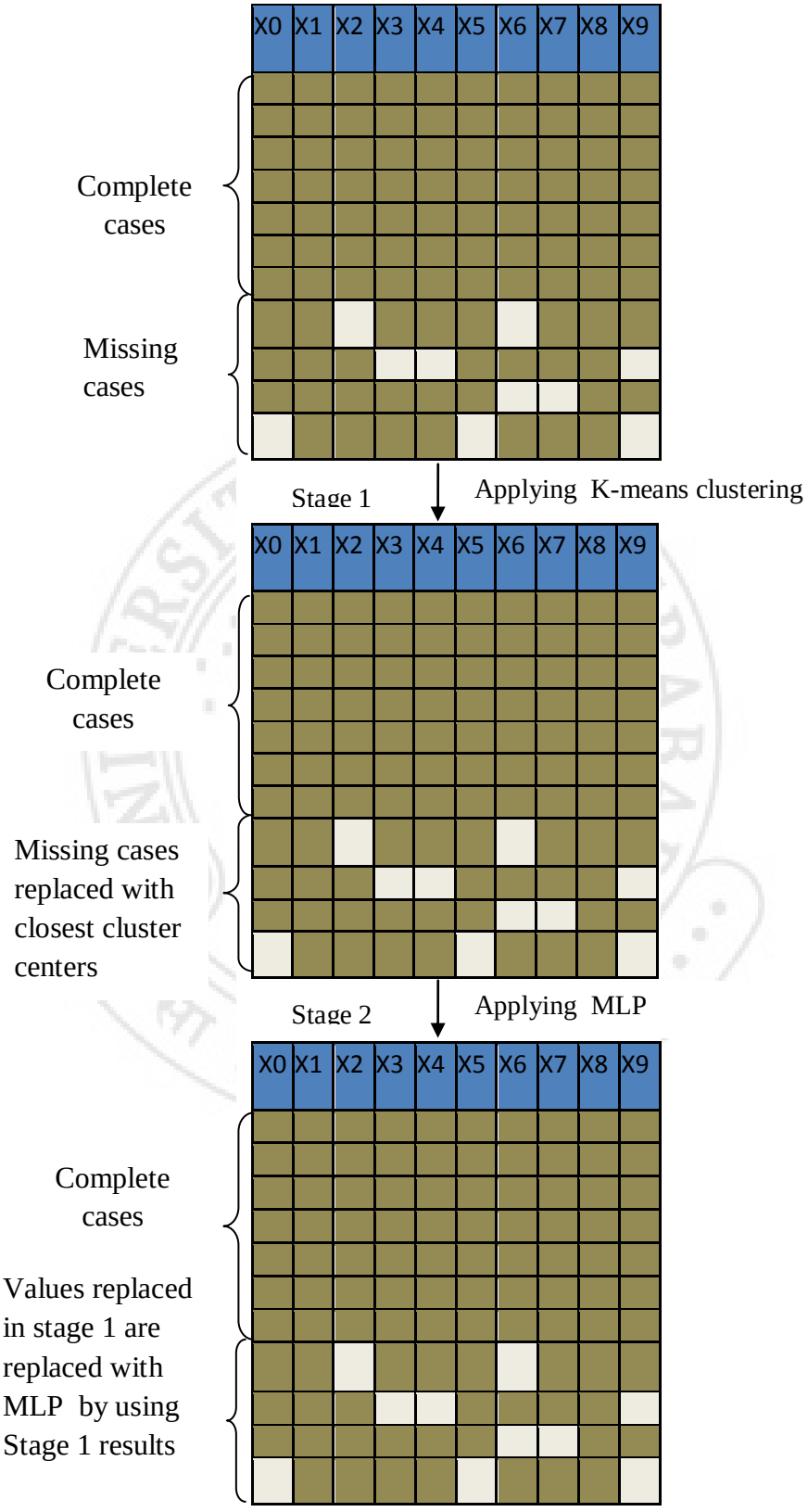


**Fig 5: Block diagram for proposed 2 stage data imputation technique (Stage 1 uses K-means clustering and Stage 2 uses Multi linear perceptron)**

# Appendix B: Tables

**Table 1: Inputs and Outputs for various datasets used for MODEA model**

| Inputs | Outputs |
|---|---|
| ***Physics Departments*** | |
| General Expenditure (like Salary etc) | Number of UG Students |
| Equipment Expenditure | Number of PG-T Students (Teaching) |
| Research Income | Number of PG-R Students (Research) |
| | |
| ***State Transport Undertakings across all the states in India*** | |
| Fleet Size (Capital..) | Passenger Kilometers |
| Total Staff | |
| Diesel Consumption | |
| | |
| ***Public libraries in Tokyo*** | |
| Area | Registered users |
| Books | Borrows |
| Staff | |
| Population | |
| | |
| ***Bank Branches ( 20 Branches in Iran)*** | |
| Staff | Deposits |
| Computers | Loans |
| Space | Income |
| | |
| ***Electric power generation companies in Japan*** | |
| Generation Capacity | Total Generation |
| Fuel Consumption | |
| Number of Employees | |
| | |
| ***Information Technology firms*** | |
| Budget | Annual Revenue Growth |
| Processer Value | Annual Income Growth |
| Training Budget | |
| | |
| ***Insurance companies (Iran)*** | |
| The number of persons | The total number of insured persons |
| The total number of computers | The number of insured persons Agreements |
| The area of the branch | The total number of life-pension receivers |
| Administrative expenses | The receipt total sum (Income) |

**Table 2: Comparison of Efficiencies from MODEA and DEA-CCR**

| ***Dataset*** | ***Model*** | ***(0.98,1]*** | ***(0.8,0.98*** | ***(0.5,0.8]*** | ***(0,0.5]*** |
|---|---|---|---|---|---|
| Physics Departments | DEA-CCR | 1 | 12 | 33 | 4 |
| | MODEA1 | 3 | 6 | 29 | 12 |
| | MODEA2 | 1 | 6 | 35 | 8 |
| | NSGA-II | 1 | 5 | 32 | 12 |
| | | | | | |
| Chemistry Departments in UK | DEA-CCR | 3 | 10 | 35 | 4 |
| | MODEA1 | 3 | 16 | 33 | 0 |
| | MODEA2 | 5 | 14 | 33 | 0 |
| | NSGA-II | 2 | 06 | 38 | 6 |
| | | | | | |
| RTC Departments in India | DEA-CCR | 2 | 11 | 12 | 4 |
| | MODEA1 | 2 | 8 | 10 | 9 |
| | MODEA2 | 2 | 9 | 11 | 7 |
| | NSGA-II | 1 | 9 | 9 | 10 |
| | | | | | |
| IT Investigation | DEA-CCR | 11 | 5 | 10 | 1 |
| | MODEA1 | 1 | 2 | 19 | 5 |
| | MODEA2 | 2 | 1 | 23 | 1 |
| | NSGA-II | 1 | 1 | 7 | 18 |
| | | | | | |
| Insurance companies | DEA-CCR | 3 | 5 | 6 | 7 |
| | MODEA1 | 2 | 4 | 7 | 8 |
| | MODEA2 | 3 | 7 | 8 | 3 |
| | NSGA-II | 1 | 4 | 5 | 11 |
| | | | | | |
| Public Libraries in Tokyo | DEA-CCR | 6 | 2 | 14 | 1 |
| | MODEA1 | 3 | 2 | 13 | 5 |
| | MODEA2 | 3 | 2 | 13 | 5 |
| | NSGA-II | 1 | 3 | 16 | 3 |
| | | | | | |
| Japanese Electric Power | DEA-CCR | 6 | 19 | 0 | 0 |
| | MODEA1 | 4 | 21 | 0 | 0 |
| | MODEA2 | 4 | 21 | 0 | 0 |
| | NSGA-II | 2 | 21 | 2 | 0 |
| | | | | | |
| Bank Branches | DEA-CCR | 5 | 4 | 5 | 6 |
| | MODEA1 | 1 | 6 | 4 | 9 |
| | MODEA2 | 1 | 4 | 10 | 5 |
| | NSGA-II | 1 | 3 | 8 | 8 |

**Table 3: Spearman's Rank correlation coefficient values**

| | ***Spearman's Rho*** | | | | |
|---|---|---|---|---|---|
| ***Dataset*** | ***MODEA1 Vs conventional DEA CCR*** | ***MODEA2 Vs conventional DEA CCR*** | ***MODEA1 Vs NSGA2*** | ***MODEA2 Vs NSGA2*** | ***MODEA1 Vs MODEA2*** |
| Physics Dept | 0.922737 | 0.914014 | 0.800624 | 0.818183 | 0.980584 |
| Chemistry Dept | 0.909588 | 0.914895 | 0.928641 | 0.959233 | 0.959546 |
| RTC | 0.97461 | 0.976252 | 0.997434 | 0.979536 | 0.998285 |
| Insurance companies | 0.873881 | 0.881105 | 0.877798 | 0.882657 | 0.995625 |
| Bank Branches | 0.968626 | 0.96623 | 0.999704 | 0.994589 | 0.999147 |
| Public Libraries in Tokyo | 0.963562 | 0.960927 | 0.961174 | 0.940382 | 0.985548 |
| Japanese Power Generation | 0.965353 | 0.963664 | 0.992981 | 0.94798 | 0.994936 |
| IT Firms | 0.931042 | 0.976122 | 0.960575 | 0.94749 | 0.960475 |

**Table 4: Comparison of ranks of Physics Departments with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 4 | 1 | 1 | 8 |
| 2 | 40 | 34 | 41 | 5 |
| 3 | 13 | 28 | 42 | 4 |
| 4 | 30 | 36 | 35 | 30 |
| 5 | 11 | 5 | 4 | 40 |
| 6 | 6 | 44 | 27 | 10 |
| 7 | 7 | 2 | 3 | 44 |
| 8 | 20 | 25 | 21 | 45 |
| 9 | 29 | 39 | 39 | 34 |
| 10 | 1 | 7 | 6 | 35 |
| 11 | 41 | 20 | 30 | 48 |
| 12 | 48 | 43 | 38 | 15 |
| 13 | 17 | 17 | 14 | 41 |
| 14 | 32 | 26 | 20 | 43 |
| 15 | 23 | 27 | 26 | 46 |
| 16 | 39 | 40 | 43 | 24 |
| 17 | 28 | 45 | 45 | 32 |
| 18 | 44 | 41 | 46 | 7 |
| 19 | 47 | 31 | 12 | 12 |
| 20 | 8 | 33 | 24 | 26 |
| 21 | 43 | 38 | 37 | 50 |
| 22 | 16 | 32 | 32 | 47 |
| 23 | 49 | 42 | 44 | 39 |
| 24 | 50 | 49 | 49 | 36 |
| 25 | 3 | 11 | 19 | 23 |
| 26 | 42 | 47 | 50 | 20 |
| 27 | 45 | 37 | 28 | 22 |
| 28 | 12 | 12 | 25 | 42 |
| 29 | 31 | 21 | 18 | 33 |
| 30 | 25 | 46 | 36 | 6 |
| 31 | 38 | 23 | 31 | 21 |
| 32 | 37 | 6 | 10 | 17 |
| 33 | 35 | 10 | 23 | 38 |
| 34 | 5 | 14 | 7 | 2 |
| 35 | 2 | 3 | 2 | 49 |
| 36 | 26 | 19 | 16 | 27 |
| 37 | 10 | 22 | 17 | 16 |
| 38 | 9 | 8 | 11 | 37 |
| 39 | 18 | 50 | 48 | 14 |
| 40 | 14 | 29 | 33 | 13 |
| 41 | 15 | 9 | 5 | 11 |
| 42 | 24 | 16 | 9 | 1 |
| 43 | 36 | 30 | 40 | 19 |
| 44 | 27 | 4 | 13 | 28 |
| 45 | 33 | 13 | 15 | 29 |
| 46 | 21 | 24 | 29 | 18 |
| 47 | 34 | 35 | 34 | 25 |
| 48 | 19 | 18 | 8 | 3 |
| 49 | 22 | 48 | 47 | 9 |
| 50 | 46 | 15 | 22 | 31 |

**Table 5: Comparison of ranks of Chemistry Departments with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA-II |
|---|---|---|---|---|
| 1 | 11 | 10 | 12 | 16 |
| 2 | 6 | 6 | 3 | 6 |
| 3 | 30 | 19 | 15 | 4 |
| 4 | 42 | 45 | 30 | 10 |
| 5 | 1 | 22 | 40 | 44 |
| 6 | 22 | 3 | 18 | 15 |
| 7 | 5 | 23 | 22 | 45 |
| 8 | 46 | 51 | 50 | 49 |
| 9 | 25 | 29 | 23 | 37 |
| 10 | 9 | 4 | 29 | 38 |
| 11 | 36 | 34 | 47 | 51 |
| 12 | 26 | 25 | 17 | 8 |
| 13 | 32 | 44 | 48 | 46 |
| 14 | 37 | 50 | 52 | 48 |
| 15 | 40 | 18 | 8 | 30 |
| 16 | 41 | 31 | 42 | 22 |
| 17 | 52 | 48 | 45 | 34 |
| 18 | 16 | 13 | 10 | 9 |
| 19 | 17 | 32 | 24 | 23 |
| 20 | 31 | 5 | 46 | 27 |
| 21 | 7 | 47 | 49 | 52 |
| 22 | 47 | 43 | 33 | 43 |
| 23 | 20 | 41 | 37 | 40 |
| 24 | 50 | 46 | 36 | 26 |
| 25 | 19 | 39 | 38 | 36 |
| 26 | 49 | 12 | 34 | 20 |
| 27 | 28 | 38 | 31 | 25 |
| 28 | 44 | 35 | 41 | 47 |
| 29 | 38 | 49 | 44 | 42 |
| 30 | 8 | 7 | 4 | 7 |
| 31 | 4 | 9 | 6 | 17 |
| 32 | 39 | 17 | 16 | 12 |
| 33 | 51 | 20 | 39 | 41 |
| 34 | 12 | 14 | 7 | 2 |
| 35 | 2 | 40 | 43 | 50 |
| 36 | 34 | 30 | 26 | 35 |
| 37 | 10 | 8 | 13 | 21 |
| 38 | 24 | 28 | 21 | 32 |
| 39 | 14 | 2 | 5 | 18 |
| 40 | 23 | 37 | 27 | 13 |
| 41 | 15 | 11 | 2 | 5 |
| 42 | 3 | 1 | 1 | 1 |
| 43 | 18 | 27 | 20 | 24 |
| 44 | 27 | 21 | 14 | 19 |
| 45 | 48 | 33 | 25 | 29 |
| 46 | 45 | 52 | 51 | 31 |
| 47 | 13 | 15 | 9 | 28 |
| 48 | 43 | 16 | 11 | 3 |
| 49 | 33 | 26 | 19 | 11 |
| 50 | 35 | 36 | 28 | 39 |
| 51 | 29 | 42 | 35 | 14 |
| 52 | 21 | 24 | 32 | 33 |

**Table 6: Comparison of ranks of STUs with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 17 | 17 | 17 | 18 |
| 2 | 11 | 14 | 14 | 15 |
| 3 | 15 | 16 | 16 | 16 |
| 4 | 19 | 18 | 19 | 19 |
| 5 | 18 | 1 | 1 | 3 |
| 6 | 14 | 15 | 15 | 14 |
| 7 | 22 | 21 | 18 | 17 |
| 8 | 7 | 8 | 8 | 8 |
| 9 | 10 | 10 | 10 | 10 |
| 10 | 1 | 13 | 13 | 12 |
| 11 | 25 | 25 | 22 | 22 |
| 12 | 24 | 24 | 23 | 23 |
| 13 | 6 | 5 | 5 | 6 |
| 14 | 13 | 11 | 11 | 11 |
| 15 | 4 | 3 | 3 | 2 |
| 16 | 8 | 7 | 7 | 7 |
| 17 | 5 | 4 | 4 | 4 |
| 18 | 9 | 6 | 6 | 5 |
| 19 | 11 | 9 | 9 | 9 |
| 20 | 1 | 2 | 2 | 1 |
| 21 | 27 | 26 | 25 | 25 |
| 22 | 28 | 27 | 27 | 27 |
| 23 | 29 | 29 | 29 | 29 |
| 24 | 26 | 28 | 28 | 28 |
| 25 | 16 | 20 | 20 | 20 |
| 26 | 23 | 22 | 26 | 26 |
| 27 | 3 | 12 | 12 | 13 |
| 28 | 21 | 23 | 24 | 24 |
| 29 | 20 | 19 | 21 | 21 |

**Table 7: Comparison of ranks of Electric generation companies with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 12 | 11 | 11 | 8 |
| 2 | 10 | 9 | 6 | 7 |
| 3 | 1 | 2 | 4 | 3 |
| 4 | 23 | 25 | 24 | 24 |
| 5 | 7 | 7 | 10 | 11 |
| 6 | 11 | 10 | 8 | 10 |
| 7 | 21 | 20 | 19 | 20 |
| 8 | 17 | 14 | 12 | 12 |
| 9 | 20 | 21 | 20 | 19 |
| 10 | 19 | 18 | 16 | 15 |
| 11 | 16 | 13 | 15 | 16 |
| 12 | 1 | 4 | 3 | 2 |
| 13 | 9 | 8 | 9 | 9 |
| 14 | 1 | 3 | 1 | 4 |
| 15 | 1 | 16 | 18 | 18 |
| 16 | 25 | 23 | 25 | 25 |
| 17 | 1 | 5 | 7 | 6 |
| 18 | 18 | 15 | 13 | 13 |
| 19 | 24 | 24 | 23 | 23 |
| 20 | 14 | 17 | 17 | 17 |
| 21 | 1 | 1 | 2 | 1 |
| 22 | 8 | 6 | 5 | 5 |
| 23 | 22 | 22 | 22 | 21 |
| 24 | 15 | 12 | 14 | 14 |
| 25 | 13 | 19 | 21 | 22 |

**Table 8: Comparison of ranks of public libraries in Tokyo with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 23 | 23 | 23 | 23 |
| 2 | 9 | 9 | 7 | 22 |
| 3 | 21 | 21 | 22 | 18 |
| 4 | 16 | 11 | 14 | 13 |
| 5 | 1 | 4 | 2 | 12 |
| 6 | 1 | 1 | 1 | 3 |
| 7 | 18 | 12 | 13 | 17 |
| 8 | 20 | 22 | 18 | 21 |
| 9 | 1 | 2 | 4 | 2 |
| 10 | 17 | 10 | 12 | 10 |
| 11 | 22 | 18 | 17 | 20 |
| 12 | 13 | 7 | 8 | 6 |
| 13 | 14 | 14 | 9 | 16 |
| 14 | 15 | 17 | 19 | 9 |
| 15 | 8 | 15 | 10 | 11 |
| 16 | 19 | 20 | 20 | 19 |
| 17 | 1 | 6 | 5 | 5 |
| 18 | 10 | 19 | 21 | 15 |
| 19 | 1 | 5 | 6 | 4 |
| 20 | 7 | 16 | 16 | 8 |
| 21 | 11 | 13 | 11 | 14 |
| 22 | 12 | 8 | 15 | 7 |
| 23 | 1 | 3 | 3 | 1 |

**Table 9: Comparison of ranks of IT firms with various models**

| DMU No / Name | Rank with DEA - CCR | Rank with MODEA1 | Rank with MODEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 8 | 11 | 5 | 12 |
| 2 | 1 | 24 | 2 | 19 |
| 3 | 24 | 19 | 30 | 24 |
| 4 | 12 | 6 | 10 | 5 |
| 5 | 22 | 26 | 8 | 18 |
| 6 | 30 | 23 | 25 | 23 |
| 7 | 21 | 20 | 15 | 17 |
| 8 | 1 | 3 | 34 | 22 |
| 9 | 23 | 21 | 16 | 9 |
| 10 | 29 | 31 | 23 | 32 |
| 11 | 16 | 25 | 6 | 10 |
| 12 | 18 | 13 | 32 | 28 |
| 13 | 25 | 18 | 18 | 16 |
| 14 | 1 | 17 | 26 | 14 |
| 15 | 20 | 16 | 24 | 21 |
| 16 | 19 | 14 | 20 | 15 |
| 17 | 1 | 2 | 3 | 2 |
| 18 | 14 | 10 | 33 | 25 |
| 19 | 34 | 30 | 36 | 35 |
| 20 | 35 | 36 | 35 | 36 |
| 21 | 28 | 22 | 31 | 29 |
| 22 | 7 | 9 | 4 | 3 |
| 23 | 27 | 33 | 19 | 30 |
| 24 | 15 | 29 | 7 | 13 |
| 25 | 36 | 32 | 27 | 34 |
| 26 | 26 | 34 | 11 | 33 |
| 27 | 13 | 7 | 12 | 6 |
| 28 | 32 | 28 | 29 | 27 |
| 29 | 9 | 5 | 17 | 7 |
| 30 | 1 | 1 | 1 | 1 |
| 31 | 11 | 15 | 21 | 11 |
| 32 | 6 | 4 | 13 | 4 |
| 33 | 33 | 27 | 22 | 26 |
| 34 | 31 | 35 | 14 | 31 |
| 35 | 10 | 8 | 9 | 8 |
| 36 | 17 | 12 | 28 | 20 |

**Table 10: Comparison of ranks of Insurance companies in Iran with various models**

| DMU No | Rank with DEA - CCR | Rank with MO-DEA1 | Rank with MO-DEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 21 | 21 | 21 | 21 |
| 2 | 10 | 7 | 7 | 20 |
| 3 | 19 | 20 | 18 | 18 |
| 4 | 14 | 9 | 9 | 13 |
| 5 | 1 | 2 | 2 | 1 |
| 6 | 1 | 1 | 1 | 3 |
| 7 | 16 | 10 | 10 | 17 |
| 8 | 18 | 19 | 19 | 19 |
| 9 | 1 | 3 | 4 | 2 |
| 10 | 15 | 8 | 5 | 10 |
| 11 | 20 | 16 | 16 | 12 |
| 12 | 11 | 6 | 6 | 6 |
| 13 | 12 | 12 | 13 | 16 |
| 14 | 13 | 15 | 15 | 9 |
| 15 | 7 | 13 | 12 | 11 |
| 16 | 17 | 18 | 17 | 7 |
| 17 | 1 | 4 | 3 | 5 |
| 18 | 8 | 17 | 20 | 15 |
| 19 | 1 | 5 | 8 | 4 |
| 20 | 6 | 14 | 14 | 8 |
| 21 | 9 | 11 | 11 | 14 |

**Table 11: Comparison of ranks of Bank branches in Iran with various models**

| DMU No | Rank with DEA - CCR | Rank with MO-DEA1 | Rank with MO-DEA2 | Rank with NSGA- II |
|---|---|---|---|---|
| 1 | 6 | 5 | 5 | 4 |
| 2 | 10 | 10 | 10 | 10 |
| 3 | 8 | 8 | 8 | 8 |
| 4 | 1 | 3 | 3 | 3 |
| 5 | 9 | 9 | 9 | 11 |
| 6 | 14 | 13 | 14 | 17 |
| 7 | 1 | 1 | 1 | 1 |
| 8 | 12 | 11 | 12 | 9 |
| 9 | 13 | 15 | 13 | 14 |
| 10 | 20 | 18 | 20 | 20 |
| 11 | 16 | 16 | 16 | 18 |
| 12 | 1 | 6 | 6 | 5 |
| 13 | 11 | 12 | 11 | 12 |
| 14 | 18 | 18 | 18 | 19 |
| 15 | 7 | 2 | 2 | 7 |
| 16 | 15 | 14 | 15 | 6 |
| 17 | 1 | 7 | 7 | 15 |
| 18 | 17 | 17 | 17 | 16 |
| 19 | 19 | 20 | 19 | 13 |
| 20 | 1 | 4 | 4 | 2 |

**Table 12: Financial Ratios of UK banks dataset**

| # | ***Predictor Variable Name*** | |
|---|---|---|
| 1 | Sales | Sales |
| 2 | Profit Before Tax/Capital Employed (%) | PBT/CE |
| 3 | Funds Flow/Total Liabilities | FF/TL |
| 4 | (Current Liabilities + Long Term Debits)/Total Assets | (CL+LTD)/TA |
| 5 | Current Liabilities/Total Assets | CL/TA |
| 6 | Current Assets/Current Liabilities | CA/CL |
| 7 | Current Assets-Stock/Current Liabilities | CA-S/CL |
| 8 | Current Assets-Current Liabilities/Total Assets | CA-CL/TA |
| 9 | LAG(Number of days between account year end and the date of annual report) | LAG |
| 10 | Age | Age |

**Table 13: Attribute description of Boston Housing dataset**

| *#* | ***Attribute Name*** | ***Attribute Type*** |
|---|---|---|
| 1 | CRIM: per capita crime rate by town | Continuous |
| 2 | ZN: proportion of residential land zoned for lots over 25,000 sq.ft. | Continuous |
| 3 | INDUS: proportion of non-retail business acres per town | Continuous |
| 4 | CHAS: Charles River dummy variable | Binary |
| 5 | NOX: nitric oxides concentration (parts per 10 million) | Continuous |
| 6 | RM: average number of rooms per dwelling | Continuous |
| 7 | AGE: proportion of owner-occupied units built prior to 1940 | Continuous |
| 8 | DIS: weighted distances to five Boston employment centres | Continuous |
| 9 | RAD: index of accessibility to radial highways | Continuous |
| 10 | TAX: full-value property-tax rate per $10,000 | Continuous |
| 11 | PTRATIO: pupil-teacher ratio by town | Continuous |
| 12 | B: 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town | Continuous |
| 13 | LSTAT: % lower status of the population | Continuous |
| **14** | **MEDV: Median value of owner-occupied homes in $1000's** | **(TARGET)** |

**Table 14: Attribute description of Forest Fires dataset**

| *#* | *Attribute Name* | *Attribute Type* |
|---|---|---|
| 1 | X - x-axis spatial coordinate within the Montesinho park map: 1 to 9 | Multi Valued Discrete |
| 2 | Y - y-axis spatial coordinate within the Montesinho park map: 2 to 9 | Multi Valued Discrete |
| 3 | Month | Multi Valued Discrete |
| 4 | Day | Multi Valued Discrete |
| 5 | FFMC - Fine Fuel Moisture Code | Continuous |
| 6 | DMC - Duff Moisture Code | Continuous |
| 7 | DC - Drought Code | Continuous |
| 8 | ISI - Initial Spread Index | Continuous |
| 9 | Temperature | Continuous |
| 10 | RH - relative humidity | Continuous |
| 11 | wind - wind speed in km/h | Continuous |
| 12 | rain - outside rain in mm/m2 | Continuous |
| **13** | **area - the burned area of the forest (in ha): 0.00 to 1090.84** | **Continuous (TARGET)** |

**Table 15: MAPE values for datasets (in %)**

| *DATASET NAME* | *MAPE after Stage1* | *MAPE after stage 2* |
|---|---|---|
| Wine | 28.84 | 21.58 |
| UK Banks | 46.45 | 32.17 |
| Boston Housing | 26.55 | 15.64 |
| Forest Fires | 37.88 | 26.61 |

**Table 16: Values of t-test statistics**

| *DATASET NAME* | *T- statistic value* |
|---|---|
| Wine | 5.56 |
| UK Banks | 1.87 |
| Boston Housing | 2.54 |
| Forest Fires | 2.69 |

**Table 17: Change in value MAPE with change in K-value**

| ***Dataset Name*** | ***MAPE*** | |
|---|---|---|
| Wine | 21.58 (K=2) | 28.32 (K=1) |
| UK Banks | 32.17 (K=2) | 40.25 (K=1) |
| Boston Housing | 15.64 (K=2) | 25.84 (K=1) |
| Forest Fires | 26.61 (K=3) | 32.15 (K=2) |

**Table 18: List of Financial variables used in assessing severity of phishing attacks**

| ***S. No*** | ***Name of the Financial Variable*** |
|---|---|
| 1 | Expenditure in advertising |
| 2 | Book value per share |
| 3 | Cost of goods sold |
| 4 | Earnings before interest and taxes |
| 5 | Total assets |
| 6 | Total inventories |
| 7 | Total liabilities |
| 8 | Total Market value in fiscal year |
| 9 | Notes payable in short term borrowings |
| 10 | Value of other Intangibles |
| 11 | Annual high price in fiscal year |
| 12 | Total receivables |
| 13 | Total assets |
| 14 | Value of tangible common equity |
| 15 | Total debt in current liabilities |
| 16 | Number of employees |
| 17 | Income before extra-ordinary items |
| 18 | Total amount of invested capital |
| 19 | Total long term debt |
| 20 | Net Loss in Income |
| 21 | Total operating Expenses |
| 22 | Total preferred/preference stock (capital) |
| 23 | General sales and administrative expenses |
| 24 | Total revenue |
| 25 | Standard & Poor core earnings |

**Table 19: Classification accuracy in assessing severity of phishing attacks**

| | ***Classification Accuracy*** | | |
|---|---|---|---|
| | | ***Missing data Imputation by using proposed 2 stage method*** | |
| | ***By using Mean Substitution*** | ***K-Means + MLP*** | ***K-Means + GRNN*** |
| **DT** | 63.88 | 70.87 | 89.38 |
| **SVM** | 56.21 | 89.2 | 89.30 |
| **NN** | 54.66 | 70.67 | 56.21 |