Hybrid Intelligence Systems
For
Data Imputation

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE DEGREE OF

Master of Technology
In
Information Technology

By

Chandan Gautam

(12MCMB03)

School of Computer and Information Sciences
University of Hyderabad, Gachibowli
Hyderabad – 500046, India

June, 2014

CERTIFICATE

This is to certify that the dissertation entitled **"Hybrid Intelligence Systems For Data Imputation"** submitted by **Mr. CHANDAN GAUTAM** bearing Reg. No. **12MCMB03,** in partial fulfillment of the requirements for the award of Master of Technology in **Information Technology,** is a bona fide 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.

Dr. V. Ravi
Supervisor
Associate Professor
IDRBT

Dean,
School of Computer and Information Sciences
University of Hyderabad

DECLARATION

I **Chandan Gautam** hereby declare that this Dissertation entitled "**Hybrid Intelligence Systems For Data Imputation**", submitted by me under the guidance and supervision of **Dr. V. Ravi, Associate Professor, IDRBT**, is a bona fide 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: **Chandan Gautam**

Signature of the Student

Regd. No.**12MCMB03**

Acknowledgements

First, I thank the Almighty God for providing me an opportunity to pursue my master degree and granting me sufficient knowledge to excel in every aspect of my life.

I would like to express my sincere gratitude to **Dr. V. Ravi** who supervised the project. I am very thankful for his continuous support not only as a guide but also as a mentor helping me in exploring new research areas and guiding me towards the right path. This work would not have been possible without the broad vision and assistance of Dr. V. Ravi.

I thank **Mr. B. Sambamurthy**, Director, IDRBT, **Prof. Arun K. Pujari,** Dean, SCIS, University of Hyderabad for extending their cooperation.

I thank IDRBT for providing me with necessary infrastructure required for the project. I thank all the faculty of IDRBT and SCIS, University of Hyderabad for all the courses they taught me which helped me in completion of my project.

I want to express my deep faith and love to my parents for their everlasting support.

I want to thank all my friends and colleagues for their necessary support and encouragement throughout my M.Tech course.

With Sincere Regards,

Chandan Gautam

List of Published And Communicated Research Papers

Published Papers in Journals

1. **C. Gautam** and V. Ravi, "*Data Imputation via Evolutionary Computation, Clustering and a Neural Network*", **Neurocomputing** (ELSEVIER), Vol. 156, pp. 134‑142, 2015.
2. C. Gautam and V. Ravi, "*Counter Propagation Auto Associative Neural Network based Data Imputation*", **Information Sciences** (ELSEVIER), Vol. 325, pp. 288‑299, 2015.

Published Paper in Conference

3. **C. Gautam** and V. Ravi, "*Evolving Clustering Based Data Imputation*", 3rd **IEEE** Conference, ICCPCT, Kanyakumari, pp. 1763-1769, 2014.

Published Book Chapter

4. C. Gautam and V. Ravi, Auto Associative Extreme Learning Machine Based Hybrids for Data Imputation, Book: Handbook of Research on Intelligent Techniques and Modeling Applications in Marketing Analytics, Publisher IGI Global, pp. 75-99.

ABSTRACT

Knowledge extraction from database is always cumbersome for researchers in various disciplines due to presence of missing data. So, missing data is an inevitable problem in many disciplines. Various methods have been proposed by many researchers to resolve the missing data problem. Many data mining algorithms cannot directly operate on a dataset having missing values. In this thesis, we developed several methods for data imputation based on computational intelligence and statistical techniques. The thesis contains three modules:

I. Four data imputation architectures for numerical data based on Extreme Learning Machine (ELM), Auto associative neural networks (AANN), Evolving Clustering Method (ECM), Principal Component Analysis (PCA) and Gray System Theory (GST).
II. Two data imputation techniques for numerical data based on covariance matrix, Particle Swarm Optimization (PSO), AANN, ELM and ECM.
III. Two data imputation techniques for numerical data based on AANN, GST and Counterpropagation Neural Network (CPNN).

We first employed auto-associative ELM architecture (developed elsewhere) for imputation. Further, since ELM depends heavily on the random weights that connect the input and hidden layers, it yields different results in different runs. Sometimes, the results could fluctuate wildly. So, in order to overcome the random behavior of ELM, we proposed two new architectures in first module. First architecture is a hybrid of ECM and AAELM in tandem and second comprises PCA and AAELM in tandem. Both architectures provided deterministic flavour to ELM. We also developed an imputation technique based on just a fast clustering algorithm, ECM, and it outperformed a hybrid algorithm K-Means+MLP. Since, ECM is controlled by *Dthr* value; we needed to find an optimum *Dthr* value. Consequently, ECM-AAELM and ECM-imputation performed better than earlier methods.. After an exhaustive experiment on ECM-AAELM and ECM-Imputation, we observed that selection of *Dthr* value strongly influences the results. Further, we developed a new architecture, ECM_PSO_COV, based on PSO and Covariance structure of matrix for selection of optimal *Dthr* value. . ECM_PSO_COV outperformed earlier ECM-imputation technique due to optimal choice of *Dthr* value. Furthermore, we pass same

Dthr value to ECM-AAELM architecture and observed that it provided more accurate imputation compared to that of ECM-AAELM. Later, we employed Gray distance based imputation technique instead of Mean imputation as a preprocessing task followed by PCA-AAELM. It also improved the accuracy of PCA-AAELM by a significant amount. Morever, we proposed two new hybrids based on CPNN viz., CPAANN and Gray+CPAANN. The proposed imputation techniques are tested on 12 benchmark datasets. The results indicate that the proposed imputation techniques approximate the missing value to a closest possible value as measured by mean absolute percentage error (MAPE). Several experiments have been conducted on several regression, classification and banking datasets to assess and compare the effectiveness of the proposed imputation techniques.

The results of the proposed methods are compared with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013), PSOAANN, PSOAAWNN, RBFAANN and GRAANN (Ravi and Krishna, 2014). We tested the effectiveness of all proposed models and implemented models on 4 benchmark classification and 4 benchmark regression datasets; 3 bankruptcy prediction datasets and one credit scoring datasets under 10-fold cross validation testing. From the experiments, we observed that the ECM_PSO_COV+ECM_AAELM and Gray+CPAANN provided better predictions for the missing values than the other models. We also performed the Wilcoxon signed rank test on our propose models with existing models to check whether our obtained results are statistically significantly different or not. It turned out that the obtained results by our proposed methods are statistically significant at 1% level of significance.

Table of Contents

CHAPTER 1
INTRODUCTION

1.1 Data Imputation

The treatment of missing data or incomplete data is an important step in the pre-processing of data. Missing data in real life data sets is an unavoidable problem in many disciplines .In statistics data imputation is defined as the substitution of some value for a missing data point or a missing component of a data point. 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). Once all missing values have been imputed then the dataset can be analyzed using standard techniques for complete data. Many data mining algorithms cannot directly operate on a dataset having incomplete data. The respondents may not give complete information because of negligence, privacy reasons or ambiguity of the survey questions. For example researchers rarely find the survey data set with complete entries (Hai and Shouhong, 2010). The missing components 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.

1.2 Reasons for missing data

In almost all the areas of research, the missing data are broadly experienced. There are many reasons may lead to missing data. In surveys, data may be missing due to procedural factors such as errors in data entry, disclosure restrictions, or failure to complete the entire questionnaire. Missing data occur when the response does not apply (e.g., questions regarding the years of marriage for a respondent who has never been married). There are also missing data due to respondent refusal to answer some sensitive questions (e.g., age, income, drug use), and variables too expensive to measure (e.g., interviewer needs to travel a long distance). In control based applications such as management of telecommunications and computer networks (Ji and Elwalid, 2000) missing values appear due to failures of monitoring or data collector equipment and traffic monitoring (Nguyen and Scherer, 2003), industrial process (Lakshminarayana et al., 2004). Missing data may occur in wireless sensor networks due to reasons like power outrage at sensor nodes, random occurrences of local inferences and higher bit error rate of the transmission (Halatchev and Gruenwald, 2005; Mohammed et al., 2006).

Speech samples that are corrupted by very high levels of noise are considered as missing data in automatic speech recognition (Cooke et.al., 1994).Incomplete data may also appear in business and financial applications. In biological research with DNA microarrays, gene data may be missing due to the reasons such as scratch on the slide that contains the gene sample and contaminated samples (Troyanskaya et al., 2001). Missing data can also occur as a result of drop outs, for example, when an experiment is run on a group of individuals over a period of time as in clinical studies.

1.3 Impact of missing data

Missing data may result in biased estimates in several ways (Roth et.al, 1999). First, the measures of central tendency may be biased upward or downward depending upon where in the distributions missing values occur. Second, measures of dispersion may be affected depending

upon which part of distribution has missing data. Third, missing data may bias correlation coefficients.

1.4Missing data Mechanisms

Missing data is categorized into 3 categories they are (i) Missing completely at random (MCAR), (ii) Missing at random (MAR) and (iii) Not missing at random (NMAR) (Little and Rubin, 2002).

i) Missing completely at random (MCAR) occurs if the probability of missing value on some variable X is independent of the variable itself and also other variables present in the dataset.For example, in a dataset that includes student marks, a student's final grade is missing, and this does not depend on his/her status (for instance if this is a graduate or undergraduate student) or final grade of other students (for instance, if the other complete final marks are low or high).

ii) Missing at random (MAR) occurs if the probability of missing value on some variable X is independent of the variable itself, but it can be computed from other variables present in the dataset.For example, student's final mark is missing, and this does depend on his/her status, but it does not depend on the final grade. Therefore, the missing final marks can be filled-in (predicted) using information about the student's status.

iii) Not Missing At Random (NMAR) occurs if the probability of missing value on some variable X is dependent on the variable itself. For instance, student's final grade is missing, and this does depend on the final grade (i.e., only grades in a special range, say 80–90%, are missing). This way, the missing value can be filled-in using the complete final marks of the other students.

1.5 Motivation

In many real life datasets missing data is present and they are frequent complications of many real-world studies. To obtain accurate inferences from the data, the data should be complete. In case the dataset contains missing values, the missing values should be imputed before performing any further analysis on the data. Statistical and Computational intelligence

techniques for data mining applications such as classification, regression, association and outlier analysis require accurate and complete data. Data imputation is of great use in such applications if the data contains missing values.

1.6 Objectives

The main objectives of the research pursued in the thesis are:

- Development of 4 data imputation architectures for numerical data based on Extreme Learning Machine (ELM), Auto associative neural networks (AANN), Evolving Clustering Method (ECM), Principal Component Analysis (PCA) and Gray System Theory (GST).
- Development of 2 data imputation techniques for numerical data based on covariance matrix, Particle Swarm Optimization (PSO), AANN, ELM and ECM.
- Development of 2 data imputation techniques for numerical data based on AANN, GST and CPNN.

1.7 Organization of thesis

The rest of the thesis is organized as follows:

Chapter 2 presents the literature review on numerical data imputation techniques.

Chapter 3 presents imputation based on ELM, GST, PCA and ECM. We proposed 4 novel imputation techniques viz. PCA-AAELM, ECM-Imputation, ECM-AAELM and Gray+PCA-AAELM.

Chapter 4 presents imputation based on ELM, ECM, Covariance matrix and PSO. We proposed 2 novel methods viz. ECM_PSO_COV and ECM_PSO_COV+ECM-AAELM.

Chapter 5 presents imputation based on CPNN, AANN and GST. We proposed 2 novel methods viz. CPAANN and Gray+CPAANN. The overall conclusions are presented in Chapter 6.

CHAPTER 2

LITERATURE REVIEW

2.1 Review of numerical data imputation techniques

Missing data is an unavoidable problem in real life datasets. The treatment of incomplete data is an important step in the preprocessing of data. Missing data handling methods for numerical data as shown in Fig. 2.1 (see Annexure) can be broadly classified into four categories: (a) deletion, (b) imputation (c) modeling the distribution of missing data and then estimate them based on certain parameters and (d) machine learning methods. Each of these techniques is discussed below.

2.1.1 Deletion Procedures

The missing data ignoring techniques or deletion techniques simply delete the cases that contain missing data. Because of their simplicity, they are widely used and tend to be the default choice for most statistics packages, but this is not an effective solution. This approach has two forms: (i) List wise deletion that omits the cases or instances containing missing values. The main drawback of this method is that the application may lead to loss of large number of observations, which may result in high error and aggravates further if the original data set itself is too small (Song and Shepperd, 2007). (ii) Pairwise deletion method that considers each feature separately.

For each feature, all recorded values are considered and missing data ignored. It is good when the overall sample size is small or missing data cases are large, (Song and Shepperd, 2007).

2.1.2 Imputation procedures

The replacement procedures are easy to perform, and some are included as an option in statistical packages. The advantages of these procedures are retention of sample size and statistical power in subsequent analysis. The earliest method of imputation is mean imputation, in which the missing values of a variable are replaced by the average value of all the remaining cases of that variable (Little and Rubin, 2002). 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 regression imputation, regression equations are computed each time by considering the attribute containing incomplete value as target variable. This method preserves the variance and covariance of missing data with other variables. The disadvantage of regression imputation is that it assumes linear relationship between the predictors and the missing variable. The technique also assumes that values are missing at random.

Hot and cold deck imputation replaces the missing values with the closest complete components, where, closest is in terms of components that are present in both vectors for each case with a missing value (Schafer, 1997). The drawback of hot deck imputation is that the estimation of missing data is based on single complete vector and thus 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, 2002). In multiple imputation procedure, each missing value is replaced by a set of reasonable and valid values, so that we get M complete data sets by replacing each value M times and by analyzing all datasets after which we can make combined inferences. According to Little and Rubin (2002), multiple imputation is better than case wise and mean substitution imputation.

2.1.3 Model-based procedures

The maximum likelihood approach to analyzing missing data assumes that the observed data are a sample drawn from a multivariate normal distribution (Desabro and Green, 1986). The

parameters are estimated by available data and then missing values are determined based on the estimated parameters. The expectation maximization algorithm is an iterative process Laird (1988). The first iteration estimates missing data and then parameters using maximum likelihood. The second iteration re-estimates the missing data based on new parameters then recalculates the new parameter estimates based on actual and re-estimated missing data (Little and Rubin, 2002).

2.1.4 Machine learning methods

In K-nearest neighbor (K-nn) approach the missing values are replaced by their nearest neighbors. The nearest neighbors are selected from the complete cases which minimize the distance function. 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. First the SOM is trained using the complete data. Second, when an incomplete pattern is presented to the SOM, its image node is chosen ignoring the distances in the missing variables: third, an activation group composed of image node's neighbors is selected; and finally each imputed value is computed based on the weights of the activation group of nodes in the missing dimensions.

In the neural network approach, MLP should be trained as regression model by using the complete cases and choosing one variable as target each time. By using appropriate MLP model, each incomplete pattern value is predicted. Several researchers Sharpe and Solly (1995), Nordbotten (1996), Gupta and Lam (1996), Yoon and Lee (1999) used MLP for missing data imputation. Ragel and Cremilleux (1999) proposed a missing value completion method. This method extends the concept of Robust Association Rules Algorithm (RAR) for databases with multiple missing values. Imputation using auto-associative neural network (AANN) is another machine learning technique. In AANN, the network is trained for predicting the inputs by taking same input variable as target (Marseguerra and Zoia, 2002) (Marwala and Chakraverty, 2006). Chen et.al, (2008) employed selective bayes classifier for classification on incomplete data with a simpler formula for computing gain ratio.

Gheyas and Smith (2010) proposed a novel nonparametric algorithm Generalized regression neural network Ensemble for Multiple Imputation (GEMI) and also developed a single imputation (SI) version of this approach—GESI. The effectiveness of the algorithms is evaluated in terms of (i) the accuracy of output classification: three classifiers (a generalized regression neural network, a multilayer perceptron and a logistic regression technique) are separately trained and tested on the dataset imputed with each imputation algorithm, (ii) interval analysis with missing observations and (iii) point estimation accuracy of the missing value imputation. Zhang et al. (2011) proposed a simple and efficient nonparametric iterative imputation algorithm (*NIIA) method* to utilize information within incomplete instances (instances with missing values) when estimating missing values. It is designed for iteratively imputing missing target values. The NIIA method imputes each missing value several times until the algorithm converges. In the first iteration, all the complete instances are used to estimate missing values. The information within incomplete instances is utilized since the second imputation iteration.

Figueroa et al. (2011) proposed a method based on an evolutionary algorithm to impute missing observations in multivariate data. A genetic algorithm based on the minimization of an error function derived from their covariance matrix and vector of means is presented. Nuovo (2011) introduced a method based on the most famous fuzzy clustering algorithm: Fuzzy C-Means (FCM) and then compared these methodologies in order to highlight the peculiar characteristics of each solution. The comparison was made in a psychological research environment, using a database of in-patients who have a diagnosis of mental retardation. The results demonstrated that completion techniques, and in particular the one based on FCM, led to effective data imputation.

Ankaiah and Ravi (2011) proposed a hybrid method for data imputation based on the K-means and Multi-layer perceptron (MLP). It is a two stage process, in first stage K-means clustering algorithm used to impute the missing values and then MLP used in second stage by taking the missing variable as target variable and remaining as inputs.

Yuan Li and Parker (2012) developed a novel Nearest Neighbor (NN) imputation method that estimates missing data in Wireless Sensor Networks (WSNs) by learning spatial and temporal correlations between sensor nodes. To improve the search time, they utilized a kd-tree data structure, which is a non-parametric, data-driven binary search tree. Instead of using traditional

mean and variance of each dimension for kd-tree construction, and Euclidean distance for kd-tree search, they used weighted variances and weighted Euclidean distances based on measured percentages of missing data.

Zhang (2012) proposed a novel k-NN (k nearest neighbor) imputation method to iteratively imputing missing data, named Gk-NN (gray k-NN) imputation to deal with heterogeneous (i.e., mixed-attributes) data. Gk-NN selects k nearest neighbors for each missing datum via calculating the gray distance between the missing datum and all the training data rather than traditional distance metric methods, such as Euclidean distance. Such a distance metric can deal with both numerical and categorical attributes.

Nishanth et al. (2012) employed a novel two-stage soft computing approach for data imputation to assess the severity of phishing attacks. The imputation method involves K-means algorithm and multilayer perceptron (MLP) working in tandem. The hybrid is applied to replace the missing values of financial data which is used for predicting the severity of phishing attacks in financial firms. After imputing the missing values, performed mining on the financial data related to the firms along with the structured form of the textual data using multilayer perceptron (MLP), probabilistic neural network (PNN) and decision trees (DT) separately.

Nelwamondoet.al. (2013) developed a novel technique for missing data estimation using a combination of dynamic programming, neural networks and genetic algorithms (GA) on suitable subsets of the input data. The proposed approach is applied to an HIV/AIDS database and the results shows that the proposed method significantly outperforms a similar method where dynamic programming is not used. Tan et.al. (2013) developed a method based on a tensor decomposition to estimate the missing value. This approach not only inherits the advantages of imputation methods based on matrix pattern for estimating missing points, but also well mines the multi-dimensional inherent correlation of traffic data. Experiments demonstrated that the proposed method achieves a better imputation performance than the state-of-the-art imputation approach even when the missing ratio is up to 90%.

França et al. (2013) proposed a novel biclustering-based approach to data imputation. This approach is based on the Mean Squared Residue metric, used to evaluate the degree of coherence

among objects of a dataset, and presents an algebraic development that allows the modeling of the predictor as a quadratic programming problem. Aydilek and Arslan (2013) utilized a fuzzy c-means clustering hybrid approach that combines support vector regression and a genetic algorithm. In this method, the fuzzy clustering parameters, cluster size and weighting factor are optimized and missing values are estimated. Recently, Nishanth and Ravi [2] proposed four hybrid methods, one online and 3 offline methods, to resolve imputation problem. They employed ECM with General regression neural network (GRNN) for online imputation, K-Means and K-Medoids with GRNN and K-Medoids with MLP for offline imputation. Most recently, Ravi and Krishna (2014) proposed four new imputation techniques based on AANN viz., PSOAANN, PSOAAWNN, RBFAANN and GRAANN. Various imputation techniques appeared in literature are presented in Table 2.1 (see Annexure).

CHAPTER 3

DATA IMPUTATION METHODS BASED ON DETERMINISTIC EXTREME LEARNING MACHINE

This Chapter presents four novel hybrid techniques for data imputation based on Extreme Learning Machine (Huang et al., 2004, 2006), Principal Component Analysis (PCA) (Pearson, 1901 and Hotelling, 1933), Evolving Clustering Method (ECM) (Song and Kasasbov, 2000, 2001) and Gray System Theory (GST) (Deng, 1982). Those proposed methods are (i) Autoassociative ELM (AAELM) with PCA (ii) GST + AAELM with PCA (iii) Evolving Clustering based data imputation (iv) AAELM with ECM. Our prime concern with AAELM was to introduce deterministic/stabilized AEELM without interrupting its performance, since the output of AAELM varies with each run due to random selection of values of parameters. Our proposed methods resolved the randomness issue of AAELM and exhibited better performance compared to AAELM. This chapter also illustrates the impact of local learning on our results through ECM and how ECM assists model to achieve better accuracy. Since the range of threshold (Dthr) for ECM could be very large, this chapter endeavored to provide a certain bound for the range of Dthr. 11 different activation functions have been employed with our

proposed methods and annotated the impact of activation function on our proposed methods. The proposed methods were tested on several regressions, classification and banking datasets, using 10 fold cross validation. The quality of the imputation is tested by using Mean Absolute Percentage Error (MAPE) value. The results of the proposed methods are compared with those of K-Means+Multilayer perceptron (MLP) imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (Krishna and Ravi, 2013). We observed that the proposed methods achieved better imputation in most of the datasets as evidenced by the Wilcoxon signed rank test to test the statistical significance of the results.

3.1 Overview of the employed techniques

The various techniques that are used in our study are extreme learning machine (ELM), evolving clustering method (ECM), principal component analysis (PCA) and gray system theory (GST).

3.1.1 Extreme Learning Machine

Extreme Learning Machine (Huang et al., 2004, 2006) is a simple and tuning free algorithm. Simple math is sufficient for implementing ELM. Autoassociative Extreme Learning Factory (AAELF) is essentially an ensemble of several AAELMs. AAELF is named by taking cue from Kernel Factory proposed by Ballings and Van den Poel (Ballings and Poel, 2013). Gradient descent based algorithms require all the weights be updated after every iteration. Therefore, gradient based algorithms are generally slow and may easily converge to local minima. On the other hand, ELM, proposed by Huang et al. (2004), randomly assigns the weights connecting input and hidden layers; and hidden biases. Then it analytically determines the output weight using the Moore-Penrose generalized inverse. It has been proved in (Huang et al., 2003) that given randomly assigned input weights and hidden biases with almost any non-zero activation function, we can approximate any continuous function on compact sets. Unlike the traditional algorithms, ELM not only achieves the minimum error but also assigns the smallest norm for the output weights. The reason for using Moore-Penrose inverse is that according to Bartlett's theory (Bartlett, 1998), smaller norm of weights results in better generalization of the feedforward neural network. The advantages of ELM over traditional algorithms are as follows:

- Simple math is required (Huang et al., 2004).
- ELM can be up to 1000 times faster than the traditional algorithm (Huang et al., 2004).
- ELM has better generalization performance as it not only reaches the smallest error but also the assigns smallest norm of weights (Huang et al., 2004).
- Our traditional algorithms work only for differential activation function but ELM also works efficiently for many non-differentiable functions (Huang et al., 2004).
- Both parameters of hidden nodes are fully independent of each other and from the training dataset (Huang et al., 2004, 2006).
- ELM resolves various issues of traditional classic gradient-based algorithms like local minima, improper learning rate, overfitting etc.

The algorithm for the ELM with architecture as shown in Fig. 3.1 (see Annexure) can be stated as follows:

Given training sample N, activation function g(x) and number of hidden neurons$\tilde{N}$,

1. Assign random input weights$w_i$and biases$b_i, i = 1 \cdots \tilde{N}$
2. Calculate the hidden layer output matrix H.
3. Calculate the output weight $\beta = H^{\dagger}T$

3.1.2Evolving Clustering Method

ECM is a one-pass, fast clustering method based on normalized Euclidean distances. It can be applied in two modes: on-line and off-line mode. The on-line method was employed in (Song and Kasasbov, 2000, 2001) for time-series prediction. The off-line ECM is an extension of on-line ECM i.e. ECM with constrained optimization. We applied on-line ECM for our experiment to resolve the problem of missing values.

Step 0: Create the first cluster center C_1 by simply taking the position of the first example from the input data stream as the first cluster center Cc_1, and setting a value 0 for its cluster radius Ru_1.

Step 1: If all examples of the data stream have been processed, the algorithm is finished. Else, the current example x_i, is taken and the distances* D_{ij}, between this example and all the n already created cluster centers Cc_j,

$D_{ij} = ||x_i - Cc_j||$, j= 1 to n, are calculated.

Step 2: If there is a cluster center (centers) Cc_j, for j= 1 to n, so that the distance value, $D_{ij} = ||x_i - Cc_j||$ is equal to, or less than, the radius Ru_j, it is assumed that the current example x_i belongs to a cluster C_m with the minimum of these distances:

$D_{im} = ||x_i - Cc_m|| = \min (D_{ij})$,

Where: $D_{ij} <= Ru_j$, j= 1 to n.

In this case, neither a new cluster is created, nor any existing cluster is updated and the algorithm returns to Step 1, else it goes to next step.

Step 3: Find a cluster C_a (with a center Cc_a and a cluster radius Ru_a) from all n existing cluster centers through calculating the values $S_{ij} = D_{ij} + Ru_j$, j=1 to n, and then select the cluster center Cc_a with the minimum value S_{ia}:

$S_{ia} = D_{ia} + Ru_a = \min \{ S_{ij} \}$, j=1 to n.

Step 4: If S_{ia} is greater than 2 * *Dthr*, the example x_i does not belong to any existing clusters. A new cluster is created in the same way as described in Step 0, and the algorithm returns to Step 1.

Step 5: If S_{ia} is not greater than 2 * *Dthr*, the cluster C_a is updated by moving its center, Cc_a, and increasing the value of its radius, Ru_a. The updated radius Ru_a^{new} is set to be equal to $S_{ia}/2$ and the new center Cc_a^{new} is located on the line connecting the new input vector x_i and the cluster center Cc_a, so that the distance from the new center Cc_a^{new} to the point x_i is equal to Ru_a^{new}. The algorithm returns to Step 1.

Normalized Euclidean distance is defined as follows:

$$\|x - y\| = \sqrt{\sum_{i=1}^{q} \left(x_i - y_i\right)^2} \Big/ \sqrt{q}$$

3.1.3 Overview of PCA (Fig. 3.2, see Annexure)

PCA performs dimensionality reduction, which transforms set of correlated variable to set of uncorrelated variable but still possesses most of the information. It generates new set of variables, called principal component (Pearson, 1901 and Hotelling, 1933). Each principal component is a linear combination of the original variables. All the principal components are orthogonal to each other, so there is no redundant information (Pearson, 1901 and Hotelling, 1933). The full set of principal components is as large as the original set of variables.

3.1.4 Overview of Gray System Theory

To select nearest neighbour for imputation, we used Gray Relational Analysis (GRA). GRA is a method of Gray System Theory (GST) which is proposed by Deng (1982). GRA measures the degree of similarity between two systems (Zhang, 2012 and Tian et al., 2013). Two things are needed to be calculated for GRA:

1. Gray Relational Coefficient (GRC)
2. Gray Relational Grade (GRG)

Formula for calculation of GRC:

$$GRC(x_{kp}^{mis}, x_i) = \frac{\min_{\forall i} \min_{\forall p} | x_{kp}^{mis} - x_{ip} | + \zeta \max_{\forall i} \max_{\forall p} | x_{kp}^{mis} - x_{ip} |}{| x_{kp}^{mis} - x_{ip} | + \zeta \max_{\forall i} \max_{\forall p} | x_{kp}^{mis} - x_{ip} |},$$

$$p = 1,2,3,.....,m.$$

$$k = 1,2,3,.....,n.$$

$$i = 1,2,3,.....,o.$$

$$0 \le \zeta \le 1$$

Where,

x_{kp}^{mis} is the kth incomplete record

p is the pth attribute with non-missing values.

x_i is the ith complete record of the dataset.

Formula for calculation of GRG:

$$GRG(x_k^{mis}, x_i) = \frac{1}{m}\sum_{p=1}^{m} GRC(x_{kp}^{mis}, x_i)$$

$$i = 1,2,....., o.$$

$$k = 1,2,....., n.$$

A larger value of GRG indicates two systems or elements are more similar and smaller value indicates less similarity of the systems or elements.

3.1.4 Overview of Transformation Functions

Varieties of activation functions have been proposed till date to improve the performance of artificial neural network. Activation function provides non-linearity, thereby becoming a necessity of neural network. Few activation functions are used in this thesis to show the fact that how it affects the output of neural network if activation function is altered. We will see that how the output fluctuated drastically due to change of activation function. We used 11 activation functions (Table 3.1, see Annexure). Sigmoid, Sin, Sinh (hyperbolic sine), Radial Basis transfer function and Gaussian are well-known activation functions. According to Glorot et al. (Glorot et al., 2011), Rectifier activation function is more biologically plausible than the popular Sigmoid activation function. Softplus activation function (Dugas et al., 2001) is a smooth approximate to the Rectifier activation function. Cloglogm is a new activation function, which is proposed by Gomes et al. (Gomes et al., 2011). Cloglogm is modified complementary log-log function and non-constant monotonically increasing function. Another variation of Sigmoid is a Bipolar Sigmoid function (Karlik and Olgac, 2001), it performed well for those types of application which produce output values in the bound of [-1, 1]. We used a Hardlim (Hard Limit) activation function; it is capable of separating an input space into two categories (0 and 1).

3.2 Architecture of the proposed methods

in this chapter, four novel methods have been proposed for data imputation task based on above discussed techniques.

3.2.1 Algorithm for Autoassociative Extreme Learning Factory

Until now ELM has been used extensively in two fields, classification and regression problems. This chapter will show the extension of ELM as an Autoassociative ELM (AAELM) and use it in data imputation. An AAELF is constructed by ensemble of several AAELMs. Architecture of AAELM (Fig. 3.3, see Annexure) consists of three layers namely input layer, hidden layer and output layer, which is same as the input layer. The number of hidden nodes in the hidden layer is an arbitrary constant defined by the user. Each input node in the input layer is connected to each node in the hidden layer and each hidden node is connected to the each of the node in the output layer. So, we used Extreme Learning Machine to train the 3-layered auto associative neural network. The architecture of the AAELF is shown in Fig. 3.3 (see Annexure). The training algorithm for AAELF of data imputation is as follows:

1. Normalize the dataset in the range of [0, 1].
2. Impute the missing value in the dataset based on Mean imputation.
3. Select the number of hidden nodes distribution for hidden nodes and activation function.
4. Initialize randomly the weight values between the input and hidden layers in the range of [0, 1]. The output nodes contain the input variables as the target variables thereby bringing in the auto associative concept.
5. Calculate the hidden layer output matrix H.
6. Calculate the output weight β.

$$\beta = H^{\dagger}T$$

7. Calculate mean absolute percentage error (MAPE) value to measure the quality of the imputation Flores (1986):

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{x_i - \widehat{x_i}}{x_i}\right|$$

   Where, n is the number of missing values in a given dataset, $\widehat{x_i}$ is predicted by the proposed AAELF for the missing values and x_i is the actual value.
8. Repeat steps 2 to 5 for 10 times for each combination of weight distribution and activation function.

Since ELM depends heavily on the random weights that connect the input and hidden layers, it yields different results in different runs. Sometimes, the results could be fluctuate wildy. The same arguments hold good for AAELM also. Hence, in order to circumvent the fluctuations in results, AAELF is designed. AAELF is essentially an ensemble of 10 independent runs of the AAELM on the same dataset. Since 10 different AAELMs are ensemble, the AAELM is called the AAELF. As the input weights and hidden biases can be chosen randomly, we have used random number following Uniform, Normal and Logistic distributions. Sigmoid and Gaussian activation functions are used in the hidden layer. So, we constructed the AAELF ensemble for each of the six possible combinations.

In order to overcome random behavior of Ensembled AAELM, following methods have been proposed:

- PCA-AAELM
- ECM based Imputation
- ECM-AAELM
- Gray+PCA-AAELM

3.2.2 Algorithm of the proposed Method: PCA-AAELM (Fig. 3.4, see Annexure)

A new technique is proposed by hybridization of autoassociative neural network, ELM and Principal Component Analysis (PCA) as shown in Fig. 3.4 (see Annexure).

1. Normalize the data in the range of [0, 1].
2. Apply Mean Imputation on incomplete dataset.
3. Perform PCA with the complete set of records.
4. Selects the maximum number of hidden nodes by determining the amount of principal components is necessary to explain the variance in the data.
5. Perform non-linear transformation on scores, the scores are the data formed by transforming the original data into the space of the principal components and apply Moore-Penrose generalized inverse after non-linear transformation to obtain $H^{\dagger}$.

6. Finally, the estimation of output layer weights is done as proposed by Huang et al. (2003), by solving the linear system Hβ = T using the Moore-Penrose generalized inverse.
7. Compute the mean absolute percentage error (MAPE) (Flores, 1986) value:

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{x_i - \hat{x}_i}{x_i}\right|$$

Where, n is the number of missing values in a given dataset, $\hat{x}_i$ is predicted by the proposed method, PCA-AAELM, for the missing values and x_i is the actual value.

3.2.3 Algorithm of the proposed Method: ECM based Imputation (Fig. 3.5, see Annexure)

ECM is one of the simplest unsupervised learning algorithms, which aids to solve missing data problem by its substantial local learning capability. The procedure of imputation is as follows:

1. Normalize the data in the range of [0, 1].
2. Apply Mean Imputation on incomplete dataset.
3. Divide a dataset in two parts: sets of complete and incomplete records.
4. Perform ECM with the set of complete records and identify all the cluster centers.
5. Attribute value, say x_k, in an incomplete record is imputed by the corresponding value of the attribute in the center of the nearest cluster by measuring the Euclidean distance between the incomplete record excluding the missing value and the cluster centers excluding the value in the same position. The Euclidean distance is measured by using the following formula:

$$D_j = \sum_{i=1; i\neq k}^{n}\left|x_i - c_j\right|^2$$

Where, j – Number of cluster centers.

n – Number of complete components in each record.

6. To measure the effectiveness of the imputation, compute the mean absolute percentage error (MAPE) (Flores, 1986) for incomplete records.

3.2.4 Algorithm of the proposed Method: ECM-AAELM (Fig. 3.6, see Annexure)

1. Normalize the data in the range of [0, 1].
2. Divide a dataset in two parts: sets of complete and incomplete records.
3. Perform ECM with the set of complete records and identify all the cluster centers.
4. Attribute value, say x_k, in an incomplete record is imputed by the corresponding value of the attribute in the center of the nearest cluster by measuring the Euclidean distance between the incomplete record excluding the missing value and the cluster centers excluding the value in the same position. The Euclidean distance is measured by using the following formula:

   a. $$D_j = \sum_{i=1; i \neq k}^{n} \left| x_i - c_j \right|^2$$

      b. Where, j – Number of cluster centers.

         i. n – Number of complete components in each record.

5. Calculate the normalized Euclidean distance from each cluster center, which is presented as hidden nodes.
6. Perform non-linear transformation by the activation function on above distance and apply Moore-Penrose generalized inverse after non-linear transformation to obtain $H^{\dagger}$.
7. Finally, the estimation of output layer weights is done as proposed by Huang et al. (2003), by solving the linear system $H\beta = T$ using the Moore-Penrose generalized inverse.
8. Compute the mean absolute percentage error (MAPE) value for missing values.

3.2.5 Algorithm for the proposed Method: Gray+PCA-AAELM (Fig. 3.7, see Annexure)

A new technique is proposed by hybridization of Gray System Theory, autoassociative neural network, ELM and Principal Component Analysis (PCA) as shown in Fig. 3.7 (see Annexure).

1. Normalize the data in the range of [0, 1].
2. Apply Gray distance based imputation instead of Mean imputation on incomplete dataset.

3. Rest of the procedure is similar to PCA-AAELM (from Step 3 to Step 7).

3.3 Experimental design

Datasets are divided into two parts: one is set of complete records and other is set of incomplete records. Complete records have been used for training process and incomplete records have been used for testing process. We also perform 10-fold cross validation in our experiment. Number of incomplete records are 10 % of the total records. Further, we calculated MAPE for missing values in incomplete records. We also performed statistical testing in order to verify the statistical significance of our obtained result from the proposed methods (Lowry; Wilcoxon, 1945; Siegel, 1956). Wherever, we employed ELM in our proposed methods, we deployed 11 activation functions with those proposed methods. We employed 11 activation functions as a part of our experiment to study the impact of activation function on our proposed methods. We also compared the average MAPE values of the proposed methods with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K-Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (Krishna and Ravi, 2013).

3.4 Results and discussions

We applied the proposed methods viz., PCA-AAELM, Gray+PCA-AAELM, ECM-Imputation, ECM-AAELM on several datasets and compared our outcomes from various methods viz., K-Means+MLP, K-Medoids+MLP, K-Means+GRNN, K-Medoids+GRNN and PSO_COV. The effectiveness of the proposed and existing methods for data imputation is tested on 4 regression, 4 classification and 4 banking datasets. We used 10 fold cross validation on all datasets and we used MAPE values to find the quality of the imputation. The average MAPE values obtained over 10-fold cross validation for the proposed method is presented in the Table 3.2, 3.3 and 3.4 (see Annexure). For online data imputation, the number of clusters obtained by ECM is dictated by a parameter known as the distance threshold *Dthr*. The *Dthr* value that yielded the best reduction in MAPE is obtained by varying the large range of *Dthr* value from 0.001 to 0.999, in steps of 0.001 and the least MAPE value thus obtained is tabulated. We kept same *Dthr* value for all folds. Similarly, PCA, which was employed in PCA-AAELM, has a parameter known as variance, which is accountable for the selection of necessary principal components for the dataset. The value of variance that yields the least MAPE is selected by varying this parameter

from 0.01 to 0.99 and the corresponding MAPE value is tabulated. In Gray+PCA-AAELM, a crucial parameter is $\zeta \in [0,1]$, we kept its value as 0.5 for all datasets.

3.4.1 Performance analysis of various activation functions

In all proposed methods viz., ECM-AAELM, Gray+PCA-AAELM, PCA-AAELM, activation function played crucial role to minimize the MAPE value.

In case of ECM-AAELM, It can be easily observed by the Table 3.2 (see Annexure) and Fig. 3.8(b) (see Annexure) that Hardlim activation function exhibited worst performance compared to other activation functions. For 3 out of 12 datasets viz., Forest fire, Prima Indian, UK credit, performance of Hardlim activation function is substantial. Except UK bankruptcy dataset, ECM-AAELM yielded satisfactory results for 10 out of 11 activation compared to other proposed methods. Only Hardlim activation function did not perform well with ECM-AAELM. Tribas activation function yielded best result among all activation function because it exhibited least MAPE value for 6 out of 12 datasets compared to other activation functions. MAPE values of Tribas activation function are merely more than 1% for rest of the 5 datasets, which can be easily observed in Table 3.2 (see Annexure).

In case of PCA-AAELM (Table 3.3 and Fig. 3.8(a), see Annexure), we obtained similar observation regarding Hardlim activation function as obtained in case of ECM-AAELM. For 4 out of 12 datasets viz., Forest fire, Prima Indian, Spectf, UK credit, performance of Hardlim activation function is substantial. It performed worst among all activation functions however, one surprised result obtained by employing this activation function with PCA-AAELM that it provided least MAPE value compared to all activation functions for UK credit dataset. Softplus activation function yielded least MAPE value among all activation functions and its MAPE value is nearly 1% more than the best value obtained by all activation function for 5 datasets. Sigmoid, Sinh, Bsigmoid, Hardlim, Radbas and Softplus activation function yielded least MAPE value for 2 datasets, 1 dataset, 2 datasets, 1 dataset, 2 datasets and 4 datasets respectively. Rest of the 3 activation functions viz., Cloglogm, Sine and Tribas were no be able to obtain least MAPE value for any of the 12 datasets.

In case of Gray+PCA-AAELM, we obtained similar observation regarding Hardlim activation function as obtained in case of ECM-AAELM and PCA-AAELM (Table 3.4 and Fig. 3.8(c), see Annexure). For 4 out of 12 datasets viz., Forest fire, Prima Indian, Spectf, UK credit, performance of Hardlim activation function is substantial. However, it did not yield least MAPE value for any of the 12 datasets. Both Softplus and sigmoid activation function yielded least MAPE value for 4 datasets among all activation functions. A variation of Sigmoid activation function viz., Bsigmoid activation function achieved least MAPE value for 3 datasets and Radbas yielded least MAPE value for 1 dataset. Rest of the 5 activation functions viz., Sinh, Cloglogm, Sine, Hardlim, Tribas, were not be able to obtain least MAPE value for any of the 12 datasets.

3.4.1 Performance analysis of our proposed methods vs. hybrid methods presented in the Table 3.5 (see Annexure)

We employed our proposed methods on 12 datasets and calculated the average MAPE value over 10 fold cross validation experiment on all datasets. Our all experimented results are presented in Table 3.5 (see Annexure). Following are the comparative discussion between our proposed methods vs. hybrid methods viz., K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (Krishna and Ravi, 2013):

For Auto MPG dataset, Gray+PCA-AAELM yielded best outcomes among all the 4 proposed methods and all the existing methods presented in the Table 3.5 (see Annexure) except K-Medoids+GRNN. The difference of MAPE value between Gray+PCA-AAELM and K-Medoids+GRNN is 0.26 only. A drastic reduction of 11.71% in MAPE value is observed by employing Gray distance based imputation at first stage and PCA-AAELM at second stage. The MAPE is reduced from 28.63% (PCA-AAELM) to 16.92% (Gray+PCA-AAELM). PCA-AAELM is worst performer among all proposed methods for this dataset.

For the Body fat dataset, the MAPE value is observed less than 10% for all the proposed imputation techniques. PCA- AAELM outperformed 5 existing methods presented in the Table 3.5 (see Annexure) viz., K-Means+MLP, K-Means+GRNN, K-Medoids+MLP and PSO_COV.

The MAPE is reduced from 6.01% to 5.41% by employing Gray Distance based imputation at first stage and PCA-AAELM at second stage. The MAPE value is reduced further from 5.41% to 5.33% by employing ECM with AAELM (ECM-AAELM). ECM-AAELM yielded best outcomes among all the proposed methods and existing methods listed in Table 3.5 (see Annexure).

For the Boston housing dataset, PCA-AAELM yielded less MAPE value compared to K-Means+MLP and PSO_COV but outperformed by rest of the methods. The value of MAPE is reduced from 20.9% to 17.46% by employing Gray distance based imputation in first stage and PCA-AAELM at second stage. It outperformed all the existing methods presented in the Table 3.5 (see Annexure). It also yielded better results compared to our two proposed methods viz., PSO_COV and PCA-AAELM. The MAPE value is further reduced from 17.46% to 16.48% by employing ECM_AAELM instead of Gray+PCA-AAELM and it outperformed all the existing methods listed in the Table 3.5 (see Annexure). PSO_COV performed worst among all the methods presented in the Table 3.5 (see Annexure).

In regards to the Forest Fire dataset, all three proposed methods viz., ECM-AAELM, PCA-AAELM and Gray+PCA-AAELM, outperformed all methods proposed by Ankaiah and Ravi (2011) & Nishanth and Ravi (2013). PCA-AAELM yielded least MAPE value among all three proposed methods.

For the Iris dataset, ECM-AAELM yielded best outcomes and PCA-AAELM yielded worst outcomes among all the methods listed in the Table 3.5 (see Annexure). The MAPE value is reduced from 10.23% to 5.79% by employing Gray+PCA-AAELM instead of PCA- AAELM. Gray+PCA-AAELM yielded better outcomes compared to all the earlier methods listed in the Table 3.5 (see Annexure) except ECM-Imputation.

For the Prima Indian dataset, Gray+PCA-AAELM is the best performer among all the methods listed in the Table 3.5 (see Annexure). The MAPE value is reduced from 23.95% to 22.06% by employing PCA-AAELM instead of ECM- AAELM. The MAPE value is further reduced from 22.06% to 22.03% by employing Gray Distance based imputation at first stage and

PCA-AAELM at second stage. So, our proposed methods outperformed all the existing methods listed in Table 3.5 (see Annexure) for this dataset.

For the Spanish dataset, the MAPE value is reduced from 30.09% to 28.06% by employing Gray+PCA-AAELM instead of PCA-AAELM. It outperformed all methods except K-Medoids+GRNN. The MAPE value is drastically reduced from 30.09% to 22.09% by employing ECM-AAELM instead of PCA-AAELM. ECM-AAELM performed better compared to all the methods listed in the Table 3.5 (see Annexure).

For Spectf dataset, our all three proposed methods viz., ECM-AAELM, Gray+PCA-AAELM, PCA-AAELM, outperformed all the existing methods presented in the Table 3.5 (see Annexure). The MAPE value is reduced from 9.11% to 8.38% by employing Gray distance based imputation at first stage and PCA-AAELM at second stage. ECM-AAELM performed best among all the proposed and existed methods listed in Table 3.5 (see Annexure).

For Turkish dataset, the value of MAPE is reduced from 30.18% to 27.38% by employing Gray distance based imputation at first stage and PCA-AAELM at second stage instead of PCA-AAELM only. A drastic reduction of 8.69% in MAPE value is observed by employing ECM_AAELM instead of PCA-AAELM. ECM-AAELM outperformed all the existing methods except K-Medoids+GRNN presented in the Table 3.5 (see Annexure). However, remaining two proposed methods are outperformed by all proposed methods presented by Ankaiah and Ravi (2011) & Nishanth and Ravi (2013). But PCA-AAELM performed better than only one method PSO_COV and Gray+PCA-AAELM performed better than PSO_COV and ECM-Imputation.

For UK bankruptcy dataset, there are no proposed methods, which performed well compared to any of the methods presented in the Table 3.5 (see Annexure). But as we experienced earlier ECM-AAELM performed better than ECM-Imputation.

For UK Credit dataset, PCA-AAELM yielded least MAPE value among all proposed methods. PCA-AAELM outperformed 4 existing methods presented in the Table 3.5 (see Annexure) viz., K-Means+MLP, K-Means+GRNN, K-Medoids+MLP and PSO_COV. It also performed better than ECM-Imputation. The MAPE value is reduced from 26.85% to 25.27% by employing PCA-AAELM instead of ECM-AAELM.

For the Wine dataset, Gray+PCA-AAELM is the best performer among all the proposed methods. The MAPE value is reduced from 16.6% to 14.78% by employing Gray Distance based imputation at first stage and PCA-AAELM at second stage. ECM-AAELM and Gray+PCA-AAELM outperformed all the methods except K-Medoids+GRNN. Nevertheless, difference of MAPE value between K-Medoids+GRNN and both of the proposed methods, ECM-AAELM and Gray+PCA-AAELM, is only 0.13 and 0.03 respectively.

We also performed the Wilcoxon two-tailed signed rank test at 1% level of significance to test the statistical significance of the results. The Wilcoxon test values for all the proposed methods are presented in Table 3.6, 3.7 and 3.8 (see Annexure). Wilcoxon test is not performed with AAELM as the proposed imputation techniques outperformed it by a large margin. The critical value from the table for N=10 is 3 at 1% level of significance. According to the Wilcoxon signed rank test, the obtained value is statistically significant if it is equal or smaller than the critical value from the table (www.sussex.ac.uk/Users/grahamh/RM1web/WilcoxonTable2005.pdf). Therefore, by observing the obtained values of the proposed and existed methods, we conclude that the obtained values from proposed methods are statistically significant compared to existing methods for all datasets

Following points have to be noted from above discussion:

i. ECM-Imputation alone outperformed many existing hybrid methods for various datasets.
ii. When we compare ECM imputation with ECM+GRNN then ECM imputation outperformed ECM+GRNN for 6 datasets. This shows that if appropriate Dthr would not be applied then even hybridization with ECM could lead to large MAPE value.
iii. For UK bankruptcy dataset, even Mean imputation outperformed all the methods proposed in this chapter.
iv. When we employed Gray distance based imputation with PCA-AAELM then the performance of Gray+PCA-AAELM degraded for 3 datasets viz., Forest fire, UK bankruptcy and UK Credit. Otherwise, Gray distance aided PCA-AAELM to enhance its performance for rest of the 9 datasets.

v. When we closely observe the MAPE value of ECM-Imputation and ECM-AAELM. ECM-AAELM always improved the accuracy for all datasets due to insertion of the concept of global approximation with ECM-Imputation through AAELM.

vi. ECM-AAELM and Gray+PCA-AAELM performed best among our four proposed methods.

vii. It is recommended to use Softplus activation function for PCA-AAELM and Gray+PCA-AAELM.

viii. ECM-AAELM outperformed PCA-AAELM for 9 out of 12 datasets because ECM-AAELM has strong local learning capability due to ECM and PCA does not perform any local learning task, it simply does linear transformation of the data.

ix. MAPE values fluctuated wildly from one activation function to other activation function in case of PCA-AAELM and Gray+PCA-AAELM. In contrast, MAPE values of ECM-AAELM are not oscillating wildly except Hardlim activation function.

x. We experimented on a large range of 999 threshold values from 0.001 to 0.999, in steps of 0.001, in order to see which value of *Dthr* performs better compared to other methods. When we observe the graph from Fig. 3.9(a) (see Annexure), we conclude that approximately after 0.399, MAPE values are constant irrespective of any threshold value which is greater than 0.399. These graphs are depicted for the behavior of the Sigmoid activation function on ECM-AAELM over a large range of *Dthr* values for 12 datasets. We experimented similar for all activation functions over a large range of *Dthr* value from 0.001 to 0.999 and experienced the same result as discussed above.

xi. Hardlim activation function did not perform well for any of the proposed methods compare to other activation function in most of the cases.

xii. It can be observed in Hardlim activation function graph (Fig. 3.9(b), see Annexure); its MAPE values yielded same MAPE value from start to end irrespective of increasing *Dthr* values. Therefore, Hardlim activation function is not benefitted by variation of threshold values in ECM.

xiii. We did not discuss the results of AAELM because all 3 proposed methods outperformed it by a large margin. It can be observed in Table 3.9 (see Annexure).

3.5 Conclusion

In this chapter, we proposed four new imputation algorithms for data imputation viz., PCA-AAELM, Gray+PCA-AAELM, ECM-Imputation and ECM-AAELM imputation. We conclude that ECM-AAELM can be used for online data imputation regardless of any activation function except Hardlim activation function because variation of activation function did not impact wildly on this method and it is best among our all four proposed methods. In contrast, Softplus activation function is recommended for PCA-AAELM and Gray+PCA-AAELM due to better imputation capability compared to other activation functions, which has been employed so far in our experiment. The results demonstrate that there is a significant reduction in MAPE after employing gray distance based imputation instead of Mean imputation with PCA-AAELM. Our proposed algorithms are fast but user's intervention is required for selection of two parameters, Dthr for ECM to get better performance. Our next chapter will be focused on selection of optimal parameters value without altering predictive efficiency of the algorithm.

CHAPTER 4

DATA IMPUTATION BASED ON OPTIMIZED EXTREME LEARNING MACHINE

In this chapter, we will resolve the problem emerged in our previous chapter. Two novel hybrid methods have been proposed for data imputation using Evolving Clustering method (ECM), Particle Swarm Optimization (PSO) (Kennedy and Eberhart, 1995) with covariance matrix and Autoassociative Extreme Learning Machine (AAELM). The results of the proposed method is compared with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013), PSOAANN, PSOAAWNN, RBFAANN and GRAANN (Ravi and Krishna, 2014). We will also compare our results from ECM imputation and ECM-AAELM in order to see the impact of optimal *Dthr* value on the proposed methods. Our proposed methods preserved the covariance structure of the data as PSO_COV and as well as yielded better performance compared to PSO_COV in most of the datasets. We also resolved the issue emerged in previous chapter that user intervention is required for selection of Dthr value in ECM based imputation. In our proposed model, we

employed PSO for finding the optimal Dthr value and to minimize the two errors in a nested form (i) Mean squared error between the covariance matrix of the set of complete records and the covariance matrix of the set of total records including imputed ones. (ii) Absolute difference between the determinants of the two covariance matrices. The concept of local learning and global approximation are hybridized by using ECM and AAELM techniques in order to obtain more accurate imputation. Further we also performed a statistical testing to ensure the credibility of our result, which is yielded by our proposed methods.

4.1 Overview of the employed techniques

In the proposed method, we used Evolving clustering method (ECM), Autoassociative Extreme learning machine (AAELM), Particle swarm optimization (PSO), covariance matrix and determinant of covariance matrix. Whereas, we already discussed ECM and AAELM in previous chapter. We also employed 11 activation functions for one of the proposed methods. We already discussed these transformation functions in previous chapter, so no need to discuss here. So we will discuss here only PSO algorithm.

4.1.1 Particle swarm optimization

Particle Swarm Optimization is an evolutionary computation based global optimization algorithm based on flocking of birds was proposed by Kennedy and Eberhart in the year 1995 (Kennedy and Eberhart, 1995). This is a population-based technique where each solution is defined by a particle in the population. Here, each particle is represented by its respective positions and velocities in the N dimensional solution space. PSO algorithm operates in three phases:

1) Initialization phase 2) Velocity and Particle position updation phase 3) Termination phase

In the initialization phase, each particle is randomly initialized to some set of positions and velocities. Each particle is associated with a neighborhood best or local best (P_{lb}) which indicates the best fitness value attained by the particle in its path. The best particle fitness attained by the movement of entire group of particles in the solution space is the global best (P_{gb}) which forms the final solution once the termination criteria is met.

In the second phase, each particle velocity updated dynamically with respect to its position (x_{old}) by using local best (P_{lb}) and global best (P_{gb}) as follows:

$$V_{new} = w * V_{old} + c_1 * rand * (p_{lb} - x_{old}) + c_2 * rand * (p_{gb} - x_{old}) \quad (1)$$

$$x_{new} = x_{old} + V_{new} \quad (2)$$

where c_1 and c_2 are two predefined positive constants also known as acceleration coefficients, w is the inertia weight value, rand is a random number generated from uniform distribution U(0,1). Eberhart and Shi (2000) suggested the value of the parameters, w, C_1 and C_2 are 0.7298, 1.49618, 1.49618 respectively.

The algorithms finally terminates once the convergence criteria is met. Convergence criteria can be any fixed number of iterations or some other criteria based on the problem requirement.

4.2 Architecture of the proposed methods

We proposed two imputation methods based on the AAELM, ECM and particle swarm optimization (PSO) algorithm to minimize the error function derived from their covariance matrix and determinant of their covariance matrix.

4.2.2 Algorithm of the Proposed Method: ECM_PSO_COV

Total data records (X_t) are needed to divide in two parts viz; complete data records (X_c) and incomplete records (X_{ic}) to train the model. As we discussed above, our proposed algorithm will have some resemblance with PSO_COV method (Krishna and Ravi, 2013) in terms of fitness function. However, other than fitness function. It is completely different methodology. The algorithm for the proposed method, which is based on the ECM, covariance matrix and PSO of the data points, is as follows:

- Compute the covariance matrix for the complete data records (X_c). Covariance matrix will be always a square matrix. As an example, if order of matrix is (k x n) then the order of the covariance matrix (X_{cov}) is (n x n).
- Performed ECM on complete data records (X_c) with randomly initialized Dthr value by PSO to obtain cluster centers.

- Performed ECM imputation for the missing records or incomplete records (X_{ic}) in the total records (X_t) of the dataset as mentioned in previous section.
- Compute the covariance matrix for the total data records (X_t) after ECM imputation of missing records. If total records (X_t) is order of (m x n) matrix then the covariance matrix (T_{cov}) of total records (X_t) will be a square matrix of order (n x n).

  If

  (MSE (X_{cov}, T_{cov}) < ϵ) and (|Det(X_{cov})-Det(T_{cov})|< ϵ) (5)

  then exit.

  Otherwise, invoke the PSO for selecting other Dthr value.

  Where,

  ϵ - Small positive value;

  MSE (X_{cov}, T_{cov}) - Mean squared error between Xcov and Tcov.

  Det(X_{cov}) - Determinant of the covariance matrix Xcov and

  Det(T_{cov}) - Determinant of the covariance matrix Tcov.

  Two fitness functions has been used to determine the convergence criteria viz., MSE (X_{cov}, T_{cov}) and | Det(X_{cov}) -Det(T_{cov}) |.
- Repeat the above two steps until convergence.

Thus, in this chapter, PSO is used to minimize the two error functions in a nested form (i) Mean squared error between the covariance matrix of the set of complete records and the covariance matrix of the set of total records including imputed ones. (ii) Absolute difference between the determinants of the two covariance matrices. The algorithm is designed to stop only when these two errors become very small across two consecutive iterations. Fig. 4.1 (see Annexure) depicts the flow chart of the proposed algorithm. After completion of the process, the model yields optimum Dthr value for which fitness function has a minimum difference. Afterwards, estimate the missing values using ECM imputation with the optimized Dthr value. Mean absolute percentage error (MAPE) is used to quantify the quality of prediction. By comparing the covariance of complete data records and the total data records after data imputation, our proposed method preserved the covariance structure of the data similar to PSO_COV (Krishna and Ravi, 2013).

4.2.2 Algorithm of the Proposed Method: ECM_PSO_COV+ECM-AAELM (Fig. 4.2, see Annexure):

1. Normalize the data in the range of [0, 1].
2. Divide a dataset in two parts: sets of complete and incomplete records.
3. Perform ECM with the set of complete records and identify all the cluster centers.
4. The *Dthr* value applied with *ECM_PSO_COV+ECM-AAELM Method* is obtained by our previous proposed method *ECM_PSO_COV*.
5. Perform ECM imputation based on step 6 instead of Mean imputation.
6. Attribute value, say x_k, in an incomplete record is imputed by the corresponding value of the attribute in the center of the nearest cluster by measuring the Euclidean distance between the incomplete record excluding the missing value and the cluster centers excluding the value in the same position. The Euclidean distance is measured by using the following formula:

$$D_j = \sum_{i=1; i \neq k}^{n} \left| x_i - c_j \right|^2$$

Where, j – Number of cluster centers.

n – Number of complete components in each record.

7. Calculate the normalized Euclidean distance from each cluster center, which is presented as hidden nodes.
8. Perform non-linear transformation by the activation function on above distance and apply Moore-Penrose generalized inverse after non-linear transformation to obtain $H^{\dagger}$.
9. Finally, the estimation of output layer weights is done as proposed by Huang et al. (2004), by solving the linear system $H\beta = T$ using the Moore-Penrose generalized inverse.
10. Compute the mean absolute percentage error (MAPE) for missing values.

4.3 Experimental Design

Datasets are divided into two parts: one is set of complete records and another is set of incomplete records. Complete records have been used for training process and incomplete records have been used for testing process. We also perform 10-fold cross validation in our experiment. Number of incomplete records are 10 % of the total records. We applied the ECM algorithm on the complete dataset and missing values of the attribute in incomplete records were imputed by the corresponding value of the attribute of the nearest cluster center. For selection of optimal *Dthr* value in ECM for each fold of the dataset, we applied PSO optimization algorithm and two fitness functions employed as mentioned by Krishna and Ravi (2013) in their paper. The value of maximum population and maximum generation size are 30 and 100 respectively. Further, we supplied same *Dthr* value to ECM-AAELM. Further, we calculated MAPE for missing values in incomplete records. We also performed statistical testing in order to verify the statistical significance of our obtained result from the proposed methods. In case of ECM_PSO_COV+ECM-AAELM, we deployed 11 activation functions as a part of our experiment to study the impact of activation function on the proposed method. So, 11 activation functions have been applied with different folds of dataset with the proposed method *ECM_PSO_COV+ECM-AAELM* and best result is utilized to compare our result to the result of other existing methods. We also compared the average MAPE values of the proposed methods with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013), PSOAANN, PSOAAWNN, RBFAANN and GRAANN (Ravi and Krishna, 2014). We will also compare our result of the proposed methods of previous chapter in order to see the impact of optimized *Dthr* value.

4.4 Results and Discussions

Our proposed methods (ECM_PSO_COV & ECM_PSO_COV+ECM-AAELM) are implemented and tested in MATLAB using computer running under Windows 7 environment with Intel processor. Following discussion will evince the better performance of the proposed algorithms over various hybrid methods by Ankaiah and Ravi (2011), Nishanth and Ravi (2013), Krishna and Ravi (2013), Ravi and Krishna (2014). We also compared our results from the

results of ECM imputation and ECM-AAELM. This comparison exhibits that how ECM imputation is efficiently dictated by optimal Dthr value in case of both the proposed methods. We also observed that how hybrid of local learning and global approximation outperformed all the existing methods presented in the Table 4.1 (see Annexure) for most of the datasets. Results of the proposed and existing methods are presented in the Table 4.2 (see Annexure).

Table 4.1 (see Annexure) shows the impact of various activation functions on ECM_PSO_COV+ECM_AAELM. After a close observation of Table 4.1 (see Annexure) and Fig. 4.3 (see Annexure), we concluded that Sigmoid is the best performer and Hardlim activation function is the worst performer among all activation functions. Hardlim (Hard Limit) activation function didn't perform well because it separates an input space into two categories (0 and 1) based on following function:

$$\text{Hardlim}(n) = 1, \text{ if } n \geq 0$$

$$0, \text{ otherwise}$$

A. Auto Mpg Dataset

ECM_PSO_COV method outperformed all the existing methods presented in the Table 4.2 (see Annexure). Only results of three hybrids viz; K-Medoids + GRNN, ECM + GRNN and GRAANN are nearby our proposed method. Accuracy of K-Medoids + GRNN, ECM + GRNN and GRAANN is lagged by 1.31%, 1.65% and 0.19% from ECM_PSO_COV respectively. It can be easily observed that ECM_PSO_COV performed better by optimal selection of *Dthr* parameter compared to ECM_Imputation for Auto_Mpg dataset.

When we passed the same obtained *Dthr* value by ECM_PSO_COV to ECM-AAELM then it reduced the error from 15.35% to 14.39%. So, it is obvious that ECM_PSO_COV+ECM_AAELM outperformed all existing and also one of our proposed methods, ECM_PSO_COV.

B. Body Fat Dataset

For this dataset also, our proposed method ECM_PSO_COV outperformed all the existing methods presented in the Table 4.2 (see Annexure) except GRAANN. ECM_PSO_COV lagged by merely 0.35% from GRAANN. ECM_Imputation did not yield better accuracy due to lack of optimal selection of *Dthr* value for Body Fat dataset.

Our other proposed method, ECM_PSO_COV+ECM_AAELM, and GRAANN performed equally. But it outperformed other existing methods and one of the proposed methods, ECM_PSO_COV.

C. Boston Housing Dataset

In the case of Boston Housing dataset, all methods are significantly outperformed by the proposed method. All existing methods except GRAANN in the Table 4.2 (see Annexure) have at least 3% more MAPE value compared to ECM_PSO_COV. Even, the difference of MAPE values between ECM_PSO_COV and various methods are more than 10%. A reduction of 3.34% in MAPE is observed when optimal *Dthr* value has been applied using ECM_PSO_COV instead of ECM_Imputation. Performance of GRAANN is only nearby our proposed method for this dataset. Difference of MAPE value between ECM_PSO_COV and GRAANN is only 0.88%.

After employing ECM+AAELM with optimum *Dthr* value obtained by ECM_PSO_COV then MAPE value is further reduced by 0.32%. Since accuracy of ECM_PSO_COV+ECM_AAELM is better than ECM_PSO_COV, so, it also outperformed all the methods listed in Table 4.2 (see Annexure).

D. Forest Fire Dataset

In regards to the Forest Fire Dataset, similar performance is observed as observed in Boston Housing Dataset. Except GRAANN, all the methods in the Table 4.2 (see Annexure) have at least 4% more MAPE value compared to ECM_PSO_COV. Accuracy of GRAANN is lagged by 0.13%. A reduction of 3.96% in MAPE is observed when ECM_PSO_COV has been applied instead of ECM_Imputation.

Our second proposed method, ECM_PSO_COV+ECM_AAELM outperformed ECM_PSO_COV and obviously, it significantly outperformed all the methods presented in the Table 4.2 (see Annexure).

E. Iris Dataset

All the existing methods in the Table 4.2 (see Annexure) are outperformed by the proposed method. Except ECM + GRNN, ECM imputation and GRAANN, all the methods presented in the Table 4.2 (see Annexure) have at least 3% more MAPE value compared to ECM_PSO_COV. The MAPE is reduced from 5.27 to 4.82 by applying optimal *Dthr* value obtained by our proposed method ECM_PSO_COV instead of ECM_Imputation.

Further, we employed our other proposed method, ECM_PSO_COV+ECM_AAELM; outperformed ECM_PSO_COV by merely 0.07% and obviously, it outperformed all the existing methods presented in the Table 4.2 (see Annexure).

F. Prima Indian Dataset

In regards to the Forest Fire Dataset, ECM_PSO_COV outperformed all the existing methods presented in the Table 4.2 (see Annexure) except PSOAANN, PSOAAWNN and GRAANN, proposed by Ravi and Krishna (2014). 2.58% of reduction in MAPE value is observed by applying ECM_PSO_COV instead of ECM_Imputation.

When we deployed our second proposed method, ECM_PSO_COV+ECM_AAELM, 1.20% of reduction in MAPE value has been observed compared to ECM_PSO_COV. It is outperformed by only one method, PSOAANN (Ravi and Krishna, 2014) among all the existing methods presented in the Table 4.2 (see Annexure).

F. Spanish Dataset

This dataset showed that how much wildly an appropriate *Dthr* value could influence the result of imputation using ECM_PSO_COV. A drastic reduction of 11.25% in MAPE value is observed by applying optimized Dthr value using ECM_PSO_COV compared to ECM_Imputation. This is an exemplary dataset, which showed that optimal selection of *Dthr*

value improved the accuracy of imputation. It also showed that an inappropriate selection of *Dthr* value could lead to totally wrong result. Except K-Medoids + GRNN and GRAANN, all the existing methods presented in the Table 4.2 (see Annexure) have 11% more MAPE value compared to ECM_PSO_COV.

When we employed ECM_PSO_COV+ECM_AAELM then MAPE value further reduced from 20.73% to 16.99% as compared to ECM_PSO_COV. Except K-Medoids + GRNN and GRAANN, all the existing methods in the Table 4.2 (see Annexure) have at least 15% more MAPE value compared to ECM_PSO_COV+ECM_AAELM. K-Medoids + GRNN and GRAANN have 9.02% and 6.29% more MAPE value respectively.

G. Spectf Dataset

For this dataset, accuracy of K-Medoids + MLP, K-Means + GRNN, K- Medoids + GRNN, ECM + GRNN, ECM_Imputation and PSO_COV are merely lagged by 0.80%, 0.76%, 0.37%, 0.50%, 0.36% and 0.49% respectively from ECM_PSO_COV. But other methods viz; K-Means + MLP, and all the methods proposed by Ravi and Krishna (2014), except GRAANN, are significantly outperformed by ECM_PSO_COV. However, GRAANN outperformed ECM_PSO_COV by 1.44% less MAPE value.

When we compared performance of existed methods with ECM_PSO_COV+ECM_AAELM then we observed that our proposed method outperformed all existed and proposed method presented in Table 4.2. (see Annexure) Except GRAANN, all the existing methods presented in the Table 4.2 (see Annexure) have at least 2% more MAPE value compared to ECM_PSO_COV+ECM_AAELM.

H. Turkish Dataset

In regards to the Turkish dataset, again a drastic reduction in MAPE value, from 27.90% to 19.28%, is observed when we employed ECM_PSO_COV instead of ECM_Imputation. Only one method, K-Medoids + GRNN yielded similar accuracy to our proposed method. Its accuracy is lagged by only 0.06%. Except GRAANN, our proposed method outperformed rest of the methods by significant amount. GRAANN performed better than ECM_PSO_COV.

When we employed ECM_PSO_COV+ECM_AAELM then MAPE value further reduced from 19.28% to 16.49% as compared to ECM_PSO_COV. It outperformed all existed and proposed method presented in the Table 4.2 (see Annexure). Except GRAANN, ECM+GRNN and K-Medoids+GRNN, all 9 methods presented in the Table 4.2 (see Annexure) have at least 9% more MAPE value compared to ECM_PSO_COV+ECM_AAELM.

I. UK Bankruptcy Dataset

Our proposed method performed worst for this dataset compared to the other 5 existing methods listed in the Table 4.2 (see Annexure). 5 methods viz., K-Medoids + MLP, K-Means + GRNN, K- Medoids + GRNN and ECM + GRNN outperformed ECM_PSO_COV by significant amount. Accuracy of K-Means + MLP is only 0.02% more than ECM_PSO_COV. Result of this dataset shows that optimal selection of *Dthr* will always improve the accuracy of imputation by ECM. A drastic reduction of MAPE value has been observed for this dataset when we employed ECM_PSO_COV instead of ECM_Imputation.

When we deployed our second proposed method, ECM_PSO_COV+ECM_AAELM, 4.09% of reduction in MAPE value has been observed compared to ECM_PSO_COV. But this method outperformed all other methods except GRAANN. It is just lagged by 0.04% of accuracy compared to GRAANN.

J. UK Credit Dataset

For this dataset also, three of the methods viz; K-Medoids + GRNN, ECM + GRNN and GRAANN outperformed ECM_PSO_COV by significant amount. Otherwise, ECM_PSO_COV performed well compared to other existing methods, which are presented in the Table 4.2 (see Annexure).

Our second proposed method, ECM_PSO_COV+ECM_AAELM outperformed ECM_PSO_COV and other 10 existing methods presented in the Table 4.2 (see Annexure). Only two methods, ECM + GRNN and GRAANN, outperformed ECM_PSO_COV+ECM_AAELM.

K. Wine Dataset

In regards to the Wine dataset, our proposed methods outperformed all the methods presented in the Table 4.2 (see Annexure). It can be observed that accuracy improved by significant amount compared to the other methods by employing ECM_PSO_COV based imputation. 2.85% of reduction in MAPE has been observed when we employed ECM_PSO_COV based imputation instead of ECM_Imputation.

When we deployed our second proposed method, ECM_PSO_COV+ECM_AAELM, 0.54% of reduction in MAPE value has been observed compared to ECM_PSO_COV. Since it outperformed ECM_PSO_COV, so, it is obvious that it outperformed all the existing and proposed methods presented in the Table 4.2 (see Annexure).

Wilcoxon two-tailed signed rank test is also performed at 1% level of significance to test the statistical significance of the results. We performed Wilcoxon test with K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K-Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013) and GRAANN (Ravi and Krishna, 2014). Table 4.3 (see Annexure) and Table 4.4 (see Annexure) represent the results of Wilcoxon signed rank test for ECM_PSO_COV and ECM_PSO_COV+ECM-AAELM respectively. The critical value from the table (www.sussex.ac.uk/Users/grahamh/RM1web/WilcoxonTable2005.pdf) for N=10 is 3 at 1% level of significance. According to the Wilcoxon signed rank test, if the computed value is less than or equal to the critical value, then it is statistically significant. This test assured that the result obtained by our proposed method is statistically significant. We didn't perform Wilcoxon test on PSOAANN, PSOAAWNN and RBFAANN because differences of MAPE values between proposed methods and these methods are very large.

4.5 Conclusion

Two novel methods have been proposed in this chapter and effectiveness of these methods are tested by experimentation on 12 datasets. Our conducted experiment evinced the following facts:

(i) ECM_PSO_COV and ECM_PSO_COV+ECM-AAELM always aided to improve the accuracy of imputation by selecting optimal *Dthr* value for ECM imputation and ECM-AAELM respectively.

(ii) Our proposed methods, ECM_PSO_COV performed better for 6 out of 12 datasets and ECM_PSO_COV+ECM-AAELM didn't perform well for only 3 out of 12 datasets in comparison to the other existing hybrid methods presented in the Table 4.2 (see Annexure).

(iii) ECM_PSO_COV outperformed three methods PSO_COV (Krishna and Ravi, 2013), RBFAANN (Ravi and Krishna, 2014) and ECM_Imputation for all 12 datasets. However, ECM_PSO_COV+ECM-AAELM outperformed 9 methods for all datasets.

(iv) One of the methods proposed by us, ECM_PSO_COV preserved covariance structure of the original data like PSO_COV however, other existing methods, except PSO_COV, did not necessarily preserve the covariance structure of the original data. Indeed, ECM_PSO_COV outperformed PSO_COV by a large margin in majority of the dataset.

(v) ECM_PSO_COV+ECM-AAELM outperformed our another proposed method ECM_PSO_COV for all datasets and 9 other existing methods which is presented in the Table 4.2 (see Annexure).

(vi) ECM_PSO_COV exhibited better imputation capability due to strong local learning capability and appropriate selection of *Dthr* value.

(vii) ECM_PSO_COV+ECM-AAELM exhibits the fact that how a hybrid of local learning and global approximation assisted our proposed method to obtain better accuracy in imputation.

(viii) It resolved the issue raised in previous chapter that "user intervention was required to select optimized *Dthr* value for ECM_Imputation algorithm". No more user intervention is required to select optimized D*thr* for ECM_Imputation in our proposed methods.

Based on above remarks, we can conclude that the proposed approach can be used as viable alternative for the data imputation. There is one drawback with our proposed method “It has long runtime during selection of optimal *Dthr* value”. Future works would be concentrated on reducing the runtime of the proposed methods.

CHAPTER5

Data Imputation Based On Counter-Propagation Neural Network

Various researchers have applied autoassociative neural network to resolve data imputation problem. In this chapter, we also hybridized the concept of autoassociativity with Counter-propagation neural network (CPNN) (Hecht-Nielsen, 1987) and proposed two novel methods. We proposed two novel hybrid methods for data imputation task using Counter-propagation neural network, Gray System Theory (Deng, 1982) and Autoassociative neural network. Novelty of our proposed methods is in this term also, that, no one applied CPNN before to resolve missing data problem. Performance of our proposed methods has been tested on 12 different datasets and the results of both the proposed methods are compared with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K-Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013) and PSOAANN, PSOAAWNN, RBFAANN & GRAANN (Ravi and Krishna, 2014). Statistical testing has been also performed to ensure the reliability of our obtained results from our proposed methods.

5.1 Overview of the employed techniques

In the proposed method, we used Counter-propagation neural network (CPNN), Autoassociative neural network and Gray system theory (GST). Whereas, we already discussed Gray system theory in earlier chapter, so please refer to Chapter 3 for GST. Therefore, we will discuss here only CPNN algorithm.

5.1.1 Overview of Counter-propagation neural network

Counter-propagation neural network (Hecht-Nielsen, 1987) is a combination of unsupervised and supervised learning i.e. semi supervised learning as shown in Fig. 5.1 (see Annexure). Unsupervised layer contains Self organization map (SOM) (Kohonen, 1988) and supervised layer contains Grossberg Outstar layer. Hecht-Nielsen proposed it in 1987. Two variation of CPNN exists:

1. Forward only CPNN
2. Full CPNN (Bidirectional CPNN)

CPNN uses a special learning technique: competitive learning, which is also called as winners-takes-it-all. Competitive learning works on the concept that if a node wins during training for one pattern of input then it will always respond as a winner for a similar pattern of input. CPNN is also similar to feed forward network in the sense that each input node is connected from each node in hidden layer and each node of hidden layer is connected from each node of output node. But it is different in terms of connection of hidden layer nodes. Inhibitory inter-connections among hidden layer nodes are used to conduct a competition for winners when training patterns are presented at the input layer (Mehrotra, 1996). Inhibitory inter-connections among hidden layer nodes are clearly depicted in Fig. 5.2 (see Annexure). Flow diagram of CPNN is depicted in Fig. 5.3 (see Annexure). This neural network has been successfully deployed for various purposes like digital image copyright authentication (Chang et al., 2010), data compression, approximation, classification tasks etc. It is heavily used by chemometric community in last decades. It is the first time, when we are employing CPNN for imputation task.

Various toolboxes have been developed to apply CPNN in various fields. Kuzmanovski and Novič (2008) developed a CPNN toolbox to handle CPNN using MATLAB (MATLAB version 7.10.0, 2010). They integrate the freely available toolbox of SOM (Vesanto et al., 2000; Vesanto, 1999) with their code for developing a CPNN toolbox. It can be downloaded from internet for free (http://www.cis.hut.fi/projects/somtoolbox/). Ballabio et al. (2009) developed "The Kohonen and CP-ANN toolbox" for handling both SOM and CPNN in 2009. This toolbox is available with interactive graphical user interface. It also added the module of Genetic algorithm for selection of optimized network setting. It is feely available on internet (http://www.disat.unimib.it/chm) with extensive description of toolbox with various examples. We opted the CPNN toolbox developed by Kuzmanovski and Novič (2008). We offer our sincere thanks to I. Kuzmanovski for sending the code of CPNN toolbox with examples.

5.2 Architecture of the proposed methods

Two novel imputation techniques have been proposed based on various existing methods viz., counter-propagation neural network, autoassociative neural network and gray system theory (GST).

5.2.1 Algorithm of the Proposed Method: Counter-propagation Auto-associative Neural Networks (CPAANN):

Earlier, Ravi and Krishna (2014) also employed the concept of auto-associativity with General regression neural network and deployed General regression auto-associative neural network (CPAANN) to resolve missing data problem. We also deployed the concept of Auto-associativity in similar fashion for CPNN and proposed a new technique to resolve missing data problem. Novelty of this technique is also in the term that no one applied CPNN before to resolve the missing data problem. Fig. 5.4 (see Annexure) depicts the architecture CPAANN. Auto-associative Neural Networks is a feedforward network, where input and output both are identical. In similar way, input and output are identical in CPAANN. By comparing Fig. 5.4 (see Annexure) from Fig. 5.2 (see Annexure), you will easily observe that number of output nodes in both diagrams is different. In Fig. 5.4 (see Annexure), number of output nodes and input nodes

are same as well as identical but it is not necessary in traditional CPNN (as shown in Fig. 5.2, see Annexure). Fig. 5.5 presents flow diagram of our proposed method.

Algorithm:

1. Divide the dataset in two parts, complete and incomplete.
2. Perform training on complete records only.
3. Weights of both the layers adjusted during training by using following function (Kuzmanovski and Novič, 2008):

$$w_{j,i}^{new} = w_{j,i}^{old} + \eta(t).a(d_j - d_c).(x_i - w_{j,i}^{old}) \quad (1)$$

$$u_{j,i}^{new} = u_{j,i}^{old} + \eta(t).a(d_j - d_c).(x_i - u_{j,i}^{old}) \quad (2)$$

Where,

x_i = Input variables.

w_j = neurons in the Kohonen layer.

$(d_j - d_c)$= Topological distance between the winning neuron c and the neuron j which weights are adjusted.

$w_{j,i}^{old}$ = Weights before adjustment in Kohonen layer.

$w_{j,i}^{new}$ = Weights after adjustment in Kohonen layer.

$u_{j,i}^{old}$ = Weights before adjustment in output layer.

$u_{j,i}^{new}$ = Weights after adjustment in output layer.

$\eta(t)$ = Learning rate

4. Selected parameters in CPNN toolbox are mentioned in Table 5.1 (see Annexure).
5. Perform Mean Imputation on Incomplete dataset.
6. Pass incomplete dataset after mean imputation through CPAANN with adjusted weight during training.

5.2.1 Algorithm of the Proposed Method: Gray+Counter-propagation Auto-associative Neural Networks (Gray+CPAANN):

In CPAANN, we performed Mean imputation before applying the incomplete records to the neural network and in Gray+CPAANN; we performed Gray distance based imputation instead of Mean imputation before presenting the incomplete records to CPAANN. Flow chart of this method is given in Fig. 5.6 (see Annexure).

1. Normalize the dataset in the range of [0 1].
2. Divide the dataset in two parts: complete and incomplete records.
3. Impute missing records using Gray distance based nearest neighbour imputation.
4. Training procedure is same as previous proposed method CPAANN.
5. Pass Gray distance based imputed records to CPAANN.

5.3 Experimental Design

Datasets are divided into two parts: one is set of complete records and another is set of incomplete records. Complete records have been used for training process and incomplete records have been used for testing process. We performed 10-fold cross validation in our experiment. Number of incomplete records are 10 % of the total records. We compared the average MAPE values of the proposed methods with those of K-Means+ MLP imputation (Ankaiah and Ravi, 2011), K-Medoids+MLP, K-Means+GRNN, K- Medoids+GRNN, ECM+GRNN (Nishanth and Ravi, 2013) and PSO_Covariance imputation (PSO_COV) (Krishna and Ravi, 2013), PSOAANN, PSOAAWNN, RBFAANN and GRAANN (Ravi and Krishna, 2014). Further, we performed statistical testing to verify the significance of our obtained results from the proposed methods.

5.4 Results and Discussions

Our both proposed methods have been applied on 12 datasets and also compared our results from the results of various existing methods presented in the Table 5.2 (see Annexure). Stage I denotes the Gray_Imputation in the last column of the Table 5.2 (see Annexure). An extensive discussion on comparison of our proposed methods from various methods is mentioned below:

In case of Auto Mpg dataset, CPAANN outperformed 7 out of 10 methods and accuracy of CPAANN is lagged by 1.66%, 1.32%, 2.78%, 0.29% from K-Medoids+GRNN, ECM+GRNN, GRAANN respectively. However, when we employed Gray distance based nearest neighbour imputation (Gray_Imputation) on Stage I instead of Mean Imputation then our new proposed method Gray+CPAANN outperformed all 11 existing methods in the Table 5.2 (see Annexure). So, MAPE value is reduced from 18.32% to 15.31% when we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN.

In regards of Body fat dataset, our one of the proposed method, CPAANN outperformed all the existing methods presented in the Table 5.2 (see Annexure) except GRAANN. It is lagged by 0.64% from GRAANN. However, our second method, Gray+CPAANN performed better than all existed methods in the Table 5.2 (see Annexure) including our one of the proposed method CPAANN. MAPE value is reduced from 18.32% to 15.31% when we employed Gray+CPAANN instead of CPAANN.

For Boston Housing dataset, our both methods outperformed all the existing methods presented in the Table 5.2 (see Annexure). Except GRAANN, accuracy of all the methods is lagged by at least 2.5% from both proposed methods. Here, CPAANN alone performed better than Gray+CPAANN. However, Gray+CPAANN is lagged by only 0.15% from CPAANN.

In case of Forest fire dataset, performance of our both the proposed methods are similar to Boston Housing dataset. They also outperformed all the existing methods presented in the Table 5.2 (see Annexure) and CPAANN alone performed better than Gray+CPAANN. However, Gray+CPAANN is lagged by only 0.94% from CPAANN. Except GRAANN, accuracy of all methods is lagged by at least 4% from both proposed methods.

In case of Iris dataset, CPAANN outperformed 8 out of 10 methods for this dataset and accuracy of CPAANN is lagged by 0.21%, 0.76%, 1.24% from ECM+GRNN, GRAANN and respectively. While, our second proposed method, Gray+CPAANN outperformed all the presented methods in the Table 5.2 (see Annexure). MAPE value is reduced from 6.51% to 4.03% when we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN.

For Prima Indian dataset, performance of our both the proposed methods is similar to Boston Housing and Forest fire dataset. They also outperformed all the existing methods in the Table 5.2 (see Annexure). However, CPAANN alone performed better than Gray+CPAANN but Gray+CPAANN is lagged by only 1.13% from CPAANN. Except PSOAANN, accuracy of all methods is lagged by at least 5% and 4% from CPAANN and Gray+CPAANN respectively. Accuracy PSOAANN is lagged by 3.51% and 2.38% from CPAANN and Gray+CPAANN respectively.

In case of Spanish dataset, both the proposed methods outperformed all the methods presented in the Table 5.2 (see Annexure) by significant amount. Except K-Medoids+GRNN and GRAANN, accuracy of all the methods is lagged by at least 14% from both proposed methods. Even, accuracy of K-Medoids+GRNN and GRAANN is lagged by significant amount 8.88% and 6.15% respectively from CPAANN and 11.80% and 9.07% respectively from Gray+CPAANN. When we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN, the MAPE value is further reduced from 17.13% to 14.21%.

For Spectf dataset, both the proposed methods outperformed all the existing methods presented in the Table 5.2 (see Annexure) except GRAANN. Accuracy of CPAANN and Gray+CPAANN is lagged by 0.20% and 0.12% respectively from GRAANN. When we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN the MAPE value is reduced from 8.61% to 8.53%.

In regards of Turkish dataset, CPAANN outperformed all the existing methods presented in the Table 5.2 (see Annexure) by significant amount. Except K-Medoids+GRNN, ECM+GRNN and GRAANN, accuracy of all the methods is lagged by at least 10.83% from CPAANN. Even, accuracy of K-Medoids+GRNN, ECM+GRNN and GRAANN is lagged by significant amount 3.27%, 6.27% and 1.18% respectively from CPAANN. However, our second proposed method Gray+CPAANN outperformed all the methods except GRAANN. Accuracy of Gray+CPAANN is lagged by 0.12% from GRAANN. Except K-Medoids+GRNN, ECM+GRNN and GRAANN, accuracy of all the methods is lagged by at least 8.53% from Gray+CPAANN. Even, accuracy of K-Medoids+GRNN and ECM+GRNN is lagged by significant amount 1.97% and 4.97%

respectively from Gray+CPAANN. CPAANN alone performed better than Gray+CPAANN but Gray+CPAANN is lagged by only 1.30% from CPAANN.

For UK Bankruptcy dataset, performance of our both the proposed methods are similar to Spanish dataset. Both the proposed methods outperformed all the existing methods presented in the Table 5.2 (see Annexure) by significant amount. Except GRAANN, accuracy of all the methods lagged by at least 6.43% and 7.81% from CPAANN and Gray+CPAANN respectively. Even accuracy of GRAANN is lagged by 4.89% and 6.27% from CPAANN and Gray+CPAANN respectively. When we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN then MAPE value is reduced from 21.96% to 20.58%.

For UK Credit dataset, CPAANN outperformed 8 out of 10 methods for this dataset and accuracy of CPAANN is lagged by 0.95% and 2.41% from ECM+GRNN and GRAANN respectively. While, our second proposed method, Gray+CPAANN outperformed all the presented methods in the Table 5.2 (see Annexure). A drastic reduction in MAPE value is observed, from 6.51% to 4.03%, when we employed Gray_Imputation on Stage I instead of Mean Imputation with CPAANN. Except GRAANN and ECM+GRNN, accuracy of all the methods lagged by at least 10.34% from Gray+CPAANN. Even accuracy of ECM+GRNN and GRAANN is lagged by 8.23% and 6.77% from Gray+CPAANN respectively.

For Wine dataset, performance of our both the proposed methods are similar to Boston Housing, Forest fire and Prima Indian dataset. They also outperformed all the existing methods presented in the Table 5.2 (see Annexure) and CPAANN alone performed better than Gray+CPAANN. However, Gray+CPAANN is lagged by only 0.17% from CPAANN. Except GRAANN, accuracy of all the methods lagged by at least 3.19% and 3.03% from CPAANN and Gray+CPAANN respectively.

We also performed the Wilcoxon two-tailed signed rank test at 1% level of significance to test the statistical significance of the results. The Wilcoxon test values for all the proposed methods are presented in the Table 5.3 and 5.4 (see Annexure). Wilcoxon test is not performed with PSOAANN, PSOAAWNN and RBFAANN as the proposed imputation techniques outperformed it by a large interval. The critical value from the table for N=10 is 3 at 1% level of significance.

According to the Wilcoxon signed rank test, the obtained value is statistically significant if it is equal or smaller than the critical value from the table (www.sussex.ac.uk/Users/grahamh/RM1web/WilcoxonTable2005.pdf). Therefore, by observing the obtained values of the proposed and existed methods, we conclude that the obtained values from the proposed methods are statistically significant for all the datasets and with all the existing methods in the Table 5.2 (see Annexure).

5.5 Conclusion

Our chapter presented two novel methods for data imputation. If you will closely observe the Table 5.2 (see Annexure) then you will find out following points:

i. CPAANN outperformed 7 out of 10 methods for all the datasets and Gray+CPAANN outperformed 9 out of 10 methods for all the datasets.

ii. Gray+CPAANN performed better compared to CPAANN and all other existing methods listed in the Table 5.2 (see Annexure) for most of the datasets. It always didn't perform better than CPAANN but where it improved accuracy of CPAANN by Gray_Imputation on Stage I, there it improved accuracy by significant amount but where it lagged by CPAANN there it is lagged by less amount in most of the datasets.

iii. Gray+CPAANN always yielded better accuracy compared to Gray_Imputation techniques alone. It shows that global approximation by CPAANN always helped to achieve better accuracy for imputation task.

So, our proposed method can be considered as a viable alternative to handle the missing value problem. Since, User intervention is required to select the parameter for both the proposed methods. So, direction of our future work will be concentrated on automatic and optimized selection of parameters.

CHAPTER 6

OVERALL CONCLUSIONS

In the first part of the study, we proposed several data imputation techniques and compared there results with the existing methods. We observed that our proposed methods outperformed existing methods by significant amount for most of the datasets. We proposed eight novel methods for data imputation based on the ELM, ECM, GST, PSO, Covariance matrix, PCA, AANN and CPNN. We resolved the randomness issue of ELM and provided a deterministic ELM, which performed better for our imputation and it can be applied for other problems in various field. We removed the requirement of user intervention for selection of *Dthr* value. Our proposed techniques can determine by themselves that - which value is optimal for clustering algorithm in our imputation task. So, our proposed method can be considered as a viable alternative to handle the missing value problem in dataset.

References

M. Abdella, T. Marwala, The use of Genetic Algorithms and Neural Networks to approximate missing data in database, Computational Cybernetics, ICCC 2005, IEEE 3rd International Conference (2005) 207-212.

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

C. Ji, A. Elwalid, Measurement-based network monitoring: missing data formulation and scalability analysis, IEEE International Symposium on Information Theory, Sorrento, Italy (2000) p78.

L. N. Nguyen, W.T. Scherer, Imputation techniques to account for missing data in support of intelligent transport system applications, Tech. Rep, University of Virginia, USA (2003).

K. Lakshminarayana, S. A. Harp, T. Samad, Imputation of missing data in industrial databases, Engineering Applications of Artificial Intelligence 11 (3) (2004) 259-275.

M. Halatchev, L. Gruenwald, Estimating missing values in related sensor data streams, International Conference on Management of Data 11 (2005) 83-94.

H. S. Mohammad, N. Stepenosky, R. Polikar, An ensemble technique to handle missing data from sensors, IEEE Sensor Applications Symposium, Houston, Texas, (2003) 101-105.

M. Cooke, P. Green, M. Crawford, Handling Missing data in speech recognition, International Conference on Spoken Language Process (1994) 1555-1558.

O. Troyanskaya, M. Cantor, O. Alter, G. Sherlock, P. Brown, D. Botstein, R. Tibshirani, T. Hastie, R. Altman, Missing value estimation methods for DNA microarrays, Bioinformatics 17 (6) (2001) 520-525.

P.L. Roth, F.S. Switzer, D.M. Switzer, Missing data in multiple item scales: a Monte Carlo analysis of missing data techniques, Organizational research methods 2 (3) (1999) 211-232.

Q. Song, M. Shepperd, A new imputation method for small software project data sets, Journal of Systems and Software 80 (1) (2007) 51-62.

R. J. A. Little, D. B. Rubin, Statistical analysis with missing data, second edition, Wiley, New York (2002).

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

W. S. Desabro, P. E. Green, J. D. Carroll, Missing data in product-concept testing, Decision Sciences 17 (1986) 163-185.

N. M. Laird, Missing data in longitudinal studies, Statistics in Medicine 7 (1-2) (1988) 305-315.

J. Jerez, I. Molina, J. Subirates, L. Franco, Missing data imputation in breast cancer prognosis, BioMed'06 Proceedings of the 24th IASTED International Conference on Biomedical Engineering (2006).

G. Batista, M.C. Monard, A study of K-nearest neighbor as an imputation method, Abraham A et al (eds) Hybrid Intelligent Systems, Ser Front Artificial Intelligence Applications 87 (2002) 251–260.

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

T. Samad and S. A. Harp, Self-organization with partial data, Network: Computation in Neural Systems 3 (1992) 205-212.

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

S. Nordbotten, Neural network imputation applied to the Norwegian 1990 population census data, Journal of Official Statistics 12 (4) (1996) 385–401.

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

S.Y. Yoon, S.Y. Lee, Training algorithm with incomplete data for feed-forward neural networks, Neural Processing Letters 10 (3) (1999) 171–179.

A. Ragel, B. Cremilleux, MVC—a preprocessing method to deal with missing values, Knowledge Based Systems 12 (1999) 285-291.

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

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

J. Chen, H. Huang, F. Tian, S. Tian, A selective Bayes Classifier for classifying incomplete data based on gain ratio, Knowledge Based Systems 21(2008) 530-534.

I.A. Gheyas, L.S. Smith, A neural network-based framework for the reconstruction of incomplete data sets, Neurocomputing 73(16-18) (2010) 3039-3065.

S. Zhang, Z. Jin, X. Zhu, Missing data imputation by utilizing information within incomplete instances, Journal of Systems and Software 84(3) (2011) 452-459.

J.C. Figueroa García, D. Kalenatic, C.A.L. Bello, Missing data imputation in multivariate data by evolutionary algorithms, Computers in Human Behavior 27(5) (2011) 1468-1474.

A.G. Di Nuovo, Missing data analysis with fuzzy C-Means: A study of its application in a psychological scenario, Expert Systems with Applications 38(6) (2011) 6793-6797.

N. Ankaiah, V. Ravi, A Novel Soft Computing Hybrid for Data Imputation, DMIN, Las Vegas, USA (2011).

Y.Y Li, L.E. Parker, Nearest neighbor imputation using spatial–temporal correlations in wireless sensor networks, Information Fusion, In Press (2012).

S. Zhang, Nearest neighbor selection for iteratively kNN imputation, Journal of Systems and Software 85(11) (2012) 2541-2552.

K. J. Nishanth, V. Ravi, N. Ankaiah, I. Bose, Soft computing based imputation and hybrid data and text mining: The case of predicting the severity of phishing alerts, Expert Systems with Applications 39(12) (2012) 10583-10589.

F.V. Nelwamondo, D. Golding, T. Marwala, A dynamic programming approach to missing data estimation using neural networks, Information Sciences 237 (2013) 49-58.

H. Tan, G. Feng, J. Feng, W. Wang, Yu-Jin Zhang, F. Li, A tensor-based method for missing traffic data completion, Transportation Research Part C: Emerging Technologies 28 (2013) 15-27.

F.O. de França, G.P. Coelho, F.J. Von Zuben, Predicting missing values with biclustering: A coherence-based approach, Pattern Recognition 46(5) (2013) 1255-1266.

I.B. Aydilek, A. Arslan, A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm, Information Sciences 233 (2013) 25-35.

K. J. Nishanth, V. Ravi, "A Computational Intelligence Based Online Data Imputation Method: An Application For Banking", Journal of Information Processing Systems (2013), vol. 9 (4), pp. 633-650.

V. Ravi, M. Krishna, "A new online data imputation method based on general regression auto associative neural network", Neurocomputing (2014), Elsevier, vol. 138, pp. 207-212.

G.B. Huang, Q. Zhu, C.Siew, Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks, International Joint Conference on Neural Networks (2004), Vol. 2, pp. 985-990.

G.B. Huang, Q. Zhu, C. Siew, Extreme Learning Machine: Theory and Applications, Neurocomputing (2006), Vol. 70, pp. 489-501.

K. Pearson,"On Lines and Planes of Closest Fit to Systems of Points in Space", Philosophical Magazine (1901), 2 (11): 559–572.

H. Hotelling, Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology (1933), 24, 417-441, and 498-520.

Q. Song and N. Kasasbov, "Dynamic Evolving Neural-Fuzzy Inference System (DENFIS): On-line Learning and Application for Time-series Prediction", Proc. 6th International Conference on Soft Computing, Iizuka, Fukuoka, Japan (2000), 696 – 701.

Q. Song and N. Kasasbov, "ECM — A Novel On-line, Evolving Clustering Method and Its Applications", Proceedings of the Fifth Biannual Conference on Artificial Neural Networks and Expert Systems, Berlin (2001), pp. 87-92.

J.L. Deng, Control problems of grey system, System and Control Letters (1982), 1,288–294.

M. Krishna, V. Ravi, "Particle swarm optimization and covariance matrix based data imputation", IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (2013), Enathi, pp. 1-6.

M. Ballings, D. Van den Poel: Kernel Factory: An ensemble of kernel machines.Expert Systems with. Applications (2013), 40(8): 2904-2913.

G.B. Huang, L. Chen, C.K. Siew, Universal Approximation using Incremental Feedforward Networks with Arbitrary Input Weights, in Technical Report ICIS/46/2003, (School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore), 2003.

P.L. Bartlett, The sample complexity of pattern classification with Neural Networks: The size of the weights is more important than the size of the network, IEEE Transactions on Information Theory (1998), Vol. 44, No.2, pp:525-536.

Shichao Zhang, "Nearest neighbor selection for iteratively kNN imputation", The Journal of Systems and Software (2012), vol. 85(11), pp. 2541-2552.

J. Tian, B. Yu, D. Yu, Shilong Ma, "Missing data analyses: a hybrid multiple imputation algorithm using Gray System Theory and entropy based on clustering", Applied Intelligence (2013), pp. 1-13.

X. Glorot, A. Bordes, Y. Bengio, "Deep Sparse Rectifier Neural Networks", International Conference on Artificial Intelligence and Statistics (2011), Fort Lauderdale, USA, vol. 15, pp. 315-323.

C. Dugas, Y. Bengio, F. Belisle, C. Nadeau, R. Garcia, " Incorporating second-order functional knowledge for better option pricing", Advances in Neural Information Processing System, MIT Press (2001), pp. 472-478.

G. S. S. Gomes, T. B. Ludermir, L. M. M. R. Lima, "Comparison of new activation functions in neural network for forecasting financial time series", Neural Computing and Applications, Springer (2011), vol. 20, Issue 3, pp. 417-439.

B. Karlik, A. V. Olgac, "Performance Analysis of Various Activation Functions in Generalized MLP Architectures of Neural Networks", International Journal of Artificial Intelligence and Expert Systems (2010), vol. 1, Issue 4, pp. 111-122.

B. E. Flores, "A pragmatic view of accuracy measurement in forecasting", Omega (1986), vol. 14(2), pp. 93-98.

R. Lowry, Concepts & Applications of Inferential Statistics.

F. Wilcoxon, Individual comparisons by ranking methods, 1 (1945) 80-83.

Sidney Siegel, Non-parametric statistics for the behavioral sciences, New York: McGraw-Hill (1956) 75-83.

www.sussex.ac.uk/Users/grahamh/RM1web/WilcoxonTable2005.pdf.

J. Kennedy, R. C. Eberhart, Particle swarm optimization, Proceeding of IEEE International Conference on Neural Networks, Piscataway, NJ, USA (1995) 1942–1948.

R.C. Eberhart, Y. Shi, " Comparing inertia weights and constriction factors in particle swarm optimization", Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla (2000), CA , pp. 84-88.

R. Hecht-Nielsen, "Counterpropagation networks", Appl Opt (1987), vol. 26, pp. 4979-4984.

T. Kohonen, Self-Organization and Associative Memory, Springer Verlag, Berlin, 1988.

K. Mehrotra, C.K. Mohan, and S. Ranka, "Elements of Artificial Neural Networks", MIT Press, Cambridge, MA, USA, 1996.

C.Y. Chang, H.J. Wang, and S,J. Su, "Copyright authentication for images with a full counter-propagation neural network", Expert Syst. Appl. (2010), vol. 37 (12), pp. 7639-7647.

I. Kuzmanovski, M. Novic, "Counter-propagation neural networks in MATLAB", Chemometrics and Intelligent Laboratory Systems (2008), vol. 90, pp. 84–91.

MATLAB version 7.10.0. Natick, Massachusetts: The MathWorks Inc., 2010.

J. Vesanto, J. Himberg, E. Alhoniemi, J. Parhankangas, "SOM Toolbox for Matlab 5", Technical Report A57, Helsinki University of Technology, 2000.

http://www.cis.hut.fi/projects/somtoolbox/.

J. Vesanto, "SOM-based data visualization methods", Intell. Data Anal., vol. 3, pp. 111–126, 1999.

The Kohonen and CP-ANN toolbox: A collection of MATLAB modules for Self

D. Ballabio, V. Consonni, R. Todeschini, "Organizing Maps and Counterpropagation Artificial Neural Networks", Chemometrics and Intelligent Laboratory Systems (2009), vol. 98, pp. 115–122.

http://www.disat.unimib.it/chm.

APPENDIX A: DESCRIPTION OF DATASETS

Boston Housing dataset

The Boston housing dataset is taken from Statlib library which is maintained at Carnegie Mellon University. The dataset describes the housing values in the suburbs of Boston. The dataset contains 506 records and 13 attributes. The description of dataset is presented in the Table A.1.The dataset is obtained from http://archive.ics.uci.edu/ml/machine-learning-databases/housing/housing.data.

Forest Fires Dataset

The forest fires dataset is taken from Cortez and Morais (2007). The description of forest fires dataset is presented in the Table A.2. The dataset is obtained fromhttp://archive.ics.uci.edu/ml/machine-learning-databases/forest-fires/forestfires.csv.

Auto MPG

The Auto MPG dataset is taken from Statlib library which is maintained at Carnegie Mellon University. The dataset is used in the 1993 American Statistical Association Exposition. The dataset concerns city-cycle fuel consumption in miles per gallon, to be produced in terms of 3 multivalued discrete and 5 continuous attributes. The description of Auto MPG dataset is presented in the Table A.3.The dataset is obtained fromhttp://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data.

Body fat dataset

The dataset lists the estimates of percentage of body fat determined by underwater weighing and various body circumference measurements for 252 men. The description of dataset is presented in table A.4. The dataset is obtained fromhttp://lib.stat.cmu.edu/datasets/bodyfat.

Wine Dataset

The wine recognition dataset contains the results of chemical analysis of wines grown in the same region of Italy but derived from three different cultivars. The analysis determines the quantities of 13 constituents found in each of the three types of wines. The numbers of instances for the three classes of wine are 59, 71 and 48 respectively. The dataset contains 13 attributes and 178 records. The attributes of wine dataset are presented in the Table A.5. The dataset is obtained from http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data.

Pima Indians Dataset

Pima Indians dataset was taken from National Institute of Diabetes and Digestive and Kidney Diseases in the year 1990. It was available at http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabete. It contains of 768 patterns of the patients in which 268 are tested positive for diabetes and 500 of the patients who are tested negative. It contains 8 features and one target class variable. Table A.6 describes the features of pima Indians dataset.

Iris dataset

The Iris plants dataset is the best known database found in pattern recognition literature. The dataset contains three classes of 50 instances each, where each class refers to a type of iris plant. The dataset contains 4 numeric attributes and a class attribute. The attribute information of iris dataset is presented in Table A.7. The dataset is obtained from http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data.

Spectf Dataset

Spectf dataset contains data on cardiac Single Proton Emission Computed Tomography (SPECT) images. Here each patient classified into two categories: normal and abnormal. .Kurgan and Cios are the donors of this dataset. It contains 267 SPECT image sets (patients) which were processed to extract features that summarize the original SPECT images. As a result, 44 continuous feature patterns were created for each patient. Table A.8 contains the features of *Spectf* dataset. The dataset is available at http://archive.ics.uci.edu/ml/machine-learning-databases/spect.

UK Credit dataset

UK credit dataset (Thomas et.al, 2002) consists of 1225 patterns of the customers applied for credit product. It contains 14 financial ratios regarding the applicants and in those we removed the 3 features: phon, aes, res which corresponds to "presence of landline or not", "applicant's employment status", "residential status" of the applicant. They are removed as some are very irrelevant and other has many categorical values. In 12225 patterns 323 are of bad customers i.e., customers with very less credit and 902 are of good customers. The financial ratios of UK Credit dataset are tabulated in table A.9.

Spanish banks dataset

The "Spanish banks" data is obtained from (Olmeda and Fernandez 1997). "Spanish banks" dataset contains the list of banks which were bankrupt and non-bankrupt, so the target variable class contains two classes: bankrupt and non-bankrupt. Spanish banking industry suffered the worst crisis during 1977-85 resulting in a total cost of 12 billion dollars. The ratios used for the failed banks were taken from the last financial statements before the bankruptcy was declared and the data of non-failed banks was taken from 1982 statements. This dataset contains 66 banks where 37 went bankrupt and 29 healthy banks. It contains 9 financial ratios and a target class variable. Table A.10 describes the financial ratios of Spanish bank dataset.

Turkish banks data set:

"Turkish banks" dataset is a bankruptcy prediction dataset where it contains patterns of several banks in which some are bankrupt and some other are non-bankrupt. It was obtained from (Canbas & Kilic 2005), which is available at (http://www.tbb.org.tr/english/bulten/yillik/2000/ratios.xls). Canbas & Kilic (2005) chose only 12 ratios as the early warning indicators that have the discriminating ability (i.e. significant level is <5%) for healthy and failed banks one year in advance. Among these variables, 12th variable has some missing values meaning that the data for some of the banks are not given. So, we filled those missing values with the mean value of the variable, is a general approach in data mining. This dataset contains 40 banks where 22 banks went bankrupt and 18 banks are healthy. Table A.10 describes the financial ratios of Turkish bank dataset.

UK Bankruptcy Dataset

The "UK bankruptcy" dataset is taken from Beynon and Peel (2001). This dataset contains 60 patterns among which 30 are healthy and 30 bankrupt. Each pattern corresponds to each bank. The dataset contains 10 financial ratios and 1 target class variable. Table A.10 contains the financial ratios of UK Bankruptcy dataset.

APPENDIX B: FIGURES

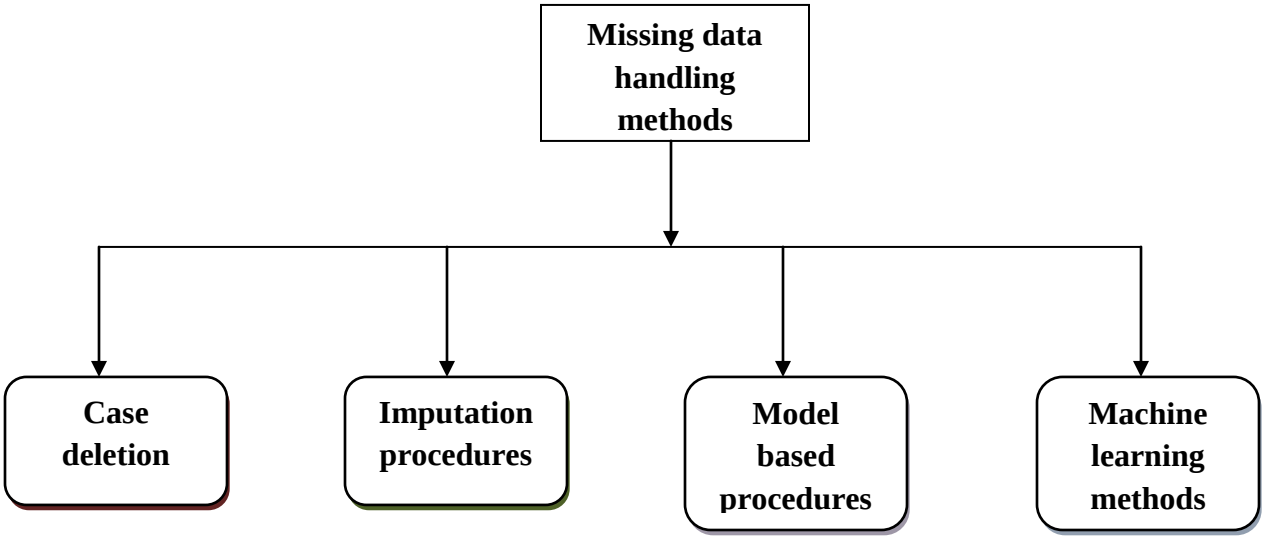

Fig 2.1: Missing data handling methods for numerical data

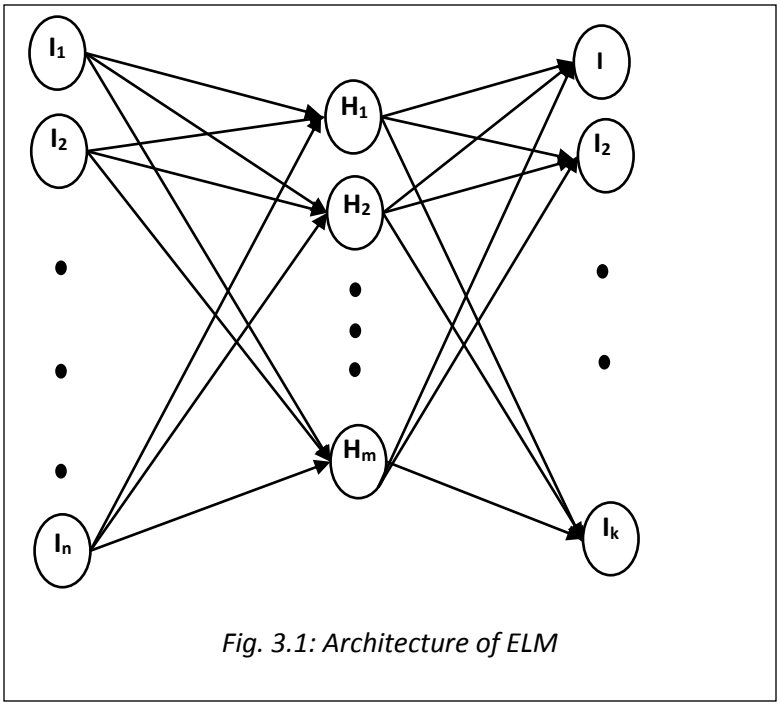

Fig. 3.1: Architecture of ELM

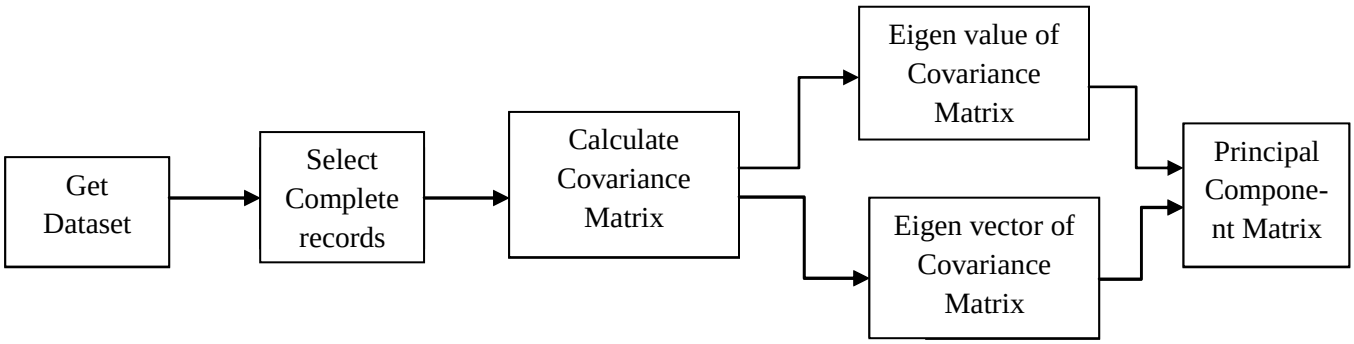

Fig. 3.2: Principal Component Analysis

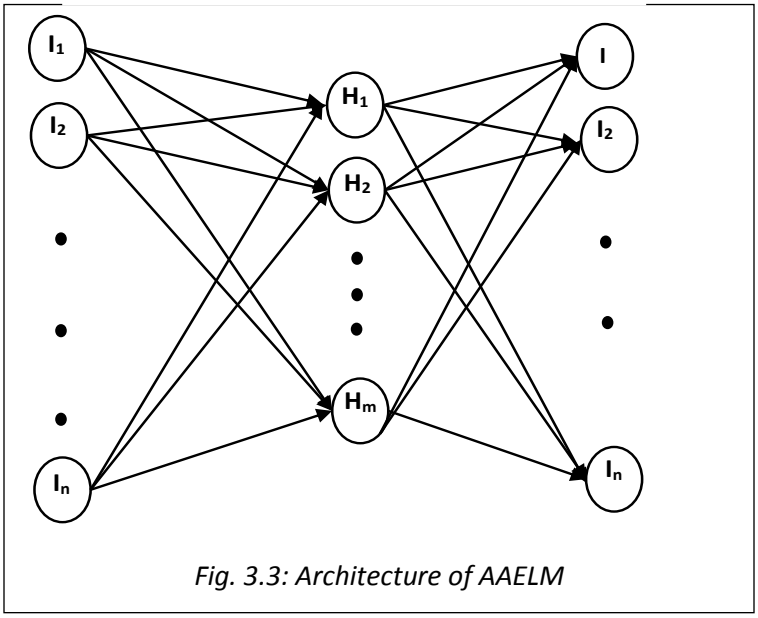

Fig. 3.3: Architecture of AAELM

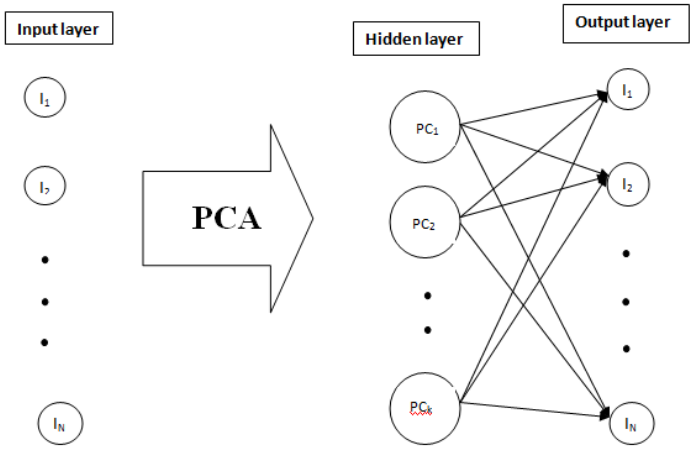

Fig. 3.4: Architecture of PCA-AAELM

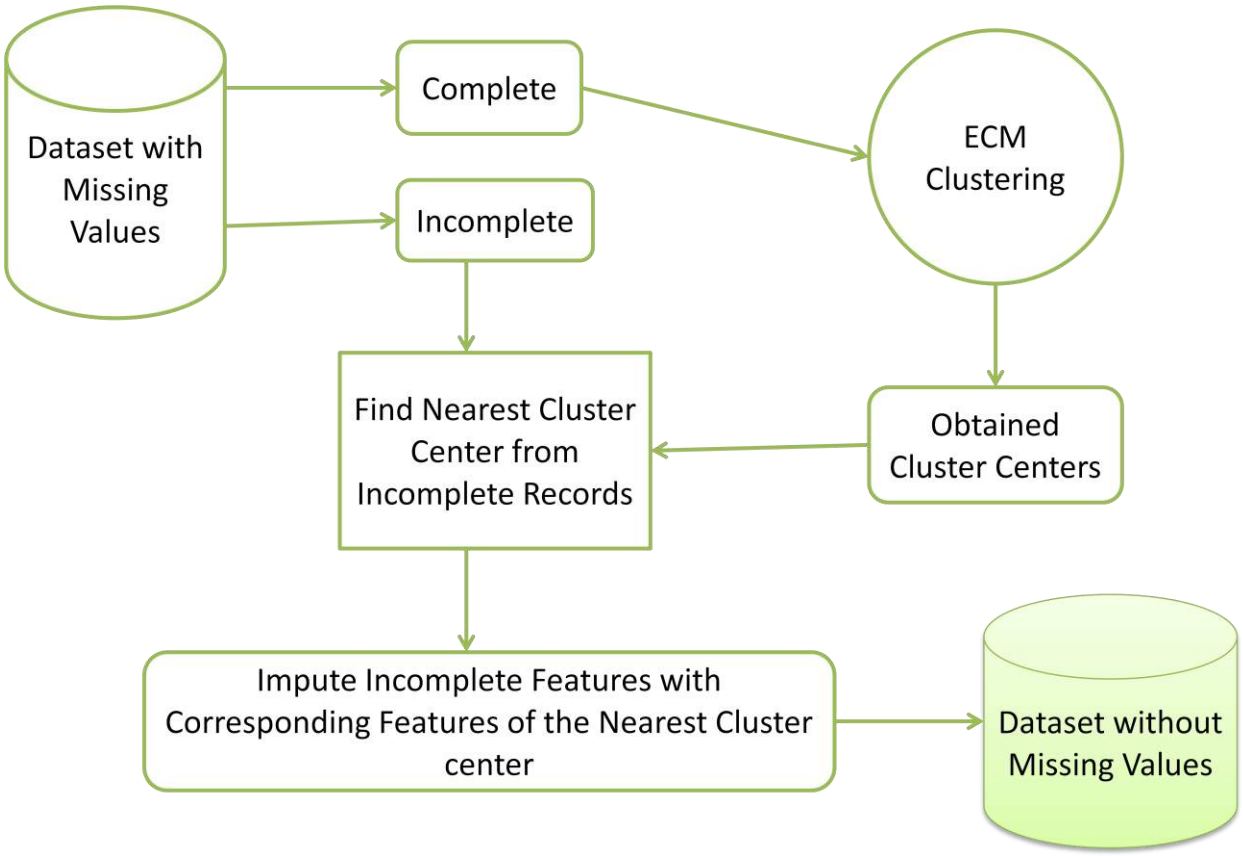

Fig. 3.5: Block Diagram of ECM-Imputation

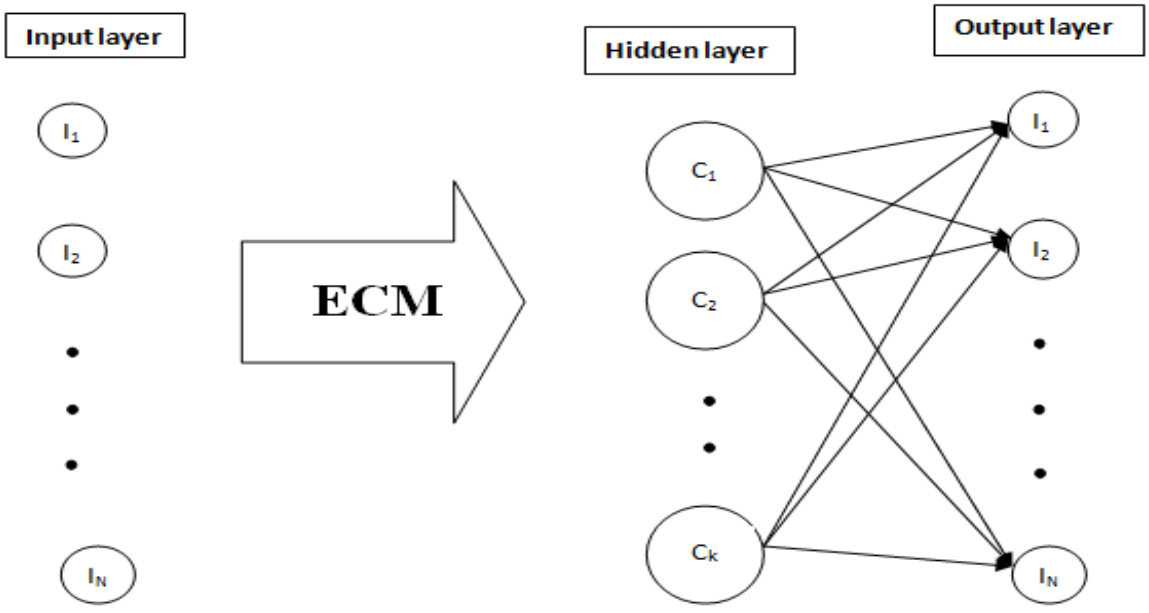

Fig. 3.6: Architecture of ECM-AAELM

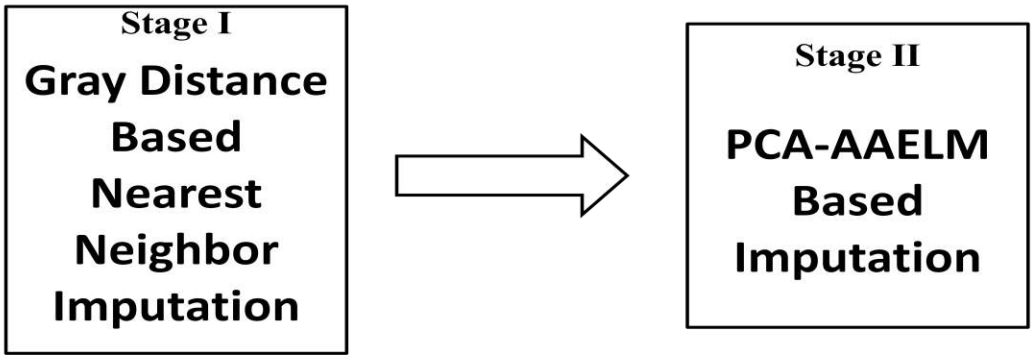

Fig. 3.7: Model of Gray+PCA-AAELM

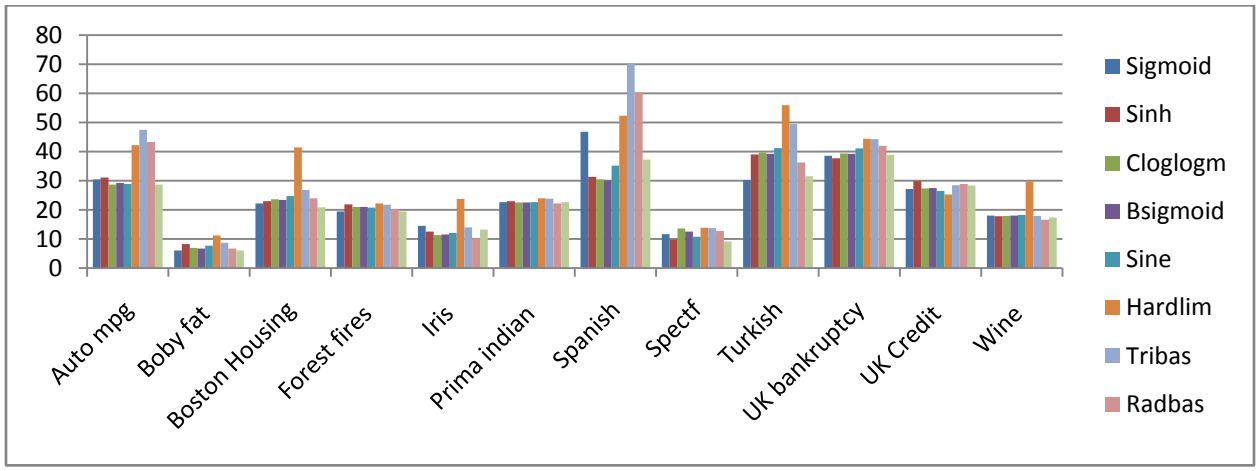

Fig. 3.8(a): Behavior of PCA-AAELM on different activation functions

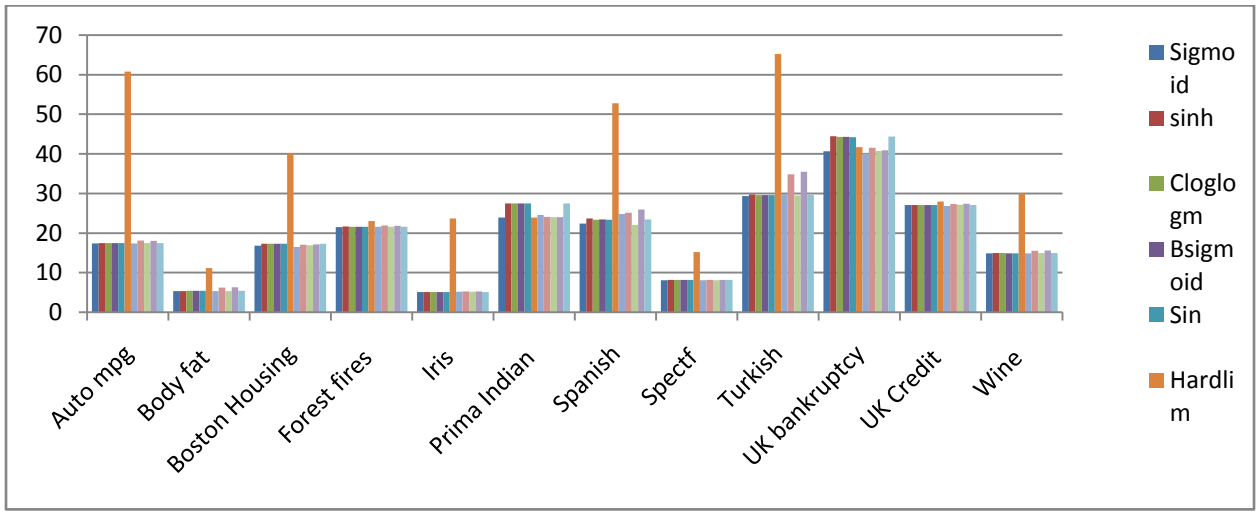

Fig. 3.8(b): Behavior of ECM-AAELM on different activation functions

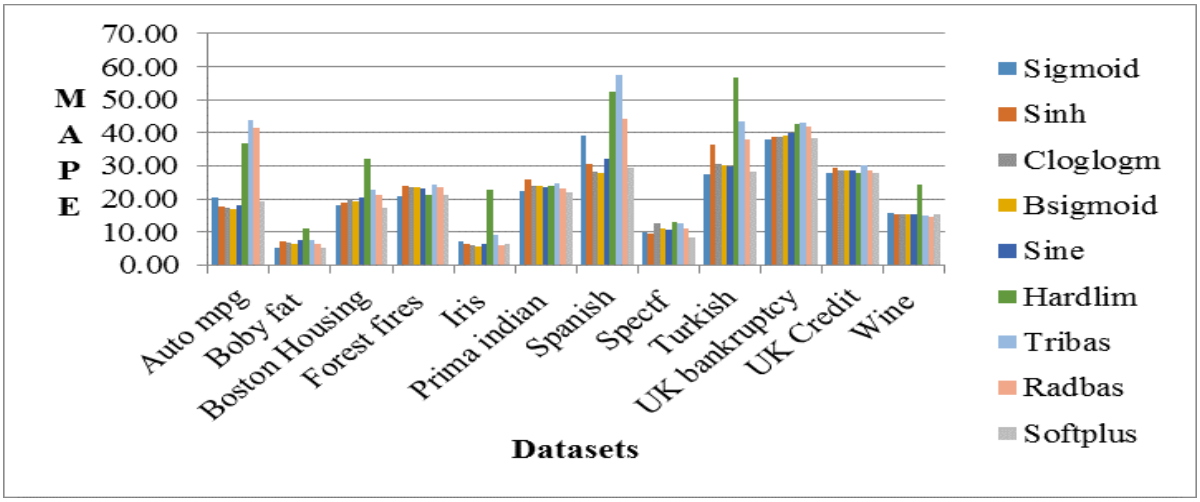

Fig. 3.8(c): Behavior of Gray+PCA-AAELM on different activation functions

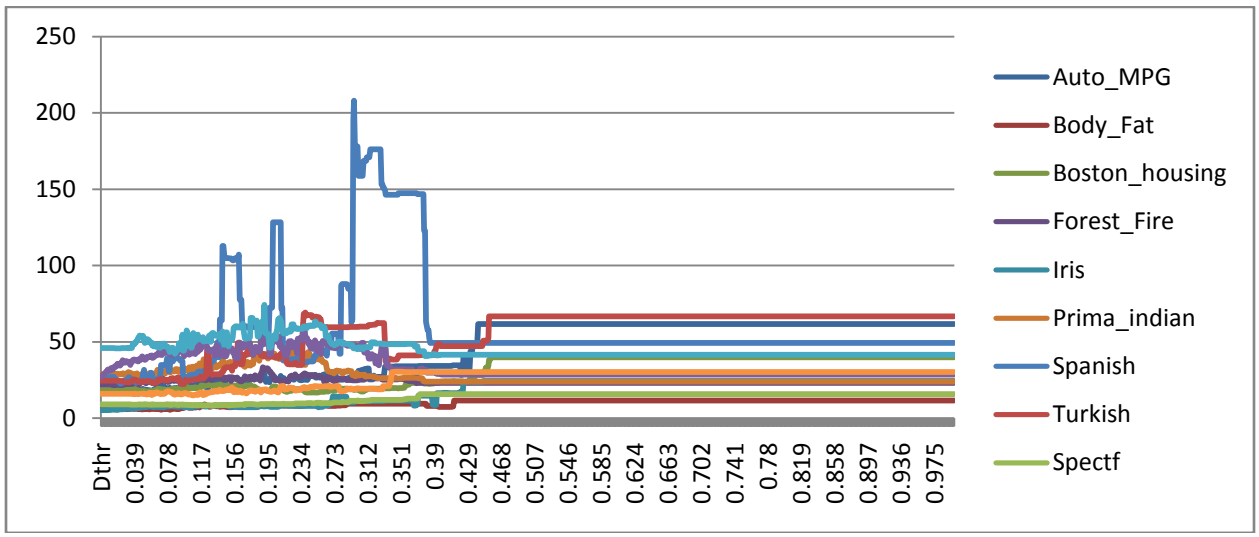

Fig. 3.9(a) Influence of Dthr value on MAPE results: ECM-AAELM

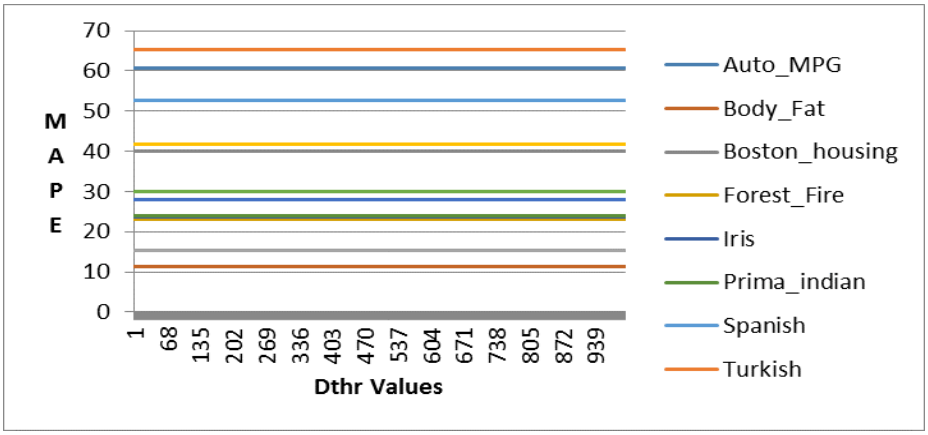

Fig. 3.9(b) Influence of Dthr value on MAPE results for Hardlim activation function

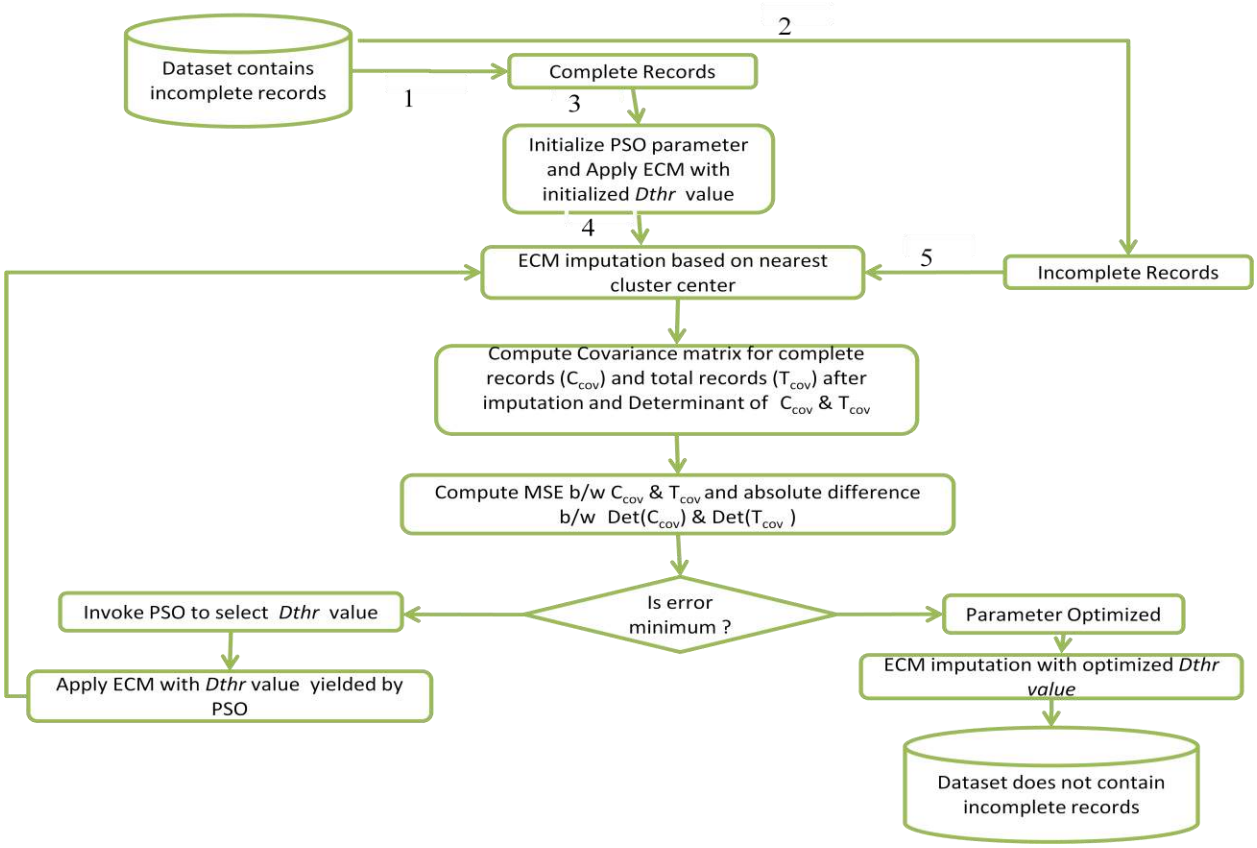

Fig. 4.1 Block Diagram of the Proposed Model ECM_PSO_COV

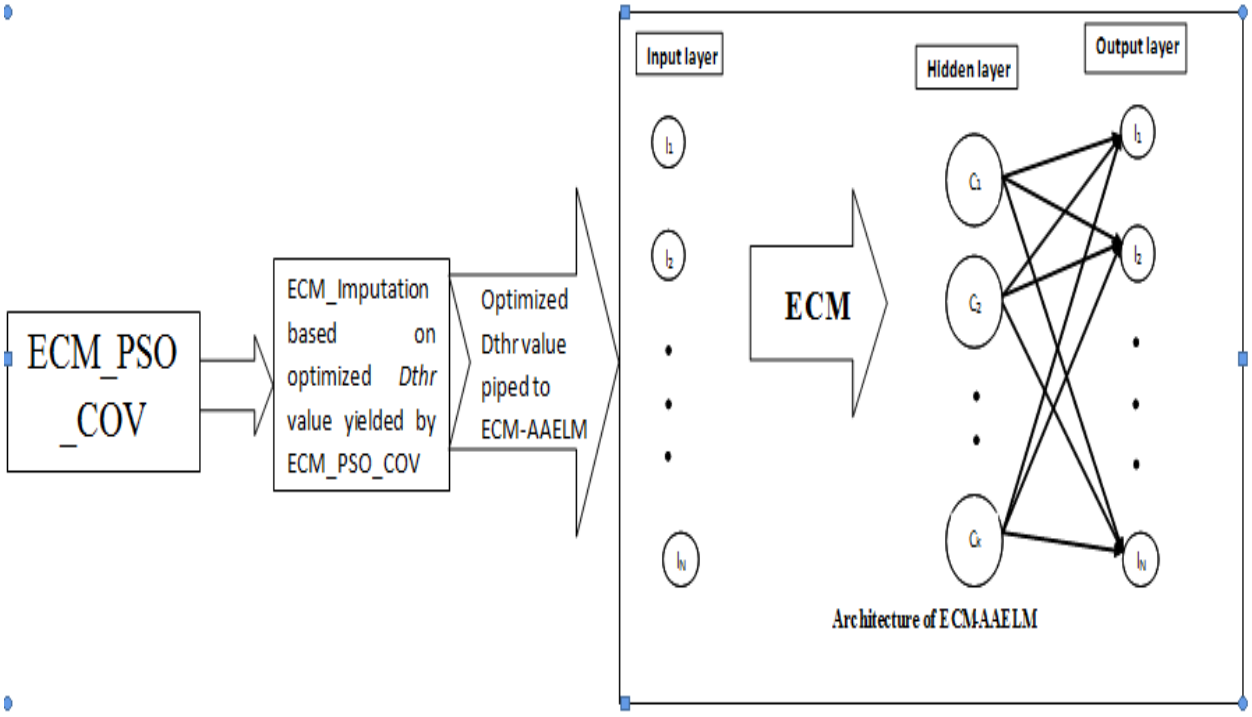

Fig. 4.2: Architecture of ECM_PSO_COV+ECM-AAELM

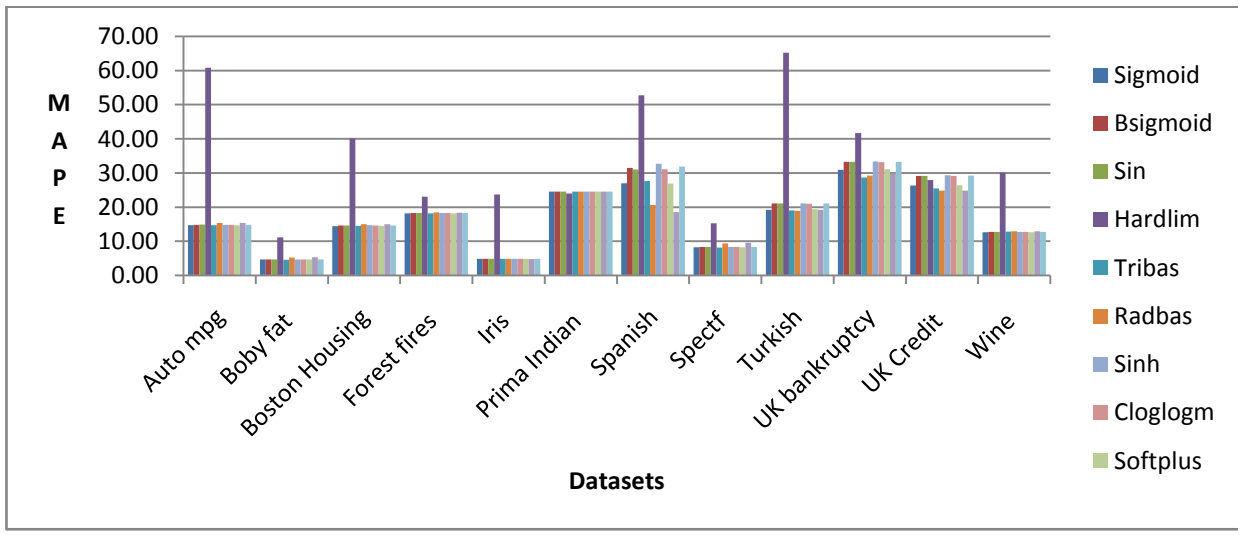

Fig. 4.3: Behavior of ECM_PSO_COV+ECM_AAELM on different activation functions

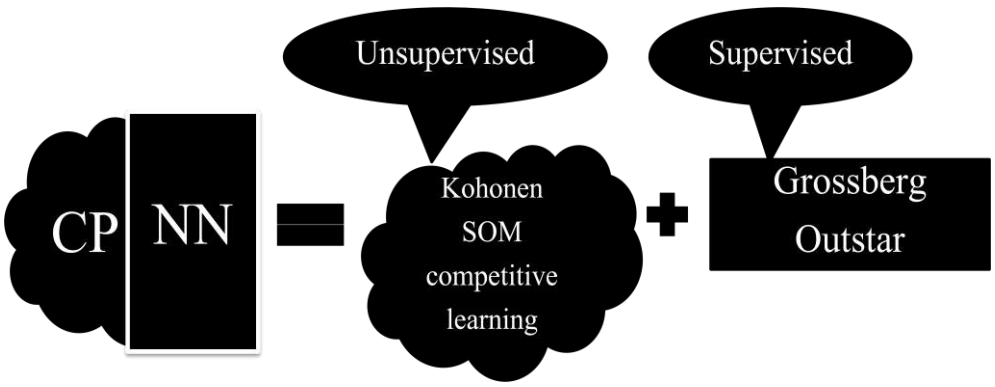

Fig. 5.1: Counter-propagation Neural Networks

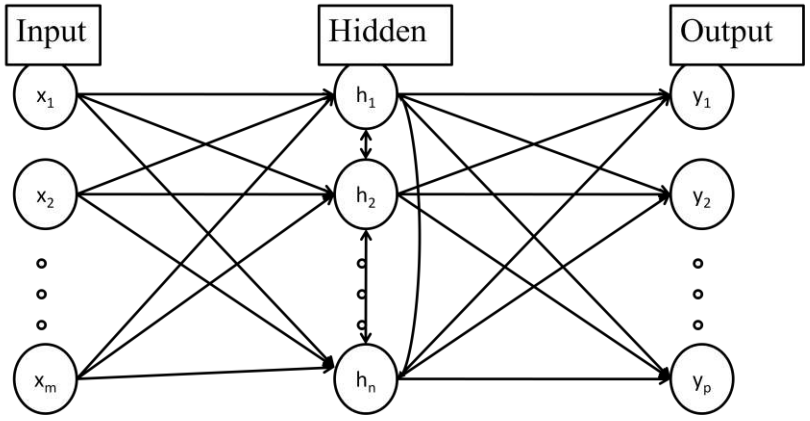

Fig. 5.2: Architecture of CPNN

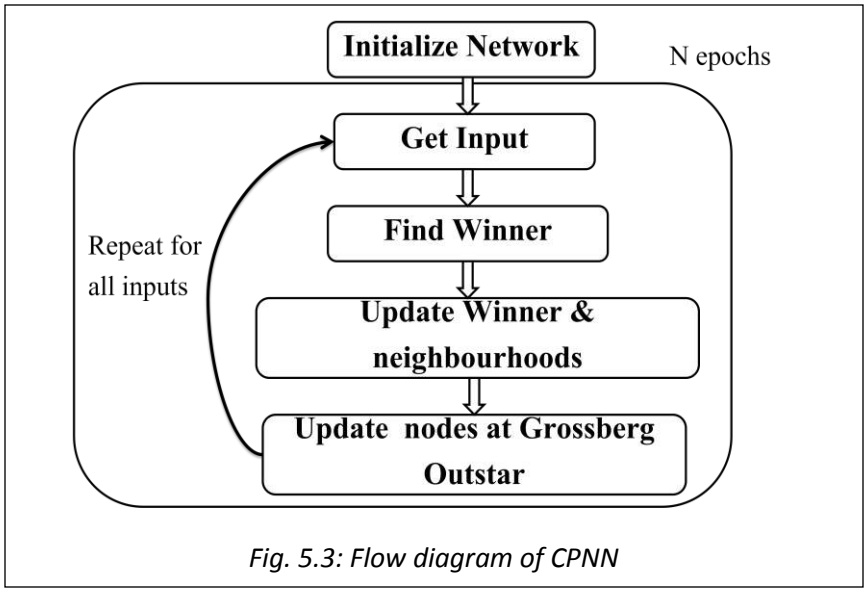

Fig. 5.3: Flow diagram of CPNN

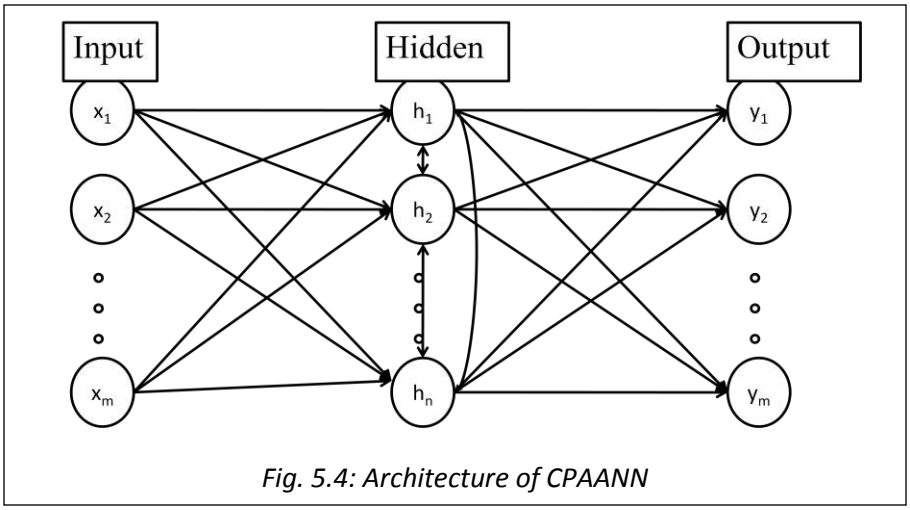

Fig. 5.4: Architecture of CPAANN

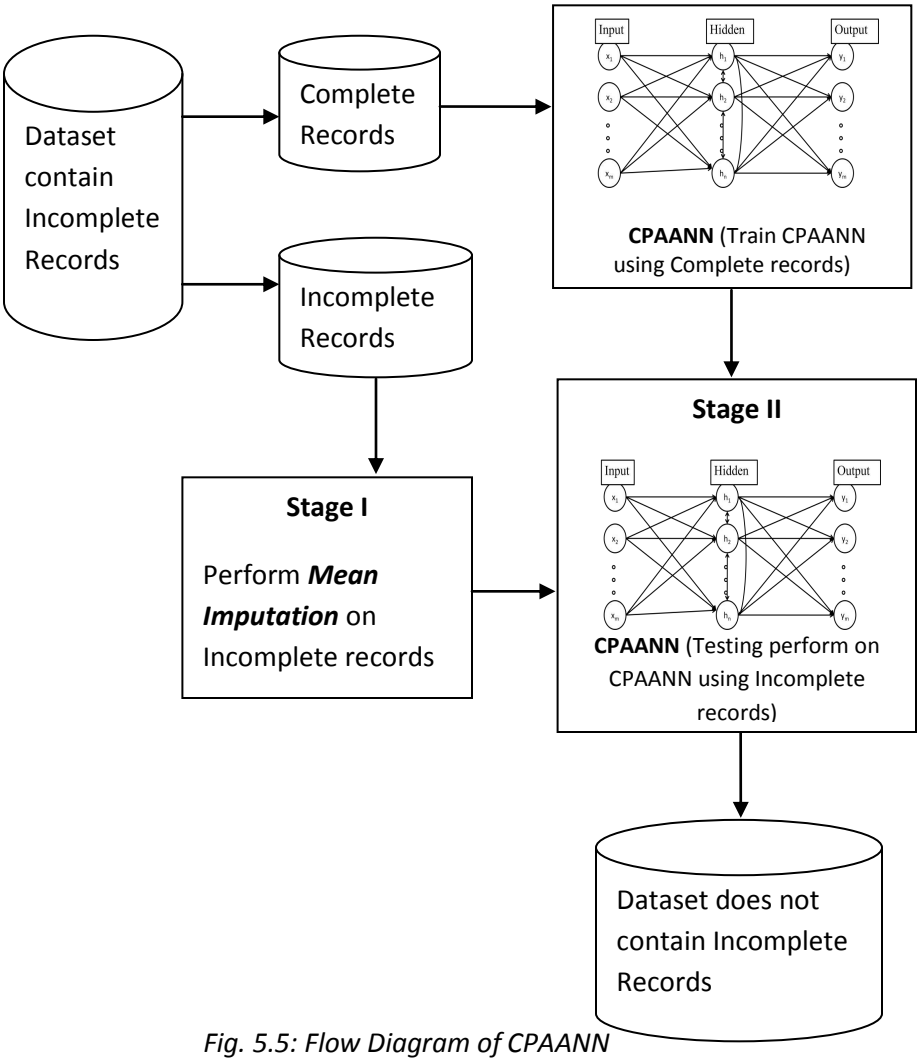

Fig. 5.5: Flow Diagram of CPAANN

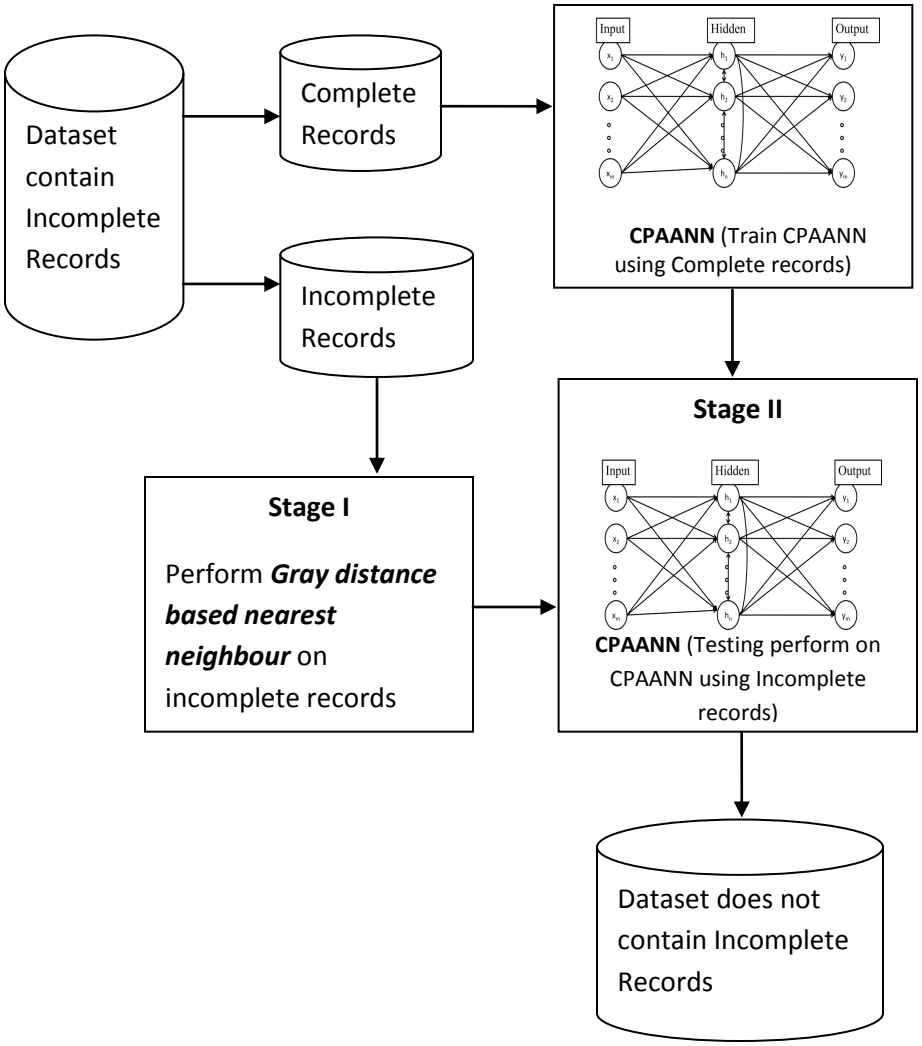

Fig. 5.6: Flow Diagram of Gray+CPAANN

APPENDIX C: TABLES

Table 2.1: Techniques for Numerical Data Imputation

TECHNIQUES WITH CITATION	BASIC PRINCPLE OF OPERATION
Deletion procedures	
Listwise Deletion (Song and Shepperd, 2007)	Eliminates all the instances with missing values.
Pairwise Deletion (Song and Shepperd, 2007)	Eliminates instances only from those statistical analyses that require the information.
Imputation procedures	
Hot-deck Imputation (Schafer, 1997)	Replaces the missing data with values from a similar complete data vector.
Mean Imputation (Little and Rubin, 2002)	Missing value is replaced by the mean.
Multiple Imputation (Little and Rubin, 2002)	Replaces each missing value with a set of plausible ones that represent uncertainty about right value to impute.
Regression Imputation (Little and Rubin, 2002)	Estimates the relationships among the variables and then uses coefficients to estimate the missing values.
Model based procedures	
Expectation Maximization (Little and Rubin, 2002)	An iterative procedure that continues until there is a convergence in parameter estimates.
Machine learning methods	
Genetic algorithms and neural networks (Marwala and Chakraverty, 2006)	Genetic algorithm is used to minimize an error function derived from an auto associative neural network.
Imputation with K-nearest neighbors (Batista and Monard, 2002)	K-nearest neighbors are selected from completed cases. The replacement value depends on type of data: the mode can be used for discrete data and mean for continuous data.
Missing Value Completion(MVC) method (Ragel and Cremilleux, 1999)	This method extends the concept of Robust Association Rules Algorithm (RAR) for databases with multiple missing values.
MLP Imputation (Gupta and Lam, 1996)	MLP is trained using only the complete cases as a regression model by taking incomplete variable as a target and remaining variables as input.
Neuro-fuzzy neural networks (Gabrys, 2002)	Missing values are processed with general fuzzy min-max neural network architecture.

SOM Imputation (Samad and Harp, 1992)	The value to be imputed based on the activation group of nodes in the missing dimensions.
Generalized regression neural network Ensemble for Multiple Imputation (GEMI) (Gheyas and Smith, 2010)	Imputation performed based on nonparametric Generalized regression neural network Ensemble for Multiple Imputation.
Nonparametric iterative imputation algorithm (*NIIA)* (Zhang et.al., 2011)	It utilize information within incomplete instances (instances with missing values) when estimating missing values.
A method based on an evolutionary algorithm (Figueroa et.al., 2011)	It used a genetic algorithm based on the minimization of an error function derived from their covariance matrix and vector of means is presented.
Fuzzy C-Means (FCM) (Nuovo, 2011)	Imputation performed based on the most famous fuzzy clustering algorithm.
Nearest Neighbor (NN) imputation (Yuan Li and Parker, 2012)	It estimates missing data in Wireless Sensor Networks (WSNs) by learning spatial and temporal correlations between sensor nodes.
*k*NN (*k* nearest neighbor) (Zhang, 2012)	Iteratively imputing missing data, named G*k*NN (gray *k*NN) imputation to deal with heterogeneous (i.e., mixed-attributes) data.
A method based on dynamic programming (Nelwamondoet.al., 2013)	Used a combination of dynamic programming, neural networks and genetic algorithms (GA) on suitable subsets of the input data for imputation.
Biclustering-based approach (França et.al., 2013)	This approach is based on the Mean Squared Residue metric, used to evaluate the degree of coherence among objects of a dataset.
Fuzzy *c*-means clustering hybrid approach (I. B. Aydilek and A. Arslan, 2013)	Utilized a fuzzy *c*-means clustering hybrid approach that combines support vector regression and a genetic algorithm.

Table 3.1: List of activation functions with their formula

Activation function names	Formula for Activation function
Sigmoid	H = 1. / (1 + exp (-x))
Sinh	H = sinh (x)
Cloglogm	H=1-2*exp (-0.7*exp (x))
Bipolar Sigmoid (Bsigmoid)	H= (1 - exp (-x)). / (1 + exp (-x))
Sin	H = sin (x)
Hardlim	Hardlim (x) = 1 if x ≥ 0 = 0, Otherwise
Tribas	H = Tribas (x) = 1 - abs(x), if -1 <= x <= 1 = 0, Otherwise
Radbas	H = Radbas (x) = exp (-x^2)
Softplus (soft)	H=log (1+exp (x))
Gaussian	H=exp (-x.*x*0.5)
Rectifier	H=max (x, 0);

Table 3.2: Average MAPE value over 10 folds ECM-AAELM

	Sigmoid	sinh	Cloglogm	Bsigmoid	Sin	Hardlim	Tribas	Radbas	Softplus	Gaussian	Rectifier	Min.
Auto mpg	**17.38**	17.49	17.47	17.44	17.43	60.78	17.4	18.07	17.46	18.05	17.46	17.38
Body fat	5.34	5.37	5.38	5.38	5.39	11.19	**5.33**	6.25	5.34	6.28	5.38	5.33
Boston Housing	16.79	17.32	17.29	17.28	17.27	40.1	**16.48**	17.07	16.9	17.15	17.29	16.48
Forest fires	**21.54**	21.63	21.61	21.58	21.56	23.06	21.59	21.93	21.62	21.84	21.59	21.54
Iris	5.13	5.12	5.11	5.11	**5.1**	23.69	5.17	5.28	5.14	5.28	5.11	5.1
Prima Indian	**23.95**	27.5	27.49	27.48	27.47	23.96	24.56	24.05	23.99	23.99	27.48	23.95
Spanish	22.41	23.67	23.38	23.43	23.37	52.8	24.79	25.12	**22.09**	25.91	23.48	22.09
Spectf	8.07	8.17	8.18	8.19	8.2	15.23	8.08	8.2	**8.05**	8.15	8.19	8.05
Turkish	22.22	22.16	22.15	22.18	22.20	65.26	**21.49**	24.85	22.20	25.81	22.17	21.49
UK bankruptcy	40.63	44.47	44.26	44.28	44.21	41.67	**40.06**	41.51	40.77	40.92	44.34	40.06
UK Credit	27.09	27.1	27.09	27.08	27.07	27.95	**26.85**	27.33	27.13	27.4	27.09	26.85
Wine	14.88	14.95	14.93	14.91	14.9	30.05	**14.88**	15.54	14.94	15.57	14.93	14.88

Table 3.3: Average MAPE value over 10 folds PCA-AAELM

	Sigmoid	Sinh	Cloglogm	Bsigmoid	Sine	Hardlim	Tribas	Radbas	Softplus	Min.
Auto mpg	30.41	31.07	28.67	29.18	28.9	42.19	47.45	43.28	**28.63**	28.63
Body fat	6.06	8.24	6.92	6.75	7.67	11.21	8.65	6.69	**6.01**	**6.01**
Boston Housing	22.17	22.96	23.64	23.41	24.78	41.46	26.86	23.92	**20.9**	**20.9**
Forest fires	**19.41**	21.89	21.03	21.03	20.82	22.19	21.74	20.22	19.45	**19.41**
Iris	14.48	12.53	11.34	11.49	12.1	23.73	14.01	**10.23**	13.17	10.23
Prima Indian	22.62	22.48	22.08	**22.06**	22.15	23.96	23.68	22.10	22.07	**22.06**
Spanish	46.81	31.27	30.4	**30.09**	35.19	52.28	70.3	60.38	37.27	**30.09**
Spectf	11.7	9.73	13.61	12.48	10.81	13.87	13.72	12.75	**9.11**	**9.11**
Turkish	**30.18**	39.03	39.67	39.22	41.25	55.91	49.43	36.29	31.56	**30.18**
UK bankruptcy	38.54	**37.7**	39.32	39.18	41.14	44.37	44.27	42.02	38.92	37.7
UK Credit	27.12	29.93	27.4	27.44	26.53	**25.27**	28.47	28.86	28.36	**25.27**
Wine	17.97	17.8	17.87	18.04	18.23	29.75	17.88	**16.6**	17.38	**16.6**

Table 3.4: Average MAPE value over 10 folds Gray+PCA-AAELM

	Sigmoid	Sinh	Cloglogm	Bsigmoid	Sine	Hardlim	Tribas	Radbas	Softplus	Minimum
Auto mpg	20.37	17.92	17.44	**16.92**	17.99	36.75	44.05	41.55	19.13	16.92
Boby fat	5.43	7.18	6.64	6.34	7.42	11.22	7.80	6.42	**5.41**	5.41
Boston Housing	18.22	18.82	19.55	19.16	20.34	32.31	22.82	21.42	**17.46**	17.46
Forest fires	**20.89**	23.92	23.70	23.68	23.38	21.31	24.19	23.44	21.30	20.89
Iris	7.27	6.50	5.87	**5.79**	6.47	22.68	8.98	6.13	6.58	5.79
Prima indian	22.43	25.92	24.04	24.02	23.51	23.96	24.89	23.05	**22.03**	22.03
Spanish	39.12	30.49	28.26	**28.06**	32.18	52.28	57.31	44.26	29.44	28.06
Spectf	10.05	9.75	12.75	11.12	10.71	12.88	12.82	11.15	**8.38**	8.38
Turkish	**27.38**	36.29	30.67	30.33	29.98	56.53	43.55	37.88	28.27	27.38
UK bankruptcy	**37.95**	38.71	38.97	39.21	39.82	42.69	43.18	41.78	38.59	37.95
UK Credit	**27.79**	29.52	28.54	28.56	28.68	28.04	30.12	28.76	27.86	27.79
Wine	15.81	15.54	15.27	15.33	15.47	24.48	14.95	**14.78**	15.43	14.78

Table 3.5: Average MAPE value over 10 folds

	Mean	K-Means+MLP[1]	K-Medoids+MLP[2]	K-Means+GRNN [2]	K-Medoids+GRNN [2]	ECM+GRNN [2]	PSO_COV	ECM_Imputation	ECM-AAELM	PCA-AAELM	Gray + PCA-AAELM
Auto mpg	59.7	23.75	20.70	20.27	**16.66**	17	24.53	18.03	17.38	28.63	16.92
Body fat	11.61	7.83	6.46	6.96	5.37	5.56	7.13	6.31	**5.33**	6.01	5.41
Boston Housing	37.77	21.01	17.69	19.57	17.68	18.08	24.85	17.84	**16.48**	20.9	17.46
Forest fires	24.72	26.61	24.46	26.21	22.97	24.38	24.85	22.29	21.54	**19.41**	20.89
Iris	23.57	9.41	9.17	8.79	8.04	6.3	8.71	5.27	**5.1**	10.23	5.79
Prima Indian	24.02	29.7	26.63	28.3	26.33	26.51	27.57	27.16	23.95	22.06	**22.03**
Spanish	55.53	39.91	32.45	37.96	26.01	34.11	33.25	31.98	**22.09**	30.09	28.06
Spectf	14.85	12.14	10.65	10.61	10.22	10.35	10.34	10.21	**8.05**	9.11	8.38
Turkish	66.00	33.01	26.90	25.9	**19.34**	22.34	30.20	27.90	21.49	30.18	27.38
UK bankruptcy	37.07	30.96	29.69	29.06	**28.39**	29.07	35.67	46.14	40.06	37.7	37.95
UK Credit	28.43	32.17	25.42	29.8	24.04	**21.93**	37.94	27.40	26.85	25.27	27.79
Wine	29.99	21.58	15.73	16.21	**14.75**	15.61	18.98	15.61	14.88	16.6	14.78

Table 3.6: Wilcoxon signed rank test values of PCA_AAELM

PCA_AAELM Vs.	K-Means+MLP	K-Medoids+MLP	K-Means+GRNN	K-Medoids+GRNN	ECM+GRNN	ECM_Imputation
Auto mpg	1.86	2.17	2.37	2.78	2.78	2.78
Body fat	1.66	0.94	1.66	0.64	0.43	0.23
Boston Housing	0.23	0.94	0.33	1.45	1.45	2.17
Forest fires	2.78	2.06	2.57	2.17	2.06	1.96
Iris	0.33	0.43	0.94	1.55	2.47	2.78
Prima Indian	2.78	2.78	2.68	2.47	1.76	2.78
Spanish	1.96	0.43	0.13	1.04	0.13	0.84
Spectf	2.47	2.06	2.37	1.96	1.45	2.27
Turkish	0.64	0.84	0.84	1.76	1.35	0.43
UK bankruptcy	1.66	2.06	2.06	1.86	1.86	1.66
UK Credit	1.76	0.43	1.15	0.54	0.94	0.84
Wine	2.57	0.54	0.33	1.15	0.54	0.74

Table 3.7: Wilcoxon signed rank test values of ECM_AAELM

	K-Means+MLP vs ECM_AAELM	K-Medoids+ MLP vs ECM_AAELM	K-Means+ GRNN vs ECM_AAELM	K-Medoids +GRNN vs ECM_AAELM	ECM+GRNN vs ECM_AAELM	ECM_Imputation vs ECM_AAELM
Auto mpg	2.27	1.55	1.35	0.03	0.03	0.64
Body fat	2.47	1.66	1.35	0.13	0.13	2.78
Boston Housing	2.57	0.84	1.55	0.94	0.84	0.74
Forest fires	2.47	1.25	1.96	0.43	1.45	2.78
Iris	2.37	2.37	2.27	1.66	1.15	2.17
Prima Indian	2.68	2.17	2.37	1.96	1.76	2.57
Spanish	2.68	2.47	1.55	0.33	0.64	2.06
Spectf	2.78	2.78	2.78	2.78	2.37	2.78
Turkish	0.43	0.13	0.84	1.76	0.54	2.47
UK bankruptcy	1.45	2.17	2.27	2.57	2.37	2.57
UK Credit	1.45	1.04	0.43	1.25	1.55	0.84
Wine	2.68	0.94	1.25	0.13	0.74	1.76

Table 3.8: Wilcoxon signed rank test values of Gray+PCA_AAELM

	K-Means+MLP vs Gray+PCA_AAELM	K-Medoids+ MLP vs Gray+PCA_AAELM	K-Means+GRNN vs Gray+PCA_AAELM	K-Medoids +GRNN vs Gray+PCA_AAELM	ECM+GRNN vs Gray+PCA_AAELM	ECM_Imputation vs Gray+PCA_AAELM
Auto mpg	2.27	1.55	1.66	0.33	0.33	0.94
Body fat	2.57	1.86	1.55	0.43	0.23	1.25
Boston Housing	2.17	0.94	1.25	0.13	0.43	0.03
Forest fires	2.47	1.45	2.06	0.74	1.55	1.86
Iris	2.37	2.06	2.06	1.25	0.23	0.94
Prima Indian	2.78	2.47	2.78	2.37	1.76	2.78
Spanish	2.27	1.25	0.74	0.54	0.33	0.13
Spectf	2.78	2.78	2.78	2.78	2.47	2.78
Turkish	0.03	0.54	0.54	1.76	1.55	0.23
UK bankruptcy	1.86	2.27	2.17	2.57	2.57	1.04
UK Credit	1.15	1.25	0.23	1.35	2.17	0.74
Wine	2.78	1.25	1.25	0.23	0.64	0.03

Table 3.9: Average MAPE value over 10 folds AAELM

	Mean	K-Means+ MLP	Uniform-Sigmoid	Uniform - Gaussian	Gaussian-Sigmoid	Gaussian-Gaussian	Logistic-Sigmoid	Logistic-Gaussian
Auto mpg	59.70	23.75	35.11	35.96	33.99	33.99	37.2	47.24
Body fat	11.61	7.83	12.92	25.91	11.68	12.09	13.03	33.66
Boston Housing	37.77	21.01	29.2	35.65	25.42	26.43	29.38	42.2
Forest fires	24.72	26.61	22	31.73	22.33	22.91	22.17	47.74
Iris	23.57	9.41	21.96	19.7	20.81	17.54	21.11	18.95
Prima Indian	24.02	29.7	23.41	26.29	23.03	24.4	22.88	29.34
Spanish	55.53	39.91	43.59	133.23	41.91	60.87	38.56	270.48
Spectf	14.85	12.14	17.05	29.20	22.87	22.89	24.29	36.52
Turkish	66.00	33.01	37.49	98.01	33.52	40.13	37.94	185.77
UK bankruptcy	37.07	30.96	36.05	42.51	38.53	37.7	38.58	49.43
UK Credit	28.43	32.17	30.84	36.04	31.92	32.19	30.83	40.41
Wine	29.99	21.58	22.65	33.27	21.26	21.92	24.07	47.59

Table 4.1: Average MAPE value over 10 folds ECM_PSO_COV+ECM_AAELM

	Sigmoid	**Bsigmoid**	**Sin**	**Hardlim**	**Tribas**	**Radbas**	**Sinh**	**Cloglogm**	**Softplus**	**Gaussian**	**Rectifier**
Auto mpg	**14.69**	14.84	14.86	60.78	14.75	15.38	14.80	14.82	14.71	15.37	14.83
Boby fat	**4.64**	4.69	4.69	11.19	4.64	5.29	4.68	4.69	4.66	5.32	4.68
Boston Housing	**14.44**	14.65	14.64	40.10	14.48	15.00	14.69	14.66	14.48	15.04	14.66
Forest fires	**18.17**	18.24	18.24	23.06	18.18	18.43	18.26	18.25	18.18	18.39	18.25
Iris	4.87	4.88	4.89	23.69	4.87	4.83	4.87	4.88	4.86	**4.83**	4.87
Prima Indian	24.55	24.54	24.54	**23.96**	24.60	24.59	24.54	24.54	24.54	24.58	24.54
Spanish	27.02	31.47	30.98	52.80	27.67	20.63	32.72	31.16	26.86	**18.53**	31.88
Spectf	8.23	8.33	8.34	15.23	**8.18**	9.41	8.31	8.32	8.22	9.54	8.32
Turkish	19.25	21.09	21.10	65.26	18.99	**18.97**	21.05	20.98	19.36	19.18	21.08
UK bankruptcy	30.93	33.29	33.28	41.67	**28.66**	29.27	33.33	33.21	31.11	29.90	33.30
UK Credit	26.38	29.18	29.13	27.95	25.51	24.80	29.36	29.15	26.41	**24.79**	29.24
Wine	**12.60**	12.74	12.73	30.05	12.80	12.96	12.79	12.76	12.65	12.92	12.76

Table 4.2: Comparison of Average MAPE value of various methods over 10 folds

	Mean	K-Means+MLP	K-Medoids+MLP	K-Means+GRNN	K-Medoids+GRNN	ECM+GRNN	*PSO_COV*	*PSOAANN*	*PSOAWNN*	*RBFAANN*	*GRAANN*	*ECM_Imputation*	*ECM_PSO_COV*	*ECM_PSO_COV+ECM_AAELM*
Auto mpg	59.7	23.75	20.70	20.27	16.66	17	24.53	37.59	38.16	62.53	15.54	18.03	15.35	14.39
Body fat	11.61	7.83	6.46	6.96	5.37	5.56	7.13	7.61	9.21	25.4	4.61	6.31	4.96	4.61
Boston Housing	37.77	21.01	17.69	19.57	17.68	18.08	24.85	24.61	30.94	98.87	15.38	17.84	14.50	14.18
Forest fires	24.72	26.61	24.46	26.21	22.97	24.38	24.85	22.69	26.62	59.24	18.47	22.29	18.34	17.66
Iris	23.57	9.41	9.17	8.79	8.04	6.3	8.71	15.84	12.83	26.93	5.75	5.27	4.82	4.75
Prima Indian	24.02	29.7	26.63	28.3	26.33	26.51	27.57	21.72	23.68	32.28	23.89	27.16	24.58	23.38
Spanish	55.53	39.91	32.45	37.96	26.01	34.11	33.25	60.95	48.81	847.02	23.28	31.98	20.73	16.99
Spectf	14.85	12.14	10.65	10.61	10.22	10.35	10.34	16.69	43.3	21.12	8.41	10.21	9.85	8.11
Turkish	66.00	33.01	26.90	25.9	19.34	22.34	30.20	53.56	33.45	188.85	17.25	27.90	19.28	16.49
UK bankruptcy	37.07	30.96	29.69	29.06	28.39	29.07	35.67	33.47	31.48	141.61	26.85	46.14	30.98	26.89
UK Credit	28.43	32.17	25.42	29.8	24.04	21.93	37.94	33.94	38.64	45.53	20.47	27.40	24.62	23.66
Wine	29.99	21.58	15.73	16.21	14.75	15.61	18.98	22.16	23.64	39.11	12.87	15.61	12.76	12.21

Table 4.3: Wilcoxon signed rank test values for ECM_PSO_COV

	K-Means+MLP vs ECM_PSO_COV	K-Medoids+MLP vs ECM_PSO_COV	K-Means+GRNN vs ECM_PSO_COV	K-Medoids+GRNN vs ECM_PSO-COV	ECM_Imputation vs ECM_PSO_COV	ECM+GRNN vs ECM_PSO_COV	PSO_COV vs ECM_PSO_COV	GRAANN vs ECM_PSO_COV
Auto mpg	2.78	2.78	2.47	0.84	2.68	0.94	2.78	0.03
Body fat	2.57	2.37	1.35	0.33	2.06	0.13	2.06	0.03
Boston Housing	2.57	2.57	2.78	2.68	2.78	2.47	2.57	1.35
Forest fires	2.78	2.78	2.78	2.78	2.78	2.47	2.17	0.54
Iris	2.78	2.78	2.47	2.06	2.17	1.15	2.37	2.27
Prima Indian	2.57	1.86	2.27	1.45	2.78	1.66	1.55	0.94
Spanish	2.17	1.35	2.06	0.13	2.68	1.45	2.06	0.54
Spectf	2.68	1.86	2.17	1.04	2.78	0.13	1.04	2.78
Turkish	1.66	0.23	1.55	0.84	2.27	0.33	1.96	1.04
UK bankruptcy	0.03	0.13	1.15	0.64	2.17	0.33	1.76	0.54
UK Credit	1.86	0.03	1.35	0.43	2.27	1.25	2.47	2.27
Wine	2.78	2.78	2.78	1.55	2.78	2.17	2.68	0.13

Table 4.4: Wilcoxon signed rank test values for ECM_PSO_COV+ECM_AAELM

	K-Means+ MLP vs ECM_PSO_COV+ECM_ELM	**K-Medoids+ MLP vs ECM_PSO_COV+ ECM_ELM**	**K-Means+GRNN vs ECM_PSO_COV+ ECM_ELM**	**K-Medoids+ GRNN vs ECM_PSO_COV+ ECM_ELM**	**ECM+GRNN vs ECM_PSO_COV+ ECM_ELM**	***ECM_Imputation vs ECM_PSO_COV+ ECM_ELM***	***PSO_COV vs ECM_PSO_COV+ ECM_ELM***	***GRAANN vs ECM_PSO_COV+ ECM_ELM***
Auto mpg	2.78	2.78	2.68	1.55	1.55	2.78	2.78	1.15
Body fat	2.68	2.47	1.86	0.94	1.25	2.78	2.27	0.64
Boston Housing	2.68	2.78	2.78	2.78	2.78	2.27	2.68	1.86
Forest fires	2.78	2.78	2.78	2.78	2.68	2.78	2.47	1.35
Iris	2.78	2.78	2.57	2.17	1.25	1.25	2.37	2.47
Prima Indian	2.68	2.37	2.47	2.27	1.76	2.78	2.17	0.64
Spanish	2.78	2.27	2.68	1.55	2.57	2.78	2.78	2.06
Spectf	2.78	2.78	2.78	2.78	2.78	2.78	2.78	1.25
Turkish	2.47	1.15	2.06	0.23	1.25	2.78	2.27	0.43
UK bankruptcy	1.15	0.94	0.13	1.55	1.25	2.78	1.76	0.13
UK Credit	1.96	0.33	1.45	0.23	1.04	2.47	2.68	2.06
Wine	2.78	2.78	2.78	2.37	2.57	2.78	2.78	0.43

Table 5.1: Parameter Selection in CPNN Toolbox

Parameter Selection for	**Parameter Selected for CPAANN**
Weights initialization function	Linear
Neighbourhood	Hexagonal
Shape of the map	Sheet
Neighbourhood function	Gaussian
Learning function	Linear
Number of the column with labels	1
Maximum Length of the Network	20
Maximum Width of the Network	20
Number of epochs in Rough training phase	18
Number of epochs in Fine training phase	230
Initial learning rate in Rough training phase	0.10
Initial learning rate in Fine training phase	0.05

Table 5.2: Average MAPE value of various methods over 10-folds

	K-Means+MLP	K-Medoids+MLP	K-Means+GRNN	K-Medoids+GRNN	ECM+GRNN	*PSO_COV*	*PSOAANN*	*PSOAAWNN*	*RBFAANN*	*GRAANN*	*CPAANN*	*Gray+CPAANN*	
												Stage I	Stage II
Auto mpg	23.75	20.70	20.27	16.66	17	24.53	37.59	38.16	62.53	15.54	18.32	16.73	**15.31**
Body fat	7.83	6.46	6.96	5.37	5.56	7.13	7.61	9.21	25.4	**4.61**	5.25	7.65	4.71
Boston Housing	21.01	17.69	19.57	17.68	18.08	24.85	24.61	30.94	98.87	15.38	**14.86**	19.28	15.01
Forest fires	26.61	24.46	26.21	22.97	24.38	24.85	22.69	26.62	59.24	18.47	**16.97**	22.89	17.91
Iris	9.41	9.17	8.79	8.04	6.3	8.71	15.84	12.83	26.93	5.75	6.51	5.34	**4.03**
Prima Indian	29.7	26.63	28.3	26.33	26.51	27.57	21.72	23.68	32.28	23.89	**18.21**	28.06	19.34
Spanish	39.91	32.45	37.96	26.01	34.11	33.25	60.95	48.81	847.02	23.28	17.13	36.29	**14.21**
Spectf	12.14	10.65	10.61	10.22	10.35	10.34	16.69	43.3	21.12	**8.41**	8.61	11.6	8.53
Turkish	33.01	26.90	25.9	19.34	22.34	30.20	53.56	33.45	188.8	17.25	**16.07**	36.63	17.37
UK bankruptcy	30.96	29.69	29.06	28.39	29.07	35.67	33.47	31.48	141.61	26.85	21.96	39.75	**20.58**
UK Credit	32.17	25.42	29.8	24.04	21.93	37.94	33.94	38.64	45.53	20.47	22.88	28.9	**13.70**
Wine	21.58	15.73	16.21	14.75	15.61	18.98	22.16	23.64	39.11	12.87	**11.56**	17.58	11.72

Table 5.3: Wilcoxon signed rank test values of various proposed methods vs CPAANN

CPAANN vs. 	K-Means+MLP	K-Medoids+MLP	K-Means+GRNN	K-Medoids+GRNN	ECM+GRNN	ECM_Imputation	PSO_COV	GRAANN
Auto mpg	2.37	0.84	1.04	0.74	0.64	0.03	2.68	2.78
Body fat	1.86	1.55	1.86	1.25	0.94	0.94	2.17	0.43
Boston Housing	2.68	2.17	2.27	2.06	2.27	1.66	2.68	0.74
Forest fires	2.78	2.68	2.78	2.68	2.68	2.47	2.68	1.25
Iris	1.86	1.86	1.66	1.25	0.03	1.76	1.66	0.94
Prima Indian	2.78	2.78	2.78	2.78	2.06	2.78	2.78	2.78
Spanish	2.78	2.06	2.68	0.94	2.27	2.57	2.78	1.96
Spectf	2.78	2.68	2.78	2.57	2.27	2.78	2.68	0.54
Turkish	2.57	1.55	2.57	0.33	1.86	2.68	2.47	0.84
UK bankruptcy	2.17	1.86	0.74	1.76	1.76	2.57	2.47	1.35
UK Credit	2.57	0.23	1.86	0.23	0.13	1.15	2.78	1.04
Wine	2.78	2.78	2.78	2.37	2.47	2.37	2.78	1.55

Table 5.4: Wilcoxon signed rank test values of various proposed methods vs Gray+CPAANN

Gray+CPAANN vs.	K-Means+ MLP	K-Medoids+ MLP	K-Means+GRNN	K-Medoids+ GRNN	ECM +GRNN	ECM_Imputation	PSO_COV	GRAANN
Auto mpg	2.47	2.06	1.96	1.25	1.35	2.27	2.57	0.13
Body fat	2.68	2.57	2.27	1.15	1.15	2.37	2.68	0.23
Boston Housing	2.47	2.68	2.68	2.57	1.96	1.86	2.47	0.43
Forest fires	2.78	2.47	2.78	2.47	2.57	2.78	2.57	0.74
Iris	2.78	2.78	2.78	2.68	2.17	2.06	2.78	2.68
Prima Indian	2.78	2.78	2.78	2.78	1.96	2.78	2.78	2.78
Spanish	2.78	2.68	2.78	2.06	2.68	2.78	2.78	2.37
Spectf	2.78	2.68	2.78	2.68	2.37	2.78	2.68	0.33
Turkish	2.37	1.15	2.27	0.23	1.25	2.37	2.17	0.23
UK bankruptcy	2.17	2.17	0.94	2.57	2.57	2.78	2.78	1.86
UK Credit	2.37	0.23	1.76	0.33	0.74	1.35	2.78	2.17
Wine	2.78	2.78	2.78	2.47	2.57	2.27	2.78	1.15

Table A.1: Attributes of Boston housing dataset

Attribute	**Description**
CRIM	*per capita crime rate by town*
ZN	*Proportion of residential land zoned for lots over 25,000 sq.ft.*
INDUS	*proportion of non-retail business acres per town*
CHAS	*Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)*
NOX	*nitric oxides concentration (parts per 10 million)*
RM	*average number of rooms per dwelling*
AGE	*proportion of owner-occupied units built prior to 1940*
DIS	*weighted distances to five Boston employment centres*
RAD	*index of accessibility to radial highways*
TAX	*full-value property-tax rate per $10,000*
PTRATIO	*pupil-teacher ratio by town*
B	*1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town*
LSTAT	*% lower status of the population*
MEDV	*Median value of owner-occupied homes in $1000's*

Table A.2: Attributes of Forest fires dataset

Attribute	**Description**
X	*x-axis spatial coordinate within the Montesinho park map: 1 to 9*
Y	*y-axis spatial coordinate within the Montesinho park map: 2 to 9*
MONTH	*month of the year: "jan" to "dec"*
DAY	*day of the week: "mon" to "sun"*
FFMC	*FFMC index from the FWI system: 18.7 to 96.20*
DMC	*DMC index from the FWI system: 1.1 to 291.3*
DC	*DC index from the FWI system: 7.9 to 860.6*
ISI	*ISI index from the FWI system: 0.0 to 56.10*
TEMP	*temperature in Celsius degrees: 2.2 to 33.30*
RH	*relative humidity in %: 15.0 to 100*
WIND	*wind speed in km/h: 0.40 to 9.40*
RAIN	*outside rain in mm/m2 : 0.0 to 6.4*
AREA	*the burned area of the forest (in ha): 0.00 to 1090.84*

Table A.3: Attributes of Auto MPG dataset

Attribute	Predictor variable name
C1	*mpg*
C2	*cylinders*
C3	*displacement*
C4	*horsepower*
C5	*weight*
C6	*acceleration*
C7	*model year*
C8	*origin*
C9	*car name*

Table A.4: Attributes of Bodyfat dataset

Attribute	**Predictor variable name**
X1	*Density determined from underwater weighing*
X2	*Percent body fat from Siri's (1956) equation*
X3	*Age (years)*
X4	*Weight (lbs)*
X5	*Height (inches)*
X6	*Neck circumference (cm)*
X7	*Chest circumference (cm)*
X8	*Abdomen 2 circumference (cm)*
X9	*Hip circumference (cm)*

X10	*Thigh circumference (cm)*
X11	*Knee circumference (cm)*
X12	*Ankle circumference (cm)*
X13	*Biceps (extended) circumference (cm)*
X14	*Forearm circumference (cm)*
X15	*Wrist circumference (cm)*

Table A.5: Attributes of Wine dataset

Attribute	Predictor Variable name
A1	*Alcohol*
A2	*Malic acid*
A3	*Ash*
A4	*Alcalinity of ash*
A5	*Magnesium*
A6	*Total phenols*
A7	*Flavanoids*
A8	*Nonflavanoid phenols*
A9	*Proanthocyanins*
A10	*Color intensity*
A11	*Hue*
A12	*OD280/OD315 of diluted wines*
A13	*Proline*

Table A.6: Attributes of Prima Indian dataset

Attributes	Predictor Variable Name
F1	*Number of times pregnant*
F2	*Plasma glucose concentration a 2 hours in an oral glucose tolerance test*
F3	*Diastolic blood pressure (mm Hg)*
F4	*Triceps skin fold thickness (mm)*
F5	*2-Hour serum insulin (mu U/ml)*
F6	*Body mass index (weight in kg/(height in m)^2)*
F7	*Diabetes pedigree function*
F8	*Age (years)*

Table A.7: Attributes of Iris dataset

Attribute	Predictor Variable name
X1	*sepal length in cm*
X2	*sepal width in cm*
X3	*petal length in cm*
X4	*petal width in cm*
CLASS	*Iris Setosa* *Iris Versicolour* *Iris Virginica*

Table A.8: Attributes of Spectf dataset

Attributes	Predictor Variable name
F1	*F1R: continuous (count in ROI (region of interest) 1 in rest)*
F2	*F1S: continuous (count in ROI 1 in stress)*
F3	*F2R: continuous (count in ROI 2 in rest)*
F4	*F2S: continuous (count in ROI 2 in stress)*
F5	*F3R: continuous (count in ROI 3 in rest)*
F6	*F3S: continuous (count in ROI 3 in stress)*
F7	*F4R: continuous (count in ROI 4 in rest)*
F8	*F4S: continuous (count in ROI 4 in stress)*
F9	*F5R: continuous (count in ROI 5 in rest)*
F10	*F5S: continuous (count in ROI 5 in stress)*
F11	*F6R: continuous (count in ROI 6 in rest)*
F12	*F6S: continuous (count in ROI 6 in stress)*
F13	*F7R: continuous (count in ROI 7 in rest)*
F14	*F7S: continuous (count in ROI 7 in stress)*
F15	*F8R: continuous (count in ROI 8 in rest)*
F16	*F8S: continuous (count in ROI 8 in stress)*
F17	*F9R: continuous (count in ROI 9 in rest)*
F18	*F9S: continuous (count in ROI 9 in stress)*
F19	*F10R: continuous (count in ROI 10 in rest)*
F20	*F10S: continuous (count in ROI 10 in stress)*
F21	*F11R: continuous (count in ROI 11 in rest)*
F22	*F11S: continuous (count in ROI 11 in stress)*
F23	*F12R: continuous (count in ROI 12 in rest)*
F24	*F12S: continuous (count in ROI 12 in stress)*
F25	*F13R: continuous (count in ROI 13 in rest)*

F26	*F13S: continuous (count in ROI 13 in stress)*
F27	*F14R: continuous (count in ROI 14 in rest)*
F28	*F14S: continuous (count in ROI 14 in stress)*
F29	*F15R: continuous (count in ROI 15 in rest)*
F30	*F15S: continuous (count in ROI 15 in stress)*
F31	*F16R: continuous (count in ROI 16 in rest)*
F32	*F16S: continuous (count in ROI 16 in stress)*
F33	*F17R: continuous (count in ROI 17 in rest)*
F34	*F17S: continuous (count in ROI 17 in stress)*
F35	*F18R: continuous (count in ROI 18 in rest)*
F36	*F18S: continuous (count in ROI 18 in stress)*
F37	*F19R: continuous (count in ROI 19 in rest)*
F38	*F19S: continuous (count in ROI 19 in stress)*
F39	*F20R: continuous (count in ROI 20 in rest)*
F40	*F20S: continuous (count in ROI 20 in stress)*
F41	*F21R: continuous (count in ROI 21 in rest)*
F42	*F21S: continuous (count in ROI 21 in stress)*
F43	*F22R: continuous (count in ROI 22 in rest)*
F44	*F22S: continuous (count in ROI 22 in stress)*

Table A.9: Description of UK Credit dataset

Attributes	**Predictor variable name**	
F1	*Year of birth*	*dob*
F2	*Number of children*	*nkid*
F3	*Number of other dependents*	*dep*
F4	*Spouse's income*	*sinc*
F5	*Applicant's income*	*dainc*
F6	*Value of Home*	*dhval*
F7	*Mortgage balance outstanding*	*Dmort*
F8	*Outgoings on mortgage or rent*	*doutm*
F9	*Outgoings on Loans*	*doutl*
F10	*Outgoings on Hire Purchase*	*douthp*
F11	*Outgoings on credit cards*	*doutcc*

Table A.10: Description of Banking datasets

SNO.*Predictor Variable Name*		
Turkish banks' data		
1	*Interest Expenses/Average Profitable Assets*	*IE/APA*
2	*Interest Expenses/Average Non-Profitable Assets*	*IE/ANA*
3	*(Share Holders' Equity + Total Income)/(Deposits + Non-Deposit Funds)*	*(SE+TI)/(D+NF)*
4	*Interest Income/Interest Expenses*	*II+IE*
5	*(Share Holders' Equity + Total Income)/Total Assets*	*(SE+TI)/TA*
6	*(Share Holders' Equity + Total Income)/(Total Assets + Contingencies &Commitments)*	*(SE+TI)/(TA+CC)*
7	*Networking Capital/Total Assets*	*NC/TA*
8	*(Salary And Employees' Benefits + Reserve For Retirement)/No. Of Personnel*	*(SEB+RR)/P*
9	*Liquid Assets/(Deposits + Non-Deposit Funds)*	*LA/(D+NF)*
10	*Interest Expenses/Total Expenses*	*IE/TE*
11	*Liquid Assets/Total Assets*	*LA/TA*
12	*Standard Capital Ratio*	*SCR*
Spanish banks' data		
1	*Current Assets/Total Assets*	*CA/TA*
2	*Current Assets-Cash/Total Assets*	*CAC/TA*
3	*Current Assets/Loans*	*CA/L*
4	*Reserves/Loans*	*R/L*
5	*Net Income/Total Assets*	*NI/TA*
6	*Net Income/Total Equity Capital*	*NI/TEC*
7	*Net Income/Loans*	*NI/L*
8	*Cost Of Sales/Sales*	*CS/S*
9	*Cash Flow/Loans*	*CF/L*
UK banks' data		
1	*Sales*	*SALES*
2	*Profit before tax/capital employed (%)*	*PBT/C*
3	*Funds flow/Total liabilities*	*FF/TL*
4	*(Current liabilities + long term debit)/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*