A hybrid approach for improving data classification based on PCA and enhanced ELM

Doaa L. El-Bably*

Department of Scientific Computing
Faculty of Computers and Informatics
Benha University, Egypt
E-mail: doaa.elbably@fci.bu.edu.eg

Khaled M. Fouad

Department of Information Systems
Faculty of Computers and Informatics
Benha University, Egypt
E-mail: kmfi@fci.bu.edu.eg

Abstract: The efficient and effective process of extracting the useful information from high-dimensional data is a worth studying problem. The high-dimensional data is a big and complex that it becomes difficult to be processed and classified. Dimensionality reduction (DR) is an important and a key method to address these problems. This paper presents a hybrid approach for data classification constituted from the combination of principal component analysis (PCA) and enhanced extreme learning machine (EELM). The proposed approach has two basic components. Firstly, PCA; as a linear data reduction, is implemented to reduce the number of dimensions by removing irrelevant attributes to speed up the classification method and to minimize the complexity of computation. Secondly, EELM is performed by modifying the activation function of single hidden layer feedforward neural network (SLFN) perfect distribution of categories. The proposed approach depends on a static determination of the reduced number of principal components. The proposed approach is applied on several datasets and is assisted its effectiveness by performing different experiments. For more reliability, the proposed approach is compared with two of the previous works, which used PCA and ELM in data analysis.

Keywords: Data mining, Data classification, Principal component analysis (PCA), Neural Network, Extreme Learning Machine (ELM).

Reference to this paper should be made as follows: El-bably, D.L. and Fouad, Kh.M. (2017) '*A hybrid approach for improving data classification based on PCA and enhanced ELM*', *Int. J. Advanced Intelligence Paradigms*, Vol. X, No. Y4, pp.000–000.

Biographical notes: Doaa L. El-Bably is an M.Sc. student and had obtained her B.Sc. in Computers and Informatics in 2012, Faculty of Computers and Informatics, Department of Scientific Computing, Benha University, successfully passed the IBM Academic Certificate exam and earned the title "Big Data Specialist with IBM Big Insights V2.1", Nov- 2015. Working now as a demonstrator in Computers and Informatics, Benha University, Egypt, her research Interests are in Big Data, Dimensionality Reduction, and Neural Network.

Khaled M. Fouad had obtained B.Sc. in 1995 and M.Sc. in 2003 and Ph.D. in 2012, Department of Systems and computers engineering, Faculty of Engineering. Working now as a lecturer in Computers and Informatics, Benha University, Egypt, His current research interests focus on Text Mining, Data Mining, Cloud Computing, Big Data, Semantic Web, and Expert Systems.

1 INTRODUCTION

The major challenge in developing of information science and data mining (YANG and WU, 2006) is utilizing an essential information gathered from high-dimensional original data and meeting the requirement of today's rapidly growing of data which took us to the world of Big Data (Lohr, 2008). Therefore, dimension reduction is an important method to address the dimensionality problem by removing the redundant or irrelevant information that was performed by many kinds of research (Fodor, 2002; Huo and Smith, 2008; Sarveniazi, 2014; Azar and Hassanien, 2014). In a proposed approach, PCA is used as dimensionality reduction method, in addition, applied as preprocessing data in order to obtain better performance of the classifier.

Up to now, extreme learning machine (ELM) has more attention for classification and regression tasks (Gautam, Tiwari, and Leng, 2015). ELM is extremely fast learning model and the capability of time processing that is recognized by many researchers (Cao, Chen, and Fan, 2016; Iosifidis, Tefas, and Pitas, 2016; Li, et al., 2016), which is based on single hidden layer feedforward neural network (SLFN). ELM was implemented by (Huang, Zhu, and Siew, 2006a), where the input weights and values of the hidden layer bias were assigned randomly and the output weights of SLFN analytically were computed using Moore-Penrose generalized inverse (Fill and Fishkind, 1998; MacAusland, 2014). However, there are many approaches have been executed through last few years to progress the performance of the classical ELM in several directions (Huang, Wang and Lan, 2011; Huang, 2013; 2014; Cao, Chen, Fan, 2014; Iosifidis, Tefas, Pitas, 2015; Iosifidis, 2015; Zhang, 2015). Many approaches handle the problem of choosing the proper number of hidden nodes by using different techniques (Huang, Chen, and Siew, 2006b). The incremental extreme learning machine (I-ELM) (Feng, 2009; Huang, Lin and Gay, 2008) which increases the number of hidden nodes until it reaches a certain error. Castaño Fernández-Navarro and Hervás-Martínez (2013) used the information that retrieved from PCA on training data to estimate the number of hidden nodes, and Memetic-ELM (Zhang et al., 2016) to get the optimal network parameters according to each task. On the other hand, there are a lot of optimization problems for several ELM variants are solved (Iosifidis and Gabbouj, 2015; Iosifidis, Tefas and Pitas, 2014).

Although, ELM is fast learning model but cannot treatment noise well, whereas PCA organized especially for preprocessing data for both noisy and high dimensional data. So, PCA considers complementary to elm to achieve the best performance for data classification

In this paper, a proposed approach is determined to improve high dimensional data processing and build effective classifier in terms of the speed and the accuracy. The proposed approach is based on PCA and ELM named enhanced ELM (PCA-EELM), which provides two main contributions. Firstly, PCA is used to reduce the dimension and remove the noisy data. Secondly, a new way is proposed to progress ELM performance by using another computation function of the hidden layer to arrive at the high accuracy rate with the minimum number of hidden nodes. Finally, several numbers of classification algorithms were implemented within and without principal component analysis (PCA) and compared with the proposed approach to prove the effectiveness and efficiency in the tasks of classification.

The following sections of this paper are presented as follows: In section 2 is the review of all previous works of PCA as dimension reduction technique, ELM as classification algorithm and the integration between them. Section 3 shows a background of PCA and classical ELM and common classification techniques. In section 4, the details of proposed approach (PCA-EELM) and its effectiveness on Big Data. The experimental results of PCA-EELM on benchmark sixteen datasets were explained in section 5. Finally, in section 6 this paper is concluded and briefly suggestion our future works.

2 RELATED WORK

2.1 Dimensionality Reduction

Zhu and others (2017) have proposed the improved dimensionality reduction method. This method was based on incremental orthogonal components analysis (IOCA) aims at handling the complex process of extracting the useful information from high-dimensional data. IOCA can perform the next four functions, which is based on an adaptive threshold policy. Firstly, it can keep learning from continually input data; secondly, achieve effective orthogonal component learning; thirdly, automatically appreciate and update the objective dimension; fourthly, achieve numerically orthogonal components.

Sharma and Saroha (2015) have proposed the method for dimensionality reduction based on principal component analysis integrated with feature ranking. Feature evaluation and ranking algorithms aimed at selecting the suitable subset of features but the authors found that these algorithms inefficient and impractical for very high dimensionality datasets. To address this issue, the output of PCA, which is a set of reduced for uncorrelated features, is applied to feature ranking and evaluation. It led to an improvement of computation time as compared to using feature evaluation and ranking for all the features. The authors have applied their proposed method on breast cancer dataset.

In the issue of anomaly detection in data traffic, Huang, Sethu, and Kandasamy (2016) have determined three defects of the traditional variance-based subspace method, when it was used in anomaly detection: (1) the structure of normal traffic is used to compute the number of reduced principal components, when the structure of the difference between the observed and the normal traffic is relevant during choosing the convenient dimensions of anomaly detection; (ii) a fixed determination of the number of reduced principal components is inappropriate, when the number of dimensions is variant during different periods of time (iii) because of the performance of anomaly detection is very sensitive to small changes in the number of dimensions, the method may present weak heuristics. The authors presented the distance-based method to anomaly detection dimensionality reduction to address these weaknesses of traditional variance-based subspace method. The proposed method was based on the metric called the “maximum subspace distance”.

In the gas identification systems, Akbar and others (2016) have analyzed the feature reduction algorithms based on a linear discriminant analysis (LDA) and PCA by using gas data. The used gas data was extracted using two types of sensors, which are an in-house fabricated 4x4 tin oxide gas array sensor and 7 commercial Figaro sensors. A decision tree (DT) classifier applied to prove the performance of the approaches PCA and LDA. Because of DT implementation simplicity and uniform behavior, it is used for classification.

The hybrid Neuro-Genetic method was proposed to predict the coronary heart disease level (Murthy and Meenakshi, 2014). The genetic algorithm was used to select feature

subset by applying the optimization of a multi-objective fitness function. The authors presented several studies which have performed the neuro-genetic method for feature subset selection. They have investigated, that initializing the weights of an artificial neural network by applying the genetic algorithm, which can exploit the advantage of optimization to overcome the faults of the slow convergence of artificial neural network and stuck in the local minima.

Omer and Khurran (2015) have presented the difference of one-dimensional component analysis (1D-PCA) and two-dimensional principal component analysis (2D-PCA). They examined two methods by applying these methods to two different types of classification techniques; support vector machines (SVM) and k-nearest neighbor (kNN). Instead of column vectors, which is used in 1DPCA, 2DPCA used 2D image matrices. The eigenvectors, which was inferred from these matrices, resulted in reduced dimensions of the images to be used for classification.

The dimension reduction, which is based on the principal component analysis (PCA) was proposed for revisiting the learning problem for pooling (Hosoya and Hyvarinen, 2016). The authors showed that by using strong dimension reduction, which is based on the principal component analysis, visual spatial pooling can be simply learned. This method aimed at ignoring a large part of the spatial structure of the input and thus appreciated a linear pooling matrix. They analyzed several different of the pooling models and discussed that pooling can be obtained from any type of linear transformation, which keeps several of the first principal components and represses the residual ones.

The algorithm of scalable supervised dimensionality reduction for a number of classification tasks has been developed (Raeder, Dalessandro, and Provost, 2013). The algorithm carried out the hierarchical clustering (Fouad and Dawood, 2016) in the parameters' model space using historical models to breakdown related features into a single dimension. This algorithm was capable to implicitly combine feature and label data of all tasks, which needs not operate directly in a large space.

2.2 Using ELM and PCA

In much traditional extreme learning machine (ELM) methods, the basic functions' parameters are randomly produced and need not be tuned, while the weights joining two layers, which are the hidden layer and the output layer, are analytically appraised (Castano, et al., 2016). The optimal parameters of basic functions, which are identified to be included in the hidden layer, are still an open issue. Cross-validation and heuristic methods are used to carry out this task. The authors depended on the principal component analysis (PCA) and ELM to assess the parameters of basic functions according to the parameters of principal components.

Singh, Chetty, and Sharma (2012) have proposed a combination of the support vector machine and the extreme learning machine based on protein structure prediction scheme. The integration of CA and the linear discriminant analysis (LDA) are used for recognition of multi-class protein fold. The validation of the proposed method experiment, which is based on a publicly available protein data set, showed a significant performance improvement of the proposed method.

The algorithm, which is based on modified ELM algorithm (P-ELM) and PCA method, was proposed in (Zhang, et al., 2015). The proposed algorithm was capable of reducing the number of hidden nodes to reduce the training time. In the simulation part, the authors artificially added the specific value of noise to the data, to prove the durability of P-ELM. In this research, PCA method was used to handle with the output matrix of hidden layer instead of the original data.

2.3 Using ELM in classification

Cao and others (2016) have mentioned that extreme learning machine (ELM) and sparse representation classification (SRC) are integrated to address the classification accuracy and computational complexity. For this reason, the authors proposed the hybrid classifier, which aimed at exploiting the benefits of ELM and SRC. The proposed classifier has two stages. In the first stage, the training by supervised learning using ELM network. In the second stage, the output of ELM is used to determine whether the image of a query can be properly classified or not. If the output is trustworthy, the classification is performed by ELM; else the query image is entered to SRC.

Krishnasamy and Paramesran (2016) provided the ELM to address semi-supervised learning problems by applying hessian regularization with ELM. Using hessian regularization in semi-supervised ELM algorithms, supported functions whose values alter linearly in relation to geodesic distance and maintains the local manifold well. Hessian regularization improved the performance of traditional ELM in semi-supervised learning. The proposed algorithm provided learning capability and the computational efficiency of traditional ELM, particularly in the case of multi-class classification problems.

Zhang and others (2015) have analyzed the pros and cons of different optimization methods for ELM. Memetic Algorithm (MA) was used to adaptively define the parameters of a network for ELM classification. MA was used to design a search to adaptively find optimal parameters of a network for each applied data set. The individual memetic calculation processes assure that the parameters of the network adaptively can be adjusted and obtain higher classification accuracy.

Jezowicz and others (2015) used extreme learning machine classification algorithm (ELM). ELM is a relatively rapid algorithm, which is based on the single hidden layer feedforward of a neural network. The objective of this research is analyzing potentiality of ELM implementation upon “CUDA” platform. They presented four various implementations of ELM on GPU.

The implementation of complete GPU of ELM method was presented and applied to hyperspectral remote sensing images for purposes of land cover (López-Fandiño, et al., 2014). In this research, different techniques, which aimed at improving the outcome of the traditional ELM classifier, were implemented on GPU. The spectral–spatial classification scheme was adopted for hyperspectral images by using a pixelwise spectral classifier addition to a spatial segmentation process by watershed utilized to the outcome of the robust color morphemically gradient (RCMG).

Xiang and others (2014) have proposed extreme learning machine method, which scales horizontally without collapsing accuracy of detection. They utilized ELM for the model of MapReduce programming. The objective of using ELM is classifying intrusion attempts.

3 PRELIMINARIES

3.1 Principal Component Analysis (PCA)

Principal Component Analysis (Jolliffe, 2002; Escabias, Aguilera, and Valderrama, 2004) is the most common linear technique available for unsupervised dimension reduction while preserving as much as possible of the current variation in the original data. Although the PCA technique is based on the linear transformation (Van der, Postma, and van den, 2008), there are many non-linear techniques, which are not better than

traditional PCA on real-world tasks. PCA is capable of computing a linear orthogonal transformation to find linearly uncorrelated components (principal components), which are a vector that represents for the largest variance (eigenvectors with the largest eigenvalues of the covariance matrix) and by removing the dimensions that have the smallest variance, then PCA allows us to execute the projection of data from a high dimensional to a low dimensional.

There are two popular methods for implementation PCA technique either using eigenvalue decomposition for covariance matrix of data or by singular value decomposition (SVD) of a patterns matrix however, the proposed approach focuses on the traditional method of (SVD) (Smith, 2002) to compute principal components that are the preferred method for numerical data accuracy. The PCA transformation of the data defined as

$$X = E^T X \quad (1)$$

Where $X = [x_{1,\dots,} x_N] \in R^{M \times N}$ *E* denotes the data matrix; and E ∈ $R^{(M \times M)}$ denotes a matrix of eigenvectors that accounts the corresponding eigenvalues, then, singular value decomposition is applied to the following equations: -

$$X = USV^T \quad (2)$$

Where $U \in R^{M \times M}$, $V \in R^{N \times N}$ are orthogonal matrix, and $S \in R^{M \times N}$

Is a diagonal matrix whose diagonal values are in descending order and other values equal zero as follows $S_1 \geq S_2 \geq \cdots \geq 0$

Finally, the covariance matrix computed as shown:

$$C = \frac{1}{(N-1)} X^T X \quad (3)$$

$$X^T X = U^T SV\, USV^T \quad (4)$$

There are various previous works exploited PCA algorithm as unsupervised dimension reduction. These works are elicited and summarized in table 1.

Table 1 PCA based approaches

PCA based Methodology	Elements	Details
ND-PCA; which is based on the matrix of covariance and the normal distribution linear additivity property (Wang et al., 2016)	**Problem scope:**	Structuring the observations in the PC space using the linear additivity property for normal distribution.
	Method features:	The method is capable of exploiting the variance information in the original data. It can obtain the analytical results rather than approximate results. Also, it has the ability to handle data of normal distribution form and other additive distributions.
Principal component	**Problem**	Concluding future ramp estimation from series of power forecast.

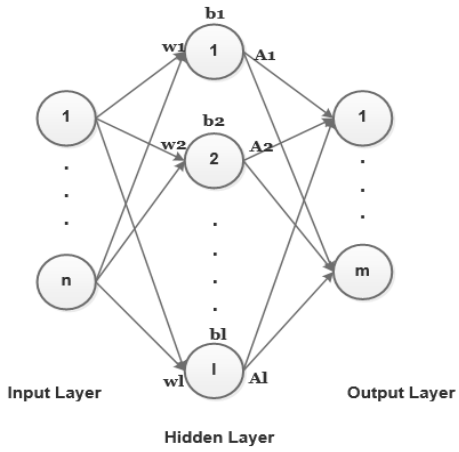
analysis of wind speed time series (Heckenbergerova et al., 2014)	**scope:**	
	Method features:	In the proposed method, numerical weather prediction (NWP) model is not required for producing wind forecasts. It can propose accurate forecasting of wind power.
	Future trends:	Other weather parameters will be resolved by linear regression models as statistical methods.
Kernel-based PCA for achieving efficient lower dimensional scheduling in variables linear parameter-varying (LPV) models. (Rizvi et al., 2016)	**Problem scope:**	Reducing the number of scheduling variables leads to a reduction of computational complexity of design and implementation of LPV controller.
	Method features:	Kernel-based PCA is desirable to obtain LPV models of interest in a rational form. It aims at extracting efficiently data components because of its capability to carry out extraction in a high dimensional feature space. The method is able to solve the problem of optimization and achieve an affine or rational representation in relation to variables of reduced scheduling.
Adaptive-PCA: novel data aggregation mechanism for WSNs (Poekaew and Champrasert, 2015)	**Problem scope:**	Data aggregation is required for reducing consumption of battery energy in sensor nodes.
	Method features:	In Adaptive-PCA, PCA is carried out dynamically using the change in data sensing. The dimensionality reduction of sensing data is used to decrease the data transmission. PCA data accuracy examiner assesses the accuracy of sensing data after sensor nodes performed the PCA procedure.
Sparse PCA modeling based on the inverse power method to gain the sparsity of PCA (Raihana et al., 2016)	**Problem scope:**	Principal components (PCs) can sometimes be complicated to explicated because PC represents the linear combinations of the variables.
	Method features:	Sparse principal component analysis (PCA) enables to address the complication problem. It is suitable to reduce the dimension of complex data. It is appropriate as feature extraction for big data since the accuracy rate is higher than input data to the classifier.
Convex Sparse Principal Component Analysis (CSPCA) applied to feature learning (CHANG et al., 2016)	**Problem scope:**	It is complicated to explicate the results of PCA. Also, the traditional PCA is vulnerable to specific noisy data.
	Method features:	Sparse model is considered as a valid measure for feature analysis. Combining robust PCA and the recent evolution of sparsity into integrated framework aims at leveraging the mutual benefit. The proposed method has a capability of mapping the invisible data during the training phase.
Parallel and distributed implementation of a hyperspectral principal component analysis (PCA) (Wu et al., 2015)	**Problem scope:**	In cloud computing environments, the development of dimensionality reduction methods can supply preprocessing of the data and efficient storage.
	Method features:	In the proposed method, a parallel model of map-reduce is used, taking full features of the high throughput access and high achievement capabilities of distributed computing in cloud computing environments. It aims at proposing the implementation

		of the PCA method on Spark platform for a cloud computing.
	Future trends:	Implementing different dimensionality reduction and classification methods in cloud computing environments.

3.2 Extreme Learning Machines (ELM)

Extreme learning machine suggested as highly fast learning method to single hidden layer feedforward neural network (SLFN) (Feng, Ong, and Lim, 2013; Feng et al., 2015) for classification and regression. The ELM contains three main characteristics; the first is that extremely learning speed treats real world classification problems and providing good accuracy rate (Huang, Ding, and Zhou, 2010). The second characteristic is randomly generating the hidden layer parameters without tuning or local minima, unlike conventional feedforward neural network, approaches such as back propagation (BP) learning algorithms to overcome local minimum problems and slow learning speed (Huang, Zhu, and Siew, 2006). The third characteristic is the output weights of an SLFN analytically calculated instead of using the standard gradient descent algorithm (Huang, Zhu, and Siew, 2004).
For these characteristics, ELM is considered as an effective learning algorithm without any learning iteration or control parameters as learning epochs; furthermore, ELM has a simple structure as shown in figure1.

Figure 1 ELM Structure



Standard ELM structure (Huang, Zhu, and Siew, 2006) requires: (1) input weights (randomly generated that connect the input layer to the hidden layer); (2) hidden layer biases (randomly assigned); (3) output weights (connect the hidden layer to output layer that are analytically computed using simple generalized inverse method "Moore-Penrose" (Serre, 2002).
Hidden layer executes any activation function depend on your purpose such as sigmoid, sine, radial basis, hard-limit, symmetric hard-limit, satlins, tan-sigmoid, triangular basis,

ridge polynomial, linear and positive linear, fuzzy inference, wavelet, etc. (Huang and Chen, 2007). The training set is supposed as:

$$\{(x_i, t_i) \mid x_i = (x_{i1}, x_{i2}, x_{in}) and\ t_i = (t_{i1}, t_{i2}, t_{im}),\ i = 1{:}N\}$$

Where N: number of instances, n: number of attributes and m: number of classes after randomly assigns the input weights hence, compute the hidden layer output matrix (H) and the output weights (A).

$$A = H^{\tau}T \ \ where\ H^{\tau} = (HH^{T})^{-1} \qquad (5)$$

Where H is the generalized pseudo inverse of $H^{\tau}T$ and T is a matrix containing a network target.

The actual output of ELM model for SLFN can be written as:

$$E = \sum_{i=1}^{l} A_i F(w_i, b_i, x_i) \qquad (6)$$

l is a number of hidden layers, F is the activation function and A are computed by using equation 5.

3.3 Common classification algorithms

In data mining, the classification was defined as supervised learning model that its objective to assign each object to its related class. (Gupte et al., 2014) Classification algorithm executed through two main phases. The first phase is the training phase to construct the model as classification rules from a training set within predefined classes. The second phase is using this model rules to classify the testing set and accuracy rate of the classifier that will determine based on the percentage of the patterns that correctly classified. There are most widely used classifiers, which are support vector machine (SVM), k-nearest neighbor (k-NN), decision tree (DT) and naïve bayes (NB). A short review to each classifier is taken in the following paragraphs.

Support Vector Machine Classifier is an algorithm, which is called the training data set support vectors due to its importance in the decision boundary (hyperplane) that maximize the separation margin between the two classes in binary classification (Parikh and Shah, 2016). SVM is effective to separate any classes that cannot separate linearly or in multi-classification task, so SVM is also kernel-based algorithm (Carmeli, De Vito, and Toigo, 2008), which transforms the input data space into a higher dimension space to separate between classes for nonlinear data using high-dimensional hyperplane, however it consumes a high computational power

K-Nearest Neighbor Classifier that its step learning algorithm is lazy because of its dependence on two parameters. The first parameter is a number of neighboring data points (k) (Dadhania and Dhobi, 2012; Baoli, SHiwen, and Qin, 2003); where the default value of k is 3 however, there are many techniques to choose the ideal values of k and the second parameter is a similarity function (Erkan et al., 2011), such as Euclidean distance that compares the training sample with test samples. Despite KNN can be effective for classification and regression due to its own simplicity, scalability, and good accuracy that achieved using its two parameters but has some restrictions such as high computation, memory requirements, low tolerance to noise, as well its lazy learner.

Decision Tree Classifier is a tree structure technique, which represents its classification rules as a decision tree that consists of the root node than attaching a set of child nodes to this node (Bose and Mahapatra, 2001; Chattamvelli, 2009). A decision tree is defined as Boolean function, which its inputs training samples (internal nodes within its own property test) and its output are the decision values ("yes, no" or class label) (Han and Kamber, 2006) corresponding to leaf nodes and when attributes test can follow a distinct branch. The decision tree is the most widely used method in classification (Azar and El-

Metwally, 2013) so; there are many decision tree types; such as C4.5, C5.0, ID3, CART, CN2 etc. Selection of attributes for internal nodes are different from each method (Dai, Zhang, and Wu, 2016) for CART is used to Gini-Index while ID3, C4.5, C5.0 uses to information gain concept. The classification accuracy of decision tree models can be improved by pruning the tree (Niuniu and Yuxun, 2010). Pruning the tree aims at removing all branches that represent noise or missing values so can overcome the overfitting of data, however, there are different levels of attributes and uncertainty with complex computation of data

Naïve Bayes Classifier is a simple classification technique that can be described as Naïve that refers to independence assumption such as class attributes or features and Bayes due to its dependence on Bayes rule (Miquelez, Bengoetxea, and Larranaga, 2004). The main idea of NB classifier is computing the posterior probabilities of classes (Dey et al., 2016; Efron, 2013), is combining a few of observed data to get the actual learning for fast classification and to able for handling streaming and discrete data but its only disadvantage that supposes independence of features.
Dimensionality reduction is preprocessing step that reduces the dimensions of data without losing. There are two reduction techniques (Vijayarani and Maria Sylviaa, 2016) feature extraction (FE) and feature selection (FS). For handling high dimensional data requires a higher memory requirements and consumption power so it is necessary to use any dimension reduction techniques to transforms data from high dimensional space to reduced dimensional space such as PCA, KPCA, ICA etc.

In this paper, principal component analysis (PCA) is focused as a step to improve the accuracy rate, but the proposed algorithm outperforms on all above classifiers with high accuracy as shown in comparison tables 5 and 6 at experimental results section.

4 PROPOSED APPROACH

4.1 Enhanced Extreme Learning Machines (EELM)

The difference between the new proposed EELM algorithm and the standard ELM algorithm lies in selection and implementation of the activation function, which it has a wonderful effect on the accuracy rate. EELM algorithm implements all basic computation functions of the standard ELM, in addition to the effective functions; which are used in proposed EELM algorithm. These functions, which are softmax function, softsig sigmoid function, and hyperbolic tangent is described in the following paragraphs.

Softmax function is used to minimize the cross-entropy or maximizes the log-likelihood (Tang, 2013) it is used as a standard function for classification problems in deep learning. That is useful to balance the output neurons of neural networks for classification model (Heaton, 2016) which is desired that the probabilities of each class sum to 1. Softmax function considers as a general form of the logistic function "sigmoid"(Bishop and Christopher, 2006). The main goal of softmax is squashing the arbitrary values of k dimensional vector (z) to the real values in the range (0, 1) while a sum of all outcomes equals 1(Wikipedia, 2017) and the equation 7 of softmax function written as:

$$\partial(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{K} e^{z_i}} \quad (7)$$

For j=1,…, K and K: number of categories

The output of function interpretable as posterior probabilities, so it used for representing categorical distribution that is useful for multi-class classification and multinomial logistic regression (Bishop and Christopher, 2006).But in this paper will focus on illustration EELM for just classification problems using the softmax function.

Softsig sigmoid function (Bergstra et al., 2009) is defined as:

$$f(x) = \frac{x}{1+|x|} \quad in\ range\ (-1,1) \quad (8)$$

Hyperbolic tangent sigmoid function is similar to softsig function (Vogl et al., 1988) but (exponential instead to polynomial) that is given as:

$$f(x) = \tanh(x) = 2g(2x) - 1 \quad in\ range\ (-1,1) \quad (9)$$

Where $$g(x) = \frac{e^x}{1+e^x} \quad (10)$$

The pseudo-code of the proposed EELM algorithm will be described in table 2.

Table 2 Pseudo code for EELM

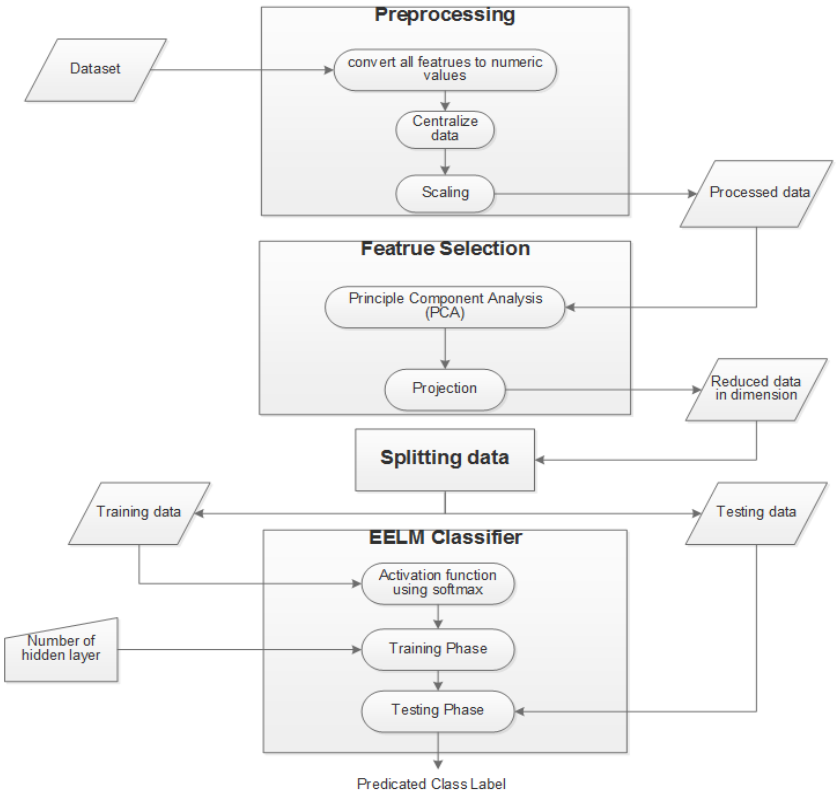
The pseudo-code of EELM
Input: data matrix(X), number of hidden nodes (l), activation function of hidden layer (F)
Process: **step1: Apply cross-validation then splitting data (X) to training set (T) and testing set** **step2: Assign randomly hidden node parameters (*wi, bi*), i=1, …, l. It refers to weight and bias, where *wi* is the input weights between the input layer and the hidden layer and *bi* is the bias of the hidden layer.** **step3: Calculate the output matrix H of the hidden layer by using the properly selected activation function according to your purpose, but softmax is the basic function for classification** **step4: Calculate the output weight A: A=ginv(H) %*% M (matrix multiplication). M is the output of the training set (T), ginv(H) is the Moore-Penrose generalized inverse of the hidden layer for output matrix H.** **step5: After SLFN (single hidden layer feedforward neural network) training, calculate the output of test set E: E=H %*%A (matrix multiplication), where E is the prediction class label.**

Output: The prediction class label E.

4.2 Principle Component Analysis and Enhanced ELM (PCA-EELM)

A conceptual view of the proposed approach inspired by both of PCA and enhancement of ELM as shown previously. PCA-EELM is considered the actually proposed algorithm that improves EELM classifier by using a dimensionality reduction phase based on PCA, which is a powerful statistical technique, to identify the features in high-dimensional data by reducing the dimensions (Good et al., 2010). The main goal is performed by applying perfection of the SLFN in classification tasks. The proposed technique has proven to work satisfactorily in (binary - multi) class classification tasks through small and large data sets; as shown in section of the experimental results. Figure 2 presents the overall description of the PCA-EELM.

Figure 2 Flowchart of PCA-EELM for Classification



The proposed approach requires the preprocessing, Feature selection and splitting data phases. Preprocessing phase is requiring before applying EELM classifier to be in the adaptable format for easily deal with it. Firstly, converting all features of data to numeric values. Secondly, centralize data using the mean centering that calculates the average of each value and subtracted from the original data. Thirdly, scaling done by dividing the (centered) columns of data by their standard deviations. Feature selection phase which involves applying principle component analysis using singular vector decomposition to

select all features that have largest variance values as shown previously then projection stage which transforms data from high dimensional space to reduced dimensional space. Splitting data phase is necessary to obtain the data divided into training data and test data rate using cross validation.

The main objective of EELM classifier is learning of the training set using softmax function which is the best activation function for classification tasks when compared with other function as explained in EELM section. Finally, the testing phase that predicts the class label for unseen data (testing data) to able calculate accuracy rate based on correctly classified data

The integration between PCA and EELM is summarized in pseudo-code of PCA-EELM in table 3.

Table 3 Pseudo code for PCA-EELM

The pseudo-code of PCA- EELM
Input: Data matrix(X), Number of hidden nodes (l), Activation function of hidden layer (F)
Process: **Phase I: Preprocessing the data and dimensionality reduction using PCA** **step1: Preprocessing data matrix (X) (convert data to numeric values - scaling and centering)** **step2: Using svd decomposition calculate principle components** $X = USV^T$ **step3: Compute conversion matrix** $C = \frac{1}{(N-1)} X^T X$ **step4: Apply PCA transformation to get output matrix(P)** **Phase II: Enhancement the classification accuracy using EELM** **step5: Apply cross-validation then splitting data (P) to training set (T) and testing set** **step6: Assign randomly hidden node parameters (wi, bi), i=1, …, l. It refers to weight and bias, where wi is the input weights between the input layer and the hidden layer and bi is the bias of the hidden layer.** **step7: Calculate the output matrix H of the hidden layer using the proper activation function you are selected according to your purpose, but softmax is the**

basic function for the multi-classification

step8: Calculate the output weight A: A=ginv(H) %*% M (matrix multiplication).

M is the output of the training set (T), ginv(H) is the Moore-Penrose generalized inverse of hidden layer output matrix H.

step9: After SLFN (single hidden layer feedforward neural network) training, calculate the output of test set E: E=H %*%A (matrix multiplication), where E is the prediction class label.

Output: the prediction class label E.

4.3 Effectiveness of PCA-EELM for big data processing

Nowadays, the huge data is collected from various online sources in many fields to serve customers (Yadav, Wang, and Kumar, 2013). This data is considered so large and difficult to be facilely processed using traditional database management and software techniques (Hassanien et al., 2015); wherefore, dimensionality reduction is a fundamental data preprocessing phase for large-scale data. Dimensionality reduction can be used to overcome big data issues regarding their structure, analysis, management, and storage. There are different algorithms aims at dealing with this data, however; each algorithm has an effective result based on data characteristics such as volume, variety, velocity, and veracity. Hence, PCA is applied as a preprocessing phase to handle big data before data classification.

In this study, the effectiveness of the proposed PCA-EELM is proven by using several datasets, it achieves good accuracy rate with the lowest computation time according to traditional techniques results for the same dataset.

5 EXPERIMENTAL RESULTS

The proposed method and all traditional techniques implemented using R (3.3.2) on computer specifications of an Intel(R) Core(TM) i3 2.40 GHz CPU and 4.00GB RAM. Experimental parameters in the experiments are executed using the values in table 10.

5.1 Datasets

Extracting useful information from a large amount of data is considered critical task. The proposed approach is evaluated by applying it to sixteen datasets, which were collected from UCI machine learning repository (UCI Repository, 2017) and Rdatasets (Rdatasets, 2016). The selected sixteen datasets have several instances, attributes and included binary and multi-class problems. Many traditional classification techniques have tested on six datasets to evaluate our proposed algorithm and ten datasets to compare it with the previous work within the similar dataset, there are different sizes of the dataset which vary from 90 to 101766 instances, and the number of attributes ranged between 3 and 50. The used datasets within the previous work were partitioned by a hold-out cross-validation procedure with (1 / 4) n instances for the testing dataset and (3/4) n instances for the training dataset where n is the total number of instance and another datasets were

partitioned using 10-fold cross validation. As well, the instances within missing values have been ignored before execution of the algorithms across the datasets, all datasets considers supervised classification problems, the data description as follows in table 4.

Table 4 Description of dataset

Datasets	**# Observation**	**# Attributes**	**Source**
Cigar	1380	8	R datasets
Snmesp-1	5904	6	
letter-recognition	20000	17	UCI Repository
credit card clients	30001	25	R datasets
InstEval	73421	8	
Diabetic data	101766	50	UCI Repository
Hepatitis	155	19	
Heart	270	13	
Vote	435	16	
German	1000	25	
Yeast	1484	10	
Ecoli	336	7	
Haberman	306	3	
Ionos	351	34	
Post-Op	90	20	
Diabetes	822	8	
Hepatitis	155	19	
Heart	270	13	
Vote	435	16	
German	1000	25	

5.2 Evaluation Methodology and Measures

In order to get a more balanced approach, cross-validation procedure (Dietterich, 1985) is used for this experiment to overcome the overfitting problem. There are many evaluation schemes such as k-fold, hold out, and leave one out cross-validation. In k-fold cross-validation the initial data is split into k subsamples, each separate sub-sample was obtained as testing dataset, and the remains k-1 samples used to form a training set, then repeats this process using a random sub-sample to get the average results for k times, therefore, after pre-processing phase, training data was conducted using 10-fold cross validation for 30 times with search for high-performance parameters and observe the average result. In another hand, leave one out cross validation procedure was used with 3/4 from the total instances for the training dataset and 1/4 instances for testing phase while comparing the proposed approach with the previous work.

The proposed algorithm is evaluated by performing many performance measures such as accuracy (Han, Kamber, andPei 2012; Bradford, Kunz, and Kohavi, 1998), sensitivity (Witten, Frank and Hall, 2011; Breiman et al., 1984), specificity (Han, Kamber, andPei, 2012), f1 score (Asch, 2013), and MCC (Yuan, Cai, and Xiao, 2007). Their main formulations are as follows:

$$\text{Accuracy (ACC)} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \tag{11}$$

$$\text{Sensitivity or True Positive Rate (TPR)} = \frac{TP}{(TP+FN)} \tag{12}$$

$$\text{Specificity or True Negative Rate (TNR)} = \frac{TN}{(TN+FP)} \tag{13}$$

$$\text{F1 score} = \frac{2(TP)}{(2TP+FP+FN)} \tag{14}$$

$$\text{Matthews correlation coefficient (MCC)} = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{15}$$

Where TP is a true positive, TN is a true negative, FP is a false positive, and FN is a false negative. In particular, the proposal approach is compared to various types of classification techniques; such as support vector machine, naïve bayes, decision tree, k-nearest neighbor and standard ELM through six datasets using 10-fold cross validation. Also, PCA-EELM is evaluated by comparing it with the previous work.

5.3 Results and Discussion

In this section, experimental results presented the proposal approach, which is based on SLFN for classification (EELM) and the common dimension reduction technique (PCA). Tables 5 and 6 illustrate the comparison between different methods, which are used in terms of binary and multi-class classification problems with more detailed results to each classifier with applying PCA or without applying PCA. The first observation, the proposed approach performs performance measures better than standard ELM before and after applying dimensionality reduction technique over all datasets. The second observation, with the small dimensional data other classifiers achieved high accuracy measures with a small-time computation rather than the proposed approach, however in high dimensional data may be reached to high accuracy but with high computational power, contrary to the proposed approach that is outperforms all other methods effectively in high dimensional data with the lowest time computation as shown in table 5.

Table 5 Performance parameters calculated based on confusion matrix with 10-fold cross validation

Datasets	**Models**	**Without Dimensionality Reduction**				**Dimensionality reduction using PCA**			
		Accuracy (%)	Time(s)	Sensitivity (%)	Specificity (%)	Accuracy (%)	Time(s)	Sensitivity (%)	Specificity (%)

Cigar	**SVM**	86.23	2.31	86.23	99.694	93.69	1.6	93.69	99.85
Snmesp-1		33.29	20.53	33.29	90.47	96.65	11.42	96.84	97.68
Letter-recognition		97.60	415.17	97.58	99.90	98.4	230.32	98.33	99.93
Credit card clients		99.94	639.37	87.25	99.98	99.94	370.51	87.25	99.98
InstEval		25.43	301.35	21.65	80.48	97.87	111.48	97.65	97.99
Diabetic_data		98.72	21973.17	99.84	98.99	99.20	1082.25	99.11	99.32
Cigar	**NB**	84.71	3.54	84.7	99.66	99.58	2.58	99.85	99.99
Snmesp-1		15.955	13.46	15.955	87.993	98.54	9.56	98.52	99.93
Letter-recognition		94.72	42.73	94.72	99.78	98.05	28.73	98.05	99.92
Credit card clients		84.297	55.06	40.666	85.516	98.76	33.68	98.73	99.99
InstEval		24.04	78.89	20.5	80.14	96.68	59.7	97.77	98.16
Diabetic_data		98.17	500.57	97.92	98.76	99.45	72.88	99.24	99.99
Cigar	**DT**	99.98	0. 72	99.98	99.99	100	0.67	100	100
Snmesp-1		15.22	0.35	15.226	87.889	93.54	0.22	94.78	94.97
Letter-recognition		95.68	8.87	95.69	96.58	96.96	1.31	96.88	97.46
Credit card clients		53.47	10.75	16.57	89.75	99.99	5.24	87.50	99.99
InstEval		23.32	17.9	19.49	79.85	59.3	9.80	60	89.8
Diabetic_data		98.97	57.06	98.76	98.99	99.50	14.16	99.48	99.62
Cigar	**KNN**	78.18	0.72	78.98	99.51	98.89	0.59	98.99	99.65
Snmesp-1		37.12	2.14	38.53	91.01	97.83	1.3	97.83	99.69
Letter-recognition		98.14	37.68	97.99	99.92	98.3	18.69	98.02	99.92
Credit card clients		66.76	179.94	67.81	93.24	98.38	108.16	80.34	90.66
InstEval		53.26	255.52	52.51	88.19	59.31	149.66	59.99	89.81
Diabetic_data		64.84	7539.16	65.90	86.45	64.88	208.60	55.73	86.42
Cigar	**ELM**	3.01	0.39	3.02	97.87	6.52	0.25	6.72	93.38
Snmesp-1		9.99	0.53	12.20	90.28	12.26	0.44	12.88	88.24
Letter-recognition		8.97	1.92	9.28	96.44	12.97	1.15	13.98	96.68
Credit card clients		39.41	4.41	12.54	87.61	40.56	2.56	12.71	87.79

InstEval		39.84	1.18	12.14	87.97	24.04	0.9	20.05	80.01
Diabetic_data		45.53	52.78	24	76	46.41	16.73	25.11	75.08
Cigar	**EELM**	65.31	1.31	67.23	99.67	98.99	1.01	98.78	99.85
Snmesp-1		99.92	1.73	99.88	99.97	99.94	1.12	99.89	99.99
Letter-recognition		21.02	2.51	22.62	98.28	99.03	1.92	99.64	99.85
Credit card clients		39.52	2.75	12.08	87.96	98.54	2.04	98.62	98.99
InstEval		97.45	2.29	97.77	98.20	98.99	2.00	99.81	99.95
Diabetic_data		46.54	28.89	26.38	76.12	96.28	6.21	95.88	99.89

Table 6 Other performance parameters

Datasets	**Models**	**Without Dimensionality Reduction**			**Dimensionality reduction using PCA**		
		MCC	AUC	F1	MCC	AUC	F1
Cigar	**SVM**	0.8628	0.9067	85.80	0.93760	0.9344	93.67
Snmesp-1		0.3622	0.5661	32.89	0.9681	0.9673	97.46
Letter-recognition		0.9751	0.9894	97.59	0.9834	0.9973	98.39
Credit card clients		0.8733	0.8911	87.34	0.9786	0.9681	98.34
InstEval		0.2037	0.5157	15.77	0.9997	0.9999	99.98
Diabetic_data		0.9888	0.9848	99.95	0.9944	0.9906	98.35
Cigar	**NB**	0.8343	0.8144	82.85	0.9985	0.9947	99.58
Snmesp-1		0.1633	0.4987	10.29	0.9948	0.9932	99.34
Letter-recognition		0.94629	0.8952	94.7410	0.9798	0.9572	98.05
Credit card clients		0.2018	0.3558	7.20938	0.9804	0.9681	97.03
InstEval		0.2085	0.5091	24.993	0.9623	0.9766	97.14
Diabetic_data		0.9846	0.991	98.756	0.9931	0.9899	98.22
Cigar	**DT**	0.9987	0.9991	99.95	0.9999	0.9997	99.99
Snmesp-1		0.1525	0.5461	9.93	0.9854	0.9826	99.46
Letter-recognition		0.9782	0.9844	98.77	0.9831	0.9882	99.20
Credit card clients		0.5980	0.6404	56.32	0.8749	0.9821	87.49
InstEval		0.1005	0.4814	10.51	0.7013	0.7598	60

Diabetic_data		0.9989	0.9988	99.99	0.9999	0.9999	99.99
Cigar	**KNN**	0.7707	0.7667	77.14	0.9837	0.9885	98.68
Snmesp-1		0.4037	0.5438	38.25	0.9755	0.9873	97.69
Letter-recognition		0.9814	0.9894	98.20	0.9878	0.9908	98.39
Credit card clients		0.3668	0.5333	38.12	0.8254	0.9599	82.72
InstEval		0.5202	0.5134	52.62	0.7012	0.5999	59.99
Diabetic_data		0.5341	0.5944	57.12	0.5372	0.5991	57.20
Cigar	**ELM**	0.0143	0.1333	1.33	0.0357	0.1850	3.50
Snmesp-1		0.0794	0.1976	4.98	0.1264	0.212	7.12
Letter-recognition		0.1123	0.3773	8.17	0.1528	0.4228	12.68
Credit card clients		0.2869	0.3597	11.76	0.2889	0.4006	11.96
InstEval		0.11389	0.4997	10.72	0.1222	0.5367	17.95
Diabetic_data		0.1325	0.4541	12.98	0.1548	0.5000	16.60
Cigar	**EELM**	0.6716	0.6577	66.72	0.9878	0.9799	98.88
Snmesp-1		0.9988	0.9854	98.85	0.9989	0.9882	98.99
Letter-recognition		0.2286	0.3940	21.96	0.9987	0.9947	99.64
Credit card clients		0.1018	0.2331	10.21	0.9865	0.9765	97.52
InstEval		0.9785	0.9744	97.99	0.9812	0.9889	99.10
Diabetic_data		0.2695	0.4970	25.01	0.9635	0.9699	95.83

The complementary strength between PCA and EELM cannot be achieved when merging PCA with any classifier techniques, therefore in all cases PCA with EELM yields the highest accuracy and the smallest time more than PCA with any other classifier as shown in table 7 that summarizes the mean percentage of performance measures that are calculated for each classification algorithm according to a given dataset by averaging computation time, accuracy, sensitivity, specificity, f1scores, area under ROC curve and Matthews's correlation coefficient. These results, which are based on the proposed approach PCA-EELM classifier, achieve the highest accuracy rate average up to 98.62 % and the lowest average computation time up to 2.38 (s). The effectiveness of PCA-EELM was ensured by calculating the average of TPR, TNR and F1 scores which are 98.77%, 99.75%, 98.32% respectively; in addition, average values for MCC and AUC achieved 0.9861 and 0.9830 respectively. Figures 3, 4, and 5 display the plots of all used performance measures of the experiments. Figure 3 shows mean computation time to each technique across six benchmark data sets, so PCA-EELM has the lowest mean computation time according to other techniques. It can be observed from figures

4 and 5 that the highest values for all performance parameters were owned to the proposed approach.

Model	Mean Performance Values						
	Accuracy (%)	**Time(s)**	**Sensitivity (%)**	**Specificity (%)**	**MCC**	**AUC**	**F1 (%)**
SVM	73.53	3891.98	70.97	94.91	0.7109	0.8089	69.89
PCA-SVM	97.62	301.26	95.47	99.12	0.9769	0.9762	97.69
NB	66.98	115.70	59.07	91.97	0.5564	0.6773	53.13
PCA-NB	98.50	34.52	98.69	99.66	0.9848	0.9799	98.22
DT	64.44	15.82	57.61	92.17	0.6378	0.7750	62.57
PCA-DT	91.54	5.23	89.77	96.97	0.9240	0.9520	91.02
KNN	66.38	1335.86	66.95	93.05	0.5961	0.6568	60.24
PCA-KNN	86.26	81.16	81.81	94.35	0.8351	0.8542	82.44
ELM	24.45	10.20	12.19	89.36	0.1232	0.3369	8.32
PCA-ELM	16.05	3.67	15.24	86.86	0.1468	0.3761	11.63
EELM	61.62	6.58	54.32	93.36	0.5414	0.6236	53.45
PCA-EELM	**98.62**	**2.38**	**98.77**	**99.75**	**0.9861**	**0.9830**	**98.32**

Table 7 The mean performance values for multi-classifiers within and without dimensionality reduction through common six datasets.

Figure 3 Mean of Computation Time for all classifiers

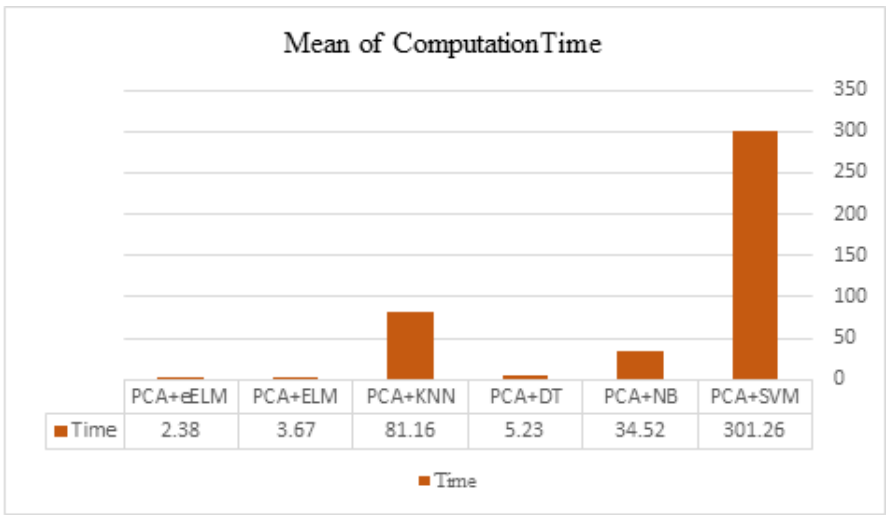
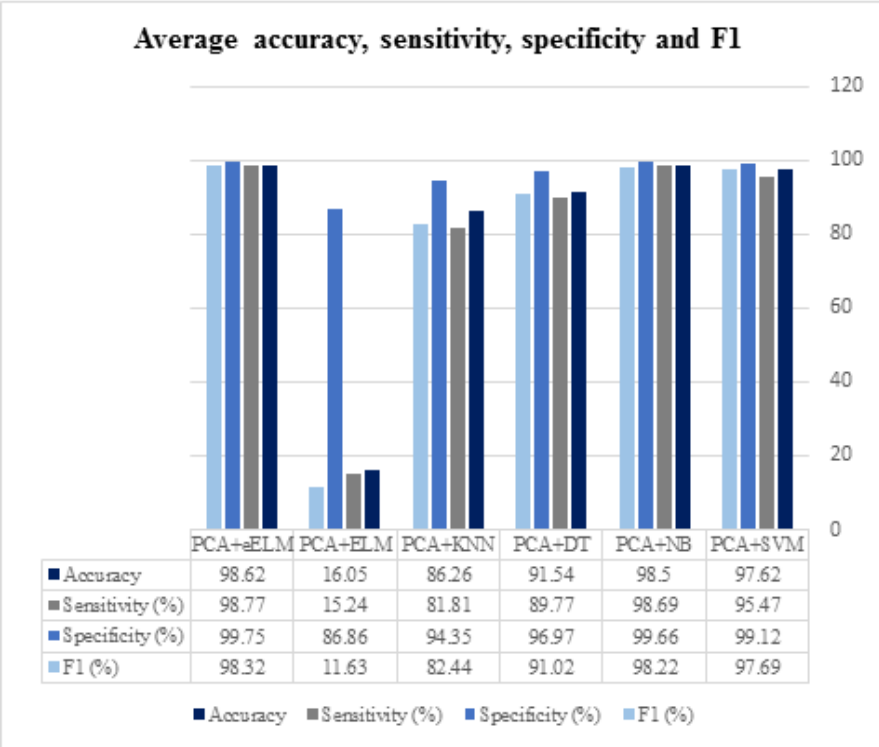
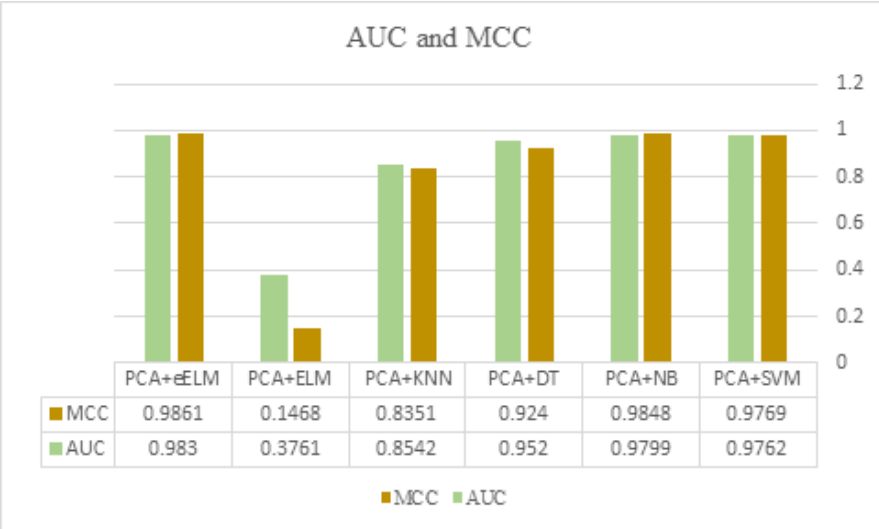


Figure 4 Average of Accuracy, Sensitivity, Specificity, and F1 scores for all classifiers



	PCA+eELM	PCA+ELM	PCA+KNN	PCA+DT	PCA+NB	PCA+SVM
Accuracy	98.62	16.05	86.26	91.54	98.5	97.62
Sensitivity (%)	98.77	15.24	81.81	89.77	98.69	95.47
Specificity (%)	99.75	86.86	94.35	96.97	99.66	99.12
F1 (%)	98.32	11.63	82.44	91.02	98.22	97.69

Figure 5 Average of AUC and MCC for all classifiers



	PCA+eELM	PCA+ELM	PCA+KNN	PCA+DT	PCA+NB	PCA+SVM
MCC	0.9861	0.1468	0.8351	0.924	0.9848	0.9769
AUC	0.983	0.3761	0.8542	0.952	0.9799	0.9762

Moreover, the performance of PCA-EELM is proved by assessing it against PCA-ELM and LDA-PCA-ELM in the previous works (Castaño, Fernández-Navarro, and Hervás-Martínez, 2013; Castaño et al., 2016) respectively. Table 8 indicates that proposed PCA-

EELM improves the accuracy on the previous work of PCA-ELM with minimum number of the hidden nodes over ten benchmark datasets, so that PCA-EELM considers competitive approach if it compared to the common technique (PCA-ELM), which was presented in (Castaño, Fernández-Navarro and Hervás-Martínez, 2013).

Table 8 Comparison between previous work and proposed algorithm

Dataset	**(PCA-ELM)**		**Proposed Approach (PCA-EELM)**	
	NHN	**Accuracy**	**NHN**	**Accuracy**
Hepatitis	12	79.487	10	89.30
Heart	8	77.941	10	83.08
Vote	11	92.885	10	94.36
German	28	76.000	10	69.96
Yeast	7	51.482	10	89.53
Ecoli	5	87.882	10	98.90
Haberman	3	76.315	10	97.23
Ionos	17	86.363	10	98.14
Post-Op	9	81.818	10	95.94
Diabetes	6	72.875	10	99.18
Mean	**10.6**	**78.304**	**10**	**91.562**

Finally, by comparing with the proposed approach and the previous work, it is observed that prediction results of PCA-ELM that visualized in figure 6 is very poor, however the sample of prediction results for PCA-EELM which visualized in figure 7 is very good because softmax function interprets The output as posterior probabilities, so it used for representing categorical distribution that is useful for multi-class classification and multinomial logistic regression, so these figures demonstrate the difference of results between before and after the enhancement of ELM, which illustrates the classification ability of PCA-EELM, is higher than PCA-ELM.

Figure 6 PCA-ELM Predications

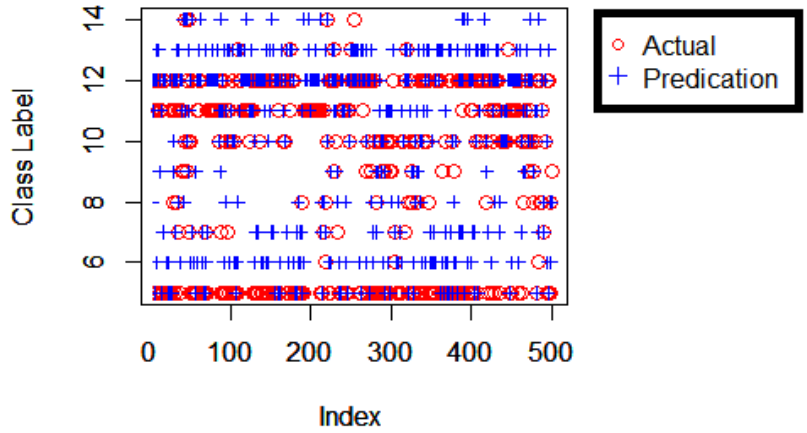


Figure 7 PCA-EELM Predications

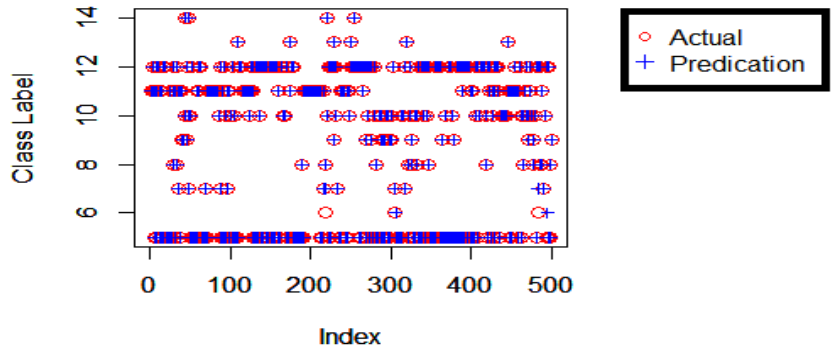


Table 9 shows the results of an improvement version of the previous work of (LDA-PCA-ELM) (Castaño et al., 2016). Our main contribution to improve LDA-PCA-ELM algorithm is using the enhanced ELM instead of the standard ELM to overcome the weakness of overall accuracy for LDA-PCA-ELM with minimum number of the hidden nodes as presented in detail to each data set in the table 9. LDA-PCA-ELM method was applied to fifteen datasets and the result is based on the average values for all data sets.

Table 9 Improvement to the previous work (LDA-PCA-ELM)

Dataset	**(LDA-PCA-ELM)**	**(LDA-PCA-EELM)**
	Accuracy	**Accuracy**
Hepatitis	88.83	97.63
Heart	85.75	98.64
Vote	94.65	97.99
German	77.94	70.25
Yeast	67.65	91.83
Ecoli	95.46	99.10
Haberman	87.21	98.81
Ionos	89.75	98.95
Post-Op	91.83	96.93
Diabetes	90.62	99.52
Mean	**86.966**	**94.965**

These results show that the integration of PCA and EELM outperforms any combination between PCA and any traditional classifier. The results prove that PCA-EELM effectively reduces the noise and irrelevant data from the original dataset as much as without losing the basic information in addition to improvement accuracy rate for the majority of classification techniques. From comparison results that were graphically represented in Figures 8, 9 have shown that time computation of standard extreme learning machine and enhancement of extreme learning machine have the lowest time because they based on single hidden layer for feedforward neural network with randomly assign hidden nodes parameters as represented in figure 8, but in figure 9 there are many algorithms that have achieved high accuracy in some cases such as Naïve base and decision tree while enhanced ELM achieved high accuracy in all cases that were used in experimental results.

Figure 8 Time comparison for all classifiers

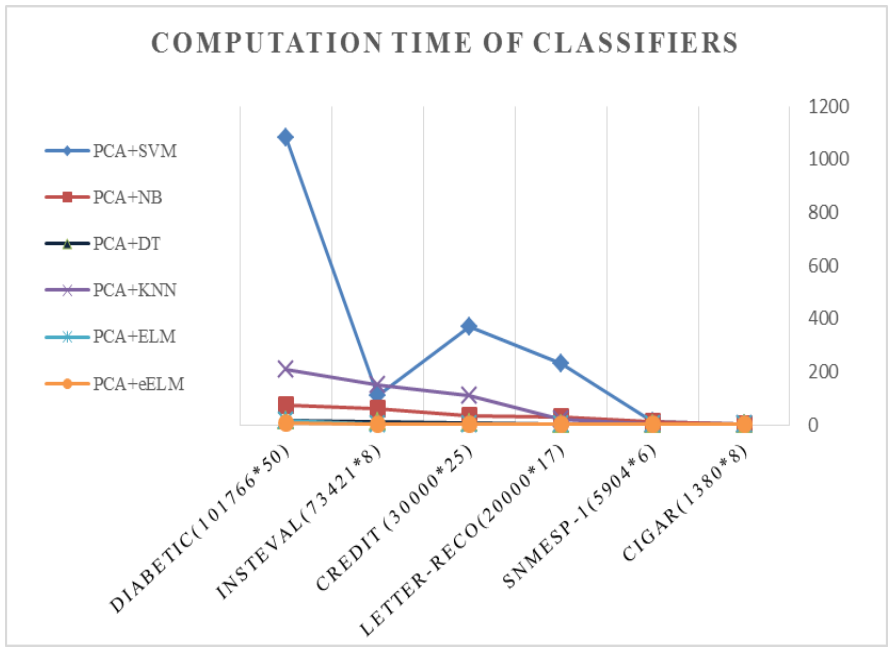
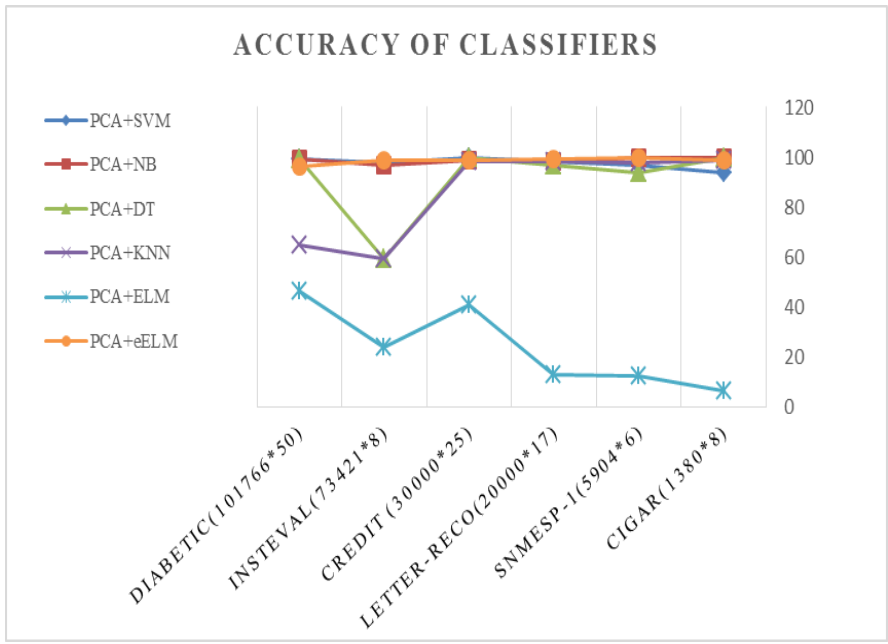


Figure 9 Accuracy comparisons for all classifiers



6 CONCLUSIONS AND FUTURE WORK

In this paper, the hybrid approach named PCA-EELM is proposed for enhancement data processing tasks. In addition to EELM can perform the experiments with low computational time and high accuracy rate, it can achieve high-speed learning using a single hidden layer for feedforward neural network with randomly assign hidden nodes parameters (weights and bias). PCA for dimensionality reduction has been applied in the pre-processing phase for mining high-dimensional data. Notably, PCA is the most fundamental method for reducing the high dimensional of linear data. The proposed PCA-EELM can be extended to deal with several problems such as nonlinear, dynamic process. In the experiments, EELM is compared with several typical classification methods through sixteen benchmark datasets in the classification issues. Experimental results show that EELM is a competitive approach in the classification performance. The basic objective is suggesting a universal approach which can be employed efficiently and effectively in various fields. The fundamental limitation of the proposed approach is that all experimental result applied on supervised classification problems, what about regression and unsupervised problems

In the future, PCA-EELM approach can be improved in the following two ways. In the first way, PCA can be improved to have the ability to deal with the nonlinear dimensionality reduction problem, by combining the kernel technique with PCA. In the second way, the parallel computing can be used, because the big data or large-scale data problems cannot be solved using conventional approaches such as SVM, NB, K-NN, and DT. Therefore, Enhanced ELM is intended due of its excellent generalization performance that reported from the results and its ability to withstand the computational complexity of ELM in SLFN with a large number of hidden layers. So, it is useful to solve very large complex dataset problems, because the matrices of the hidden layer will be too large for big data problems, which make the computation time impossible. Therefore, parallel computation with MapReduce framework can be applied to overcome this issue. In the third way, design optimization technique to improve EELM network parameters (bias, weight, number of hidden layer) or test EELM on multi-layer feedforwad neural network.

REFERENCES

Akbar, A., Ali, S., Amira, A., Bensaali, F., Benammar, M., Hassan, M., and Bermak, A. (2016). An Empirical Study for PCA and LDA Based Feature Reduction for Gas Identification. *IEEE Sensors Journal*, Vol. 16, No. 14, pp. 5734 – 5746**.**

Azar, A.T., and El-Metwally, S.M. (2013). Decision Tree Classifiers for Automated Medical Diagnosis. *Neural Computing and Applications, Springer,* Vol. 23, Nos. 7-8, pp. 2387-2403. DOI: 10.1007/s00521-012-1196-7.

Azar, A.T., and Hassanien, A.E. (2014). Dimensionality Reduction of Medical Big Data Using Neural-Fuzzy Classifier. *Soft computing, Springer,* Vol. 19, No. 4, pp. 1115-1127. DOI 10.1007/s00500-014-1327-4

Baoli L., SHiwen, Y., and Qin, L. (2003). An Improved k-Nearest Neighbor Algorithm for Text Categorization. *In: Proc. of the 20th International Conference on Computer Processing of Oriental Languages*, *Shenyang, China*, pp. 1–7.

Bengio, Y. (2009). Learning deep architectures for AI. Foundations and Trends in Machine Learning, Vol. 2, No. 1, pp. 1–127. DOI: 10.1561/2200000006.

Bishop, F.R., and Christopher, M. (2006). Pattern Recognition and Machine Learning, Linear Models for Classification, Springer, pp.179-210.

Bose, I., and Mahapatra, R.K. (2001). Business data mining - a machine learning perspective Information and Management, Vol. 39, pp. 211-225.

Bradford, J.P., Kunz, C., and Kohavi, R. (1998). Pruning decision trees with misclassification costs. *In: Proceedings of the 10th European conference on machine learning, Chemnitz, Germany*, pp. 131–136.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). Classification and regression trees, Wadsworth and Brooks, CA. Since 1993 this book has been published by Chapman and Hall, New York.

Cao, J., Chen, T., & Fan, J. (2016). Landmark recognition with compact bow histogram and ensemble ELM. Multimedia Tools and Applications, Vol. 75, pp. 2839–2857.

Cao, J., Chen, T., Fan, J. (2014). Fast online learning algorithm for landmark recognition based on bow framework: IEEE Conference on Industrial and Electronics Applications.

Cao, J., Zhang, K., Luo, M., Yin, C., and Lai, X. (2016). Extreme learning machine and adaptive sparse representation for image classification. Neural Networks 81, Elsevier Ltd, pp. 91–102.

Carmeli, C., De Vito, E., and Toigo, A. (2008). Reproducing Kernel Hilbert Spaces and Mercer theorem. v1, *eprint arXiv 0807.1659*. Doi: math/0504071.

Castaño, A., Fernández-Navarro, F., and Hervás-Martínez, C. (2013). PCA-ELM: A Robust and Pruned Extreme Learning Machine Approach Based on Principal Component Analysis, *Neural Process Lett* 37:377–392, DOI: 10.1007/s11063-012-9253-x.

Castano, A., Navarro, F., Riccardi, A., and Martınez, C. (2016). Enforcement of the principal component analysis–extreme learning machine algorithm by linear discriminant analysis. *Neural Computing and Applications, Springer,* 27, pp. 1749–1760. DOI: 10.1007/s00521-015-1974-0.

CHANG, X., NIE, F., YANG, Y., ZHANG, CH., and HUANG, H. (2016). Convex Sparse PCA for Unsupervised Feature Learning. *ACM Transactions on Knowledge Discovery from Data*, Vol. 11, No. 1, Article 3, pp. 1-16.

Chattamvelli, R. (2009). Data Mining Methods. Oxford, UK: *Alpha Science International Ltd, pp.* 185-264.

Dadhania, S., and Dhobi, J. (2012). Improved k-NN Algorithm by Optimizing Cross-validation: *International Journal of Engineering Research and Technology*, Vol. 1, No. 3, pp. 1–6.

Dai, Q., Zhang, C., and Wu, H. (2016). Research of Decision Tree Classification Algorithm in Data Mining: *International Journal of Database Theory and Application*, Vol. 9, No. 5, pp. 1-8.

Dey, L., Chakraborty, S., Biswas, A., Bose, B., and Tiwari, S. (2016). Information Engineering and Electronic Business: Sentiment Analysis of Review Datasets Using Naïve Bayes'and K-NN Classifier, (4), pp. 54-62.

Dietterich, T. (1985). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation* 10(7).

Efron, B. Mathematics. (2013). Bayes' theorem in the 21st century. Science .340:1177-8. Doi:10.1126/science.1236536.

Erkan, G., Hassan, A., Diao, Q., and Radev, D. (2011). Improved Nearest Neighbor Methods for Text Classification. Technical Report CSE-TR-576-11. DOI: 10.14311/NNW.2016.26.003.

Escabias, M., Aguilera, A., and Valderrama, M. (2004). Principal component estimation of functional logistic regression: discussion of two different approaches. *J Nonparametric Stat*, Vol. 16, Nos. 3-4, pp. 365–384.

Feng, G., Huang, G.B., Lin, Q., and Gay, R. (2009). Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Trans Neural Network*, Vol. 20, No. 8, pp. 1352–1357.

Feng, L., Ong, Y. S., Lim, M.-H., and Tsang, I. (2015). Memetic search with interdomain learning: A realization between CVRP and CARP. *IEEE Transactions on Evolutionary Computation*, Vol. 19, No. 5, pp. 644–658.

Feng, L., Ong, Y.S., and Lim, M.-H. (2013). ELM-guided memetic computation for Fodor, I. K. (2002). A survey of dimension reduction techniques. *Neoplasia*, Vol. 7, No. 5, pp. 475–485.

Fill, J.A., and Fishkind, D.E. (1998). The Moore Penrose Generalized Inverse for Sums of Matrices, Vol. 21, No. 2. DOI: 10.1137/S0895479897329692. 2-5

Fouad, K., and Dawood, M. (2016). Adaptive Optimized Clustering for Veterans' Administration Lung Cancer. *8th Cairo International Biomedical Engineering Conference (CIBEC 2016). IEEE.* DOI: 10.1109/CIBEC.2016.7836127.

Gupte, A., Joshi, S., Gadgul, P., and Kadam, A. (2014). Comparative Study of Classification Algorithms used in Sentiment Analysis: *International Journal of Computer Science and Information Technologies*, Vol. 5, No. 5, pp. 6261-6264.

Han, J., and Kamber, M. (2006). Data mining Concepts and Techniques. Elsevier, (2nd ed., pp. 285-378).

Han, J., Kamber, M., and Pei, J. (2012). Data Mining Concepts and Techniques, *Elsevier Inc*, 3rd ed., pp. 327-383.

Hassanien, A.E., Azar, A.T., Snasel, V., Kacprzyk, J., and Abawajy, J.H. (2015). Big Data in Complex Systems: Challenges and Opportunities, Studies in Big Data, *Springer-Verlag* GmbH Berlin/Heidelberg. ISBN 978-3-319-11055-4, (9).

Heaton, J. (2016). Regression and Classification: Predictive Analytics and Futurism, (13),pp. 40-41

Heckenbergerov, J., Musilek, P., Marek, J., and Rodway, J. (2014). Principal Component Analysis for Evaluation of Wind Ramp Event Probability. *Electrical Power and Energy Conference, IEEE.*

Hosoya, A., and Hyvarinen, A. (2016). Learning Visual Spatial Pooling by Strong PCA Dimension Reduction: *ACM Journal of Neural Computation*, Vol. 28, No. 7, pp. 1249-1264.

Huang, G.B. (2013). Extreme Learning Machine, Springer.

Huang, G.B. (2014). An insight into extreme learning machines: random neurons, random features and kernels, Cogn.Comput. Vol. 6, No. 3, pp. 3376-390.

Huang, G.B., and Chen, L. (2007). Convex incremental extreme learning machine, Neurocomputing70 (16–18), pp. 3056–3062.

Huang, G.B., Chen, L., and Siew, C. (2006). Universal approximation using incremental constructive feedforward networks with random hidden nodes, *Neural Network 17*, pp. 879-892.

Huang, G.B., Ding, X., and Zhou, H. (2010). Optimization method based extreme learning machine for classification, *Neurocomputing74* (1), pp. 155–163.

Huang, G.B., Li., M.B., Chen, L., and Siew, C. (2008). Incremental extreme learning machine with fully complex hidden nodes, Neurocomputing71, pp. 576-583.

Huang, G.B., Wang, D., Lan, Y. (2011). Extreme learning machine: a survey, Int.J.Mach. Learn. Cybern, Vol. 2, No. 2, pp. 107-122.

Huang, G.B., Zhu, Q.Y., and Siew, C.K. (2006). Extreme learning machine: Theory and applications, Neurocomputing70 (1-3), pp. 489–501.

Huang, G.B., Zhu, Q.Y., and Siew, CK. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks: *IEEE International Conference of Neural Network*, Conf. Proc 2, pp. 985–990.

Huang, T., Sethu, H., and Kandasamy, N. (2016). A New Approach to Dimensionality Reduction for Anomaly Detection in Data Traffic. *IEEE transactions on network and service management*, Vol. 13, No. 3, pp. 651-665.

Huo, X. M., and Smith, A. K. (2008). A survey of manifold-based learning methods: In Mining of Enterprise Data, pp. 691–745.

Iosifidis, A. (2015). Extreme learning machine based supervised subspace learning, Neurocomputing167, pp. 158-164.

Iosifidis, A., Gabbouj, M. (2015). On the kernel extreme learning machine speedup, PatternRecogn.Lett.68, pp. 205–210.

Iosifidis, A., Tefas, A. Pitas, I. (2014). Minimum class variance extreme learning machine for human action recognition, IEEETrans.CircuitsSyst.VideoTechnol, Vol. 23,No. 11, pp. 1968-1979.

Iosifidis, A., Tefas, A., Pitas, I. (2015). Drop ELM: fast neural network regularization with dropout and dropconnect, Neurocomputing162, pp. 57-66.

Jezowicz, T., Gajdos, P., Uher, V., and Snasel, V. (2015). Classification with Extreme Learning Machine on GPU: *IEEE International Conference on Intelligent Networking and Collaborative Systems*, pp.116-122.

Jolliffe, IT. (2002). Principle components analysis: Springer-Verlag, New York, 2nd ed, pp. 29-43. DOI: 10.1007/b98835.

Krishnasamy, G., and Paramesran, R. (2016). Hessian semi-supervised extreme learning machine. Neurocomputing207, pp. 560–567.

Li, S., You, Z.-H., Guo, H., Luo, X., & Zhao, Z.-Q. (2016). Inverse-free extreme learning machine with optimal information updating. IEEE Transactions on Cybernetics,Vol. 46, No. 5, pp. 1229–1241.

Lohr, S. (2008). The age of big data. New York Times, Vol. 16, No. 4, pp. 10–15.

López-Fandiño, J., Quesada-Barriuso, P. Heras, D., and Argüello, F. (2014). Efficient ELM-Based Techniques for the Classification of Hyperspectral Remote Sensing Images on Commodity GPUs: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, Vol. 8, No. 6, pp. 2885-2891. DOI. 10.1109/JSTARS.2014.2384133.

MacAusland, R. (2014). The Moore-Penrose Inverse and Least Squares. University of Puget Sound MATH 420: *Advanced Topics in Linear Algebra*, pp. 2-8

Miquelez, T., Bengoetxea, E., and Larranaga, P. (2004). Evolutionary Computation based on Bayesian Classifier, Vol. 14, No. 3. pp. 335 –349

Murthy, H., and Meenakshi, M. (2014). Dimensionality Reduction Using Neuro-Genetic Approach for Early Prediction of Coronary Heart Disease: Proceedings of International Conference on Circuits, Communication, Control and Computing (I4C 2014) *IEEE*, pp. 329 - 332

Niuniu, X., and Yuxun, L. (2010). Review of Decision Trees. *3rd IEEE International Conference on Computer science and information technology (ICCSIT), Vol.* 5, pp.105-109.

Omer, A., and Khurran, A. (2015). Facial Recognition using Principal Component Analysis based Dimensionality Reduction: IEEE International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering, pp. 434-439. DOI: 10.1109/ICCNEEE.2015.7381408.

Parikh, K., and Shah, T. (2016). Support Vector Machine – a Large Margin Classifier to Diagnose Skin Illnesses: 3rd International Conference on Innovations in Automation and Mechatronics Engineering, Procedia Technology, Vol. 23, pp. 369 – 375, pp. 853–862.

Poekaew, P., and Champrasert, P. (2015). Adaptive-PCA: An Event-Based Data Aggregation Using Principal Component Analysis for WSNs: *IEEE International Conference on Smart Sensors and Application (ICSSA)*, pp. 50-55 . DOI:10.1109/ICSSA.2015.7322509.

Raeder, T., Dalessandro, B., and Provost, F. (2013). Scalable Supervised Dimensionality Reduction Using Clustering: *19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1213-1221.

Rdatasets. (2016). Retrieved from https://vincentarelbundock.github.io/rdatasets/datasets.html . (Accessed in Dec 2016)

Rizvi, S., Mohammadpour, J., Tóth, R., and Meskin, N. (2016). A Kernel-Based PCA Approach to Model Reduction of Linear Parameter-Varying Systems. *IEEE Transactions on Control Systems Technology*, Vol, 24, pp. 1883-1891

Sarveniazi, A. (2014). An actual survey of dimensionality reduction. *American Journal of Computational Mathematics*, Vol. 4, No. 2, pp.55–72.

Serre, D. (2002). Matrices: Theory and Applications. *Springer Verlag, New York Inc*, pp. 109-275.

Sharma, N., and Saroha, K. (2015). A Novel Dimensionality Reduction Method for Cancer Dataset using PCA and Feature Ranking. 978-1-4799-8792-4/15

Singh, L., Chetty, G., and Sharma, D. (2012). A Novel Approach to Protein Structure Prediction Using PCA or LDA Based Extreme Learning Machines. ICONIP 2012, Part IV, LNCS 7666, *Springer-Verlag Berlin Heidelberg*, pp. 492–499.

Smith, L. I. (2002). A tutorial on principal components analysis. Cornell University.2-21

Tang, Y. (2013). Deep Learning using Linear Support Vector Machines: *International Conference on Machine Learning 2013*: Challenges in Representation Learning Workshop. Atlanta, Georgia, USA, pp. 1-5.

Tiwari, A., and Leng, Q. (2015). On the Construction of Extreme Learning Machine for Online and Offline One Class Classification-An Expanded Toolbox. Indian *Institute of Technology Indore Microsoft ATC*.

UCI Repository. (2017). Retrieved from https://archive.ics.uci.edu/ml/datasets.html . (Accessed Jan 2017)

Van Asch, V. (2013). Macro- and micro-averaged evaluation measures, pp. 1-14.

Van der Maaten, L.J.P., Postma, E.O., and van den Herik, H.J. (2008). Dimensionality reduction, a comparative review, Neurocomputing, vehicle routing. *IEEE Intelligent Systems*, Vol. 28, No. 6, pp. 38–41.

Vijayarani, S., and Maria Sylviaa, S. (2016). Comparative Analysis of Dimensionality Reduction Techniques: *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 4, No. 1, pp. 2320-9801. DOI: 10.15680/IJIRCCE.2016.0401006.

Vogl, T.P., Mangis, J.K., Rigler, A.K., Zink, W.T., and Alkon, D.L. (1988). "Accelerating the convergence of the backpropagation method," Biological Cybernetics, Vol. 59, pp. 257–263

Wang, H., Chen, M., Shi, X., and Li, N. (2016). Principal Component Analysis for Normal-Distribution-Valued Symbolic Data: *IEEE TRANSACTIONS ON CYBERNETICS*, Vol. 46, No. 2, pp. 356-365.

Wikipedia. (2017). Retrieved from https://en.wikipedia.org/wiki/Softmax_function#cite_note-bishop-1. (Accessed Feb 2017)

Witten, I., Frank, E., and Hall, M. (2011). Data Mining Practical Machine Learning Tools and Techniques, Third edition, *Elsevier Inc*, Chapter 5, pp. 147-187.

Wu, Z., Li, Y., Li, J., Xiao, F., and Wei, Z. (2015). Parallel and Distributed Dimensionality Reduction of Hyperspectral Data on Cloud Computing Architectures. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.*

Xiang, J. Westerlund, M., Sovilj, D., and Pulkkis, J. (2014). Using Extreme Learning Machine for Intrusion Detection in a Big Data Environment. *AISec '14 Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop, ACM*, pp. 73-82.

Yadav, Ch., Wang, Sh., and Kumar, m. (2013). Algorithm and approaches to handle large Data: *IJCSN International Journal of Computer Science and Network*, Vol. 2, No. 3. DOI: 2277-5420.

YANG, Q., and WU, X. (2006).10 CHALLENGING PROBLEMS IN DATA MINING RESEARCH: *International Journal of Information Technology and Decision Making*, Vol. 5, No. 4, pp. 597–604.

Yuan, Q., Cai, C., and Xiao, H. (2007) Diagnosis of breast tumours and evaluation of prognostic risk by using machine learning approaches, Vol. 2, pp. 1250–1260. Doi: 10.1007/978-3-540-74282.

Zhang, H., Yin, Y., Zhang, S., and Sun, C. (2015). An Improved ELM Algorithm Based on PCA Technique: *Springer International Publishing Switzerland, Proceedings of ELM-2014*, v2.98-103

Zhang, L., Zhang, D. (2015). Domain adaptation extreme learning machines for drift compensation in E-Nose systems, IEEE Trans.Instrum.Meas, Vol. 64, No. 7, pp. 1790-1801.

Zhang, Y., Cai, Z., Wu, J., Wang, X., andLiu, X. (2015). A Memetic Algorithm Based Extreme Learning Machine for Classification, *IEEE*. 978-1-4799-1959-8/15.

Zhang, Y., Wu, J., Cai, Z., Zhang, P., and Chen, L. (2016). Memetic Extreme Learning Machine. *Pattern Recognition58*, pp.135–148.

Zhu, T., Xu, Y., Shen, F., and Zhao, J. (2017). An online incremental orthogonal component analysis method for dimensionality reduction. *Neural Networks 85, Elsevier Ltd*, pp. 33–50.