Optimization of Neural Network Classifier for Thermal-Deceive Data Set by Genetic algorithm.

Zeyu Gong
Research School of Computer Science, Australian National University
u6146094@anu.edu.au

Abstract. The purpose of this report is to explore the application of neural networks to thermal-deceive data sets. Genetic Algorithm is used to feature selection and parameters optimization. The neural network is also compressed by the pruning technique. It was found that small mutation rate should be used in genetic algorithm to fulfill the task of exploring search space. Retaining the best performing genotypes can improve the performance of the genetic algorithm. Finally, the accuracy of the classifier is 0.97, which is much better than the 0.87 of the original paper. It was also found that proper threshold is very important for pruning techniques. Pruning can compress the network size without compromising performance.

Keywords: Neural Network, Pruning, Feature Selection, Genetic algorithm (GA), Evolutionary algorithm

1. Introduction

The data in this report are mainly from research "Network physiology of 'fight or flight response in facial superficial blood vessels" [1]. The pattern of blood flow changes in the cutaneous superficial blood vessels of the face can reflect a certain degree of subjects' psychological states [1]. The subjects were asked to deceive the researchers about several questions. The camera recorded blood flow data in five different parts of their faces (periorbital, forehead, cheek, perinasal and chin) as they answered the questions. There was a total of 20 different directions of blood flow. The data were normalized to (0,1). There were 31 subjects, 16 of whom were asked to deceive the questioner. With only 31 observations, this data set is comparatively small. However, each observation has 20 features. The dimensions of the data are comparatively high. It is challenging to apply neural network to such high-dimensional data with small sample size. In this case, feature and parameter selection may be necessary work. This report concentrates on using evolutionary algorithms to select features and parameters for training neural network.

Evolutionary algorithm (EA) has been applied in many fields and achieved good results [2]. The GA, one of the evolutionary algorithms can be used in feature selection and gets better results than traditional feature selection methods. [3] In addition to feature selection, some studies indicate that the parameter selection of neural networks is also very important. For example, determining the appropriate number of hidden neurons to prevent overfitting is critical in the prediction problem [4]. If the number of hidden neurons is too small, the model cannot reflect all the information contained in the data sets. If the number of hidden neurons is too high, the model will bind too tightly to the data, learning useless features. Genetic algorithms can also be used for parameter selection by encoding the parameters of a neural network into the chromosome. At the same time, genetic algorithms also need to balance the two cornerstones of search problems, exploration and exploitation [5]. This involves the selection of parameters of the genetic network itself - the selection of mutation rates, whether to retain the mother generation after crossover and mutation, whether to retain the best individuals of each generation and so on. This report will also discuss the different results of these parameters.

In this report, a four-layer feedforward neural network is used for training. There are two hidden layers, both using the tanh function as the activation function. Genetic algorithms are used for feature selection and parameter adjustment. The performance of the genetic algorithm is evaluated by comparing other feature selection methods such as information entropy and variance.

At the same time, a pruning technique was also used in the model. Gedeon and Harris [6] provided a pruning technique that used the distinctiveness of units as a criterion to reduce the number of hidden neurons. Using many parameters (more parameters than observations) is helpful for neural network training, but it means more training and storage costs. [7]. These computing and memory resources can be stressful at times. The limitation of resources requires reducing the size of the neural network while maintaining the performance of it. The pruning technique is a technique to compress the size of the neural network. This report is concerned with the influence on the performance of neural network using Gedeon's technique.

2. Method

2.1 The Basic Information of the Neural Network Used for Training

This is a simple three-layer neural network with two hidden layers. Adam is used as the optimizer, and tanh function is used as the activation function. The learning rate is set to 0.02. Cross-Entropy is used as the loss function. The first step in training is to adjust the number of neurons in the two hidden layers.

2.2 Validation Set and Training Set Selection

The data set used in this report is small. Because there is only 31 observation, it is difficult to divide them into a training set, a validation set, and a test set according to conventional methods. This will lead to a tiny test data set or the validation data set too, and stochastic segmentation will seriously affect the accuracy of the model in these data sets. For example, when the test data set has only three observations, a model with 100% accuracy may not perform better than a model with 0% accuracy. The large differences in the accuracy of these models may just be due to chance. Therefore, the method of cross-validation is used here. This method will cycle training the model 31 times, each time two observation value is extracted as the validation set and the rest as the training set. The final accuracy is the average accuracy of validation sets. In this way, the performance of different models can be fairly compared.

2.3 Basic Feature selection methods.

Reducing the number of features can significantly improve the prediction speed [8]. At the same time, reducing the feature dimension can effectively improve the prediction accuracy and avoid the curse of dimensionality [9]. In this report, two different basic feature selection techniques are applied: 1. selection based on the chi-square statistics between the feature vectors and the label vector, 2. selection based on relative entropy. 3. selected based on variance. They will be compared with the outcome of genetic algorithm.

2.4 Genetic algorithm.

a genetic algorithm needs to define the following components:

1) Chromosome: This report defines a binary string as a gene. There are 30 bits in the string. The first 20 bits reflect the selection of 20 features. Bits 20-30 reflect the size of the two hidden layers of the neural network, ranging from 2 to 64. Twenty random chromosomes were created initially.
2) Fitness function: the max accuracy of training was used as fitness value. The fitness value is used to rank the chromosome.

3) Selection: tournament selection was used. Select 3 individuals each time and add the best among them to the next generation.
4) Crossover and mutation algorithms: Crossover allow good genes to be recombined to eliminate less important ones. mutation prevents convergence to local optimum. The crossover method in this paper is to randomly select a pair of chromosomes from populations and generate four off springs, two of which are generated by multi-site recombination and two by one-site recombination. Mutation is controlled by the mutation rate, which is the probability that each gene spot on a chromosome may change.

Other hyperparameters: 1. population size: control the number of populations in each generation, which is set to 10 in this paper (due to limited computing power). 2. remove parent: when set to true, the parent of crossover will be removed from the population (allowing for diversification and renewal) 3. remove origin: when set to true, the original chromosome of mutation will be removed from the population (allowing for diversification and renewal) 4. Add number of mutate and crossover: Control the number of new offspring created. 5 save the best: when set to true, the best chromosome of this generation will save to the next generation.

2.4 Pruning technology

This article uses a pruning technique provided by Gedeon. The distinctiveness of neurons is evaluated by calculating the angles between the activation vectors generated by these neurons. If the angles are similar (the Angle difference is close to 0°) or opposite (the Angle difference is close to 180°) [6], it means that the neurons do not provide more information and it is not helpful for the model calculation. The thresholds set in the original text are 15° and 165°. The higher the threshold, the more neurons were deleted. This paper will discuss the influence of different thresholds on the performance of neural networks.

3. Results and Discussion

3.1 Feature Selection

Ranking	Variance	Chi-2	Relative entropy
1	17 CN -> PO	05 FH -> PO	05 FH -> PO
2	06 FH -> CK	19 CN -> CK	18 CN -> FH
3	01 PO -> FH	01 PO -> FH	20 CN -> PN
4	03 PN -> PO	18 CN -> FH	01 PO -> FH

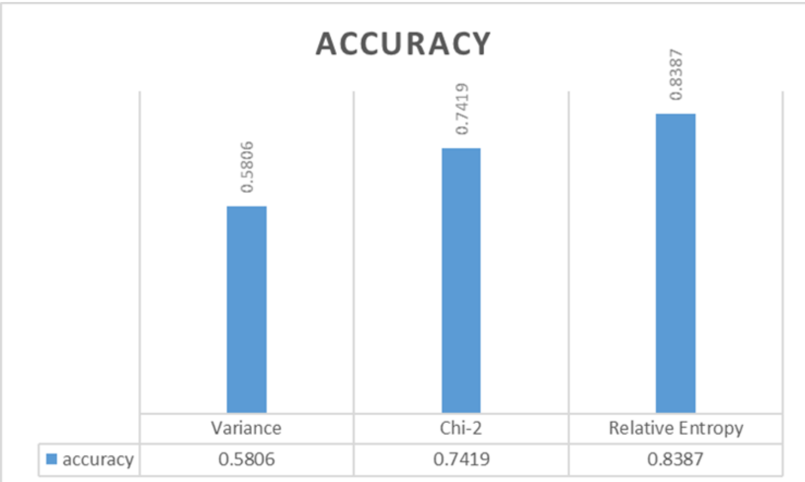

Figure 1

The table exposes the ranking of features based on different criteria. The data illustrates that the variance of features should be used as a threshold filter rather than as a method to extract several features. Both chi-square and relative entropy improve the performance of the model. After selecting the two highest-ranking features to retrain the model (40 neurons in the first layer and 10 neurons in the second layer), the accuracy reached 0.84. The selection of features greatly improves the performance

3.2 **Compare with the outcomes of GA**

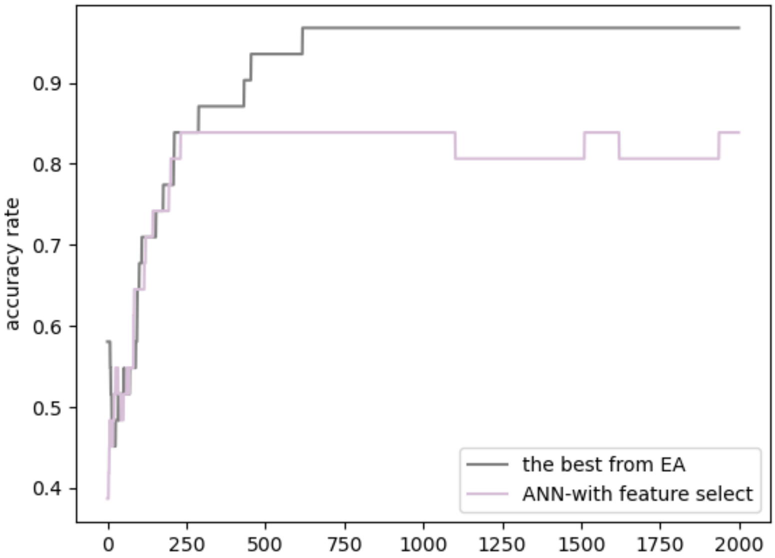

Figure 2

From Figure 2, we can see that genetic algorithm can significantly improve the performance of the classifier even though we have greatly improved the performance of the classifier by using the traditional feature selection method. The accuracy of the optimal solution of genetic algorithm is 0.97.

Compared with artificial feature selection using some criteria, genetic algorithm can better find the optimal solution of the whole search space. While these criteria can be useful, they can also cause people to miss out on better possible solutions. Genetic algorithm does not introduce a priori knowledge but only results oriented. Although it requires more computing power, it is more effective. The results of genetic algorithms can even be used to gain some new knowledge. For example, the features selected by relative entropy is features 5 and 18, while the features selected by GA is 3,4,5,6,8,10,13,15 and 19. This might mean that features 3,4,6,8,10,13,15,19 and feature 18 might contain similar information.

3.3 Compare GA with different parameters

1) mutation rate

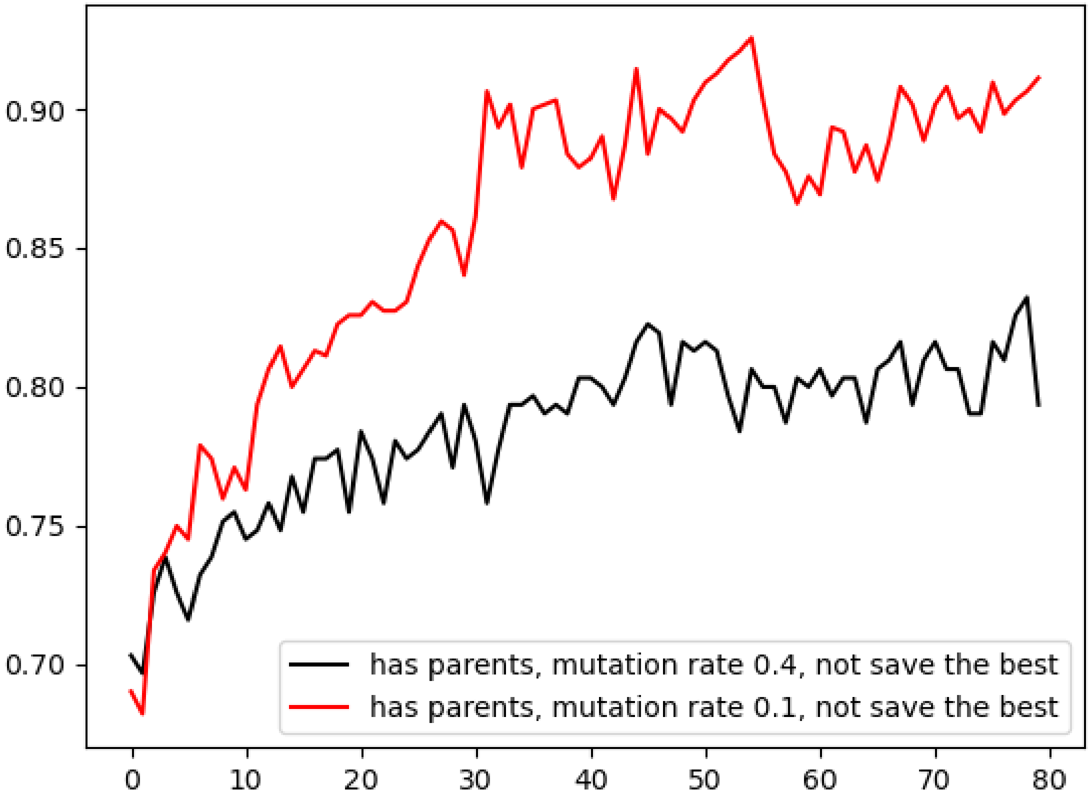

Figure 3

In Figure 3, the y-axis is the average accuracy of the off springs, and the x-axis is the number of iterations. It is important to determine the appropriate mutation rate. Because a high mutation rate is not exploring but jumping in the search space.

The high mutation rate breaks the link between the source chromosome and the new chromosome produced by the mutation. Figure 3 shows that the effect of training with a mutation rate of 0.1 is far better than that with a mutation rate of 0.4.

2) the influence of save the best

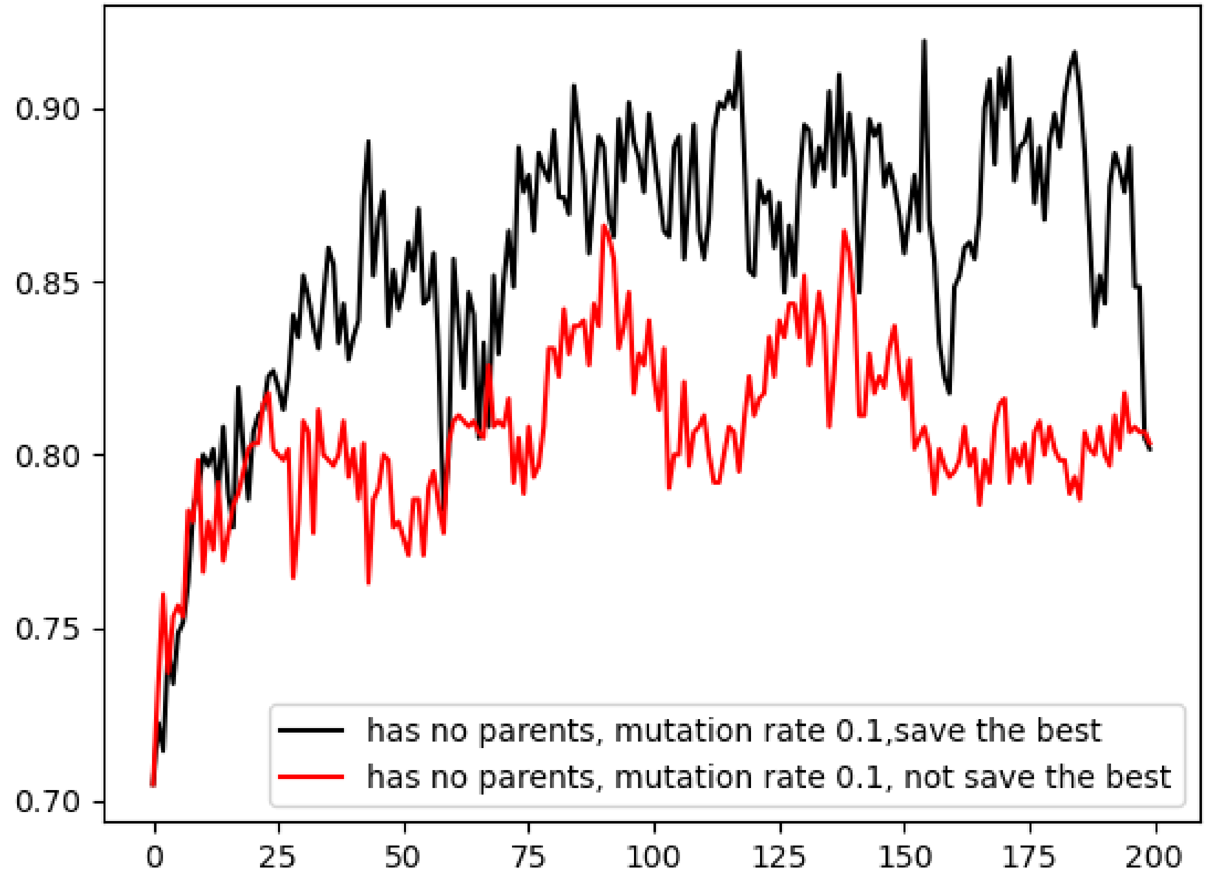

Figure 4

In Figure 4, the y-axis is the average accuracy of the off springs, and the x-axis is the number of iterations.

Save the best means to preserve the best genotypes from each generation to the next generation. When selecting remove the parents and origin genotypes, preserving the best genotype ensures that the current superior solution will not be deleted in an iteration. From figure 4, it can be concluded that This approach makes the performance of genetic algorithm better.

3) the influence of removing the parents

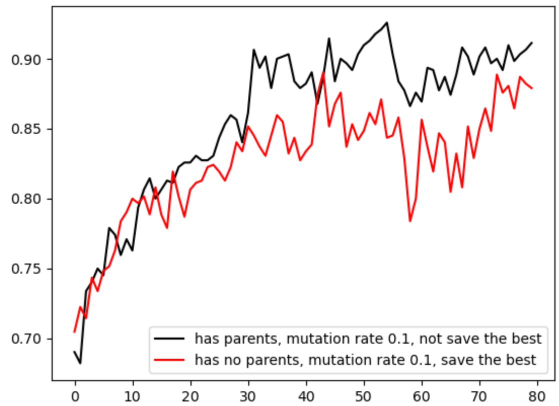

figure 6

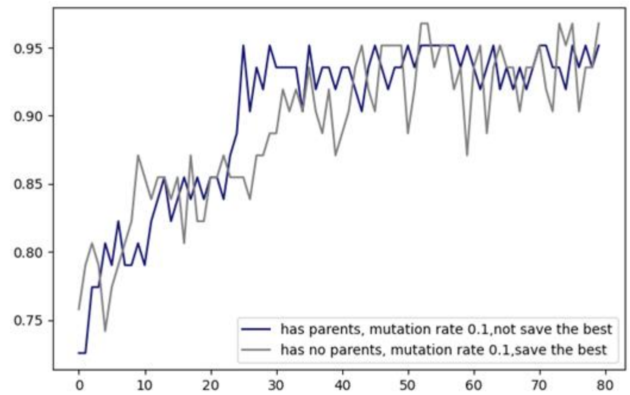

figure 5

In Figure 6, the y-axis is the average accuracy of the off springs, and the x-axis is the number of iterations. In figure 5, the y-axis is the max accuracy of the off springs. The effect of the genetic algorithm is similar in these two cases. Given the savings in storage and computing space by not keeping parents, save the best is a better approach.

3.4 Pruning

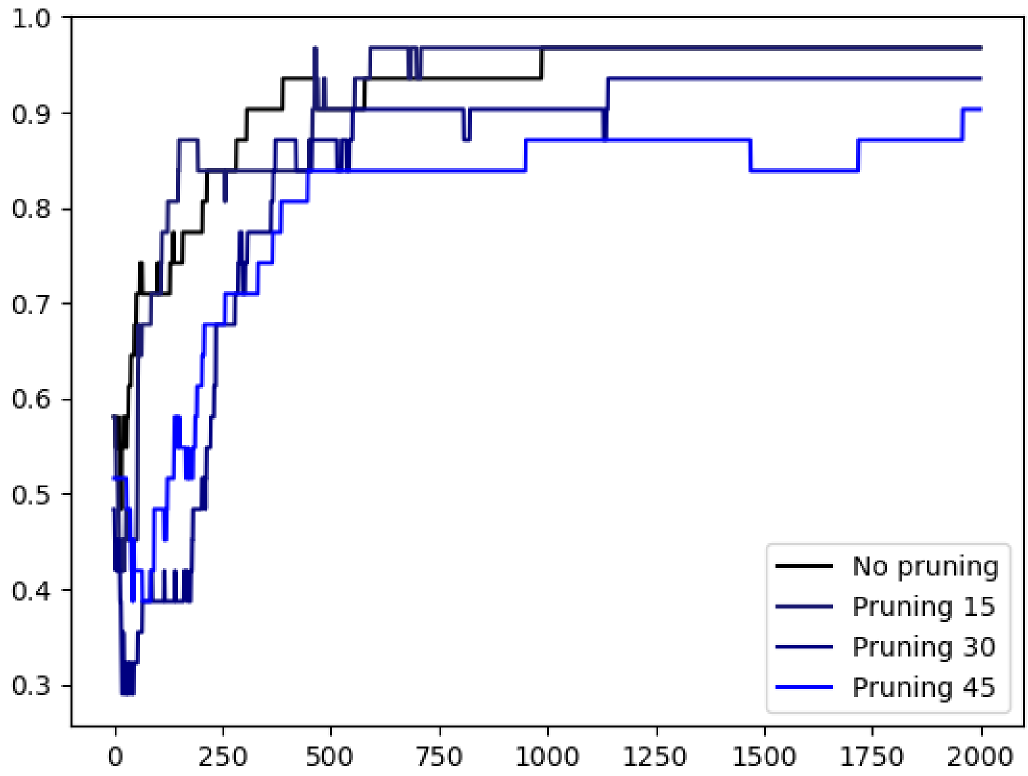

figure 7

The choice of threshold of angles between hidden neurons' weights affects the number of neurons deleted. As shown in Figure 7, when the threshold is greater than 30°, the performance of the neural network begins to decline. This is because some less similar weights have been pruned. It was found that even if the threshold were equal to 30°, about 35 neurons could still be pruned on a 40-2 double-hidden-layer neural network. The pruning technology is applied very well and greatly reduces the scale of the network.

4. Conclusion and Future Work

Genetic algorithm can be applied to feature selection problems and can get better results than ordinary methods. Neural networks that adjust parameters and feature selection through genetic algorithms have more powerful performance. The accuracy of the classifier applied to the dataset in this paper is 0.97, while that in the original paper is only 0.87. Small mutation rate should be used in genetic algorithm to fulfill the task of exploring search space. Retaining the best performing genotypes can improve the performance of the genetic algorithm. Threshold selection plays an important role in pruning application. Pruning can compress the size of the neural network without affecting its performance. Follow-up work can be combined with the knowledge obtained by genetic algorithm and the real world to explore why these parameters have better results.

5. Reference

[1] A. Derakhshan, M. Mikaeili, A. M. Nasrabadi and T. Gedeon, “Network physiology of ‘fight or flight’ response in facial superficial,” *Physiological Measurement,* 2019.

[2] C. G. C. W. ,. J. Y. Benjamin Doerr, “The (1+λ) Evolutionary Algorithm with Self-Adjusting Mutation Rate,” *Algorithmica, vol* 81, pp. 539-631, 2019.

[3] M. Lester, M. Guerrero 和 J. Burge, “Using evolutionary algorithms to select text features for mining design rationale,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing , vol* 34, pp. 132-146, 2020.

[4] D.-C. Li and C.-W. Liu, “A neural network weight determination model designed uniquely for small data set learning,” *Expert Systems with Applications,* vol. 36, no. 6, pp. 9853-9858, 2009.

[5] Č. Matej, L. Shih-Hsi 和 M. Marjan, “Exploration and exploitation in evolutionary algorithms: A survey,” *ACM Computing Surveys,* 2013.

[6] T. Gedeon and D. Harris, “PROGRESSIVE IMAGE COMPRESSION,” in *International Joint Conference on Neural Networks*, 1992.

[7] S.-K. Yeom, P. Seegerer, S. Lapuschkin, A. Binder, S. Wiedemann, K.-R. Müller and W. Samek, “Pruning by explaining: A novel criterion for deep neural network pruning,” *Pattern Recognition,* vol. 115, 2021.

[8] M. Thomas, K. D. Brabanter and B. D. Moor, “New bandwidth selection criterion for Kernel PCA: Approach to dimensionality reduction and classification problems,” *BMC Bioinformatics,* vol. 15, p. 137, 2014.

[9] N. Pochet, F. D. Smet, J. A. K. Suykens and B. L. R. D. Moor, “Systematic benchmarking of microarray data classification: assessing the role of non-linearity and dimensionality reduction,” *Bioinformatics,* vol. 20, no. 17, pp. 3185-3195, 2004.

[10] K. Gnana Sheela and S. N. Deepa, “Review on Methods to Fix Number of Hidden Neurons in Neural Networks,” *Mathematical Problems in Engineering,* 2013.

[11] M. Saygın and S. Zübeyde, “Deep Neural Network Initialization Methods for Training Sample Support,” *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS VOL. 14, NO. 12,* pp. 2462-2466, 2017.