Int J Hum Genet, 17(1): 38-42 (2017)
DOI:10.31901/24566330.2017/17.01.06

Comparative Study on Gene Expression for Detecting Diseases Using Optimized Algorithm

J. Sumitha[1], T. Devi[2] and D. Ravi[3]

[1,2]Department of Computer App, Bharathiar University, Coimbatore 641 046, Tamil Nadu, India
[3]PG and Research Department of Botany, Govt Arts College, Coimbatore 641 018, Tamil Nadu, India
E-mail: [1]<sumivenkat2006@gmail.com>, [2]<tdevi5@gmail.com>, [3]<dravi_botany@hotmail.com>

KEYWORDS Breast Cancer. Classification. Confusion Matrix. Micro-array Gene Expression Data

ABSTRACT The main objective is to detect the disease-causing gene from microarray data and predict the results from the gene expression value. Many computer-assisted algorithms developed to predict the characteristic of a gene is done using machine learning and other bio-inspired algorithms. In this paper, seven works are proposed and compared to estimate the efficiency. The Support Vector Machine (SVM) optimized neuro-expert algorithm is developed to optimize these predictive results of both machine learning and bio-inspired algorithms and proven its effectiveness and efficiency in detecting the disease-causing gene than ever before.

INTRODUCTION

Gene expression in microarray data has been used for identifying genes causing diseases located in the DNA of human body. Inherited diseases have been affecting human beings to a greater extent and many databases related to this exist in the web for research (Carl 2016). The dataset used for this research is PIMA dataset and the software which is used for implementation is Matlab. This dataset consists of the information which is related to the patients who are affected by breast cancer with gene ID (Cho-Jui 2014) and the results are predicted based on this gene value. There are two types of breast cancer datasets in which one is malignant cancer and another is prognostic. The breast cancer dataset used for this research is prognostic dataset and the efficiency can be estimated on the basis of confusion matrix method in terms of accuracy, precision, recall and F- measure.

The main aim is to predict the disease-causing gene using an optimized technique. In this paper, a newly developed neuro-expert system is proposed to optimize the performance of the other existing algorithm prevailed in this field. The feature of this algorithm is that it is based on the classification background and inhibits the characteristics of neural network algorithm. So classification of data classes and class labels are made easier to analyze the efficiency using gene expression value.

The objective of this paper is to find the disease-causing gene based on gene expression value and predict the effectiveness of the optimized algorithm over the data. The breast cancer dataset BRCA (prognostic) taken from UCI repository, URL. Matlab is the software that has been used for implementing this work. Especially, the prognostic breast cancer dataset is selected for this research.

This paper is organized as follows: Section 2 presents proposed methods to solve the task of identifying the cancer-causing gene. Section 3 contains results obtained and discussions. Finally, section 4 contains conclusions.

METHODOLOGY

Dataset Description

The breast cancer (WDBC) dataset taken from the URI repository had 761 data with gene ID or gene expression value. From this, 380 data are taken as training data and 381 data are taken as testing data. But this performance criterion for the classifiers in disease detection (Chopin 2013) is based on confusion matrix to analyze the performance criterions have been computed from this dataset.

Table 1: Pseudocode for training and testing data

```
Line1 dt=load('wdbc.data');
Line2 traindata=dt(1:380,3:end);
Line3 traincl=dt(1:380,2);
Line4 testdata=dt(381:end,3:end);
Line5 testcl=dt(381:end,2);
```

Table 1 represents the pseudocode for classifying the training data and testing data from the dataset where dt is the dataset variable, and traincl is the class variable of the training data. First 380 data from class label to the end is loaded as training in line 3 and line 4 and remaining data from the first class label to the end of the data in the dataset is loaded as testing data.

Sequential Algorithm

In this research, it is for finding the differences between expression values of a pair of genes in the prognostic dataset (Carl 2016). For example: g1,g2…..g10 are genes in the dataset in which the gene pair(g4g8g2) and the other gene pairs (g8g4g2) are found to be identical (Yuhaizhao 2014). It then implies that these two gene pairs generate the same disease in the human body (Thompson 2016). The linear sequences of predicting each gene in this dataset are taken as an input for predicting results (Chopin 2013). The pseudo code for this algorithm is shown in Table 2.

Pseudo Code for BAT Algorithm

Table 2: Pseudocode for BAT algorithm

```
for t=1:N_gen,
    for i=1:n,
        Q(i)=Qmin+(Qmin-Qmax)*rand;
        v(i,:)=v(i,:)+(Sol(i,:)-best)*Q(i);
       S(i,:)=Sol(i,:)+v(i,:);
        Sol(i,:)=simplebounds(Sol(i,:),Lb,Ub);
      if k>3
          k=1;
      end
      if rand>r
           S(i,:)=best+0.001*randn(1,d);
end
       Fnew=Fun(k,data,cl);
       if (Fnew<=Fitness(i)) & (rand<A) ,
          Sol(i,:)=S(i,:);
          Fitness(i)=Fnew;
      end
      if Fnew<=fmin,
          best=S(i,:);
          fmin=Fnew;
      end
   end
    N_iter=N_iter+n;
   k=k+1;
end
```

DCKSVM (Divide and Conquer Kernal Support Vector Machine)

DCKSVM is used to improve classification prediction of data in the dataset (Del 2015). The purpose of using DCKSVM algorithm is to divide the main clusters of data into sub-clusters and make its prediction reliable on that clusters (Cho-Jui 2014). Since it is a good classifier (Ken 2011), its efficiency is better compared to Sequential algorithm (Michael 2015). It removes the common group of data from the dataset (Yang 2016) and the efficiency is calculated (Yuchen 2013).

Algorithm 1: Divide and Conquer SVM

Step 1: Partitioning dual variables into k subsets {v1….vk}

Step 2: Time complexity for solving subproblems are reduced to $O(k^*n/k)2=O(n2/k)$ with space complexity, where n is the variable and k is the cluster subset.

Step 3: Concatenate them to form solution for whole problem ab = (a1….ak)

Step 4: A bound is derived on $||ab-a^*||$ where ab is the optimal solution by adding cluster-kernel values.

Step 5: Minimizing the off-diagonal values of the kernel matrix with a balancing normalization.

Step 6: For each cluster k, go to Step 2 for partitioning data and computing Step 4 for absolute scale.

HRBFNN (Hybrid Radial Bias Neural Network)

To enhance the capabilities of data granulation in the dataset, HRBFNN is applied to this breast cancer dataset which inhibits the characteristics of data granulation and PCA for preprocessing the data (Wei 2014). Weight bias is multiplied with the training data and the output is calculated. This output data is then matched with the testing data and the actual output is determined. The data which prone to lesser error rate is taken for training data to calculate the efficiency of algorithm. This algorithm is having capable of enhance the best network topology which is vital for developing the performance of the results. In future, HRBFNNs may be enhanced by constructing fuzzy or with the help of multi-objective evolutionary algorithms, it may be used to optimize the HRBFNNs (Wei 2014).

Algorithm 2: HRBFNN

Step 1: Pre-process the data set using PCA. To obtain dimensionality reduction, principal component analysis is used to pre-process data sets for feature extraction and reduction of data.

Step 2: Training and testing data sets are formed.

Step 3: The generic parameters used in the conclusion part are decided.

Step 4: System's input variables are determined.

Step 5: PFNs are designed. For the selecting r inputs, the number of nodes (PFNs) generated in each layer becomes equal to k = n!/n!(n-r)!r!, where, n is the number of total inputs and i stands for the number of the chosen input variables and c is the clusters.

Step 6: Check the termination criterion.

Step 7: Select the best predictive capability nodes and construct their corresponding layer. To select the highest predictive capability nodes (PFNs), the following steps are used.

Step 7.1: The polynomial coefficient parameters (a0, a1. . . a5) of each PFN are estimated by the subset of the training data and testing data.

Step 7.2: The identification error (EPI) of each PFN is determined with the help of the testing data set.

Step 7.3: All PFNs are sorted and rearranged in descending order based on their performance of the identification error (EPI1, EPI2… EPIn! / (n-r)! r!).

Step 7.4: Select the best x nodes, for constructing the next layer of PFNs, where x is the best predictive capabilities nodes. All nodes (PFNs) are first rearranged in a descending order based on their performance (EPI1, EPI2 . . . EPI n! / (n-r)! r!), and then some nodes will be selected.

The representations are concise as that the PI represents the performance index of training data set and the EPI represents the performance of the testing data set (Wei 2014).

ONMTF (Orthogonally Non-Matrix Trifactorization) Algorithm with BAT

ONMTF algorithm (Stra•ar 2016) with BAT to cluster the rows and columns in the breast cancer dataset along with Bio-inspired algorithm, (Vijay Kumar 2016) BAT for optimizing the results drawn from the machine Learning Algorithms (DCKSVM and HRBFNN), BAT (Yubao 2016) is applied to this prognostic dataset to predict the performance (Paul 2016). The performance results drawn when compared to the results of machine learning algorithms is the peak. It gives a high percentage of accuracy than the machine learning algorithms (Sequential model, DCKSVM, BAT) than ever before.

The steps of the BAT algorithm are as follows:
Initialization;
Repeat
New solutions generated;
Local searching;
New solution generated by flying randomly;
Finding the current best solution;
Until (getting optimized solutions)

MNTF with BAT

This algorithm determines multi-variant approach over the data and predicts the value of the gene (Del 2015). The multi-variant may be more than one factor taken as a predictive element for processing the results. It is evaluated with multiple factors that correlate the efficiency and takes more than one factor for predicting results in the dataset other than a single factor in the dataset.

Algorithm: Multi-label Nonnegative Matrix Tri-Factorization (MNMTF)

1: Input: Nonnegative matrix X and binary label matrix Y; Weighting parameter for label correlation Ω; the number of bases J;

2: Output: Non –negative matrices U and S minimizing ||X-USY||2 F + Ωtr (SLST);

3: Initialize U and S by random positive values;

4: repeat 5 and 6.

5: $U = U^{*} \frac{XYTST}{USYYTST}$

6: $S = S^{*} \frac{UTXYT + \Omega S}{UTUSYYT + \Omega SD}$

7: until convergence criterion met

The results are evaluated with hybridization of BAT (Yubao 2016) with MNTF algorithm. The accuracy is estimated by comparing the results of ONMTF and MNTF with BAT algorithm.

SVM- Optimized Neuro-expert Algorithm

This proposed algorithm inhibits the features of both Support vector machine and the Neural Networks. Since SVM is a good classifier, this system gives the better result when compared with those above existing methods.

The steps for the SVM- Optimized NE algorithm:

Input: Dataset D;

Output: OP is the Outcome.

Step 1: Support Vector Machine algorithm is executed for removing the common group of data.

Step 2: Neural Network Algorithm is implemented. The selected reduced error rate data items (OP) are prone to calculate with weight bias (wtb) to the training data, then

OP= Data * wtb;

Step 3: OP is applied to the testing data in which the matched weight bias of data from testing data and the actual data is selected.

Step 4: For each iteration, i converges step 1 to step 3 until the condition met.

RESULTS AND DISCUSSION

In the first step, sequential algorithm and DCKSVM is applied to the wdbc dataset and it is compared for predicting the efficiency. Then, after predicting the results of these algorithms, it is evaluated with the results of HRBFNN. It is proved that among machine learning algorithms that applied over the dataset, HRBFNN gives better performance than the sequential and DCKSVM. Then, bio-inspired algorithm, BAT is tried with ONMTF algorithm and its result is compared with DCKSVM. Now, it is proved that ONMTF with BAT algorithm gives better prediction than the other algorithms. Again, MNTF with BAT is tried to estimate prediction results and it shows a high predictive performance than the other algorithms that applied before. SVM-optimized neuro expert algorithm is proposed for optimizing this performance and proven that this proposed work gives a tremendous high performance than the other algorithm used for detecting the disease-causing gene and the efficiency in terms of accuracy, precision, recall and f-measure calculated based on confusion matrix method.

Accuracy is the percentage of correct predictions from the dataset and is proportional to the total number of predictions that were correct. It can be calculated using equation,

Accuracy= a1+a4/ (a1+a2+a3+a4)..... (1)

The recall is the proposition of positive cases that are correctly identified, as estimated using equation,

Recall=a3/ (a3+a4)..... (2)

Precision (P) is the proposition of the predicted positive cases that were correct, as evaluated using equation

Precision=a4/ (a2+a4)..... (3)

F-measure (F) can be calculated using formula,

F= 2 *(Precision*recall) /(precision+ recall)...(4)

Where a1 is the number of correct predictions that an instance is negative, that is, correctly predicted genes which are not disease-causing gene, a2 is the number of incorrect predictions that an instance is positive, that is, incorrectly predicted which are diseases causing gene, a3 is the number of incorrect of predictions that an instance negative, that is, incorrectly prediction of non–diseased gene and a4 is the number of correct predictions that an instance is positive, that is, correctly predictions of disease-causing genes in the dataset.

The efficiency of Sequential, DCKSVM, HRBFNN, ONMTF, BAT, MNTF, MNTF with BAT, and SVM- optimized neuro –expert algorithm is shown in the Table 1 and it is proven that the proposed SVM optimized neuro-expert algorithm is effective than other algorithms such as Sequential, DCKSVM, HRBFNN, ONMTF, Bat, MNTF, MNTF with Bat.

It is to be noted in Table 3 the values of f-measure and the recall of both HRBFNN and ONMTF are the same. When the findings of ONMTF with BAT and the MNTF with BAT are compared, it is observed that the MNTF with BAT has 94.7 percentage shows the greater efficiency than ONMTF with BAT with 89.4 percentage in terms of accuracy, precision, recall and f-measure as shown in Table 3. These results then get optimized by newly developed SVM optimized neuro-expert algorithm and yield

Table 3: Results of the algorithms

	Sequence model	*DCKSVM*	*HRBFNN*	*ONMTF*	*ONMTF with Bat*	*MNTF*	*MNTF with Bat*	*SVM-optimized neuro-expert*
Accuracy	77.7778	80.4233	85.1852	86.7725	89.4180	92.0635	94.7090	96.2963
Precision	0.7103	0.7504	0.7937	0.8085	0.8396	0.8722	0.9081	0.9340
Recall	0.7659	0.8323	0.8713	0.8734	0.8987	0.9322	0.9576	0.9678
F-measure	0.7371	0.7892	0.8307	0.8397	0.8682	0.9012	0.9322	0.9506

results as 96.2 percent. This is the peak performance than the other machine learning and bio-inspired algorithms.

CONCLUSION

After implementing each algorithm over the cancer dataset, the efficiency performance noted between the sequential algorithm, DCKSVM, HRBFNN, ONMTF with Bat, MNTF, and Hybrid MNTF with Bat and finally the proposed SVM-optimized neuro-expert algorithm. The results proved that this proposed algorithm SVM-optimized neuro-expert gives better efficiency than the other algorithms on detecting cancer gene. Since it stimulates the nature of both support vector machines and the neural networks, it is proved in itself as the best effective predictor on investigating the gene. In future, it can enhance with a newly developed algorithm or converge with a new algorithm to predict the disease-causing gene.

REFERENCES

Carl H 2016. Sequential pattern mining – Approaches and algorithms. *ACM,* 20: 1-46.

Cho-Jui H 2014. A divide-and-conquer solver for kernel support vector machines. *Mach Learn,* 12: 56-68.

Chopin N 2013. SMC2: An efficient algorithm for sequential analysis of state space models. *J R Statist Soc,* 75: 397-426.

Del B 2015. Non-negative matrix tri-factorization for co-clustering: An analysis of the block matrix. *Inform Sciences,* 301: 13–26.

Ken K 2011. A sequential pattern mining algorithm using rough set theory. *Int J Approx Reason,* 52: 881–893.

Michael A 2015. A preliminary model of design as a sequential decision process. *Procedia Computer Science,* 44: 174-183.

Paul S 2016. Orthogonal symmetric non-negative matrix factorization under the Stochastic Block Model. stat.ML, arXiv: 1605. *Machine Learning (stat. ML),* arXiv:1605.05349 [stat.ML].

Stra•ar M 2016. Orthogonal matrix factorization enables integrative analysis of multiple RNA binding proteins. *Bioinformatics,* 32(10): 1527-1535.doi: 10.1093/bioinformatics/btw003.

Thompson JV 2016. Optimization of focusing through scattering media using the continuous sequential algorithm. *J Mod Optic,* 63: 80-84. DOI: 10.1080/09500340.2015.1073804.

Vijay Kumar B 2016. Bat algorithm and firefly algorithm for improving dynamic stability of power systems using UPFC. *IEEE T KNOWL DATA EN,* 8: 126-134.

Wei Huang 2014. Design of hybrid radial basis function neural networks (HRBFNNs) realized with the aid of hybridization of fuzzy clustering method (FCM) and polynomial neural networks (PNNs). Neural Netw. 60:166-181.

Yang P 2016. High-dimensional black-box optimization via divide and approximate conquer. *Artificial Intelligence (cs.AI),* cs.AI, arXiv:1603.03518v2.

Yuhaizhao 2014. Learning phenotype structure using sequence model. *IEEE T KNOWL DATA EN,* 26: 179-185.

Yuchen Z 2013. Divide and conquer kernel ridge regression. *Int J Approx Reason,* 30: 1–26.

Yubao L 2016. Improved bat algorithm for reliability-redundancy allocation problems. *Int J Approx Reason,* 10: 156-163.

Paper received for publication on February 2016
Paper accepted for publication on February 2017