International Journal of Biomedical Science & Bioinformatics
Volume 3 : Issue 2 *[ISSN 2475 - 2290]*
Publication Date : 31 August, 2016

Experiments on weighted classifier fusion for autism detection using genetic data

Fuad M. Alkoot

***Abstract*—Research in the health related field involves the use of high dimensional data where microarray gene expression data are used for the classifier based detection of diseases and abnormalities. Many machine learning tools and methods have been presented and proposed to detect diseases from microarray gene expression datasets where the overwhelming majority of work is for the detection of cancer. However, less attention is made to the detection of autism using such data. We experiment with autism detection using five gene expression data sets from five chromosomes. This data includes a low number of samples and a high number of features that reach tens of thousands. The task is difficult due to the large dimension of the data set and the high overlap in the class distributions. Therefore, a feature selection stage is necessary before the classifier and combiner design stages. We experiment with four feature selection methods, five classifier types and two existing combiner methods. Additionally, we propose six variants of a weighted fusion method, where this proposed method influences the classifier decision on a test sample based on its previous performance on the validation set. This is achieved by multiplying its decision by a predetermined weight. Results show that it outperforms or is equal to existing methods. This is achieved when the feature set size is very low reaching 50 or less**

Keywords*—Classifier combination; bagging, Autism, Gene expression data, big data

I. Introduction

Many machine learning tools and methods have been presented and proposed to detect diseases from microarray gene expression datasets. Most are comparing the performances due to different types of feature selection and classifiers or combiners. The different researches show that some methods are not always successful and suffer from drawbacks. The overwhelming majority of work is for the detection of cancer. No previous work was found for the detection of autism using microarray gene expression data.

The task is difficult due to the large dimension of the data set and the high overlap in the class distributions. Therefore, the autism detection process involves several stages, starting from the data preprocessing stage, then the feature selection stage followed by the classifier design stage and ending at the classifier combination or decision fusion stage. Using the preprocessed data provided by [1] we experiment with four different feature selection methods to find which yields best results. These are variants derived from two basic existing methods of PCA and clustering. At the next stages our experiments involve five classifiers and fourteen combiner methods. Classifiers that we experiment with are k-nearest neighbor, 1-nearest neighbor, neural networks and two types of support vector machine classifiers. Combiner methods used are bagging [2], random subspace method [3], six types of the weighted classifier selection combiner and six types of the single feature weighted classifier combiner. The last two combiners are proposed by us here. In all the combiners the classifier decisions are fused using the sum soft fusion strategy [4], [5]. We experiment with different feature set sizes that are less than 50.

In the next section, we present the data used in this paper for autism detection. This is followed by a section on experimental methodology. In section 4 we present results and the report is brought to conclusion in the last section.

Fuad M Alkoot *(Author)*
Higher Institute of Telecommunication & Navigation / PAAET
Kuwait

II. The Data Set

A. *Data Description*

Even though the DNA copy numbers variations occur frequently in the genome of normal people, especially in the segmental duplication regions (SDs), it has been demonstrated that some variations are associated with behavioral and developmental abnormalities such as cognitive impairment, autism, mental retardation, and possibly psychiatric diseases. Different studies tested the whole genome and detected autism-related abnormalities in 5 SD-rich intervals [7]. Therefore, autism is correlated with DNA copy number variations (DCV). Our study is confined to analyze and detect the recurrent variations across these 5 intervals which have a total length of 75Mb using finely-tiled oligonucleotide arrays. shows the genomic locations of each interval. This data includes samples of 71 autistic children and 71 typically developing children.

B. *Data Preprocessing*

Before any feature selection and classification is performed, at the first stage, we need to preprocess the data to improve its quality using the preprocessing method proposed by [1].

III. Experimental Methodology

To find the best machine learning system that detects autism we experiment with several types of feature reduction, classification and combination methods. Simulation

experiments are conducted using Matlab. The data is partitioned in two training and test sets based on the 10 fold cross validation method. Furthermore, the training set is divided in two equal sets; training and validation sets. Feature selection is conducted on the full training set only while the test set is set aside and treated as data of new patients. Classifier and combiner methods are designed using the training and validation sets. The test set is used after designing the final system to measure its classification rate. The classification rate is found by dividing the total number of correctly classified test samples by the total number of test samples.

We repeat the experiments using five types of classifiers as described below, and two combiner methods for each type of the classifiers. The fusion method used to combine the classifiers is Sum [4][5] fusion method. Therefore, our system for autism detection consists of the following stages after conversion of genetic information found in a human sample to digital genetic data using microarray sequencing of gene expression levels.

1- Preprocessing of genetic data based on the method of [1].
2- Feature selection to reduce data dimensionality.
3- Classifier design or training.
4- Classifier combination and decision fusion.

For some combiners the third and fourth stages are merged in one step. For the first stage we use the method of [1] as described in the previous section. The methods of stages 2 to 4 are described next.

A. Dimensionality Reduction Methods

- Clustering

Data clustering is commonly used to find clusters, or classes, of data in an unsupervised classification problem. All methods start by defining a temporary cluster center that is gradually moved as relevant samples are assigned to the cluster. The methods differ in the techniques used to assign samples to clusters. Additionally, several methods are used to merge or divide clusters. We don't aim to find classes or clusters because the classes are known. However, we aim to use clustering tools to find the most distinguishing features. Therefore, we attempt to use clustering tools to find features that yield the largest distance between the means of the two clusters and yield clusters with smallest standard deviation. This can be found using the following equation, known as fisher score [7].

$$f_{1,2} = \frac{(\mu_1 - \mu_2)^2}{(\sigma_1 - \sigma_2)} \quad (1)$$

Based on their fisher scores we sort the features in a descending order. We experiment with taking the best 50, 30 and 10 features, which are referred to as size 1, 2 and 3 in the tables of results.

- Clustering-PCA:

Here we sort features according to the clustering method which used the fisher score equation 1. Then apply PCA to the best features to obtain a new representation of the feature space where features are moved to a more distinguishing representation. Best eigenvectors are found by finding eigenvalues that are greater than 0.01, 0.1 and 0.5.

- Staged Clustering, "2nd, 3rd":

This is a feature selection method that is proposed by us and is based on clustering but we take the most different features by measuring the Euclidean distance between features. The furthest 1000 are taken and clustering is applied to them. Next, from this sorted list the furthest 500 are taken and clustering is applied to them to create the 2nd stage sorted cluster set. Next for this sorted list the furthest 100 are taken and clustering is applied to them to create the 3rd stage cluster set. These are referred to as 2nd stage and 3rd stage. For each of these feature selection methods we consider three feature set sizes that are used by the classification system. The feature set sizes are 50, 30 and 10 features, referred to as size 1, 2 and 3 respectively.

B. Classifier Types

For the nearest neighbor classifier we experiment with two values of k set at 1 and $\sqrt{N}$, where N is the square root of the number of training samples. The distance metric used is the mahalanobis metric. The neural network classifier used here consists of three layers. The transfer function or output of the first two layers is log-sigmoid, while that of the output or third layer is purelin. The network training function used is backpropagation. The number of neurons in the first layer is equal to the number of features, while that for the hidden (second) layer is set at 5. The number of neurons at the output layer is equal to the number of classes, which is two. For the support vector machine, SVM, we experiment with two SVMs; one with RBF sigma and box constraint values set to 1, and a second with these parameter values calculated using the training set and set to 0.3.

C. Existing Combiner Methods

Bagging predictors proposed by Breiman [2], is a method of generating multiple versions of a predictor or classifier, via bootstraping and then using those to get an aggregated classifier. We set the number of multiple versions of classifiers to 25, as recommended by Breiman [2]. The total number of samples in each bootstrap set is equal to those of the original training set. The second combiner RSM [3] aims at creating diverse classifiers by assigning different features to each classifier. The number of features is set at a fixed value, m, less than the total number of features. Each classifier is assigned a subset of features that are randomly selected without replacement from the full feature set. This results in classifiers having different views of the data space. We set m to equal 67 percent of the total number of available features. In comparison to 50% recommended by [3] we found better rates are achieved at 67%. The number of combined classifiers is set similar to bagging at 25

D. *Proposed Combiner Method*

1. Weighted classifier combination, WC#N-R:

This method is proposed by us for this project. It aims at learning the performance of each classifier using the validation set. The weight assigned to each classifier based on its performance is found using the probability estimate of a classifier for an input sample.

For a test sample each classifier decision is multiplied by a weight factor found using one of the three methods below. We experiment with each method alone and refer to them as WC1, WC2 and WC3 combiners.

Weight for class i is found according to one of the three methods below:

- Wi1= $P(\omega_i/x)$ where $P(\omega_i/x) = \max_i P(\omega_i/x)$ is the classifier estimate for a validation sample x.
- Wi2= $P(\omega_i/x)$ if x belongs to class ω_i otherwise 1- $P(\omega_i/x)$
- Wi3= 1 if x belongs to class ω_i otherwise 0.

The weight table is created from the decisions on the validation set. This can be used directly to find the final weights for a test sample, hence named WC#R and # is 1, 2 or 3. Or can be regenerated using a neural network, hence named WC#N. This is done by training a neural network to generate the table of weights using the weight table generated from the validation set and the training samples. The maximum number of combined classifiers is 25. Therefore, the combiner estimate for class i is $P(\omega_i/x) = W_{ij} \times P(\omega_i/x)$, for i=1, 2 classes, and j = 1, 2 or 3 depending on the weight method used.

2. Single Feature Weighted Classifier Combination, SFC#N-R:

In this method we experiment with creating single feature classifiers that are combined using the weight table of the previous subsection. Therefore, as in the previous section we name them as SFC#R or SFC#N. This method may generate very strong classifiers on subsets of the data space, in addition to very weak classifiers that are excluded using the weight table.

IV. Results

Tables 2 to 4 show the classification rates achieved for each chromosome under the various parameters and scenarios. Given the feature selection method, a chromosome and a feature set size, or eigen value, we present the best rate and the classifier and combiner that yields this rate. In addition to the maximum rate we also present all rates that are insignificantly lower than the maximum. Calculation of significance is made by finding rates that are lower by less than five percent of the amount needed for the maximum rate to reach perfect classification, i.e. 100%. This can be found through an equation which finds the lowest classification rate considered insignificantly lower than the highest rate achieved:

$$C_{Low} = \text{Highest rate} - (5\% \times (100 - \text{Highest rate})) \quad (2)$$

Where C_{Low} is the lowest acceptable classification rate.

1. Clustering feature reduction method, "Clstr"

At this feature selection method for each chromosome the highest rates achieved at the different feature set sizes were in the upper 70's or low 80's, as shown in table I. Best results over the three feature sizes were achieved using N17. Here, N10 results improve compared to other feature reduction methods. We find that at each chromosome and feature set size, one of our proposed methods yields the highest rate.

TABLE I. RESULTS OF THE CLUSTERING FEATURE SELECTION METHOD

Chromosome	Feature size	Combiner system	Classifier	Classification rate
N7	1	WC1R	kNN	74.29
	2	bagging	KNN	69.29
		RSM, WC3N	KNN	68.57
		RSM	SVM.3	68.57
	3	WC3N, SFC3N	Neural	71.43
		SFC1R	Neural	70
N10	1	WC1N,WC2N&R,WC3N	Neural	71.43
		RSM, SFC1N,	Neural	70.71
		SFC1N&R	SVM.3	70.71
		RSM, WC1N&R, SFC1N&R	SVM1	70
		SFC2N&R	SVM1	70
		SFC2N, SFC3N	Neural	70
			SVM.3	70
	2	Bagging, WC1N, WC3N	Neural	71.43
		RSM, WC1R, WC2N&R, SFC1R.	Neural	70.71
		SFC1N&R, SFC2N	SVM1& SVM.3	70.71
	3	Single, WC1N.	Neural	70.71
		Single, RSM, WC1N&R	SVM1	70.71
		SFC1N&R, SFC2N	SVM.3	70.71
		WC1R, WC2N&R, WC3N	Neural	70
		WC3N, SFC1N&R	SVM1	70
		SFC3N	SVM.3	70
N15	1	WC2N	SVM1	76.43
		Bagging, WC1R, WC3N,	kNN	75.71
		WC3N	SVM1	75.71
	2	Single, WC1R, WC2N, WC3N	Neural	75
		WC2N, WC3N	SVM1	75
		WC1N, WC2R	Neural	74.29
	3	Single, SFC1N&R,	kNN	74.29
		WC2N	kNN	73.57
N17	1	RSM	kNN	82.14
	2	Single	kNN	80
		RSM	RSM	80
		WC2N	kNN	79.29
		Single, WC1N&R	SVM1	79.29
	3	WC3N	SVM1	81.43
		RSM	SVM.3	80
		Single, RSM, WC1N&R	SVM1	80
N22	1	Single	Neural	76.43
		RSM	kNN	75
	2	WC2R	Neural	75
		WC2N	Neural	74.29
		Bagging, WC2N	kNN	74.29
	3	Single	Neural	71.43
		Bagging	Neural	70
N all	1	WC2N, WC3N	kNN	78.57
	2	WC3N	kNN	78.57
		Bagging	kNN	77.86
	3	Single, WC3N	kNN	77.14
		WC1N		76.43

TABLE II. RESULTS OF THE CLUSTERING-PCA FEATURE SELECTION METHOD

Chromosome	Eigen	Combiner system	classifier	Classification rate
N7	1	WC1N&R WC2N&R Single, WC3N	Neural	75.71 75 74.29
	2	Single	Neural	68.57
	3	SFC3R	kNN Neural	72.14 70
N10	1	Bagging WC1R WC2N	Neural Neural	70.71 70 69.29
	2	RSM	SVM1	72.14
	3	SFC3N	SVM.3	67.86
N15	1	bagging	Neural	82.86
	2	WC1N&R, WC2R. Bagging, WC2N, WC3N	Neural	80 79.29
	3	WC1R WC2R	kNN SVM1	70 70
N17	1	Bagging Single, RSM, WC1N&R	1NN	73.57 72.86
	2	Single, WC1N, WC1R	1NN	73.57
	3	SFC3N SFC3N	Neural kNN	69.29 68.57
N22	1	WC1R, WC2R WC1N, WC3R *Bagging, WC2N, WC3N*	Neural	72.86 72.14 *71.43*
	2	Single, bagging, WC1R, WC3N WC2R	Neural Neural	70.71 70
	3	WC3N	Neural	70
N all	1	bagging, WC1N&R,WC2R WC3R	1-NN 1-NN	76.43 75.71
	2	WC1R Bagging, WC3N WC3R	Neural Neural 1-NN	73.57 72.86 72.86
	3	SFC3N WC2R	SVM.3 kNN	70 68.57

2. Clustering-PCA feature reduction method "Clstr-PCA":

Here, the best combiners were mostly bagging and WC1 except at eigen size 3 where the SFC3 combiners yields better results. For the different chromosomes and eigen sizes the rates were mostly in the 70's, except for N10 that yields lower rates in the 60's.

3. Two stage Clustering, 2nd stage:

Overall rates for this feature selection method at the different chromosomes and feature sizes are mostly in the lower 70's.

4. Three stage Clustering feature reduction, "3rd stage":

Overall the maximum rates at each chromosome and feature size were mostly in the upper 60's and lower 70's, except for N15 that yielded rates in the 80's and upper 70's. Looking at table IV we find that SFC is the best at chromosome N22, while at all other chromosomes WCC is best.

TABLE III. RESULTS OF THE 2ND STAGE FEATURE SELECTION METHOD

Chromosome	Eigen	Combiner system	classifier	Classification rate
N7	1	Single, RSM WC2R, WC3N Single, bagging, WC1R WC2N SFC3R	Neural Neural kNN neural kNN	70 69.29 69.29 68.57 68.57
	2	WC3N RSM, WC1N, WC2N&R, WC3R Single, WC1R	Neural	70.71 70 69.29
	3	WC1R	kNN	70.71
N10	1	RSM	kNN	57.86
	2	SFC2N WC3R, SFC2R	k-NN 1-NN	59.29 58.57
	3	Single	Neural	60
N15	1	WC3R	neural	78.57
	2	WC3R WC2N	Neural	77.86 77.14
	3	WC3R Bagging, WC1N, WC2R, WC3N SFC1N	Neural Neural kNN	67.86 66.43 66.43
N17	1	Bagging, RSM WC1N&R, WC2N&R	Neural	84.29 83.57
	2	WC2N RSM	Neural	85.71 85
	3	WC2N	Neural	85
N22	1	Bagging, WC2N WC1R, WC3N SFC3R SFC3R WC1N Single, RSM, WC1N&R	kNN kNN SVM.3 neural kNN SVM1	71.43 70.71 70.71 70.71 70 70
	2	bagging, WC2N SFC2N SFC2R WC1N, SFC2R RSM,SFC3N&R bagging, RSM, WC1N&R, WC3R SFC3N&R	kNN kNN SVM.3 kNN neural SVM1 SVM.3	71.43 70.71 70.71 70 70 70 70
	3	Single WC1R	Neural kNN	74.29 73.57
N all	1	WC1N Bagging, WC3N	Neural	77.14 76.43
	2	WC2N	Neural	75.71
	3	WC3N Bagging, WC1N&R, WC2N, WC3R	Neural	70 69.29

v. Conclusion:

In a machine learning environment, microarray gene expression data are commonly used to detect different types of cancer. We use such data to design an autism detection system. However, due to the large dimension of the data in addition to the high class overlap it is impossible without preprocessing and reducing the data dimensionality. Therefore, for the given data set from five chromosomes, we preprocess the data then apply feature reduction techniques before using the data for designing the classifiers. We experiment with four clustering techniques and five classifiers that are combined using two existing combiner methods.

TABLE IV. TABLE 1BEST CLASSIFICATION RATE ACHIEVED AT EACH FEATURE REDUCTION METHOD

Chromo	Feat. size	Clstr	2nd stage	3rd stage	Clstr-PCA
N7	1	74.29	70	66.43	**75.71**
	2	69.29	**70.71**	65.71	68.57
	3	71.43	70.71	65	**72.14**
N10	1	**71.43**	57.86	69.29	70.71
	2	71.43	59.29	68.57	**72.14**
	3	**70.71**	60	67.86	67.86
N15	1	76.43	78.57	**83.57**	82.86
	2	75	77.86	**81.43**	80
	3	74.29	67.86	**77.86**	70
N17	1	82.14	**84.29**	76.43	73.57
	2	80	**85.71**	75	73.57
	3	81.43	**85**	74.29	69.29
N22	1	**76.43**	71.43	70.71	72.86
	2	**75**	71.43	69.29	70.71
	3	71.43	**74.29**	69.29	70
N all	1	**78.57**	77.14	77.14	76.43
	2	**78.57**	75.71	76.43	73.57
	3	**77.14**	70.70	71.43	70

Additionally we propose several weighted combiners that differ in the method for creating the weight table. All types of our proposed WCC combiners yielded highest rate at different chromosomes and feature set sizes. However, no single variant yields a consistent highest rate. The SFC also yielded high results however not consistently where at some chromosomes it yielded low rates that reached zero performance. This indicates the weight table had zero values due to misclassification of classifiers at the validation stage. Further experiments with other feature reduction methods, feature set sizes are needed to achieve better rates. Additionally further investigations are required to find when each variant of the WCC combiner achieves the maximum rate.

Acknowledgment

The author sincerely thanks PAAET for financially supporting this project under grant number TR-14-02.

References

[1] Abdullah Alqallaf and Ahmed Tewfik, "Maximum Likelihood Principle for DNA Copy Number Analysis," IEEE Int'l Conference on Acoustics, Speech, and Signal Processing, IEEE/ICASSP, Taipei, Taiwan, April, 2009.

[2] L. Breiman. Bagging predictors. Machine Learning, 24:123–140, 1996.

[3] T. Ho. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):832{844, 1998.

[4] F. M. Alkoot and J. Kittler. Experimental evaluation of expert fusion strategies. Pattern Recognition Letters, 20(11-13):1361–1369, 1999.

[5] J. Kittler, M. Hatef, R. Duin, and J. Matas. On combining classifiers. IEEE Transaction on Pattern Analysis and Machine Intelligence, 20(3):226–239, 1998.

[6] Yves Moreau , Frank De Smet , Gert Thijs , Kathleen Marchal , Bart De Moor. Functional bioinformatics of microarray data: from expression to regulation. Proceedings of the IEEE, Volume:90 Issue:11.

[7] Mohammed Uddin, Kristiina Tammimies, Giovanna Pellecchia, Babak Alipanahi, Pingzhao Hu, Zhuozhi Wang, Dalila Pinto, Lynette Lau, Thomas Nalpathamkalam, Christian R Marshall, Benjamin J Blencowe, Brendan J Frey, Daniele Merico, Ryan K C Yuen, & Stephen W Scherer, Brain-expressed exons under purifying selection are enriched for de novo mutations in autism spectrum disorder. Nature Genetics, Vol. 46, Pp: 742–747 :2014.

TABLE V. RESULTS OF THE 3RD STAGE FEATURE SELECTION METHOD

Chromosome	Eigen	Combiner system	classifier	Classification rate
N7	1	SFC1R	1-NN	66.43
		SFC1N, SFC2R, SFC3R	1-NN	65.71
		WC3R	kNN	65.71
		SFC3R	neural	65
		SFC2N, SFC3N	SVM1	65
	2	SFC1N&R	1-NN	65.71
	3	SFC2R	1-NN	65
N10	1	SFC1N	neural	69.29
		SFC1R, SFC3N	neural	68.57
		SFC1N&R	SVM1	68.57
		WC1R, SFC2R	Neural	67.86
	2	SFC1N	Neural	68.57
		SFC1R, SFC2N&R,	Neural	67.86
		SFC3N	SVM.3	67.14
		SFC3N		
	3	RSM, WC1N,	Neural	67.86
		SFC1N&R, SFC2N&R	SVM1	67.86
		SFC2N	Neural	67.14
		WC3N	SVM1	67.14
		RSM, SFC1N&R		
N15	1	WC3N	kNN	83.57
		Single	neural	82.86
	2	WC3N	neural	81.43
		WC1R, WC2N		80.71
	3	WC1R	kNN	77.86
		Bagging, WC1N	kNN	77.14
N17	1	SFC1N&R	SVM1 & SVM.3	74.29
	2	WC2R	neural	75
		bagging	SVM1	74.29
	3	WC1N&R	SVM1	74.29
		RSM, WC2N, WC3N	SVM1	73.57
		SFC1N	neural	73.57
N22	1	SFC3N	SVM.3	70.71
		SFC2R	1-NN	70
		SFC1R	kNN	70
		SFC1R	Neural	70
		SFC2R, SFC3N	kNN	69.29
		SFC1N	neural	69.29
		SFC3N	SVM1	69.29
	2	SFC3N	SVM.3n	69.29
		SFC3N	eural	69.29
		SFC1N&R	Neural	68.57
		WC3R, SFC2R	SVM1	68.57
		SFC2N&R	Neural	67.86
		WC3N	SVM1	67.86
	3	SFC2N&R	Neural	69.29
		SFC3N	Neural	68.57
		SFC1N	Neural	67.86
		SFC2R	kNN	67.86
N all	1	Bagging	Neural	77.14
	2	Bagging	Neural	76.43
		WC1N, WC2R	Neural	75.71
	3	RSM, WC3R	Neural	71.43
		WC3N	Neural	70.71
		WC2R, WC3R	SVM1	70.71
		Bagging, WC1N, WC2R	Neural	70
		Bagging	SVM1	70

[8] A.R. Webb. Statistical pattern recognition, 2nd ed., John Wiley and sons. 2002