Abstract—Scleroderma is an autoimmune disease of the connective tissues, which thickens and hardens the affected areas. Recently, researchers have found evidence that genes are important factors for this disease, and there exist consistent differences in the patterns of gene expressions of skin biopsies from affected and non-affected individuals. In this paper, we apply genetic programming (GP) on the gene expression data of scleroderma and normal biopsies to evolve the classification rules that can differentiate between them. In these evolved rules, we have found six genes that have differential gene expression levels in scleroderma and normal biopsies and thus individually can classify all the samples correctly. In addition to these genes, we have also found some simple rules containing two or more genes that can classify all the samples perfectly.

I. INTRODUCTION

Scleroderma is an autoimmune disease that affects the blood vessels and connective tissues. The most evident symptom is the hardening of the skin and associated scarring. Typically the skin appears reddish or scaly in appearance. Blood vessels may also be more visible. Where large areas are affected, fat and muscle wastage will weaken limbs and affect appearance. In USA, scleroderma affects approximately 300,000 people and it is more common in women than in men [1].

According to MedicineNet, Inc. [2], there are two most common subtypes of scleroderma: diffused and limited. The diffused form of scleroderma is the most severe form of the disease and life-threatening; it rapidly progresses to hardening of a wide skin area and can cause damage to blood vessels as well as to the internal organs. It thickens the skin of the extremities, face, chest, abdomen, back or flanks and affects internal organs including esophagus, bowels, lungs with scarring (fibrosis), heart, and kidneys. High blood pressure can be a troublesome side effect. The limited form of scleroderma tends to be confined to the skin of the fingers and face. The skin changes and other features of disease tend to occur more slowly than in the diffuse form [2]. This limited form of scleroderma is sometimes called the CREST variant of scleroderma. Morphea/linear scleroderma is another form of scleroderma; it involves isolated patches of hardened skin but no internal organ [1].

Though the exact cause of the disease is yet unknown, researchers have found evidence that genes are important factors and there exist consistent differences in the patterns of gene expressions of skin biopsies from affected and non-affected individuals [2]. To get a better understanding of the disease and to identify the possible bio-markers for the disease, Whitfield et al. [3] analyzed the gene expressions patterns of scleroderma and normal biopsies through clustering of the genes of DNA microarray data.

Many computational methods have been proposed for the classification of gene expression data. Since the number of available training samples is smaller compared to the huge number of genes, most researchers first reduce the dimensionality of the problem by filtering some informative genes, and then apply classifiers. Widely used technique for gene selection is rank-based method that ranks genes using a suitable score like signal-to-noise ratio (SNR) [4]. However, these rank-based methods selects genes independently of the classifier. In addition to rank-based method, many evolutionary computation methods, which employ wrapper approaches [5] of gene selection, have been proposed to select informative subsets of genes [6]–[17]. As classifiers, clustering [4], [18]–[23], support vector machine (SVM) [6], [24]–[26] or k-nearest neighbor (kNN) [6], [12], [22]–[24], to name a few prominent ones, are used. However, over-fitting is a dominant concern of these gene selection methods and classifiers.

Recently, genetic programming [27], an evolutionary computation method based on natural selection and evolution, has been applied to classification of gene expression data [28]–[31]. In its typical implementation, using training data, a single rule or a single set of rules is evolved with GP, and then it is applied to test data to get generalized test accuracy. The main advantage of GP is that it can act as a classifier as well as a gene selection algorithm. Unlike SVM and other classifiers, the use of GP as a classifier is transparent in the sense that the mechanism used to classify patient samples is available for inspection [31]. In this paper, we apply this genetic programming to evolve classification rules for classification of scleroderma microarray data set [3]. Then we analyze those evolved rules to identify the possible biomarkers of the disease.
II. Methods

A. Genetic programming

Genetic programming [27] is an extension of the genetic algorithm (GA) in which genetic population consists of computer programs. The basic difference between genetic algorithm and genetic programming is the representation of an individual in the genetic population. In GA, an individual is usually a fixed-length string of symbols whereas in GP, an individual is a variable length tree composed of functions and variables. Thereby, the crossover and the mutation operators are applied in different ways. In gene expressions based classification, the individuals in a GP population are S-expression of classification rules consisting of functions and variables corresponding to the genes of a microarray data set. Let the expression of a rule be represented by $R_{expr}$, and its output on a sample $Y$ is a real-valued number for a sample. In the typical implementation of a genetic programming classifier, the class of test sample $Y$ is predicted as follows:

$$Class(Y) = \begin{cases} 
'\text{A}' & \text{if } R_{expr}(Y) \geq 0; \\
'\text{B}' & \text{if } R_{expr}(Y) < 0.
\end{cases}$$

An example of $R_{expr}$ is as follows:

$$(2 \times X2474 - X1265/X1223)$$

where $X2474$, $X1265$ and $X1223$ correspond to the expression levels of genes 2474, 1265 and 1223, respectively. There are many parameters like population size, maximum depth of a rule, crossover depth, probability of crossing over, etc. associated with genetic programming. Detailed descriptions on genetic programming can be found in [27]. Using Figs. 1 and 2, we have shown how the fitness calculation of an individual varies in genetic algorithm and genetic programming for classification of gene expression data.

1) Components of a GP S-expression: Each S-expression in a population consists of randomly chosen functions and genes. Each gene in the rule is followed by an 'X' represented by the gene number. For example, $X1314$ represents gene 1314 of a data set. Functions, arithmetic and/or logical functions can be used for the evolution of classification rules. During the coding of genetic programming, we have to choose an appropriate function set depending on the targeted output. If we want Boolean outputs, we consider only arithmetic and logical functions like $\{+,-,*,/,$ $sqr,$ sqrt, exp, and, or, not,$ $>, <, <=, =\}$ or only Boolean functions like $\{and, or, not,$ xor, $\geq, <, <=, =\}$. If our targeted output is real, we consider only arithmetic functions like $\{+,-,*,/,$ $sqr,$ sqrt, ln, exp, power, sin, cos, tan\}$. An example of generating a S-expression of a rule of maximum depth 5 from the function and terminal sets of $\{+,-,*,/,$ $sqr\}$ and $\{X1, X2, X3\}$ is shown in Fig. 3.

2) Evaluation of a rule: The success of an evolutionary computation method is very much dependent on the fitness function used to measure the goodness of an individual. For classification problems, the accuracy or the error rate of a predicting program can be used as a fitness measure; however, these methods may not get the optimum fitness. Matthews [32] proposed correlation between the predicted and the observed reality as the measure of raw fitness of a predicting program. For a binary classification problem, the correlation ($C$) is calculated as follows:

$$C = \frac{N_{tp}N_{tn} - N_{fp}N_{fn}}{\sqrt{(N_{tn} + N_{fp})(N_{tn} + N_{fn})(N_{tp} + N_{fn})(N_{tp} + N_{fp})}}$$

where $N_{tp}$, $N_{tn}$, $N_{fp}$ and $N_{fn}$ are the number of true positives, true negatives, false positives and false negatives, respectively. When the denominator of equation (1) is 0, $C$ is set to 0. The standardized fitness of a rule is calculated as follows:

$$fitness\text{(rule)} = \frac{1 + C}{2}.$$  

Since $C$ ranges between -1.0 and +1.0, the standardized fitness ranges between 0.0 and +1.0, the higher values being better and 1.0 being the best. The ultimate objective of GP is to find a rule that can classify all the samples correctly and thus has fitness = 1.0. Steps required in calculation of fitness of a genetic programming rule are shown in Fig. 2.

During execution of the expression of a rule on a sample, we take precautions so that the two functions ‘sqr’ and ‘/’ do not produce undefined results. In the case of undefined results, we treat them as follows: $\frac{x}{0} = 1$, and $\sqrt{x} = 0$ if $x < 0$. Note that after adjustment, $\sqrt{|x|^2} \neq \sqrt{x^2}$. For example, if $x = -3$, then $\sqrt{(x)^2} = 3$ while $\sqrt{|x|^2} = 0$. Similarly, $z \times (x/y) \neq (z \times x)/y$; if $y = 0$, then $z \times (x/y) = z$ while $(z \times x)/y = 1$.

3) Creation of new rules: New rules are created by applying crossover and mutation to the rules selected from the old population. Since the population size is very large in genetic programming classifier, we apply the greedy-over selection method [27] to select two parents (individuals) for branch-type crossover. For mutation, a node from the tree of the selected parent is randomly chosen. If the node is a function, it is replaced with another function of the same
type; if it is a terminal, it is replaced with another terminal.

III. EXPERIMENTS

A. Microarray data set

The scleroderma data set [3] contains the expression levels of more than 12000 genes across 27 oligonucleotide microarrays of systemic sclerosis (SSc) and normal biopsies. Out of these 27 oligonucleotide microarrays, 12 are signal amplification replicates. The full set of data is available at http://genome-www.stanford.edu/scleroderma. From the website, we downloaded the preprocessed data set containing 7777 genes. Then we divided this data set into two mutually exclusive training and test subsets containing 22 and 5 samples. The five test samples are: G.U95A, I.U95A2, K.U95A, O.U95A and T.U95A.

B. Values of different parameters

The values of different genetic programming parameters were: population size=4000; maximum number of nodes (components) in the tree of a rule=100; maximum number of generations in a run=100; maximum crossover depth=7; maximum initial depth=6; mutation probability=0.1; crossover probability=0.9, and reproduction probability=0.1. The initial population of each run was generated using the ramped half-and-half method [27]. We used elitism so that the best found rule of a population survived for the next generation. In a run, the algorithm terminated when either all the training samples were correctly classified or maximum number of generations had passed.

C. Results

In this section, we present the experimental results on the scleroderma data set. In the results, during describing an important gene, we follow a notation like X1234 (RPS) [M30845] where X1234 is the serial number of the gene in the preprocessed data set containing 7777 genes, RPS is the official symbol of the gene, and M30845 is the GenBank accession number of the mRNA sequence of the gene.

1) Perfect classification rules using arithmetic functions:

First, we performed experiments on the data divided into mu-
Expression of the rule

\[ X_1 \geq X_2 \text{ THEN } 'SSc' \text{ ELSE } 'Normal'. \]

Out of 200 rules, we found 5 single gene rules (\( R_1 : R_5 \) in Table I) that have differential gene expression values in normal and scleroderma samples. These five genes include X1746 (ARRB2) [AF106941], involved in agonist-mediated desensitization of G protein-coupled receptors [33]; X1996 (LGALS3BP) [L13210], a beta-galactoside-binding protein implicated in cell transformation and cancer metastasis [34]; X2381 (MX1) [M33882], a cellular protein that becomes upregulated in epithelial cells in response to andes virus infection [35]; X4534 (FKBP1A) [M34539], a cis-trans prolyl isomerase involved in protein folding that binds the immunosuppressive drugs FK506 and rapamycin [36]; and X6329 (PFN1) [J03191], a ubiquitous actin monomer-binding protein gene that is thought to regulate actin polymerization in response to extracellular signals [37]. These genes are expressed at a higher level in scleroderma tissues relative to normal tissues, and they could be the markers for the disease. In addition to these genes, genetic programming also evolved four other rules that can perfectly classify all the samples. In Table I, rules \( R_6 : R_9 \) represent these perfect rules. Among the genes involved in these rules, X1636 (COL15A1) [L25286] in rule \( R_6 \) is of particular biological interest because its deficiency in mouse tissues is known to be associated with muscle and microvessel deterioration [37].

Next we, performed experiments on the whole (training+test) data to evolve perfect classification rules. In addition to single gene rules, we got four other rules (\( R_{10} : R_{13} \)) that can classify all the samples correctly. Since there may exist some expressions that have positive output in normal samples and negative output in scleroderma samples, we performed experiments on the whole data with the following decision rule:

\[ \text{IF (expression(Y) \geq 0) THEN Normal ELSE SSc.} \]

The perfect rules found in this way are \( R_{14} : R_{29} \) in Table I. Here also we got a gene: X5238 (LTBP4) [Y13622] that has positive gene expression levels in normal sample while negative gene expression levels in scleroderma samples, and therefore, it is another potential candidate for the biomarker of the disease. One interesting observation here is that the gene X1982 (TOMM7) [AI557912] in \( R_{23} \) is an irrelevant gene—it has no effect on classification accuracy.

2) Majority voting accuracy: In the previous section, we have presented 29 rules that can perfectly classify all the five test samples. However, all the evolved genetic programming rules are not able to classify all the test samples correctly; some rules can classify 3 out of 5 samples correctly. To get better test accuracies, we apply majority voting genetic programming classifier (MVGPC) [38] to predict the labels of the test samples. In its typical implementation, we evolve multiple rules in different GP runs, apply them one by one on a test sample and count their votes in favor of a particular class. Then the sample is assigned to the class that gets the highest number of votes in favor of it. However, the success of majority voting depends on the number of members in a voting group. We performed different experiments with number of rules per ensemble: 3, 5, and 22. We got the best test accuracies when the number of rules in each ensemble of MVGPC was equal to the number of training samples (=22). In Table II, we present these best results. For each experiment, the average accuracy of 22 single rules is presented with the standard deviation under the column \( Average \pm Stdev \). Using majority voting technique, we got 100% test accuracy in 24 cases of 25.

To verify that MVGPC is not biased towards any fixed splitting of the data set into training and test subsets, we performed 25 additional experiments where the five test samples were selected randomly (however, the numbers of systemic sclerosis and normal samples in the test subset were always 3 and 2, respectively), and the number of rules in each ensemble of MVGPC was equal to the number of training samples (=22). Here in 21 cases out of 25, the ensembles...
of 22 rules could classify all the five test samples perfectly, and in the remaining four cases, the ensembles could classify four test samples correctly. These evidences strongly suggest that majority voting technique is an appropriate method for prediction of the labels of test samples.

3) Evolution of rules using logical functions: In subsection III-C.1, we have predicted the label of a sample based on the real-valued output. However, the output value may not be very much interesting to biologists. For example, if the outputs of two rules A and B are 345.67 and 0.1, respectively, both will predict the class as 'SSc' (or 'Normal' based on the case) but the output of rule A is stronger than that of rule B. To overcome this limitation, we can use logical functions instead of arithmetic functions to evolve rules; in this case, the output will be a Boolean value—either true or false.

We performed additional experiments on the data set with seven logical functions: {>,<,>=,<=, AND, OR, NOT}. By running genetic programming with these logical functions and the training data set (fixed split), we did not find any rule that classified the five test samples correctly; all the rules could classify 3 out of 5 test samples correctly. However, MVGPC with 22 rules in the ensemble could classify all the five test samples accurately in 18 times out of 25 experiments; in the remaining cases, it could correctly classify four test samples.

To get some perfect classification rules, we executed again genetic programming with the logical function and using all the available samples as training data. Some of the evolved rules that can perfectly classify all the samples are shown in Table III. Here most of the evolved rules are very simple.

4) Biological Inference: One interesting observation from the results presented before is that the six genes that have differential expression levels in SSc and normal samples are not included in the multi-gene perfect rules (except rule R23). This implies that the single genes in the multi-gene perfect rules may not have the capability to cause the scleroderma disease individually but in joint interactions with other genes, they may become the active agents for the disease. Therefore, to get better understanding of the scleroderma disease, we should first perform intensive study on the six single genes that have differential expression levels in scleroderma and normal samples; if that does not work, we should then study other genes in other perfect rules for possible biomarkers.

IV. CONCLUSION

In this paper, we analyzed the Scleroderma microarray gene expression data with genetic programming classifier. By performing experiments, we have identified six genes that have differential gene expression levels in scleroderma and normal samples, and they are the potential candidate for the biomarkers of the disease. In addition to these genes, we have presented some other rules that can perfectly classify all the samples.

However, when the data is divided into mutually exclusive training and test subsets, all the evolved rules using the training data are not able to classify all the test samples correctly. We have shown that we can get higher test accuracies when majority voting technique is employed to classify the test samples. By performing different experiments with majority voting, we have observed that the best test accuracies can be achieved when the number of rules per ensemble is equal to the number of training samples.

REFERENCES


