

Prediction of Sickle Cell Anemia Patient's Response to Hydroxyurea Treatment Using ARTMAP Network

Hongyu Xu, Faramarz Valafar, Marko Vuskovic
Department of Computer Science
San Diego State University
San Diego, CA 92182-7720

Abstract

This paper discusses the application of the newly developed Mahalanobis distance-based ARTMAP (MART) network to the prediction of sickle cell anemia patients' response to Hydroxyurea treatment. Mahalanobis distance-based ARTMAP is a modified version of the Fuzzy ARTMAP networks where the activation and choice function are the same and equal to the Mahalanobis distance between the pattern and the template. A recurrent formula has been derived to update the inverse of a covariance matrix after it resonates with an input pattern. The full covariance matrices provide a better coverage of samples than the diagonal covariance matrices. The first part of the paper describes the MART network, with a short introduction of fuzzy ARTMAP and Gaussian ARTMAP networks, from which it was derived. The second part of the paper describes the clinical data and the procedures for data preprocessing, training and testing. The classification accuracy achieved with MART classifier was as high as 92.59%

I INTRODUCTION

In the United States, about 1 in 500 African Americans develops sickle cell anemia [5]. In Africa, about 1 in 100 individuals develop the disease. Sickle cell anemia is a genetic disorder, caused by single point mutation in the beta globin gene that changes from CCTGAGG to CCTGTGG. As a result of this mutation, the molecules of sickle cell hemoglobin adhere to each other and polymerize into long fibers that distort red blood cells (RBC) into a sickle shape. The sickled cells tend to stick in narrow blood vessels, blocking the flow of blood. Many sickle cell patients die before the age of 20. Current research suggests that increasing fetal hemoglobin (HbF) level may be effective treatment for sickle cell anemia, which can prolong these individuals' lives into their 40s and 50s.

In 1983, a drug called hydroxyurea (HU) was first used on sickle cell patients [5], [1]. Patients who responded to HU with an increasing amount of HbF in their F-cells or the proportion of F-cells experienced less frequent and severe painful crises. However, HU can be quite toxic when used continuously to maintain

elevated levels of fetal hemoglobin and also can increase the risk of leukemia [10]. Furthermore, not all patients show an increase in HbF expression to levels that would alleviate any of the disease's symptoms. In this paper we, therefore, aim at developing a system that can predict which patients will respond positively enough to the medication so that the number (or level) of their disease-related crises is reduced. The goal here is to only administer HU to the "positive responders" and not subject those who will not benefit from the medication to the negative side effects of the drug.

A group of scientists at University of Georgia and the Medical College of Georgia has engaged five years in studying the effectiveness of HU treatment. Their work is summarized in paper [11]. The prediction of patient's response to HU treatment has been formulated as a pattern recognition problem. There are two general approaches in pattern prediction (or classification): statistical and neural network approach. The latter approach is very attractive because it provides better generalization. Several two-layer perceptrons with back propagation have been investigated and used in [11]. It was noticed that for certain labeling methods (which will be discussed later in section IIIB) the neural networks have failed to reach a reasonable level of accuracy.

In this paper we use the same data set obtained from the authors of [11] to which we have applied an entirely new type of neural networks that belong to the class of adaptive resonance theory (ART) networks, originally invented by Grossberg and Carpenter and their collaborators [8], [2]-[4] from the Boston University. There are several advantages of ART networks over the back-propagation networks which have motivated the work in this paper: (a) ART networks can be used on line with a stream of experimental data, while BP networks are naturally batch oriented networks. This ability is called "incremental learning"; (b) ART network are generally plastic and stable, which means that they are able to learn new knowledge and retain the previously learned knowledge; (c) ART networks are easy to train, their training and classification is faster; (d) ART networks can generally achieve better accuracy at smaller number of processing nodes.

In order to achieve the maximum performance, we have used a new version of ART network called "Mahalanobis

distance based ARTMAP network" (MART), which was developed at SDSU [13], [14] and [16]. The initial results have proven to be very encouraging. Since the implementation of the network is sequential, rather than parallel, we will refer to the network as "classifier" or "algorithm".

In the rest of the paper, the original ART and ARTMAP classifiers will be shortly discussed, which will give a good base for presenting the newly developed MART classifier. Following this presentation, we will discuss the application of the classifier to the sickle data, and to the prediction of patient's response to HU treatment.

II CLASSIFICATION BASED ON ART NETWORKS

Since MART network was developed as a generalization of Gaussian ARTMAP network [15], and the latter was based on the original ART and ARTMAP networks [2]-[4], we will shortly discuss these networks first, in the following two sections.

A. Fuzzy ARTMAP

The original ART-based network was proposed by Grossberg [8], and by Carpenter and Grossberg [2]. The network, named ART-1, was intended for unsupervised clustering of binary data. The highly parallel network architecture was based on the model of human cognition process. Later, the same authors have enabled the network for classification of analog patterns (Fuzzy ART-1). The fundamental achievement of the ART-1 and Fuzzy ART-1 was in resolution of the "plasticity-stability dilemma", that is, the networks were able to acquire new knowledge without forgetting the previously learned behaviors. The network is essentially adaptive and on-line, which made it very attractive for many practical applications. Carpenter, Grossberg and their associates have extended the ART networks to supervised clustering, i.e. to classification of binary and analog patterns [3], [4]. The new network was named Fuzzy ARTMAP network. These networks also have sequential implementations, which in some papers, for example [9], are called "simplified fuzzy ARTMAP networks". In order to describe the network proposed by authors of this paper, we first present the sequential version of the fuzzy ARTMAP.

Let X be a set of samples (patterns) $\mathbf{x} = (x_1, x_2, \dots, x_D)$, where x_i is feature, and D is dimension of the feature space. Suppose there is a set of K classes $C = \{c_1, c_2, \dots, c_K\}$ such that each pattern from X is associated with one of the classes from C . If the set X is used for network training, then we assume that patterns $\mathbf{x} \in X$ are labeled, i.e. $label(\mathbf{x}) \in C$.

The goal of the network training is to find a set of m labeled templates $W = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m\}$, $label(\mathbf{w}_j) \in C$, such that $d(\mathbf{x}, \mathbf{w}_j) < d(\mathbf{x}, \mathbf{w}_k)$ for each

$j \neq k$ implies $label(\mathbf{x}) = label(\mathbf{w}_j)$, where $d(\mathbf{x}, \mathbf{w}_j)$ is some measure of closeness (distance) of \mathbf{x} to \mathbf{w}_j . In other words, the templates \mathbf{w}_j best represent clusters of samples. Once trained, the network should be able to classify any pattern $\mathbf{y} \notin X$ with an unknown label. This is achieved by finding the template \mathbf{w}_y closest to \mathbf{y} , which implies $label(\mathbf{y}) = label(\mathbf{w}_y)$. The algorithm uses two additional sets. The first set maps templates into labels $L = \{l_1, l_2, \dots, l_m\}$, $l_j = label(\mathbf{w}_j)$, while the other set $N = \{N_1, N_2, \dots, N_m\}$ represents the cluster sizes, i.e. the number of samples from X that are associated with templates. Since the sets are ordered, we will use the following notation in the further discussions: $W(j) = \mathbf{w}_j$, $L(j) = l_j$ and $N(j) = N_j$. It is important to note that sets W , L and N are essentially dynamic arrays, which can grow in the process of training.

```

C = {};           Initialize set of classes (categories)
L = {};           Initialize the map (set of labels)
W = {};           Initialize set of templates
N = {};           Initialize the cluster sizes
while (X not empty) Learning loop
{ get x;           Get a labeled pattern from X
  new = true;      Set flag "new node needed"
  if (label(x) ∉ C) If the class hasn't been seen so far
    C := C ⊕ label(x); Add new class to C
  else
  { loop j = 1, m   Compute activation value
    t_j = T(x, w_j); for all templates
    loop i = 1, m   Search for resonance
    { Find template with highest
      J = arg max_{j ≤ m} T_j; activation value
    }
    if (label(w_j) = label(x))
    { If the label of the winner match
      if (M(x, w_j) > ρ and the resonance occurs
      { w_j := U(w_j, x); update the template
        new = false; No new node needed
        break; Stop the search for
          the resonant node
      }
      else If no match
        ρ = T_j + ε; Increase vigilance
      }
      T_j = 0; w_j is no longer a candidate
    }
    Continue search for resonant node
  }
  NEWNODE(NEW); Create new node if needed
} End of learning loop

```

Figure 1: Pseudo code of ARTMAP clustering algorithm

NEWNODE(new) in Figure 1 is a macro routine that creates a new node (template):

```

if new == true
{ m := m + 1; Increment current number of templates
  N := N ⊕ 1; The new cluster has one sample
  wm = x; The new template is initially equal to x
  W := W ⊕ wm; Add new template to the set W
  L := L ⊕ label(x); Add label of x to the set L
}

```

The three functions, $T()$, $M()$ and $U()$, used in the algorithm above are defined as follows:

$T(\mathbf{x}, \mathbf{w}_j)$ is the *choice function* which is used as a closeness measure [2]-[4]:

$$T(\mathbf{x}, \mathbf{w}_j) = \frac{\|\mathbf{x} \cap \mathbf{w}_j\|}{\alpha + \|\mathbf{w}_j\|}, \quad (1)$$

where α , $\alpha > 0$, is a choice parameter.

$M(\mathbf{x}, \mathbf{w}_j)$ is the *match function* used to quantify how good is the closeness of \mathbf{x} to \mathbf{w}_j

$$M(\mathbf{x}, \mathbf{w}_j) = \frac{\|\mathbf{x} \cap \mathbf{w}_j\|}{\|\mathbf{x}\|}, \quad (2)$$

The function is used in conjunction with the *vigilance parameter* $\rho \in (0, 1]$, where $M(\mathbf{x}, \mathbf{w}_j) > \rho$ means a good match (resonance). The vigilance parameters is the most important network parameter that determines its resolution: larger vigilance value normally yield larger number of output nodes.

The third function is the *update function*:

$$U(\mathbf{w}_j, \mathbf{x}) = (1 - \beta)\mathbf{w}_j + \beta(\mathbf{x} \cap \mathbf{w}_j) \quad (3)$$

where β is learning rate, $0 < \beta < 1$. Higher values of β result in faster learning.

The operators in equations (1) through (3) are used in ARTMAP for classifying binary data: $\mathbf{a} \cap \mathbf{b} = (a_1 \text{ AND } b_1, a_2 \text{ AND } b_2, \dots, a_D \text{ AND } b_D)$ and $\|\mathbf{a}\| = \sum a_i$. In Fuzzy ARTMAP the bitwise AND operator \cap , is replaced by the fuzzy AND operator \wedge : $\mathbf{a} \wedge \mathbf{b} = (\min(a_1, b_1), \min(a_2, b_2), \dots, \min(a_D, b_D))$.

The classification of an unlabeled pattern \mathbf{y} is achieved by maximization of the activation function, i.e.

$$class = L(\arg \max_{j \leq m} T_j(\mathbf{y}, \mathbf{w}_j)) \quad (4)$$

B Gaussian ARTMAP

Based on fuzzy ARTMAP algorithm, Williamson [15] has proposed Gaussian ARTMAP, which uses

multivariate independent Gaussian distribution function as activation and match functions:

$$M(\mathbf{x}, \mathbf{w}_j, \boldsymbol{\sigma}_j) = -\frac{1}{2} \sum_{i=1}^D \left(\frac{x_i - w_{ji}}{\sigma_{ji}} \right)^2 \quad (5)$$

$$\begin{aligned} T(\mathbf{x}, \mathbf{w}_j, \boldsymbol{\sigma}_j) &= \ln P(j | \mathbf{x}) = \\ &= M(\mathbf{x}, \mathbf{w}_j, \boldsymbol{\sigma}_j) - \ln \left(\prod_{i=1}^D \sigma_{ji} \right) + \ln(P(j)) \end{aligned} \quad (6)$$

where $\boldsymbol{\sigma}_j$ and \mathbf{w}_j are parameters of normally distributed conditional probability density function of samples \mathbf{x} given cluster j , $P(j)$ is prior probability of cluster j , $P(j|\mathbf{x})$ is posterior probability of j given \mathbf{x} . The cluster updates require updates of three parameters:

$$(n_j, \mathbf{w}_j, \boldsymbol{\sigma}_j) := U(n_j, \mathbf{w}_j, \boldsymbol{\sigma}_j) \quad (7)$$

where $U()$ represents:

$$\begin{aligned} n_j &:= n_j + 1; \\ w_{ji} &:= (1 - \beta_j)w_{ji} + \beta_j x_i; \quad i = 1, 2, \dots, D \\ \sigma_{ji} &:= \sqrt{(1 - \beta_j)\sigma_{ji}^2 + \beta_j(w_{ji} - x_i)^2}; \end{aligned} \quad (8)$$

The learning rate β_j is different for different clusters and depends on the cluster size N_j , i.e. $\beta_j = 1/N_j$

Since several clusters can be associated with the same class c_k , the posterior probability of the class given sample \mathbf{x} can be expressed as:

$$P(c_k | \mathbf{x}) = \sum_{j, L(j)=c_k} P(j | \mathbf{x}) = \sum_{j, L(j)=c_k} \exp(T(\mathbf{x}, \mathbf{w}_j, \boldsymbol{\sigma}_j)) \quad (9)$$

Consequently, the maximum likelihood estimate of the class of an unknown (unlabeled) sample \mathbf{y} will be given as:

$$class = \arg \max_{c_k} (P(c_k | \mathbf{y})) \quad (10)$$

Williamson has shown [15] in several benchmarks that the Gaussian ARTMAP has better performance than fuzzy ARTMAP in terms of the hit rate and the number of required output nodes.

C Mahalanobis distance based ARTMAP

Mahalanobis distance based ARTMAP (MART) is a generalization of Williamson's Gaussian ARTMAP. It utilizes full covariance matrices of clusters instead of their diagonal part. The algorithm was first proposed in [13], then further investigated and improved in [14] and [16]. The new algorithm uses for the activation function and for the match function the same quantity, the Mahalanobis distance between the sample \mathbf{x} and the template \mathbf{w}_j :

$$T(\mathbf{x}, \mathbf{w}_j, \mathbf{Q}_j) = (\mathbf{x} - \mathbf{w}_j)^T \mathbf{Q}_j (\mathbf{x} - \mathbf{w}_j), \quad (11)$$

where \mathbf{Q}_j is the inverse of the covariance matrix of the cluster j . This has contributed to a significant speedup, which will be discussed shortly below. The algorithm is apparently more numerically intensive since the full covariance matrix uses more parameters, but provides much better coverage of clusters at fewer output nodes. It was shown in [13], [14] and [16], that in some instances the

algorithm provides even lower time overhead despite the fact that the computation of covariance matrices requires more floating point operations, due to fewer required output nodes. Numerous experiments on several benchmarks have shown that using the identical function for activation and match doesn't diminish the classification-hit rate; in many cases it even improves the ability of the network to classify. The pseudo code of the algorithm is as follows:

```

C = {};           Initialize set of classes (categories)
L = {};           Initialize the map (set of labels)
W = {};           Initialize set of templates
M = 0;           Initialize the current number of templates
N = {};           Initialize the cluster sizes
while (X not empty)           Learning loop
{
  get x;           Get a labeled pattern from X
  new = true;      Set flag "new node needed"
  if (label(x) ∉ C)   If class hasn't been seen so far
    C := C ⊕ label(x);   add new class to C
  else
  {
    loop j = 1, m           Compute activation value
    {
      if (label(wj) = label(x))   for templates with
        tj = T(x, wj);           the same labels as x
    }
    J = arg minj ≤ m T(x, wj, Qj); Find closet template for x
    if (tJ < ρ)           If the winner is close enough
    {
      (NJ, wJ, QJ) := U(NJ, wJ, QJ); Update the
      new = false; template. No new node needed
    }
  }
  NEWNODE(new);       Create new node if necessary
}
End of learning loop

```

Figure 2: Pseudo code of MART clustering algorithm

As shown, this algorithm has three important differences with respect the code in Figure 1. First, the j -loop, which searches for resonant node, has been eliminated. This was possible since the activation function and the match function are the same, thus the most active node is at the same time the closest node to the sample \mathbf{x} . Therefore if that node doesn't satisfy the match criterion, there is no need to continue with the search for the resonance among other candidates. This speeds up the algorithm considerably. Secondly, the activation function is computed only at the label match, which has reduced many function evaluations. Thirdly, the vigilance is kept constant. Numerous experiments have confirmed that the change of the vigilance didn't improve the network performance.

The update functions are implemented as:

$$\mathbf{w}_J := (1 - \beta)\mathbf{w}_J + \beta\mathbf{x}, \quad (12)$$

$$\mathbf{Q}_J := \beta_J \left(\mathbf{Q}_J - \frac{\mathbf{g}_J \mathbf{g}_J^T}{\beta_2 + T_J} \right), \quad (13)$$

where

$$\beta_1 = \frac{1 - \beta}{1 - 2\beta}, \quad \beta_2 = \frac{1}{\beta\beta_1}, \quad \beta = \frac{1}{N_J} \quad (14)$$

$$\mathbf{g}_J = \mathbf{Q}_J(\mathbf{x} - \mathbf{w}_J) \quad (15)$$

In practical implementation the learning rates β_j are kept constant for very small and very large values of N_j , typically $\beta = 1/10$ and $\beta = 1/1000$, for values of N_j less than 10 and greater than 1000 respectively. It should be noted that the algorithm is making use of the inverse covariance matrices $\mathbf{Q}_j = \mathbf{S}_j^{-1}$, $\mathbf{S}_j = \text{cov}(X_j)$, with X_j being the subset of X associated with the j -th cluster. The derivation of the recurrent equation (13) was based on the Sherman-Morrison-Woodbury matrix identity [7].

The *NEWNODE()* macro is now given an additional line: $\mathbf{Q} := \mathbf{Q} \oplus \mathbf{Q}_o$, where \mathbf{Q}_o is the initial value of covariance matrices \mathbf{Q}_j . It was found in [14] that this matrix can be simply $\mathbf{Q}_o = \mathbf{I}R_o^{-2}$, where \mathbf{I} is D -dimensional identity matrix. In other words \mathbf{Q}_o represents an initial sphere with radius R_o .

In the light of the discussion above, it can be said that the MART classifier has two scalar parameters, ρ and R_o . The former parameter, the network vigilance can be viewed as $\rho = \chi^2(D, p)$, since the function (11) has chi-square distribution with D degrees of freedom for normally distributed \mathbf{x} . The parameter p is the probability that a sample \mathbf{x} is contained within the ellipsoid with parameters $\{\mathbf{w}, \mathbf{Q}, \rho\}$.

III EXPERIMENTAL RESULTS

A. The Clinical Data

The sickle cell patient data contains the information from 92 patients [11]. For each patient 26 parameters were collected. The parameters are listed in TABLE I.

TABLE I: Parameters Collected from Patients

Parameter	Description
1 Age	Age of patient at the time of analysis
2 Sex	Male/Female
3 NAGG	α Globin gene number
4 SBAN	Number of BAN haplotypes
5 SBEN	Number of BEN haplotypes
6 SCAM	Number of CAM haplotypes
7 SSEN	Number of SEN haplotypes
8 TotalTx	
9 WGT	Weight of patient
10 %HbF	Fetal hemoglobin, as % of total hemoglobin
11 HbF	Fetal Hemoglobin, absolute value
12 Hb	Total hemoglobin concentration

13	RBC	Red blood cell count
14	RDW	% Variation in the size of red cells
15	PCV	Packed cell volume (hematocrit)
16	Retic	Reticulocytes
17	MCV	Mean cell (erythrocyte) volume
18	MCH	Mean cell hemoglobin
19	WBC	White cell count
20	Polys	Polymorphonuclear leukocytes
21	Plats	Platelet count
22	Bili	Bilirubin concentration in blood
23	SNRBC	Nucleated RBC seen in peripheral blood
24	%fHbF	Max. percentage of HbF (during treatment)
25	fHbF	Maximum value of HbF (during treatment)
26	Duration	Duration of treatment a patient received to arrive at the maximum %HbF level

B. Data Labeling

The list of parameters (features) in TABLE I has no indication about the label. In order to enable prediction of patient's response to HU treatment, the available patterns have to be properly labeled. For this purpose the data set will be divided into two classes: patients who have responded to HU treatment, and the patients who have not responded. There are two widely accepted criteria to define responders and non-responders [11]. The first criterion considers the responding patients those who have level of fHbF increased during the treatment at least two times over the initial value HbF. We will call this method of labeling the "Double Rule". The second criterion marks the responders if their final %HbF (%fHbF) is over 15%, provided that their initial value %HbF is under 15%. The criteria are illustrated in Figure 3.

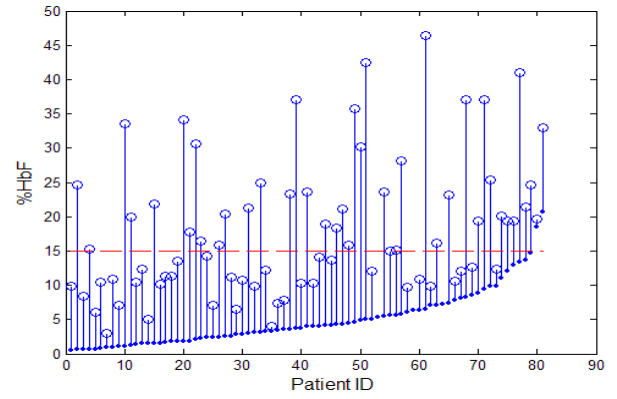
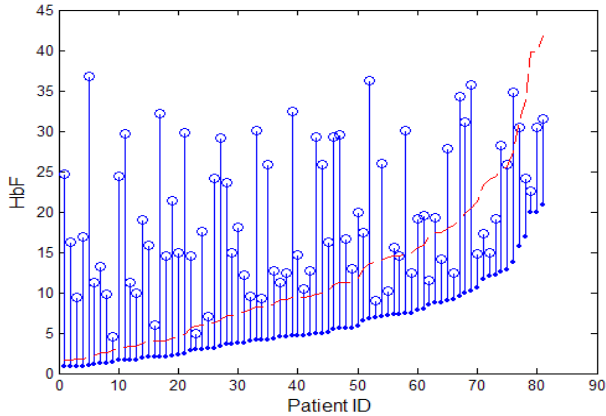


Figure 3: Label assignments: double Rule (above) and 15% rule (below). Figure show patient's HbF data sorted according to initial values of HbF or %HbF. The circles represent the final values of HbF or %HbF. The dashed lines define the label partitioning (patients with fHbF above dashed lines are labeled as responders)

The distribution of responders and non-responders corresponding to the two criteria is shown in TABLE II.

TABLE II: The Distribution of Responders and Non-responders in the Sample

Criterion	Responders	Non-responders	Total
Double Rule	63	18	81
15% Rule	45	31	76

It should be noted that the patients with incomplete data were excluded from the data set considered in training and testing, as shown below.

C. Treatment of Linearly Dependent and Missing Data

During the data processing it was noticed that some features are linearly dependent. This can be described as follows. Suppose that the pattern $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \dots, x_D^{(i)})$ represents the i th patient. Then the vector $\xi_k = (x_k^{(1)}, x_k^{(2)}, \dots, x_k^{(N)})$ represents the same feature across the entire set of patients. We consider that features i, j and k are linearly dependent if the vectors ξ_i, ξ_j and ξ_k are linearly dependent. It can be shown [16] that the linear dependency of features can be revealed by examining the eigenvectors and eigenvalues of the covariance matrix of the entire data set. Using this approach it was determined that features 4, 5, 6 and 7 (see TABLE I) are linearly dependent. Therefore, only the feature 4 was retained, while the others were dropped.

Similar procedure based on eigenvectors and eigenvalues of the covariance matrix was used to rank the features according to their importance [16]. The final list of features, reduced and reordered according to their importance, is shown in TABLE III. The table doesn't show the features labeled "fHbF" and "%fHbF" which are dropped because they were used for labeling.

Rank	Features
1	SNBRC
2	HbF
3	TotalTx
4	Retic
5	Bili
6	Age
7	RBC
8	Plats
9	Hb
10	RDW
11	SBAN
12	MCH
13	Sex
14	PCV
15	MCV
16	WGT
17	NAGG
18	WBC
19	Polys

In addition to feature reduction, all samples with two or more missing features were discarded. The samples with only one missing feature were retained, with the missing feature replaced by the mean value of the corresponding feature.

TABLE III: The Reduced and Reordered List of Features

D. Data Preprocessing

It is known from the experience [12] that the skewness and kurtosis of data and therefore their separability can be improved by nonlinear transformation of data. A common transformation is $\log(1+x)$, and the process is sometimes called "data gaussianization". Experiments performed in this paper have confirmed this experience; the hit rates with transformed data were couple of percents higher. Usage of the logarithm transformation eliminates the need for data normalization. The 3D plot of labeled and log-transformed samples is shown in Figure 4 and Figure 5.

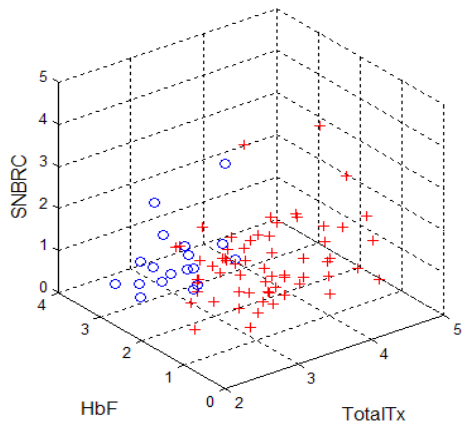


Figure 4: Plot of samples along three most significant dimensions of the feature space. Samples are labeled by the double rule (o - responders, + - non-responders)

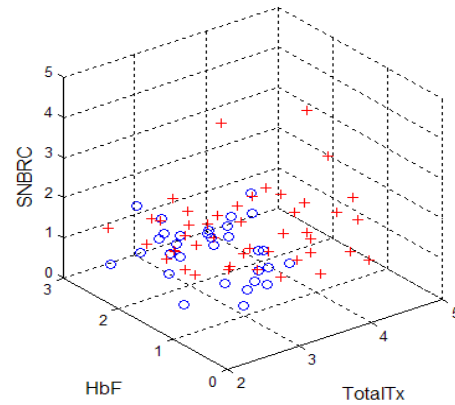


Figure 5: Same as Figure 4, only the 15% rule was used for sample labeling.

E. Training and Testing

The leave-one-out method [6] is used to test the performance of the MART network. The procedure consists of the following: A sample is extracted from the sample set. The leftover set is then used for the network training. After the network is trained, the extracted sample (which did not participate in training) is used for testing. This procedure is repeated N times, each time with a different sample left out, until all samples in data set are used for extraction and testing. The average hit rate and number of output nodes are then calculated. The results are shown in TABLE IV for two labeling methods and for two different number of features used. As seen, the results for 15% labeling rule were inferior to the double rule. The reason for this is high overlap of the clusters for the two classes, which makes the labeling method ineffective. This can be easily concluded after examining Figure 4 and Figure 5. Interestingly, usage of the subset of three most significant features (SNBRC, HbF and TotalTx) in case of double rule has given considerably better result. This can be explained by the fact that the other features have practically no relevance and act as noise. As the table shows, the classification accuracy for responders was 96.82% while the non-responders have scored 77.77%. This is considered to be a very encouraging result.

TABLE IV: Testing Results on MART Classifier

	15% Rule		Double Rule	
	# of Features		# of Features	
Accuracy of Predicting Responders	19	84.4% (38/45)	8	68.89% (31/45)
Accuracy of Predicting Non-Responders	19	45.16% (14/31)	3	100% (63/63)
Global Hit Rate	19	68.42% (14/31)	3	96.82% (61/63)
# of Output Nodes	3.50	68.42%	3.32	79.01%
				92.59%
				2.06

IV CONCLUSION

The Mahalanobis distance-based ARTMAP network, called MART classifier developed at SDSU [13],[14], [16] has been presented. Since MART has evolved from Fuzzy and Gaussian ARTMAP, these approaches were also briefly discussed. The new network was applied to the prediction of sickle cell anemia patient's response to Hydroxyurea (HU) treatment. A database for 92 patients has been obtained from a previous project [11] and used in this research. The MART classifier has shown to be very effective for the classification of HU responders, giving a classification hit rate of 92.59%. In the process of data analysis, it has been discovered that some of the features in the database are linearly dependent. These features can be therefore removed from the database. In addition, it is shown that the features can be ranked according to their relevancy to the discrimination. It is noticed that reducing the features from 19 independent features to only three most significant ones, can even improve the classification accuracy.

In the future work, the linear transformations of data, including simultaneous diagonalization, will be explored. The data labeling will also be investigated, since the currently used methods, recommended by physicians, appear to be deficient from the discrimination point of view. The data processing technology used in this work will provide a good basis for the new recommendations for data labeling.

ACKNOWLEDGEMENT

Authors would like to acknowledge Dr. Homayoun Valafar from Structural Genomics Group, University of Georgia, for his extraordinary work in collecting and preprocessing of data on sickle cell anemia patients, and for making the database available for this research.

REFERENCES

- [1] Bonds, D. (contact): "Clinical Alert: Drug Treatment for Sickle Cell Anemia," National Heart, Lung, and Blood Institute, Bethesda, MD 20892, January 1995.
- [2] Carpenter G. A. and Grossberg, S.: A massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine," *Computer Vision, Graphics, and Image Processing*, Vol. 37, 54-115, 1987.
- [3] Carpenter G.A., S. Grossberg, and J.H. Reynolds. 1991. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4:565-588.
- [4] Carpenter G.A., S. Grossberg, N. Markuzon, J.H. Reynolds, and D.B. Rosen. Sep. 1992. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3(5): 698-713.

- [5] Chowning, J. T.: "Sickle Cell Anemia Case Study: Summary", BioLab, Seattle, WA, October 2000, available at: <http://genetics-education-partnership.mbt.washington.edu>
- [6] Duda, R.O., P.E. Hart, and D.G. Strok, "Pattern Classification," John Wiley & Sons, Inc, 2001.
- [7] Golub, G. and C. Van Loan: "Matrix Computations:", John Hopkins Univ. Press, Baltimore MD, 1983.
- [8] Grossberg, S.: "Adaptive Pattern Classification and Universal Recoding, I: Parallel Development and coding of neural feature Detectors," *Biological Cybernetics*, Vol. 23, 121-134, 1976.
- [9] Kasuba, T.: "Simplified Fuzzy ARTMAP," *AI Expert*, pp18-25, November 1993.
- [10] Sickle Cell Disease Scientific Research Group: "Patient fact sheet: The Multicenter Study of Hydroxyurea in Sickle Cell Anemia (MSH)", National Heart, Lung, and Blood Institute, Bethesda, MD 20892, January 1995.
- [11] Valafar H., F. Valafar, Alan Darvill, Peter Albersheim, Abdullah Kutlar, Christy Woods, and John Hardin, 2000. Predicting the effectiveness of Hydroxyurea in Individual Sickle Cell Anemia Patients. *Journal of Artificial Intelligence in Medicine*. 18 (2): 133-148 February 2000.
- [12] Vuskovic, M., A.L. Pozos, and R. Pozos: "Classification of Grasp Modes Based on Electromyographic Patterns of Pre-shaping Motions," *IEEE Internat. Conference on Systems, Man and Cybernetics*, Vol. 1, pp 89-95, Vancouver, BC, October 22-25, 1995.
- [13] Vuskovic, M. and S. Du: "Classification of EMG Patterns with Simplified Fuzzy ARTMAP Networks," *Proc. Of the 2002 International Joint Conference on Neural Networks, Networks*, Honolulu, HI, May 12-17, 2002.
- [14] Vuskovic, M., H. Xu and S. Du: "Simplified ARTMAP Network Based on Mahalanobis distance" *Proc. Of the 2002 International Conference on Mathematics and Engineering Techniques in Medicine and Biological Science*, Las Vegas, Nevada, June 27-27, 2002.
- [15] Williamson, J. R. : "Gaussian ARTMAP: A Neural Network for Fast Incremental Learning of Noisy Multidimensional Maps," *Neural networks*, Vol.9, No.5, pp. 881-897, 1996.
- [16] Xu, H.: "Mahalanobis Distance Based ART Networks", M.S. thesis, Department of Computer Science, SDSU, 2003 (to appear in August)