Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 275831, 12 pages
http://dx.doi.org/10.1155/2015/275831

Review Article

On Feature Selection and Rule Extraction for High Dimensional Data: A Case of Diffuse Large B-Cell Lymphomas Microarrays Classification

Narissara Eiamkanitchat,[1] Nipon Theera-Umpon,[1,2] and Sansanee Auephanwiriyakul[2,3]

[1] *Department of Electrical Engineering, Faculty of Engineering, Chiang Mai University, Chiang Mai 50200, Thailand*
[2] *Biomedical Engineering Center, Chiang Mai University, Chiang Mai 50200, Thailand*
[3] *Department of Computer Engineering, Faculty of Engineering, Chiang Mai University, Chiang Mai 50200, Thailand*

Correspondence should be addressed to Nipon Theera-Umpon; nipon.t@cmu.ac.th

Received 28 April 2015; Accepted 13 September 2015

Academic Editor: Chunlin Chen

Neurofuzzy methods capable of selecting a handful of useful features are very useful in analysis of high dimensional datasets. A neurofuzzy classification scheme that can create proper linguistic features and simultaneously select informative features for a high dimensional dataset is presented and applied to the diffuse large B-cell lymphomas (DLBCL) microarray classification problem. The classification scheme is the combination of embedded linguistic feature creation and tuning algorithm, feature selection, and rule-based classification in one neural network framework. The adjustable linguistic features are embedded in the network structure via fuzzy membership functions. The network performs the classification task on the high dimensional DLBCL microarray dataset either by the direct calculation or by the rule-based approach. The 10-fold cross validation is applied to ensure the validity of the results. Very good results from both direct calculation and logical rules are achieved. The results show that the network can select a small set of informative features in this high dimensional dataset. By a comparison to other previously proposed methods, our method yields better classification performance.

1. Introduction

An innovation in computational intelligence mechanism, not only to develop the high accuracy mechanisms but also to be interpreted easily by human, is an interesting research topic. In order to achieve the interpretability purpose, linguistic features are more desirable than other types of features. An algorithm for finding appropriate symbolic descriptors to represent ordinary continuous features must be developed for classification mechanism. Enormous research works in neural networks are accomplished in classification accuracy [1–4]. The better performance of rules generated from neural network than that from the decision tree in noisy conditions was demonstrated [1]. The subset method which conducted a breadth first search for all the hidden and output nodes over the input links was proposed [2]. Knowledge insertion was applied to reduce training times and improve various features of the neural networks [3, 4]. However, these algorithms are difficult to comprehend due to the large number of parameters and the complicated structure inside the networks. The methods to extract rules from neural network without the consideration of linguistic feature have been proposed in some research works [5, 6]. Fuzzy sets are appropriate choice in preparing linguistic data to more interpretable information for humans [7–11]. The methods to transform numeric data into linguistic terms before training and then extracting the rules were proposed [12–18]. A supervised type of neural network with a structure which supported the simplicity of the rule extraction was proposed [12]. Rules were extracted from neural networks using structural learning based on the matrix of importance index [13]. Other rule extraction methods were proposed by simply determining the typical fuzzy membership functions using the expectation maximization (EM) algorithm [14] or determining the context-dependent membership functions for crisp and fuzzy linguistic variables which allowed different linguistic variables in different rules [15]. The logical rule extraction from data was proposed with an assumption that a set of symbolic or continuous valued predicate functions has been defined for some objects, thus providing values of features for categorization of these objects [16]. Nice examples of neurofuzzy methods for a medical prediction problem and a biometric classification problem can be found in [17] and [18], respectively. Nevertheless to discover proper linguistic features for continuous data representation in these methods is the tradeoff between simplicity and accuracy.

Due to linguistic feature requirement, in some situations depending on classification models, input features are m times increased corresponding to m linguistic terms for each original feature. This problem is more serious in high dimensional datasets. The high dimensional feature vectors may contain noninformative features and feature redundancy that, in turn, can cause unnecessary computation cost and difficulty in creating classification rules. Therefore, an algorithm for informative feature selection must be developed. Although some neurofuzzy methods were utilized for feature selection in high dimensional datasets, they are not widely used. On the other hand, there are several research works on the use of neurofuzzy methods as a classifier [19, 20]. In [21], a neurofuzzy method was utilized to select good features by utilizing a relationship between fuzzy criteria. In [22], a neurofuzzy scheme which combines neural networks and fuzzy rule base systems was proposed for simultaneous feature selection and fuzzy rule-based classification. There were three subprocesses in the learning phase. The network structure was changed in phases 2 and 3, and the parameters of membership functions were fine-tuned in phase 3. Although we have similar purpose with [22], the network structures and learning methods are different. In addition, our method can automatically create linguistic features and all parameters are modified automatically in one learning phase without changing the network structure or retraining network, while in [22] there is more than one learning phase, and that algorithm cannot fine-tune parameters without retraining network with different structure.

We initially proposed a neurofuzzy method that could select features and simultaneously create rules for low-dimensional datasets [23, 24]. Although the method in this paper is similar, the training process for a high dimensional dataset in this paper is different. To emphasize its usefulness, Chen and Lin have adopted our method into skin color detection [25]. With the consideration of the uncomplicated network structure, the main components including linguistic feature creation and tuning algorithm, feature selection, and rule-based classification were embedded in one neural network mechanism. The problem of using linguistic feature is to define a particular linguistic set for each feature in a given dataset. The fuzzy membership function was embedded in our network for linguistic feature creation rather than using the ordinary feature. The reason of this combination model is the applicability of the neural network's learning algorithms developed for fuzzy membership function. The original features were transformed to linguistic features and then classified to informative and noninformative classes, that is, either +1 or −1. Features with high weight values referred to as the informative features were therefore selected.

In this paper, we investigate the usefulness of our proposed neurofuzzy method by applying it to the high dimensional diffuse large B-cell lymphomas (DLBCL) microarray classification problem. The number of features in this microarray dataset (7,070 features) is much larger than what we have tried previously. Moreover, the number of samples is very small (77 samples). Therefore, this problem is very challenging and it is interesting to see whether our method would work in this dataset with huge number of features, but very small number of samples. The findings of informative features and rules will be useful in diagnosis of this kind of cancer. The results will also indicate the generalization of our method.

This paper is organized as follows. A neurofuzzy method with feature selection and rule extraction and its training scheme designed for a high dimensional dataset is described in Section 2. The experimental setup, data description, experimental results, and discussion are given in Section 3. Section 4 concludes the paper.

2. Neurofuzzy Method with Feature Selection and Rule Extraction

Three requirements are concerned in the design of a neural network structure for rule extraction. Less complication of network structure is the first requirement. Therefore, only three layers (an input, a hidden, and an output layers) in a neural network are constructed. Consistent with the first requirement, the small number of linguistic variables is the second requirement. The set of linguistic terms {small, medium, large} is sufficiently understood. The final requirement is that there is only one best combination rule used for classification. The combination rule is created with the consideration of the class order described in the next section.

The neural network designed based on the aforementioned requirements is displayed in Figure 1. The original features are fed forward to the input layer. Each original feature is reproduced m times corresponding to the number of specified linguistic terms and used as the input to the hidden layer. Instead of using the static linguistic feature from preprocessing, we add the hidden layer with fuzzy logic membership functions. A Gaussian membership function is used to represent membership value of each group. In addition to weight updating equations, the modifications of the center c and the spread σ are required during the training process. The second layer is the combination of fuzzification and informative linguistic feature classification. The class of linguistic features is decided and fed as the input to the output layer.

The number of nodes constructed in the input layer is the same number of original input features. For the forward pass, each node is reproduced m times. In our structure m is equal to 3. The number of outputs from this layer is therefore triple of that of the original input features and represented by

$$y_{ij}^{\mathrm{in}} = x_i, \tag{1}$$

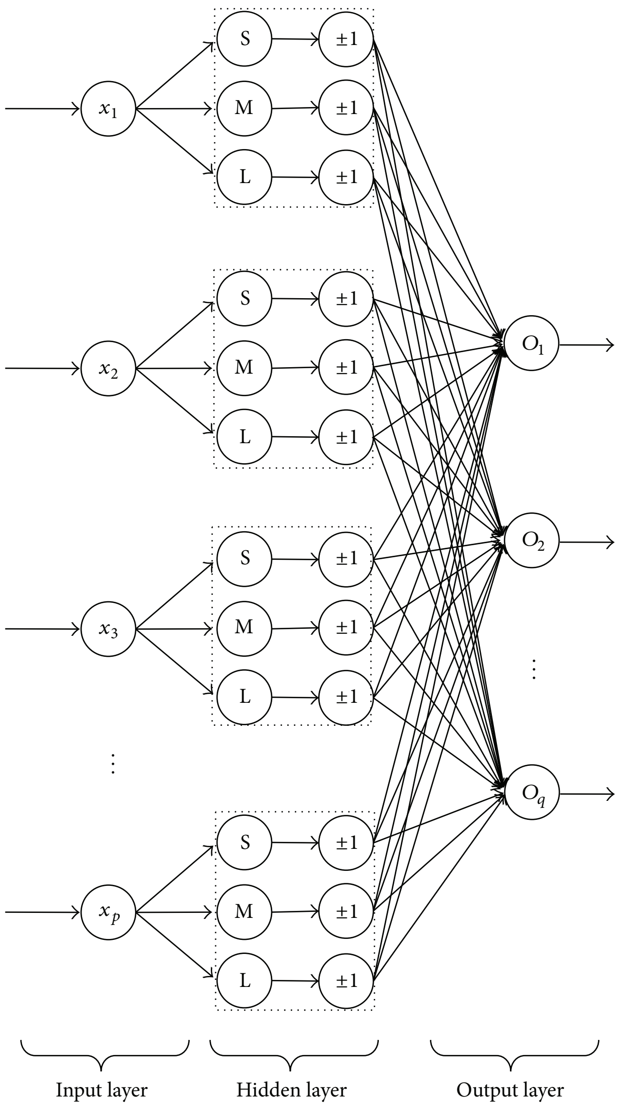


Figure 1: Our neurofuzzy classification model.

where $i = 1, 2, \ldots, p$ denotes the order number of the original input and $j = 1, 2, 3$ denotes the order number of the linguistic feature created for each original input. Between the input layer and hidden layer, all connection weights are set to unity.

The next layer is the hidden layer. In addition to calculating fuzzy values corresponding to the specified linguistics of the original input, the maximum membership value is classified to the informative class. Since we use 3 linguistic terms, that is, small (S), medium (M), and large (L), for each original feature, the number of nodes constructed in this hidden layer is 3 times of that of the input layer. Each node uses Gaussian membership function to specify the membership values of the original feature; that is,

$$\mu_{ij} = e^{(-(1/2)(y_{ij}^{\text{in}}-c_j)^2/\sigma_j^2)}. \tag{2}$$

μ_{ij} is the membership value of y_{ij}^{in} which is the original input i from the linguistic node j. Each node in this layer has two parameters to consider. The first parameter is the spread σ_{ij} for each original feature i. The initial values of σ_{ij} for j = S, M, and L come from the spreads divided by 3 ($\sigma_{ij}/3$) of all data points in linguistic term j. The second parameter is the center c_{ij}. The initial values of c_{ij} for j = S, M, and L are set to $(c_{ij}-(\sigma_{ij}/3))$, c_{ij}, and $(c_{ij}+(\sigma_{ij}/3))$, respectively, where c_{ij} is the center of the membership set of linguistic term j of original feature i.

For each original input, the most informative linguistic feature is defined as the feature with maximum membership value corresponding to the input value y_{ij}^{in}. The parameters c_j and σ_j which are mean and standard deviation of only the most informative linguistic variable are modified. Consider two linguistic terms (classes) of S (the most informative linguistic term) and T (the noninformative linguistic term); the output y_{ij}^{h} of the hidden layer is equal to +1 if the input y_{ij}^{in} belongs to class S. Alternatively, y_{ij}^{h} is equal to −1 if the input belongs to class T, for $j = 1, 2, 3$. Therefore, the output from the hidden layer is ±1. Each original feature has one informative linguistic term represented by +1 and two noninformative linguistic terms represented by −1. All outputs with identified informative values are used as input to the final layer.

In the output layer, the number of nodes is equal to the number of the classes in the dataset. Weights are fully connected between this layer and hidden layer. Hence, we utilize this layer for feature selection purpose. The importance of linguistic features is specified by the corresponding weight values. Sigmoid function is used as the activation function in this layer. Therefore the output is

$$y_k^o = \varphi(z_k) = \frac{1}{(1+e^{-z_k})}, \tag{3}$$

where

$$z_k = \sum_j w_{jk} y_j^h. \tag{4}$$

The subscript k, ordering from 1 to q, represents the class indices of the dataset. w_{jk} represents the weight connected between node j in the hidden layer and node k in the output layer. The summation of the product between weights and outputs from the hidden layer is represented by z_k.

2.1. Parameter Tuning. The algorithm for parameter tuning of the proposed model is slightly different from the conventional algorithm as Algorithm 1.

2.1.1. Weight Modification. The standard error backpropagation algorithm is used in the backward pass of the proposed model. Consider the kth neuron of the output layer at iteration n; the error signal is defined by

$$e_k(n) = d_k(n) - y_k^o(n), \tag{5}$$

where $e_k(n)$ and $d_k(n)$ represent the error signal and desired output, respectively. The output signal of neural k in the output layer is represented by $y_k^o(n)$. The instantaneous error

```
While (performance ≤ threshold)
    Compute the delta values (local gradient values) at the output nodes using (7)
    Update weights between hidden and output layers using (8)
    For each input features
        If (hidden node connecting to input feature belongs to informative class)
            Compute the delta values (local gradient values) at the output nodes using (9)
            Update mean and standard deviation using (10)
        Else
            Retain original values
        End If
    End For
End While
```

Algorithm 1

energy value of neuron k is defined as $(1/2)e_k^2(n)$. The total error energy of neuron k can be calculated by

$$E(n) = \frac{1}{2}\sum_k e_k^2(n). \tag{6}$$

Backpropagating from the output layer, the delta value (local gradient value) is defined as

$$\delta^o(n) = e_k \varphi'(z_k(n)), \tag{7}$$

where $\varphi'(z_k(n)) = \partial y_k(n)/\partial z_k(n)$. Given the learning rate η_w, the connected weights between the hidden layer and the output layer are updated by

$$w_{jk}(n+1) = w_{jk}(n) - \eta_w \delta^o(n)\, y_j^h(n). \tag{8}$$

2.1.2. Membership Function Parameter Modification. In the hidden layer, the update process follows the parameter tuning algorithm displayed at the beginning of this section. Only membership functions of the informative class are updated. Since Gaussian membership functions are chosen for the classified informative linguistic feature, we update 2 parameters, that is, c and σ. Similar to the output layer, we perform the backpropagation algorithm in this hidden layer with the delta value defined as

$$\delta^h(n) = \varphi'\left(\mu_{ij}(n)\right)\sum_k \delta^o(n)\, w_{jk}(n). \tag{9}$$

The parameters c and σ belonging to the informative linguistic features at iteration $(n+1)$ are updated by

$$\begin{aligned} c_{ij}(n+1) &= c_{ij}(n) + \Delta c_{ij}(n), \\ \sigma_{ij}(n+1) &= \sigma_{ij}(n) + \Delta\sigma_{ij}(n), \end{aligned} \tag{10}$$

where Δc_{ij} and $\Delta\sigma_{ij}$ are defined by

$$\begin{aligned} \Delta c_{ij}(n) &= -\eta_c \delta^h(n)\, y_{ij}^{\text{in}}, \\ \Delta\sigma_{ij}(n) &= -\eta_\sigma \delta^h(n)\, y_{ij}^{\text{in}}. \end{aligned} \tag{11}$$

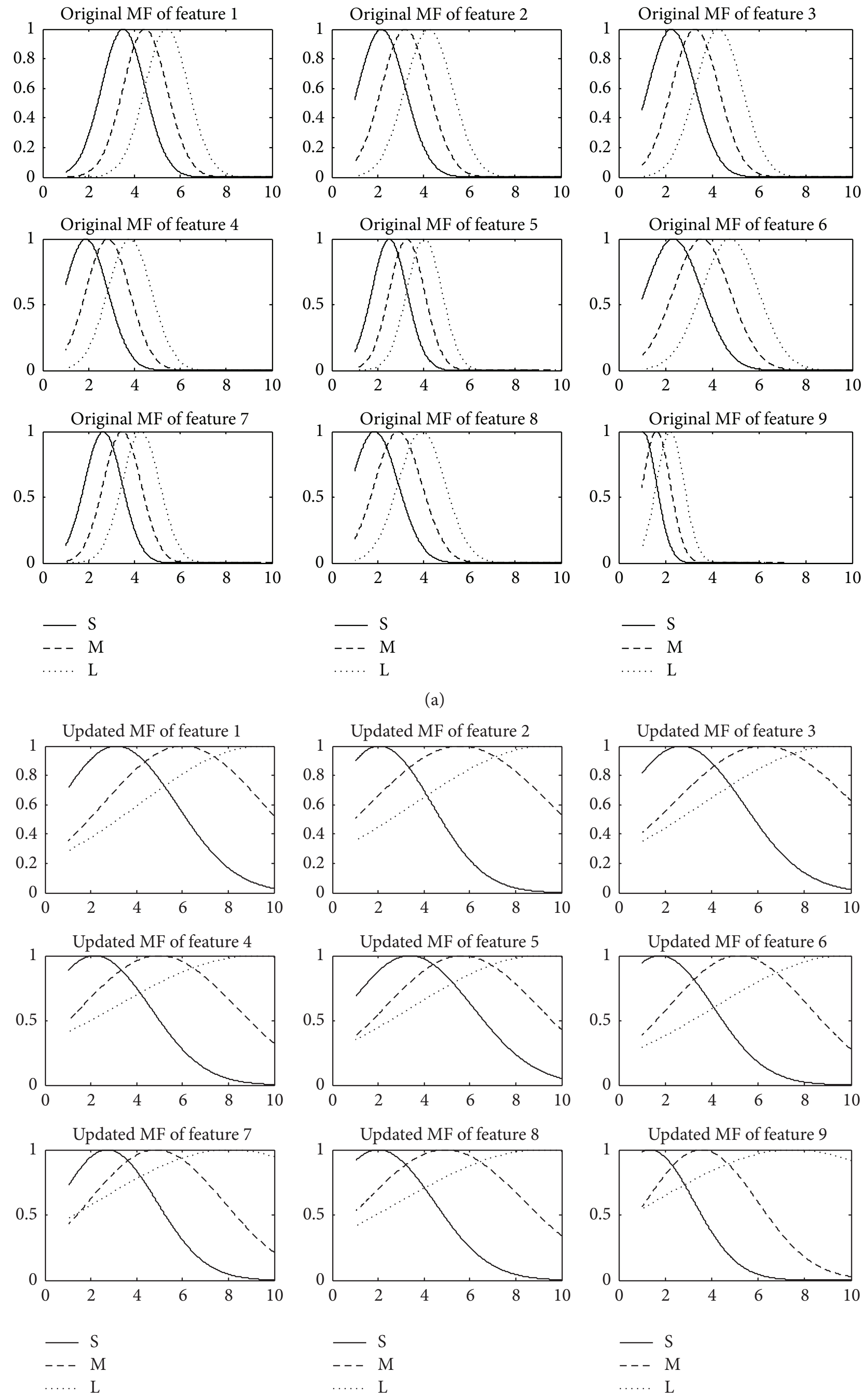
η_c and η_σ are the learning rates for the parameters c and σ, respectively.

Table 1: Genes corresponding to the selected features.

Feature index	Gene
83	MDM4
87	STX16
207	NR1D2
355	DCLRE1A
450	PARK7
546	ATIC
931	HG4263-HT4533_at
2164	CSRP1
2360	NASP
2479	PGK1
6264	HLA-DPB1_2
7043	HLA-A_2
7052	ITK_2
7057	PLGLB2

In Figure 2(a), the Wisconsin breast cancer dataset with 9 original features and 683 data points is used to illustrate the initial membership functions. Figure 2(b) shows the illustrators of all corresponding parameters tuned after training. For the DLBCL dataset used in this research, the number of original features is too large. Therefore, we cannot display the initial membership functions of all features. However, the means and standard deviations of a set of 14 selected features' initial membership functions of are shown later in Table 1.

2.2. Rule Extraction Methods. For the typical direct calculation in a neural network, the output is y^o as shown in (3). The class decision is simply the class with the corresponding maximum output. For the rule extraction purpose, however, the weight values are used to verify the importance of the features after the training process. In each fold of the 10-fold cross validation, after the learning phase, the connection weights between the hidden layer and the output layer are sorted to prioritize the informative features to be described in more detail below.



(a)

(b)

Figure 2: (a) Initial membership functions of the Wisconsin breast cancer dataset. (b) Updated membership functions after training by the neurofuzzy classification model.

Table 2: Means (c_j) and variances (σ_j) of membership functions {S, M, L} of 14 selected features before training for the DLBCL dataset.

	Initial values					
Feature	c_S	c_M	c_L	σ_S	σ_M	σ_L
83	185.636	254.651	323.665	69.014	69.014	69.014
87	200.414	255.633	310.852	55.219	55.219	55.219
207	−164.730	−89.712	−14.693	75.018	75.018	75.018
355	−22.187	11.675	45.538	33.862	33.862	33.862
450	3384.316	4126.221	4868.126	741.905	741.905	741.905
546	1810.664	2396.104	2981.544	585.440	585.440	585.440
931	−148.675	−26.558	95.558	122.117	122.117	122.117
2164	865.360	1032.987	1200.614	167.627	167.627	167.627
2360	1107.608	1373.013	1638.418	265.405	265.405	265.405
2479	25.192	42.494	59.795	17.301	17.301	17.301
6264	5419.876	6722.623	8025.370	1302.747	1302.747	1302.747
7043	11535.574	12983.805	14432.037	1448.232	1448.232	1448.232
7052	259.998	393.052	526.106	133.054	133.054	133.054
7057	422.074	516.844	611.614	94.770	94.770	94.770

The algorithm selects the informative linguistic features twice. The first selection is done by considering the bipolar output from the hidden layer. The linguistic feature with output of +1 is considered an informative feature. The second selection is done by considering the weight values between the hidden layer and the output layer. The larger weight value indicates the more informative feature. Consider the proposed network for logical rule extraction, the IF part is extracted from the hidden layer. As mentioned previously, we use 3 membership functions representing the linguistic terms $\{S, M, L\}$ for each original feature. The summation of the product of weights and output from the hidden layer is interpreted to the logical OR in the extracted rules. The final classified class from the output layer is interpreted to THEN in the classification rules. After finishing the training phase, the weights are sorted. We use the final values of weights to select the "Top N" informative features.

From the structure described earlier, the rule extracted from our network can be interpreted by 2 approaches. Both approaches combine all conditions to only 1 rule. The first approach is the "*simple OR*" rule. All N features are used to create a *simple OR* rule of each class, for example, when a 2-class problem is assumed and N is set to 3. Considering class 1, if the first informative order is "Feature 1 is Large," the second informative order is "Feature 5 is Medium," and the third informative order is "Feature 3 is Small." Considering class 2, if the first informative order is "Feature 10 is Small," the second informative order is "Feature 5 is Small," and the third informative order is "Feature 6 is Large." In case of the class order "Class 1 then Class 2," the rule automatically generated from the *simple OR* approach of our proposed method is as Rule 1.

In case of the class order "Class 2 then Class 1," the rule will be slightly modified to Rule 2.

The second approach creates a rule with the consideration of the order of informative linguistic features and class order. We call this approach the "*layered*" rule. All N linguistic features are created with the consideration of the order of informative features and class order. In case of the class order "Class 1 then Class 2," the rule automatically generated from the *layered* approach for the same scenario as above is as Rule 3.

In case of the class order "Class 2 then Class 1," the extracted rule will be Rule 4.

2.3. Neurofuzzy Method with Feature Selection and Rule Extraction for High Dimensional Dataset via Iterative Partition Method. An iterative partition method is designed for the application on a dataset that has a large amount of features. The idea is to partition the entire set of features into subclusters. Each cluster is used in informative feature selection. The structure of the algorithm is displayed in Figure 3. The first step in the algorithm is to define the desired number of features (F) to be used in the final step. The original dataset is partitioned into n subset. Each subset is used as an input to the neurofuzzy method. All selected features are then combined to create the dataset with selected features. The partitioning and feature selection is iteratively performed until the desired number of informative features is achieved.

3. Experimental Results and Discussion

The diffuse large B-cell lymphomas (DLBCL) dataset consisting of 77 microarray experiments with 7,070 gene expression levels [27] was utilized in this research. It was made available to the public at http://www.broadinstitute.org/cgi-bin/cancer/datasets.cgi. There are two classes in the dataset, that is, diffuse large B-cell lymphoma (DLBCL) is referred to as class 1 and follicular lymphoma (FL) is referred to as class 2. These 2 types of B-cell lineage malignancies have very different clinical presentations, natural histories, and response to therapy [27]. Because DLBCLs are the most common lymphoid malignancy in adults, a method that can efficiently classify these 2 lymphomas is, therefore, very

TABLE 3: Means (c_j) and variances (σ_j) of membership functions {S, M, L} of 14 selected features after training for the DLBCL dataset.

	Final values					
Feature	c_S	c_M	c_L	σ_S	σ_M	σ_L
83	42.217	244.422	448.788	70.018	68.702	68.640
87	89.961	244.306	401.299	55.559	53.710	53.294
207	−316.059	−88.806	138.408	73.797	75.289	75.806
355	−84.419	12.311	109.255	31.586	31.900	37.146
450	1811.843	4047.102	6272.187	746.321	748.685	750.148
546	594.454	2361.230	4125.430	590.479	591.615	595.343
931	−404.615	−26.100	348.347	124.371	124.241	125.706
2164	552.324	1048.629	1545.862	168.960	168.791	173.626
2360	566.830	1351.743	2133.887	262.543	263.800	265.277
2479	−10.591	41.380	95.801	17.087	18.011	23.355
6264	2972.543	6890.328	10810.781	1308.614	1309.833	1308.325
7043	8683.574	12940.488	17192.838	1424.231	1424.378	1422.902
7052	−21.680	387.083	797.666	137.781	137.389	139.518
7057	217.169	507.375	803.842	100.208	102.060	98.289

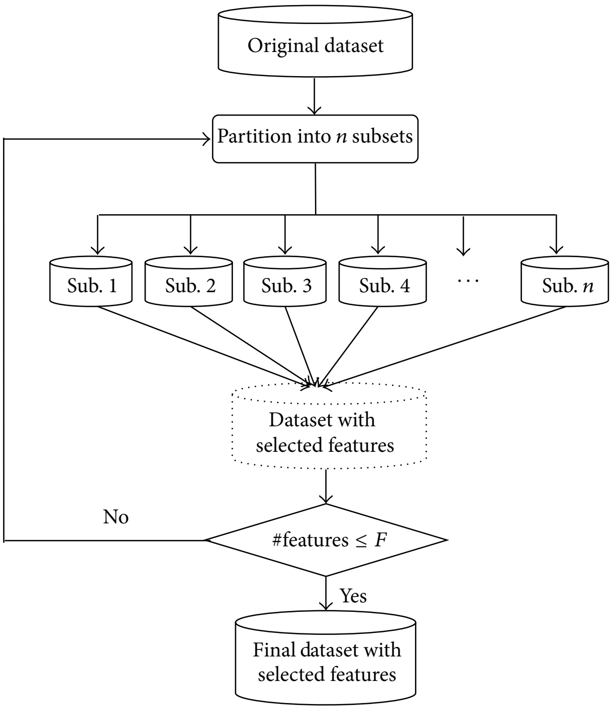

Original dataset
Partition into n subsets
Sub. 1
Sub. 2
Sub. 3
Sub. 4
· · ·
Sub. n
Dataset with selected features
No
#features ≤ F
Yes
Final dataset with selected features

FIGURE 3: Iterative partition method for neurofuzzy method with feature selection and rule extraction for large dataset.

desirable. In the dataset, there are 58 samples of DLBCL class, and 19 samples of FL class.

The selected informative features from the learning phase are used for classification task in both direct calculation and logical rule. The 10-fold cross validation is performed in the experiments. The results displayed are the average results on validation sets over 10 cross validations. During the training step, because the number of features was too large, the original features were divided into small subsets, rather than using the entire 7,070 features at once. The 10-fold cross validation was performed on each subset. The informative features from each subset were selected and form the final informative features by combining them together. The learning rate for weight updating was set to 0.1, and the learning rates used in updating c and σ were set to 0.001.

3.1. Classification Results on DLBCL Microarrays by Direct Calculation Using Selected Features. We tried several choices of the number of selected features (N) and found that $N = 14$ was adequate for this problem. After training, 14 features were automatically selected by our method. The set of features that yielded the best results on validation sets among those in 10-fold cross validation consisted of features 83 (MDM4), 87 (STX16), 207 (NR1D2), 355 (DCLRE1A), 450 (PARK7), 546 (ATIC), 931 (HG4263-HT4533_at), 2164 (CSRP1), 2360 (NASP), 2479 (PGK1), 6264 (HLA-DPB1_2), 7043 (HLA-A_2), 7052 (ITK_2), and 7057 (PLGLB2). The genes corresponding to the selected features are shown in Table 1. The values of means and standard deviations of all membership functions of the 14 selected features before and after training are shown in Tables 2 and 3, respectively.

The classification rates on the validation sets of 10-fold cross validation achieved by using the direct calculation were 92.21%, 89.61%, 84.42%, and 84.42%, when the numbers of selected linguistic features were set to 14, 10, 5, and 3, respectively. These results show that the proposed method can select a set of informative features out of a huge pool of features. As shown in Table 4, the classification performance is comparable to those performed by previously proposed methods [26, 27]. However, rather than using random initial weights connecting between the hidden layer and the output layer, we used the weights achieved in the current cross validation to be the initial weights for the next cross validation. This constrained weight initialization yielded 100.00%, 97.40%, 90.91%, and 92.21% using the direct calculation, when the numbers of selected linguistic features were set to 14, 10,

```
IF "Feature 1 is Large" OR "Feature 5 is Medium" OR "Feature 3 is Small"
          THEN "Class is 1"
ELSEIF "Feature 10 is Small" OR "Feature 5 is Small" OR "Feature 6 is Large"
          THEN "Class is 2"
END
```

Rule 1

```
IF "Feature 10 is Small" OR "Feature 5 is Small" OR "Feature 6 is Large"
          THEN "Class is 2"
ELSEIF "Feature 1 is Large" OR "Feature 5 is Medium" OR "Feature 3 is Small"
          THEN "Class is 1"
END
```

Rule 2

```
IF "Feature 1 is Large" THEN "Class is 1"
ELSEIF "Feature 10 is Small" THEN "Class is 2"
ELSEIF "Feature 5 is Medium" THEN "Class is 1"
ELSEIF "Feature 5 is Small" THEN "Class is 2"
ELSEIF "Feature 3 is Small" THEN "Class is 1"
ELSEIF "Feature 6 is Large" THEN "Class is 2"
END
```

Rule 3

```
IF "Feature 10 is Small" THEN "Class is 2"
ELSEIF "Feature 1 is Large" THEN "Class is 1"
ELSEIF "Feature 5 is Small" THEN "Class is 2"
ELSEIF "Feature 5 is Medium" THEN "Class is 1"
ELSEIF "Feature 6 is Large" THEN "Class is 2"
ELSEIF "Feature 3 is Small" THEN "Class is 1"
END
```

Rule 4

5, and 3, respectively. To ensure that the set of all parameters achieved here could get 100% correct classification on this dataset, we tried to use the networks to classify all 77 microarrays (rather than considering the results from 10-fold cross validation in which the outputs could be different when the random groups are different.) We found that each of all 10 networks from 10-fold cross validation still yielded 100% correct classification on the entire dataset.

3.2. Classification Results on DLBCL Microarrays by Logical Rule Using Selected Features. One of the good features of our method is the automatic rule extraction. Even though this approach usually does not yield as good performance as the direct calculation, it provides rules understandable for human. This is more desirable from the human interpretation aspect than the black-box based direct calculation.

Table 4: Comparison between the proposed method and other algorithms on the DLBCL dataset.

Method	Number of features selected	Classification rate (%)
Naïve Bayes [26]	3–8	83.76
Our method without constrained weight initialization	10	89.61
Decision trees [26]	3–8	85.46
Our method without constrained weight initialization	14	92.21
k-NN [26]	3–8	88.60
Weighted voting model [27]	30	92.20
VizRank [26]	3–8	93.03
Our method with constrained weight initialization	10	97.40
SVM [26]	3–8	97.85
Our method with constrained weight initialization	14	100.00

The N selected linguistic features were used to create rules using both *simple OR* and *layered* approaches as mentioned in Section 2.2. The classification rate of 90.91% on validation sets using only 5 selected linguistic features was achieved using the *simple OR* rule with the class order "Class 1 then Class 2," where Class 1 and Class 2 denote the DLBCL class and FL class, respectively. For more details of the results, Table 5 shows the top-10 linguistic features for the DLBCL dataset selected by our method in each cross validation of the 10-fold cross validation. The classification rates in all 10 cross validations were 75.00%, 100.00%, 100.00%, 75.00%, 100.00%, 87.50%, 75.00%, 100.00%, 100.00%, and 100.00%, respectively. The classification rate from the *layered* rule was 81.82% using the same top 5 linguistic features. The details of classification rates in all 10 cross validations were 75.00%, 87.50%, 100.00%, 87.50%, 62.50%, 75.00%, 75.00%, 85.71%, 85.71%, and 85.71%, respectively. When using the aforementioned constrained

```
IF "HLA-A_2 is Small" OR "NASP is Large" OR "MDM4 is Small" OR "ATIC is Medium" OR "STX16 is Small"
        THEN "Class is DLBCL"
ELSEIF "HLA-A_2 is Large" OR "ATIC is Small" OR "STX16 is Large" OR "MDM4 is Medium" OR "NASP is Small"
        THEN "Class is FL"
END
```

Rule 5

Table 5: Top-10 linguistic features for DLBCL dataset selected by our method in 10-fold cross validation (most informative: right, least informative: left).

Cross validation	Class	Feature type	Feature ranking									
1	1	Linguistic	S	M	M	S	L	S	M	L	S	S
		Original	7057	207	931	87	2479	7052	546	2360	83	7043
	2	Linguistic	S	M	L	L	M	S	L	L	L	S
		Original	450	7052	6264	207	83	2360	83	87	7043	546
2	1	Linguistic	L	L	M	M	S	S	S	M	L	S
		Original	355	2479	207	931	87	7052	83	546	2360	7043
	2	Linguistic	S	M	L	L	L	M	S	L	L	S
		Original	355	7052	83	6264	207	83	2360	87	7043	546
3	1	Linguistic	S	M	S	M	L	S	M	L	S	S
		Original	7052	931	7057	207	2479	87	546	2360	83	7043
	2	Linguistic	L	S	S	L	S	L	M	S	L	L
		Original	7057	2479	450	207	2360	83	83	546	87	7043
4	1	Linguistic	M	L	M	L	M	S	S	S	L	S
		Original	931	2164	207	2479	546	7052	87	83	2360	7043
	2	Linguistic	S	L	S	L	L	S	M	L	S	L
		Original	2479	6264	450	207	83	2360	83	87	546	7043
5	1	Linguistic	L	M	L	M	S	S	M	L	S	S
		Original	2164	931	2479	207	7052	87	546	2360	83	7043
	2	Linguistic	S	S	L	S	L	L	M	L	S	L
		Original	2479	450	6264	2360	83	207	83	87	546	7043
6	1	Linguistic	L	M	S	M	L	S	S	M	L	S
		Original	355	207	7052	931	2479	87	83	546	2360	7043
	2	Linguistic	L	L	S	L	M	L	S	L	S	L
		Original	7057	6264	450	207	83	83	2360	87	546	7043
7	1	Linguistic	S	S	M	M	L	S	M	S	L	S
		Original	7052	7057	931	207	2479	87	546	83	2360	7043
	2	Linguistic	L	S	S	L	L	S	M	L	S	L
		Original	6264	2479	450	207	83	2360	83	87	546	7043
8	1	Linguistic	L	M	M	S	S	S	L	S	L	S
		Original	355	546	931	7052	87	7057	2479	83	2360	7043
	2	Linguistic	L	L	S	M	S	L	L	L	S	L
		Original	7057	207	2360	83	2479	6264	83	87	546	7043
9	1	Linguistic	S	S	M	L	M	S	M	S	L	S
		Original	7057	7052	931	2479	207	87	546	83	2360	7043
	2	Linguistic	L	S	S	L	L	S	M	L	S	L
		Original	6264	450	2479	207	83	2360	83	87	546	7043
10	1	Linguistic	M	L	M	S	S	S	M	S	L	S
		Original	207	2479	931	87	7052	7057	546	83	2360	7043
	2	Linguistic	L	L	S	L	S	L	M	L	S	L
		Original	6264	7057	450	207	2360	83	83	87	546	7043

IF "HLA-A_2 is Small" *THEN* "Class is DLBCL"
ELSEIF "HLA-A_2 is Large" *THEN* "Class is FL"
ELSEIF "NASP is Large" *THEN* "Class is DLBCL"
ELSEIF "ATIC is Small" *THEN* "Class is FL"
ELSEIF "MDM4 is Small" *THEN* "Class is DLBCL"
ELSEIF "STX16 is Large" *THEN* "Class is FL"
ELSEIF "ATIC is Medium" *THEN* "Class is DLBCL"
ELSEIF "MDM4 is Medium" *THEN* "Class is FL"
ELSEIF "STX16 is Small" *THEN* "Class is DLBCL"
ELSEIF "NASP is Small" *THEN* "Class is FL"
END

Rule 6

weight initialization, the classification rates were the same. We also tried to increase the number of selected features to 10 but it did not help. The results were the same as that using 5 features.

From Table 5, it can be seen that the sets of informative features for class 1 and class 2 are different across the cross validations. That will result in a number of different rules. However, in the real application we will have to come up with the best rules among them. We propose to use the summation of the feature informative level in all 10 cross validations. The feature informative level is simply defined by the informative order. For example, if only 10 linguistic features are considered, the most informative one will have the feature informative level of 1.0, the second one will have that of 0.9, and so on. Hence, the tenth most informative feature will have the feature informative factor of 0.1, and the remaining will get the feature informative level of 0.0. The informative levels of each feature are then summed across all cross validations to yield the overall informative level of that feature.

We show the overall feature informative levels from the 10-fold cross validation using top-10 features in Table 6. In this case, the highest possible informative level is 10.0. Out of 42 linguistic features for each class, there were only 12 and 13 linguistic features with nonzero informative level for class 1 and class 2, respectively. That means the remaining features did not appear at all in the top-10 list of any cross validation. The results showed that, for class 1, the first 5 most informative linguistic features ranking from the most informative to the less informative were "Feature 7043 (HLA-A_2) is Small," "Feature 2360 (NASP) is Large," "Feature 83 (MDM4) is Small," "Feature 546 (ATIC) is Medium," and "Feature 87 (STX16) is Small," respectively. For class 2, the first 5 most informative linguistic features were "Feature 7043 is Large," "Feature 546 is Small," "Feature 87 is Large," "Feature 83 is Medium," and "Feature 2360 is Small," respectively. It is worthwhile noting that the last statement can also be "Feature 83 is Large" because it has the same feature informative level of 5.5 as for "Feature 2360 is Small." This information was used to create rules as described in Section 2.2. Hence, the rule extracted using the *simple OR* approach is as Rule 5.

Using the *layered* approach, the extracted rule is in Rule 6.

Table 6: Feature informative levels from the 10-fold cross validation using top-10 features.

Original feature	Linguistic term	Informative level
	Class 1	
7043	S	10.0
2360	L	8.7
83	S	8.1
546	M	6.5
87	S	5.5
2479	L	4.2
7052	S	3.9
207	M	2.8
931	M	2.8
7057	S	1.9
355	L	0.3
2164	L	0.3
	Class 2	
7043	L	9.8
546	S	9.1
87	L	8.1
83	M	6.2
2360	S	5.5
83	L	5.5
207	L	4.1
6264	L	2.3
450	S	2.0
2479	S	1.4
7057	L	0.5
7052	M	0.4
355	S	0.1

4. Conclusion

The classification problem of diffuse large B-cell lymphoma (DLBCL) versus follicular lymphoma (FL) based on high dimensional microarray data was investigated by our neurofuzzy classification scheme. Our direct calculation method could achieve 100% classification rate on the validation sets of 10-fold cross validation by using only 14 out of 7,070 features in the dataset. These 14 features including genes MDM4, STX16, NR1D2, DCLRE1A, PARK7, ATIC, HG4263-HT4533_at, CSRP1, NASP, PGK1, HLA-DPB1_2, HLA-A_2, ITK_2, and PLGLB2 were automatically selected by our method. The method could also identify the informative linguistic features for each class. For the DLBCL class, the first 5 most informative linguistic features were "HLA-A_2 is Small," "NASP is Large," "MDM4 is Small," "ATIC is Medium," and "STX16 is Small," respectively. For class 2, the first 5 most informative linguistic features were "HLA-A_2 is Large," "ATIC is Small," "STX16 is Large," "MDM4 is Medium," and "NASP is Small," respectively. The terms Small, Medium, and Large of each original feature were automatically determined by our method. The informative linguistic features were used to create rules that achieved 90.91% classification rate on the validation sets of 10-fold cross validation. Even though this rule creation approach yielded worse classification performance than the direct calculation, it is more desirable from the human interpretation aspect. It can be seen that very good results are achieved in this standard high dimensional dataset. A set of selected informative genes will be useful for further investigation in the fields of bioinformatics or medicines. To ensure that this set of selected features can be used in general, it should be applied to more DLBCL versus FL cases.

Conflict of Interests

The authors declare no conflict of interests.

References

[1] G. G. Towell, J. W. Shavlik, and M. O. Noordenier, "Refinement of approximate domain theories by knowledge based neural network," in *Proceedings of the 8th National Conference on Artificial Intelligence*, pp. 861–866, Boston, Mass, USA, July-August 1990.

[2] L. M. Fu, "Knowledge-based connectionism for revising domain theories," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 1, pp. 173–182, 1993.

[3] L. M. Fu, "Learning capacity and sample complexity on expert networks," *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1517–1520, 1996.

[4] S. Snyders and C. W. Omlin, "What inductive bias gives good neural network training performance?" in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '00)*, pp. 445–450, Como, Italy, July 2000.

[5] D.-X. Zhang, Y. Liu, and Z. I.-Q. Wang, "Effectively extracting rules from trained neural networks based on the characteristics of the classification hypersurfaces," in *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '05)*, vol. 3, pp. 1541–1546, IEEE, Guangzhou, China, August 2005.

[6] L. Fu, "Rule generation from neural networks," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 8, pp. 1114–1124, 1994.

[7] F. Beloufa and M. A. Chikh, "Design of fuzzy classifier for diabetes disease using modified artificial bee colony algorithm," *Computer Methods and Programs in Biomedicine*, vol. 112, no. 1, pp. 92–103, 2013.

[8] J. Luo and D. Chen, "Fuzzy information granulation based decision support applications," in *Proceedings of the International Symposium on Information Processing (ISIP '08)*, pp. 197–201, Moscow, Russia, May 2008.

[9] A. T. Azar, S. A. El-Said, and A. E. Hassanien, "Fuzzy and hard clustering analysis for thyroid disease," *Computer Methods and Programs in Biomedicine*, vol. 111, no. 1, pp. 1–16, 2013.

[10] C.-H. L. Lee, A. Liu, and W.-S. Chen, "Pattern discovery of fuzzy time series for financial prediction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 5, pp. 613–625, 2006.

[11] R. Yang, Z. Wang, P.-A. Heng, and K.-S. Leung, "Classification of heterogeneous fuzzy data by choquet integral with fuzzy-valued integrand," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 5, pp. 931–942, 2007.

[12] P. Hartono and S. Hashimoto, "An interpretable neural network ensemble," in *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON '07)*, pp. 228–232, Taipei, Taiwan, November 2007.

[13] T.-G. Fan and X.-Z. Wang, "A new approach to weighted fuzzy production rule extraction from neural networks," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 3348–3351, Shanghai, China, August 2004.

[14] M. Gao, X. Hong, and C. J. Harris, "A neurofuzzy classifier for two class problems," in *Proceedings of the 12th UK Workshop on Computational Intelligence (UKCI '12)*, pp. 1–6, Edinburgh, UK, September 2012.

[15] W. Duch, R. Adamczak, and K. Grąbczewski, "A new methodology of extraction, optimization and application of crisp and fuzzy logical rules," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 277–306, 2001.

[16] W. Duch, R. Setiono, and J. M. Zurada, "Computational intelligence methods for rule-based data understanding," *Proceedings of the IEEE*, vol. 92, no. 5, pp. 771–805, 2004.

[17] A. T. Azar, "Adaptive network based on fuzzy inference system for equilibrated urea concentration prediction," *Computer Methods and Programs in Biomedicine*, vol. 111, no. 3, pp. 578–591, 2013.

[18] A. Ouarda, A. Ahlem, B. Brahim, and B. Kheir, "Comparison of neuro-fuzzy models for classification fingerprint images," *International Journal of Computer Applications*, vol. 65, no. 9, pp. 12–16, 2013.

[19] Z. Yang, Y. Wang, and G. Ouyang, "Adaptive neuro-fuzzy inference system for classification of background EEG signals from ESES patients and controls," *The Scientific World Journal*, vol. 2014, Article ID 140863, 8 pages, 2014.

[20] M. S. Al-Batah, N. A. M. Isa, M. F. Klaib, and M. A. Al-Betar, "Multiple adaptive neuro-fuzzy inference system with automatic features extraction algorithm for cervical cancer recognition," *Computational and Mathematical Methods in Medicine*, vol. 2014, Article ID 181245, 12 pages, 2014.

[21] J.-A. Martinez-Leon, J.-M. Cano-Izquierdo, and J. Ibarrola, "Feature selection applying statistical and neurofuzzy methods to EEG-based BCI," *Computational Intelligence and Neuroscience*, vol. 2015, Article ID 781207, 17 pages, 2015.

[22] D. Chakraborty and N. R. Pal, "A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification," *IEEE Transactions on Neural Networks*, vol. 15, no. 1, pp. 110–123, 2004.

[23] N. Eiamkanitchat, N. Theera-Umpon, and S. Auephanwiriyakul, "A novel neuro-fuzzy method for linguistic feature selection and rule-based classification," in *Proceedings of the 2nd International Conference on Computer and Automation Engineering (ICCAE '10)*, vol. 2, pp. 247–252, Singapore, February 2010.

[24] N. Eiamkanitchat, N. Theera-Umpon, and S. Auephanwiriyakul, "Colon tumor microarray classification using neural network with feature selection and rule-based classification," in *Advances in Neural Network Research and Applications*, vol. 67 of *Lecture Notes in Electrical Engineering*, pp. 363–372, Springer, Berlin, Germany, 2010.

[25] C.-H. Chen and C.-J. Lin, "Compensatory neurofuzzy inference systems for pattern classification," in *Proceedings of the International Symposium on Computer, Consumer and Control (IS3C '12)*, pp. 88–91, IEEE, Taichung, Taiwan, June 2012.

[26] M. Mramor, G. Leban, J. Demšar, and B. Zupan, "Visualization-based cancer microarray data classification analysis," *Bioinformatics*, vol. 23, no. 16, pp. 2147–2154, 2007.

[27] M. A. Shipp, K. N. Ross, P. Tamayo et al., "Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning," *Nature Medicine*, vol. 8, no. 1, pp. 68–74, 2002.