软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software
©中国科学院软件研究所版权所有.

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

Classifying Pediatric Brain Tumor Subtypes with Sparse Mixture of Experts Architecture

DARMAWAN Jason Rich[1]

[1](School of Computer Science, Northwestern Polytechnical University, Xi'an 710129, China)

Abstract: Classifying pediatric brain tumor subtypes based on microarray data analysis require to know the specific genes associated with the tumor subtypes. Existing methods solved this problem, but are prone to outliers. This paper proposed 4 key components for classifying tumor subtypes: 1) Data preprocessing techniques. 2) A neural network with Sparse Mixture of Experts (SMoE) architecture. 3) A load balancing loss. 4) A contrastive loss. In addition, the paper demonstrated how to detect outliers in a microarray data using approximation of squared Mahalanobis distances between data points and the centroids. The proposed method achieved 94.29% accuracy on the validation set.

Key words: mixture of experts, mahalanobis distances, contrastive loss, pediatric brain tumor classification

Code: https://github.com/jasonrichdarmawan/learn-data-mining-techniques

Training a neural network on microarray data to classify tumor subtypes may assist clinicians in choosing less toxic therapy [1]. Since microarray data measure the fluorescence intensity of many genes [2], a neural network maybe be able to reveal hidden gene patterns for each tumor subtype by using non-linear functions [3].

Traditional methods for classifying tumor subtypes—known as microarray data analysis—involve measuring the fluorescent intensity of specific genes [4]. Based on the measurements, the tumors are grouped into subtypes [4]. However, a limitation of this method is the requirement to know the specific genes associated with tumor subtypes. To this end, existing methods [5] proposed to use K-Means clustering to classify tumor subtypes. These methods solved the problem.

Microarray data is small in quantity and non-error outliers are can negatively affect microarray data analysis [6] and existing methods are prone to outliers. To solve this problem, this paper proposed methods without the need to process the non-error outliers from the dataset but still achieved 94.29% accuracy on the validation set.

This paper proposed 4 key components for classifying tumor subtypes: data preprocessing techniques, a neural network with SMoE architecture [7], a load balancing loss, and a contrastive loss. In addition, this paper demonstrates the presence of outliers in the microarray data from Dr. Bremer [8] by approximating the squared Mahalanobis distances between data points and the centroids of each label using methods proposed by Shieh et. al., 2009 [6].

Section 1 of this paper introduced the existing methods for classifying tumors subtypes. Section 2 introduced the basics required for this paper, including the methods used to demonstrate the presence of outliers. Section 3 introduced the proposed methods, including the data preprocessing techniques, the neural network with the SMoE architecture, and two loss functions. Section 4 verify the proposed methods with comparative experiments. Section 5 concludes the significance of the research and opportunity for further research.

1 Related Work

[9] proposed to preprocess the microarray data to remove redundant genes based on maximum-to-minimum value of each gene to remove genes with low variability, which may not contribute to distinguish samples with different labels [9], [10]. However, removing genes based on maximum-to-minimum value of each gene ratios may remove genes that could distinguish samples with different labels when used with other genes. Furthermore, [11] demonstrates Welch's t-test limitations when applied to small sample sizes.

[5] proposed to use K-Means clustering to cluster the samples. However, a limitation of K-Means clustering is the requirement to know the number of clusters associated with the specific dataset. Furthermore, K-Means clustering assumes that clusters are spherical, an assumption that may not hold for microarray data where non-error outliers are common [6] and can lead to non-spherical clusters.

2 Basic Concepts Required

2.1 Detecting Outliers by Estimating Mahalanobis Distances

K-Means clustering assumes that clusters are spherical, an assumption that may not hold for microarray data where the presence of non-error outliers can lead to non-spherical clusters. Although research have been carried out on microarray data from Dr. Bremer [8], no single study exists to detect outliers in the dataset.

To detect the presence of outliers, the method proposed by Shieh et. al., 2009 [6] was used. The fundamental of the method is the normality assumption, where the squared Mahalanobis distances between data points and the centroids should follow a straight line, except the outliers [6]. The straight line is constructed based on χ^2 probability point function. If the data points deviate from the straight line, then the outlier detection method is unreliable [6].

Demonstrating the presence of outliers consist of six steps: 1) Transform the microarray data with base-2 logarithm to reduce the variance of each gene. 2) Normalize each sample to have the same distribution of gene with quantile normalization. 3) Reduce the microarray data dimensions using the Principal Component Analysis (PCA). 4) Estimate the optimal number of principal components $\hat{q}$ based on the profile log-likelihood of q. 5) Transform the microarray data with PCA. 6) Approximate the Mahalanobis distances between data points and the centroids with the Minimum Covariance Determinant. The Mahalanobis distances are approximated because the sample mean, and covariance are susceptible to outliers [6]. 7) Compare the outlier threshold value with the squared Mahalanobis distances. The outlier threshold value is based on critical value of the χ^2 probability point function with level of significance of 0.975 and degrees of freedom following the estimated $\hat{q}$ to ensure that only 2.5% of the data points are falsely flagged as outliers. 6) Plot the quantile-quantile plot to verify the normality assumption.

2.2 Base-2 Logarithm Aimed at Reducing Variance of Each Gene

Outliers may negatively affect microarray data analysis [6]. The fundamental of applying a base-2 logarithm transformation to microarray data is to stabilize variance across different value level because genes with high values tend to have higher variance. By reducing the influence of extreme values, the log transformation tries to ensure that all genes contribute equally. Consequently, this preprocessing step is crucial for statistical modeling.

2.3 Neural Network with SMoE Architecture

Mislabeled samples may decrease the accuracy of the neural network [12]. The fundamental of SMoE architecture is that the Expert modules are more likely to output the same label than to output different label over a specific sample [12]. The SMoE architecture consists of a Router module to select the Expert modules for a specific sample and Expert modules to output the scores of each label. The common misconception in Co-Teaching [13], [12] is that the neural network is limited to two Expert modules. The SMoE architecture remove this limitation by allowing a neural network to have as many Expert modules as required, but only use few of the available Expert modules for a specific sample.

However, training a neural network with the SMoE architecture on an imbalanced dataset are more likely to over-relying on specific experts [7], which could result in the undertraining of other Expert modules. Consequently, it is necessary to implement a load balancing loss function to penalize over-relying on specific experts.

The significance of a neural network with the SMoE architecture outside this research topic is that it enables the neural network to have larger parameters, but only use small amount of the neural network parameters during inference. Consequently, a neural network with the SMoE architecture can be scaled up to learn more but still use the same compute budget as before the scale up.

3 Robust Neural Network using Data Preprocessing Techniques and the SMoE Architecture

The proposed methods consist of four components: data preprocessing techniques, the SMoE architecture, a load balancing loss function, and a contrastive loss. The following definitions are used for the three subsections below. Suppose a neural network with the SMoE architecture has E Expert modules trained on a microarray data with N samples and microarray data with L labels is given as $D = \{x_i, y_i\}_{i=1}^{N}$, where x_i is the i-th sample index with its label as $y_{i,e} \in \{1, \ldots, L\}$ where e indexes the Expert modules.

3.1 Data Preprocessing Techniques

The data preprocessing techniques consist of four steps: 1) Assigns any value less than 20 to 20 and any value greater than 16,000 to 16,000. This is a specific step for the microarray data from Dr. Bremer [8] due to a known issue with the MAS Version 4 software which generates negative values [9], [10]. 2) Remove genes based on the maximum-to-minimum value of each gene with ratio less than 2 to remove genes with low variability, which may not contribute to distinguish samples with different labels [10]. 3) Transform the microarray data with base-2 logarithm to reduce the variance of each gene. 4) Standardize each gene to have a mean of 0 and a standard deviation of 1 to ensure the genes with higher values do not disproportionately influence the model.

3.2 The SMoE Architecture

The neural network uses the SMoE architecture [7], which have four Expert modules but only use two Expert modules during inference. Consequently, only 58.3% of model parameters are used during inference.

The SMoE module consists of a Router module and Expert modules. The Router module outputs the scores of each Expert $r_i \in \{1, \ldots, E\}$ where i indexes the sample and e indexes the Expert modules. The Router module output r_i is used as values to select the two best Expert modules, a higher value is better. The Expert module outputs the scores of each label $y_{i,e} \in \{1, \ldots., L\}$ where L is the tumor subtype labels. The SMoE module is used to sum the weighted scores of each label $y_{i,e}$ from the two best Expert modules for each specific sample. The weighted scores are calculated by multiplying the scores r_i from the Router module with the scores $y_{i,e}$ from the Expert modules.

The Router module consist of a single linear transformation layer. To encourage the balanced use of all available Expert modules within a training batch, a Gaussian noise—sampled from a normal distribution with a mean of 0 and a variance of 0.1—is added to the Router module output r_i.

The Expert module consist of a single linear transformation layer and a dropout layer. The dropout layer is used to mitigate overfitting during training by randomly zeroes some of the elements of the Expert module output $y_{i,e}$ with probability of 0.5.

3.3 Loss Functions

The neural network uses two loss functions: 1) A load balancing loss is used to penalize the Router module for not using all the Expert modules within the training batch. 2) A contrastive loss is used to penalize the Expert module when there are different outputs between Expert modules. The contrastive loss is adapted from [12]. The difference lies in the ability to scale up or down the term following the number of Expert modules used to process a specific sample. The scale up is when the maximum number of Experts to process each sample is increased. The scale down is when an Expert is over-used within a training batch.

The loss function is constructed as follows: $\ell(r, y) = \ell_{\text{load balance}}(r) + \ell_{\text{contrastive}}(y)$ where r is the Router module output and y is the Expert modules output.

The load balance loss function outputs 0 when the Expert modules are used equally within a training batch. The load balance loss function uses the Router module output r as input and is constructed as follows: $\ell_{\text{load balance}}(r) = \sigma(S)$ where σ is the unbiased standard deviation, and S is the sum of the Router module output r_i across all samples, $S = \sum_{i=1}^{N} r_i$, $\sigma(S) = \sqrt{\frac{1}{E-1}\sum_{i=1}^{E}(S_i - \bar{S})^2}$, S_i is the sum of the Router module output of each Expert module and $\bar{S}$ is the Router module output mean across all Expert modules.

The contrastive loss function outputs 0 when each of the Expert module outputs are identical. The contrastive loss function uses the Router module output y_i for a specific sample as input. The loss function below is only for academic purpose, the paper's source code is optimized to process all samples. The loss function is constructed as follows: $\ell_{\text{contrastive}}(y_i) = \sum_{p=1}^{E} \sum_{q=p+1}^{E} D_{\text{KL}}(y_{i,p} || y_{i,q}) + D_{\text{KL}}(y_{i,q} || y_{i,p})$ where D_{KL} is the Kullback-Leibler divergence term, $D_{\text{KL}}(y_{i,p} || y_{i,q}) = y_{i,p} \log \frac{y_{i,p}}{y_{i,q}}$ and $D_{\text{KL}}(y_{i,q} || y_{i,p}) = y_{i,q} \log \frac{y_{i,q}}{y_{i,p}}$. The $\sum_{p=1}^{E} \sum_{q=p+1}^{E}$ term is summation over all pairs of the Expert modules (p, q). This term enables the contrastive loss function to scale up or down following the number of Expert modules used for a specific sample within a training batch.

4 Experimental Analysis

4.1 Experimental Data

The pediatric brain tumor subtypes microarray data from Dr. Bremer [8] is used for the experimental analysis. The microarray data measured 7070 genes and each sample may have one of the 5 possible labels, which are MED, MGL, RHB, EPD, and JPA. The microarray data consist of 69 samples and the labels are imbalanced as follows: 39 samples are labelled MED, 7 samples are labelled MGL, 7 samples are labelled RHB, 10 samples are labelled EPD, and 6 samples are labelled JPA.

4.2 Evaluation Indicators and Benchmark Models

The evaluation indicator used is the accuracy of the model to assess whether the model can correctly predict unseen data. The higher accuracy is better and is constructed as follow: $\text{Accuracy} = \frac{\text{True Prediction}}{\text{All Prediction}}$

The benchmark model used are the K-Means clustering [5] and Random Forest [14] methods. The K-Means clustering method [5] proposed to preprocess the microarray data to remove redundant genes based on sorted neighborhood method, use an algorithm to perform dimension reduction and use the K-Means clustering to cluster the samples into clusters of each tumor subtype. The Random Forest method [14] proposed three steps for classifying tumor subtypes: 1) Select the top N genes per label based on Welch's t-test values, where $N \in \{2,4,6,8,10,12,15,20,25,30\}$. 2) Combine the top N genes per label into sets. 3) Find the best combination of classifier and gene set.

4.3 Experimental Methods

The microarray data from Dr. Bremer [8] is randomly divided equally into training and validation sets, each set maintaining the original label distribution.

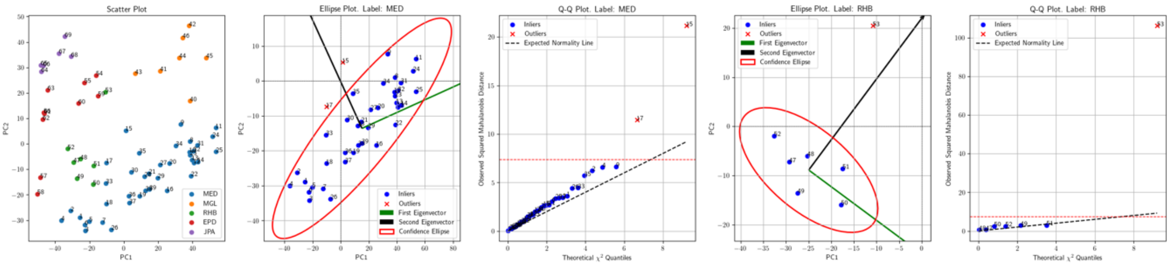
4.4 Experimental Results and Analysis

Table 1 Performance Comparison

Target	Training Set Accuracy	Validation Set Accuracy	Microarray Data Accuracy
K-Means clustering [5]	-	-	0.565
Random Forest [14]	-	-	1.0
Ours	1.0	0.943	0.971

The K-Means clustering method [5] does not mention dividing the microarray data into training and validations sets and the Random Forest method [14] removed genes before splitting the microarray data into training and validation sets. As a result, both the Random Forest method [14] and the K-Means clustering method [5] may have a data leakage problem—where the model performance during evaluation is high because the model was trained on samples which are used for evaluation.

The Random Forest method [14] outperformed our model by 2.9%. In this case, our model misclassified 2 samples out of 35 samples in the validation set. Specifically, the patient with index 53 true label is RHB but the predicted label is MED and patient index 21 with true label MED but the predicted label is MGL.



(a) (b) (c) (d) (e)

Figure 1 Scatter, Ellipse and Q-Q Plots of Principal Components

Figure 1(a) is constructed using the microarray data transformed by the PCA which was explained in Subsection 2.1 . Figure 1(a) shows the patient with index 53 may be hard to classify because the patient with index 53 is far from the other patients with true label RHB. However, the patient with index 21 should be easy to classify but the neural network misclassified it, indicating an overfitting problem. This suggests that there is a research opportunity to train a robust neural network against overfitting problem.

4.5 Parameter Impact Analysis

Table 2 Parameter Impact

	Parameters	Active Parameters	Ratio
Router	16,816	16,818	1.0
Experts	84,080	42,040	0.500
Total Parameters	100,896	58,856	0.583

The neural network uses the SMoE architecture [7], which have four Expert modules but only use two Expert modules during inference. Consequently, only 58.3% of model parameters are used during inference.

4.6 Outliers Analysis

Figure 1(b)-(e) is constructed based on the approximated squared Mahalanobis distances explained in Subsection 2.1 . Figure 1(c) shows that the data points with true label MED violate the normality assumption. Consequently, the outlier detection method is unreliable for data points with true label MED. However, Figure 1(e) shows that the data pints with true label RHB do not violate the normality assumption. Therefore, the data point with index 53 and true label RHB is an outlier.

5 Summary

Research for classifying pediatric brain tumor subtypes may support clinicians in making more informed decision in choosing less toxic therapies. The proposed methods demonstrated how to classify tumor subtypes without the need to know the specific genes associated with tumor subtypes. The proposed contrastive loss equation can scale up and down following the number of Expert modules used to process a specific sample. However, the experiment demonstrated the neural network misclassified an outlier. The experiment found the presence of outliers in the microarray data from Dr. Bremer [8]. This suggests there is a research opportunity to train a more robust neural network against outliers and misclassified samples.

References:

[1] BIRKS D K, BARTON V N, DONSON A M, 等. Survey of MicroRNA expression in pediatric brain tumors[J/OL]. Pediatric Blood & Cancer, 2011, 56(2): 211-216. DOI:10.1002/pbc.22723.

[2] HELD G A, GRINSTEIN G, TU Y. Relationship between gene expression and observed intensities in DNA microarrays--a modeling study[J/OL]. Nucleic Acids Research, 2006, 34(9): e70-e70. DOI:10.1093/nar/gkl122.

[3] LECUN Y, BENGIO Y, HINTON G. Deep learning[J/OL]. Nature, 2015, 521(7553): 436-444. DOI:10.1038/nature14539.

[4] NATARAJAN J, BERRAR D, DUBITZKY W, 等. Text mining of full-text journal articles combined with gene expression analysis reveals a relationship between sphingosine-1-phosphate and invasiveness of a glioblastoma cell line[J/OL]. BMC Bioinformatics, 2006, 7(1): 373. DOI:10.1186/1471-2105-7-373.

[5] LI X, JIANG K, WANG H, 等. A novel K-means classification method with genetic algorithm[C/OL]//2017 International Conference on Progress in Informatics and Computing (PIC). IEEE, 2017: 40-44. DOI:10.1109/PIC.2017.8359511.

[6] SHIEH A D, HUNG Y S. Detecting Outlier Samples in Microarray Data[J/OL]. Statistical Applications in Genetics and Molecular Biology, 2009, 8(1): 1-24. DOI:10.2202/1544-6115.1426.

[7] JIANG A Q, SABLAYROLLES A, ROUX A, 等. Mixtral of Experts[A/OL]. (2024). https://arxiv.org/abs/2401.04088.

[8] PIATETSKY-SHAPIRO G, PARKER G, BREMER E. Data Mining Course Datasets[A/OL]. 2004: 1[2025-01-21]. https://web.archive.org/web/20041031102841/http://www.kdnuggets.com/dmcourse/data_mining_course/data/index.html.

[9] PIATETSKY-SHAPIRO G, PARKER G. Data Mining Course Final Project[A/OL]. 2004: 1. https://web.archive.org/web/20041109215952/http://www.kdnuggets.com/dmcourse/data_mining_course/assignments/final-project.html.

[10] PIATETSKY-SHAPIRO G, KHABAZA T, RAMASWAMY S. Capturing best practice for microarray gene expression data analysis[C/OL]//Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: Association for Computing Machinery, 2003: 407-415. https://doi.org/10.1145/956750.956797. DOI:10.1145/956750.956797.

[11] MPINDI J P, SARA H, HAAPA-PAANANEN S, 等. GTI: A Novel Algorithm for Identifying Outlier Gene Expression Profiles from Integrated Microarray Datasets[J/OL]. PLOS ONE, 2011, 6(2): 1-12. https://doi.org/10.1371/journal.pone.0017259. DOI:10.1371/journal.pone.0017259.

[12] WEI H, FENG L, CHEN X, 等. Combating Noisy Labels by Agreement: A Joint Training Method with Co-Regularization[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.

[13] HAN B, YAO Q, YU X, 等. Co-teaching: robust training of deep neural networks with extremely noisy labels[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc., 2018: 8536-8546.

[14] ZHAO T. Predict disease classes using genetic microarray data[A/OL]. 2020: 1. https://github.com/zyz9066/Data-Analysis/blob/31623dbc70b7f4e5525af4ec1a4f6dc0453d1f1d/Predict%20disease%20classes%20using%20genetic%20microarray%20data/Project3.pdf.

DARMAWAN Jason Rich (1999—), Male, Master’s Degree Candidate, Main research areas are Computer Vision, Medical Imaging.