Sparse Model for Leukemia Gene Selection and Subtype Classification

Changxu Luo, Department of Electrical Engineering, Columbia University

***Abstract*—Classifying disease type by monitoring gene microarrays is critical for proper cancer treatments. Gene selection technique is used to reduce the gene features size for a target disease type classification problem in order to improve the algorithm performance and better understanding the relations between the gene expressions and disease. In this project, a sparse model for gene selection and subtype classification is proposed. The model implements l_1 norm to reconstruct the latent sparsity embedded in the gene microarrays. The result of this model applied to two Leukemia gene microarrays datasets indicates that the model can capture the microarrays pattern well by maintaining the similar classification error with the result of using the entire gene features, while decreasing the size of gene features by two to three orders of magnitude, greatly improving the efficiency of the classification algorithm[1].**

***Keywords*—Sparse Model, Gene Selection, Classification**

I. Introduction

CANCER treatment remains to be a great challenge for medical organization. One of the reason why cancer treatment is hard to optimize is the technical challenge on tumor type classification, while the correct classification contribute a lot to the specific proper therapy. For this reason, it is critical to develop a classification method that is both robust and efficient. Current classification methods include monitoring the tumor's morphological, clinical, and molecular variables[1]. Researchers have shown that, the DNA microarrays can express differently in normal tissues and cancers[2], so that it is possible to classify the type of tumors according to the expression difference from their DNA sequences. Many studies in this topic have proved this result by clustering analysis of tumor and normal tissues' DNA arrays[3]. However, using the entire DNA array for solving this problem is not the best solution for several reasons. First, a typical DNA microarray sample can have more than 20,000 probe values, which makes it difficult to extract the latent pattern for a specific type of a cancer due to the lack of the samples. Also, different genes may have different expression weights on the progression of various tumors, and for a certain type of tumor, most of the genes may perform a similar pattern to those same genes in the normal tissues, which brings a great noises in this classification problem and wastes a lot of computational power, negatively influencing the performance of classification algorithm. For these two reasons, if we can shrink the size of the sample and find the genes that contributes most to the expression differences between the tumors and the normal tissues, it can greatly improve the performance of the classification and help medical professionals in understanding the disease. Perform a gene selection step, that is, finding the genes that correlated most to the distinction problem before sending the gene data to the classification algorithm is a good way for shrinking the sample size.

The traditional way for gene selections applies the statistical processing on the data, ranking the gene features by introducing a way of calculating their importance[4][5][6]. But these models doesn't guarantee that the chosen genes represents a best sparse selected features in terms of the mathematical structure. In this project, a method that applies the sparse regression model for gene selection is introduced and tested on the problem of differentiating different subtypes of Leukemia by the DNA microarrays samples. The method introduces the l_1 norm to guarantee the sparsity of the solution, that is, to push the contribution of most gene features to the type classification to zero. The remaining parts of the introduction will first give a brief sketch on the Leukemia subtypes classification problem, then show why this sparsity structure can work in this problem, and provide the description on the two datasets used in this project.

A. Leukemia Subtypes Classification Problem

Leukemia is a cancer of blood or bone marrow that has very poor prognosis in clinical situation[7]. There are four major subtypes of Leukemia: Acute Lymphoblastic Leukemia(ALL), Chronic Lymphoctic Leukeia(CLL), Acute Myelogenous Leukemia(AML), and Chronic Myelogenous Leukemia(CML), in which the acute subtypes are more general to see in clinical than the chronic types. It is important to differentiate the correct subtype of Leukemia for the patients to get proper treatment[8]. Since it has high heterogeneity in its clinical manifestations, monitoring the change in patient's gene samples becomes a critical way for assisting the diagnosis. Studies have shown that it is possible to distinct the subtypes of Leukemia by observing the patterns in patient's DNA microarrays[9]. Some studies also showed that some rare type of Leukemia can also be distinguished by the gene expression profile[8].

B. Sparsity Structure

In a study of using genes expression values for Acute Leukemia subtype distinction[9], researchers have shown that for Acute Leukemia there are roughly 1100 genes that were highly correlated with the AML-ALL class distinction. Compared to the total number of genes in one sample which is

[1]Source codes and data of this project can be found at https://github.com/camelboat/6878_ELEN_Sparse_and_Low-Dimensional_Models_for_Hi-Dimensional_Data_Project

greater than 20,000, this number shows that the genes that actually matter in this classification problem take a small proportion of the total genes, and the selected genes are sparse in the complete genes pool.

C. Background of the Datasets

The project applies two different datasets in testing the performance of the gene selection algorithm. The first dataset is the gene expression dataset[10] from Golub et al.'s research in 1999[9]. This is one of those earliest researches which showed that the AML-ALL subtypes can be classified by monitoring the gene microarray. The dataset contains 72 samples for AML and ALL from bone marrow and peripheral blood. The gene microarrays have been preprocessed to maintain equivalent intensities for each chip. For the version used in this project, each sample has 7129 gene values.

The other dataset is the E-TABM-185 dataset, which comes from the research "A Global Map of Human Gene Expression"[11] by the European Bioinformatics Institute, and can be found on the ArrayExpress Website[12]. It "collects 5372 human samples representing 369 different cell and tissues types, disease states and cell lines"[11]. Among these samples, there are 722 samples whose disease type is Leukemia, in which there are 315 ALL, 342 AML, and 65 CML samples. Each sample has 22,283 different gene probes in its microarrays.

II. Technical Approach

A. Gene Selection via Sparse Regression

Suppose the dataset we use has n DNA microarray samples from different patients that diagnosed with different subtypes of Leukemia, and each sample has d genes expression. For a specific sample, we model its subtype class as the result of the superposition of all its genes expression as

$$\mathbf{X}\beta = y \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{N\times D}$ is the dictionary of the observed gene data samples organized by horizontally stacking the samples $X_i \in \mathbb{R}^D$ as

$$\mathbf{X} = \begin{bmatrix} — & X_1 & — \\ — & X_2 & — \\ \cdots & \vdots & \cdots \\ — & X_N & — \end{bmatrix}, \quad X_i = [x_{i1}, x_{i2}, \cdots, x_{iD}] \tag{2}$$

$\beta \in \mathbb{R}^D$ is the gene coefficient vector, in which value β_i represents the contribution of the gene expression at location i to the subtype class. We expect that most elements in β should be zero, and the non-zero elements' indexes are the gene indexes that we want to select for classification. $y \in \mathbb{R}^N$ is the subtype class vector, and y_n represents the class of the n-th sample. In this project, we assign the class of ALL, AML, CLL, and CML to class 0, 1, 2, and 3 respectively.

Since the number of genes d is much larger than the observed samples number n, there will be infinitely many possible solutions to this equation. The sparse structure that we expect β to have can solve this problem. In order to get a sparse β, i.e., to minimize the number of the non-zero elements in β so that the number of selected genes for the classification problem is small, while mostly persisting the latent pattern that is related to the subtypes classification problem, we introduce a l_1 minimization problem, that is, to find the β that has the minimum l_1 norm.

$$\min_\beta \|\beta\|_1 \text{ s.t. } \mathbf{X}\beta = y \tag{3}$$

l_1 norm can correctly recovers sparse feature[13]. To solve with the noise that encountered when measuring the the microarrays, we can extend the l_1 minimization problem to the Lasso problem[14]

$$\min_\beta \lambda\|\beta\|_1 + \frac{1}{2}\|\mathbf{X}\beta - y\|_2^2 \tag{4}$$

in which λ is a regularization parameter to trade off between the sparsity of the result and the distance between $\mathbb{X}\beta$ and y. Since l_1 is a non-smooth function, this optimization problem cannot apply simple gradient descent method. To solve this problem, technique called proximal gradient descent is applied. Here we provide a brief description for how it works under this setup[2].

The proximal gradient method tries to minimize convex objective function $F(\beta)$ that can be written as the sum of a smooth convex function $f(\beta)$ and a convex but potentially non-smooth function $g(\beta)$

$$F(\beta) = f(\beta) + g(\beta) \tag{5}$$

which, for the Lasso problem we want to solve in this project, $f(\beta) = \frac{1}{2}\|\mathbf{X}\beta - y\|_2^2$ and $g(\beta) = \lambda\|\beta\|_1$. Then the proximal gradient method is

$$\beta^{(k+1)} = \text{prox}_{\alpha_k g}\left(\beta^{(k)} - \alpha^{(k)}\nabla f\left(\beta^{(k)}\right)\right) \tag{6}$$

where the proximal operator prox is

$$\text{prox}_g(w) = \underset{\beta}{\text{argmin}}\left\{g(\beta) + \frac{1}{2}\|\beta - w\|_2^2\right\} \tag{7}$$

We then iteratively update the value of w and the proximal operator until the Lasso objective doesn't change much. For each iteration, we first update w by taking a gradient descent with step size $\alpha^{(k)}$

$$w^{(k)} = \beta^{(k)} - \alpha^{(k)}\nabla f(\beta^{(k)}) \tag{8}$$

in which for Lasso problem, we can calculate ∇f as

$$\nabla f(\beta) = \mathbf{X}^*(\mathbf{X}\beta - y) \tag{9}$$

where $\mathbf{X}^*$ is the conjugate of $\mathbf{X}$. The step size $\alpha^{(k)}$ can be chosen as the reciprocal of the Lipschitz constant. We can show that for Lasso problem here ∇f is $\|\mathbf{X}\|^2$-Lipschitz, where $\|\mathbf{X}\|^2$ is the operator norm of $\mathbf{X}$, so that we choose $\alpha^{(k)} = 1/\|\mathbf{X}\|^2$.

Then we calculate the proximal operator

$$\text{prox}_{\alpha^{(k)}g}\left(w^{(k)}\right) = \underset{\beta}{\text{argmin}}\left\{\alpha^{(k)}g(\beta) + \frac{1}{2}\|\beta - w\|_2^2\right\} \tag{10}$$

[2]Detailed descriptions and proofs can also be found in [14] and [15].

For the scaled l_1 norm, the closed-form expression for the proximal mapping can be calculated as

$$\left[\text{prox}_{\alpha\|\cdot\|_1}(w)\right]_i = \begin{cases} w_i - \alpha & \text{if } w_i > \alpha \\ 0 & \text{if } |w_i| \leq \alpha \ ) \\ w_i + \alpha & \text{if } w_i < -\alpha \end{cases} \tag{11}$$

where $\alpha = 1/\|\mathbf{X}\|^2$.

After running the proximal gradient descent until the objective function value doesn't change much, we will get a sparse β as the result, in which the indexes of the non-zero element are the indexes of the selected genes. For each dataset, different value of λ is chosen for testing, and the result for each λ is tested for classification, in which the one with the lowest error rate will be chosen as the best λ.

B. Classification

To test the result of gene selection, we use the selected genes of randomly chosen 75% of the complete data as the training set to build a classifier, and test the accuracy on the remaining 25% data. To build the classifier, we first use the selected genes in the training set to construct the new $\hat{\mathbf{X}}$. Suppose we have selected k genes from the d genes, then $\hat{\mathbf{X}}$ is constructed as

$$\hat{\mathbf{X}} = \begin{bmatrix} — & \hat{X}_1 & — \\ — & \hat{X}_2 & — \\ \cdots & \vdots & \cdots \\ — & \hat{X}_{\lfloor 0.75N \rfloor} & — \end{bmatrix}, \quad \hat{X}_i = [x_{i1}, x_{i2}, \cdots, x_{ik}] \tag{12}$$

and solve the Lasso problem

$$\min_{\hat{\beta}} \lambda\|\hat{\beta}\|_1 + \frac{1}{2}\|\hat{\mathbf{X}}\hat{\beta} - \hat{y}\| \tag{13}$$

by the proximal gradient descent method, where $\hat{\beta} \in \mathbb{R}^k$ is the coefficient vector to use in the prediction. For a sample with the selected genes value $x \in \mathbb{R}^k$ in the training set, we predict its class j as

$$j = \underset{j}{\text{argmin}} \left| x\hat{\beta} - j \right|, \quad j \in \{0, 1, 2, 3\} \tag{14}$$

For each λ, the classification model is built for multiple times for differently chosen random training set in order to calculate the average value of accuracy.

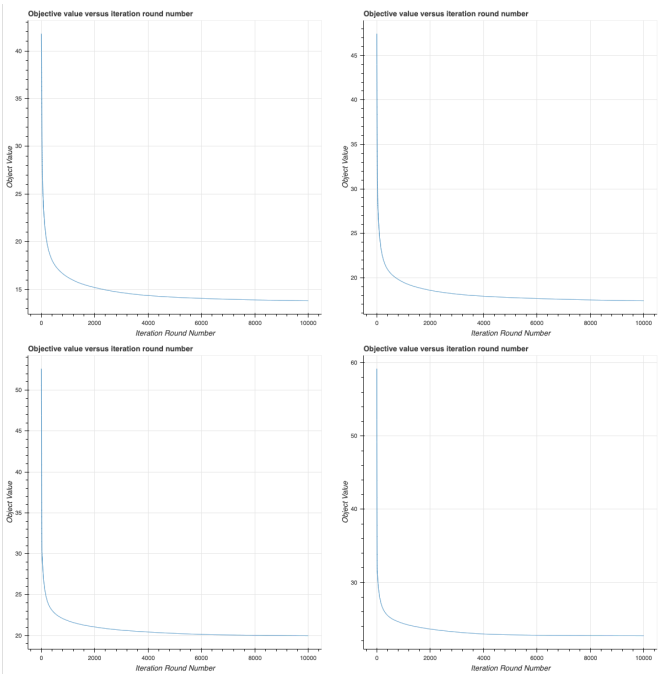

Fig. 1. Objective function value versus the iteration round number on Golub et al.'s dataset. From left to right then top to bottom, $\lambda = 0.003, 0.005, 0.007$, and 0.01.

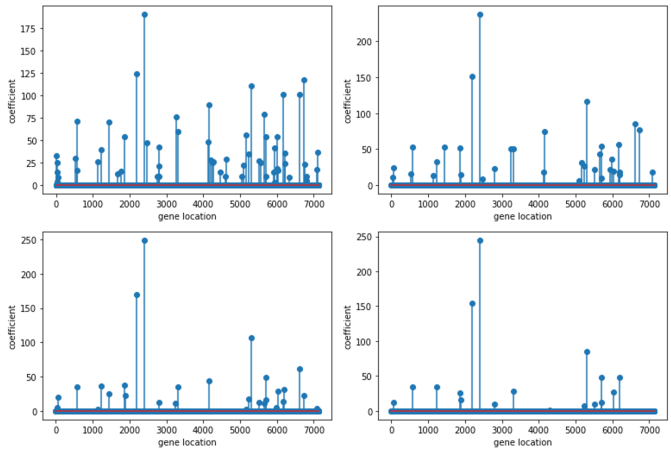

Fig. 2. Resulting β after 10000 rounds on Golub et al.'s dataset. From left to right then top to bottom, $\lambda = 0.003, 0.005, 0.007$, and 0.01.

III. Experimental Results

A. Golub et al.'s Gene Expression Dataset

For Golub et al.'s dataset, we choose λ =0.0001, 0.0003, 0.0005, 0.0007, 0.001, 0.003, 0.005, 0.007, 0.01, 0.03, and 0.05 to run the proximal gradient descent, each for 10,000 iterations. We choose $\lambda = 0.003, 0.005, 0.007$, and 0.01 to show the example plot of objective value versus the iteration round number, and the stem plot of the resulting β in figure 1 and 2.

From these result, we can observe that the objective function value decreases monotonically and dramatically with the increasing of the round number, which means that the proximal gradient descent method works well here. With the increasing of λ, the number of non-zero elements in β decreases, however, from figure 2 we can observe that, several genes are always selected no matter what λ is chosen. The 14 genes that are always chosen when λ is $0.003, 0.005, 0.007$, and 0.01 are 'M25079_s_at', 'AFFX-M27830_5_at', 'L20688_at', 'L06499_at', 'Z84721_cds2_at', 'M96326_rna1_at', 'L04483_s_at', 'HG1428-HT1428_s_at', 'U49869_rna1_at', 'U14973_at', 'M69043_at', 'M26602_at', 'D86974_at', 'D79205_at'. If we use these 14 genes to build the classifier, the average error rate would be about 18.33%(20 rounds), which is about the lowest error rate in this linear model setting. This means that the gene selection algorithm successfully chooses the genes that contribute most to the disease type distinction.

The result of number of non-zero elements in β, and the average error rate in classification(20 rounds) for corresponding λ is shown in table I. Each λ's classifier is built for 10 times for getting the average error rate. The plot of both number of non-zero element in β and error rate versus λ is shown in figure 3. We can see when λ is in a certain range (between 0.0001 and 0.01), the number of non-zero elements in β doesn't have a high influence on the average error rate, which again shows that the sparse regression algorithm successfully eliminates the genes that doesn't really correlated with the disease subtype classification[3].

TABLE I
THE NUMBER OF NON-ZERO ELEMENTS IN RESULTING β AND AVERAGE CLASSIFICATION ERROR RATES FOR EACH λ ON GOLUB ET AL.'S DATASET.

λ	# $\beta_d > 0$	Error Rate(%)
0.0001	1370	22.5
0.0003	458	22.5
0.0005	290	17.8
0.0007	222	25.8
0.001	159	25.6
0.003	59	33.9
0.005	34	26.4
0.007	28	28.6
0.01	17	23.9
0.03	5	36.1
0.05	2	34.4

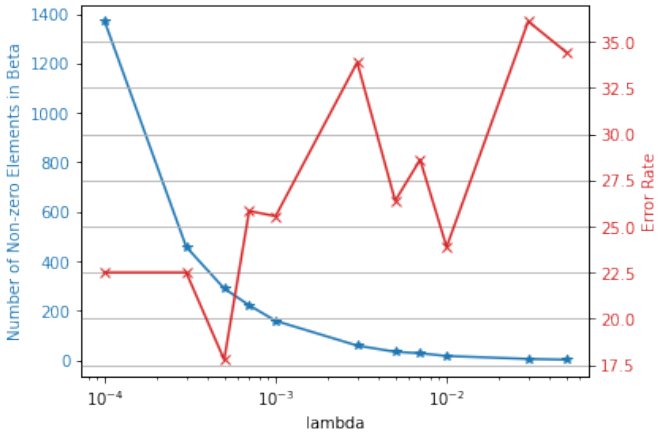

Fig. 3. Number of non-zero elements in resulting β and average error rates versus λ on Golub et al.'s dataset.

B. E-TABM-185 Dataset

For E-TABM-185 dataset, we choose λ =0.07, 0.1, 0.3, 0.5, 0.7, 1.0, 1.3, 1.5, 2.0, 2.5, 3.0, and 5.0, and iteration number of 10,000 to run the sparse regression model. We choose the $\lambda = 0.5, 0.7, 1.0$ and 1.3 as the typical result to show their plots of objective function value versus iteration number in figure 4 and the resulting β value in figure 5. These results show that the sparse regression model also converges efficiently with the proximal gradient descent methods on this dataset. With the increasing of λ, the number of non-zero coefficients in β decreases, but some of the genes coefficients keep to be selected when λ is between a certain range. For λ equals to the four values shown here, there are 26 genes that are always selected, and they are the same genes selected when $\lambda = 1.3$, which are "218618_s_at", "204951_at", "204777_s_at", "205382_s_at", "206871_at", "204891_s_at", "221558_s_at", "212012_at", "206111_at", "203591_s_at", "207826_s_at", "214575_s_at", "210487_at", "220416_at", "206120_at", "209930_s_at", "201360_at", "222044_at", "200742_s_at", "203799_at", "203948_s_at", "221349_at", "206001_at", "203949_at", "204170_s_at", and "205267_at". If we only use these 26 genes to build the linear classifier, it would have an average error rate of 2.12%, which is about only 1.2% worse than result got by using 1001 genes. This means that on E-TABM-185 dataset, the sparse regression model also succeeds in finding the genes that contribute most to the Leukemia subtype classification.

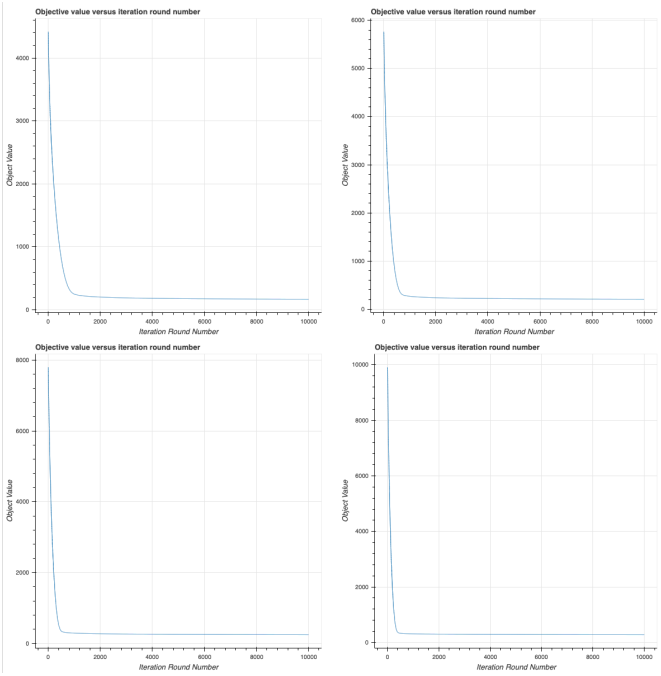

Fig. 4. Objective function value versus the iteration round number on E-TABM-185 dataset. From left to right then top to bottom, $\lambda = 0.5, 0.7, 1.0$, and 1.3.

Table II shows the data of λ and its corresponding selected gene size and average error rate in 10 rounds. These data are also plot in figure 6. This figure shows that, the size of selected genes shrinks dramatically with the increasing of λ, while the error rate of the classification model built on those selected genes remains to be low ($<3\%$) when the selected genes size is larger than 10. When size selected genes is less than 10, for example, when $\lambda = 2.5$, and the selected genes size is 3, the error rate of the classification model jumps to 42.2%, which becomes unacceptable for the classification.

[3]The error rate of the simple support vector machine(SVM) model on the original Golub et al.'s dataset is about 33%, so that these selected genes almost all positively contribute to the classification accuracy

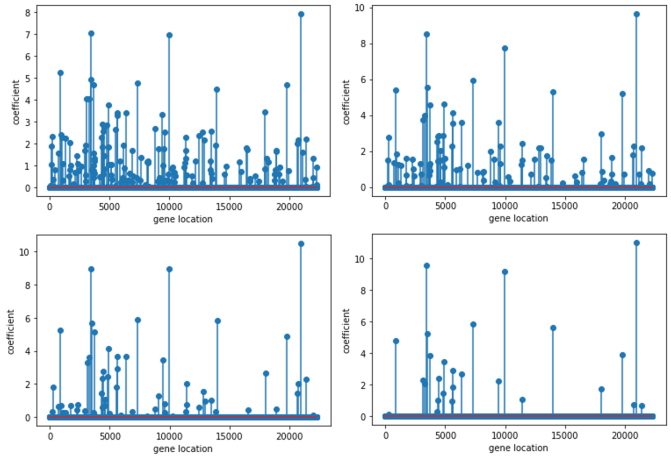

Fig. 5. Resulting β after 10000 rounds on E-TABM-185 dataset. From left to right then top to bottom, $\lambda = 0.5, 0.7, 1.0$, and 1.3.

TABLE II
THE NUMBER OF NON-ZERO ELEMENTS IN RESULTING β AND AVERAGE CLASSIFICATION ERROR RATES FOR EACH λ ON E-TABM-185 DATASET.

λ	# $\beta_d > 0$	Error Rate(%)
0.07	1531	1.10
0.1	1001	0.85
0.3	275	0.97
0.5	164	1.70
0.7	105	1.45
1.0	62	1.82
1.3	26	1.21
1.5	19	2.18
2.0	9	2.67
2.5	3	42.18
3.0	1	48.73
5.0	0	N/A

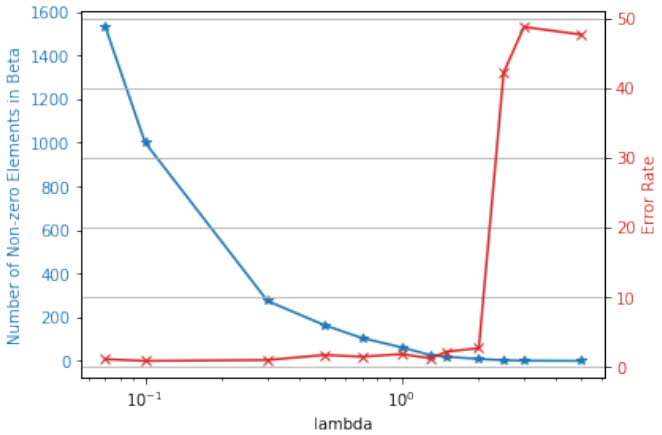

Fig. 6. Number of non-zero elements in resulting β and average error rates versus λ on E-TABM-185 dataset.

IV. DISCUSSION

The results of the sparse regression model on both Leukemia genes microarray datasets show that the sparse model succeeds in selecting the genes that are most related to the disease subtype classification problem, and the linear classification model built on the selected genes performs well in testing. On Golub et al.'s dataset, the classification model using the 14 most related genes have the average classification error rate of 18.33%, and on E-TABM-185 dataset, the classification model built on the 26 most highly contributed genes can reach the error rate of 2.12%, and this value can be decreased to less than 1% when the size of selected genes are set to be at 300. This result conforms well with what is indicated in Dudoit et al.'s study[1] that "for Leukemia datasets, increasing the number of genes (up to $p = 200$) did not affect greatly the performance of the various predictors".

As a comparison, we also run a simple support vector machine(SVM) model on both datasets for all gene features and the minimum selected 14 and 26 genes(for each case, the model is run for 100 times on different random choices on the training set and testing set to get the average error rate). On Golub et al.'s dataset, the error rate for the entire genes is 33.39%, and 19.56% for the 14 selected genes. On the other dataset, these two values are 0.47% and 1.44% respectively. These values show that, our selected genes don't lose much information on the disease subtype classification(it even improves the classification accuracy on Golub et al.'s dataset). Although the linear classification model doesn't provide the best performance in this problem, its performance is high enough to see the benefits of the gene selection step. For better classification accuracy, more complicated model and techniques such as neural networks and boosting can be used as the classification model.

There are still some issues remain to solve. The first one is that, the CLL and CML data are very little in the two chosen datasets, which can bring a huge bias to the model, affecting the model performance. To see if the selected genes also work well on those two subtypes, more gene microarrays data is needed. Also, in future works, the sparse regression model can be run on each subtype of the Leukemia data, and it remains to check whether the indicator genes are the same. Whether this sparse regression model can work on disease discrimination other than Leukemia should also be tested. These works, along with explaining the mechanism of the chosen genes and associating the gene selection results on two datasets, require collaborations with medical professionals.

Another probable improvement for the model may on the genes redundancy elimination. Researches have shown that for genes microarray, some genes may express in the similar way, which brings the redundancy in the features[16]. In the classification part of the current sparse regression model, the weights for all genes are learned after the selection step, and the redundancy problem is not considered, so that other methods need to apply in order to detect and eliminate the redundancy problem.

V. Conclusion

In this project, we proposed a sparse regression model for selecting the genes that has the strongest correlation with the disease type classification problem. The sparse regression model applies a linear regression model with l_1 norm to reconstruct the sparsity of the gene features in the classification problem. The model was tested on the Leukemia subtypes classification problem with Golub et al.'s dataset and E-TABM-185 dataset. The testing results indicate that the genes selected by the sparse regression model captures the latent pattern in the classification problem well by maintaining the similar classification error with the entire gene features, while the sparse regression model dramatically decreases the size of the genes by two to three orders of magnitude. In some cases, classification by the selected genes can even provide a better accuracy than using the whole gene features.

Acknowledgment

This is a report for course project of ELEN E6878 Sparse and Low-Dimensional Models for Hi-Dimensional Data at Columbia University. The author would like to thank the instructor Prof. John Wright and Teaching Assistant Tim Wang for their great suggestions and helps during the entire process of the project as well as all their diligent works in the whole semester.

References

[1] S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Journal of the American statistical association*, vol. 97, no. 457, pp. 77–87, 2002.

[2] S. Volinia, G. A. Calin, C.-G. Liu, S. Ambs, A. Cimmino, F. Petrocca, R. Visone, M. Iorio, C. Roldo, M. Ferracin, *et al.*, "A microrna expression signature of human solid tumors defines cancer gene targets," *Proceedings of the National Academy of Sciences*, vol. 103, no. 7, pp. 2257–2261, 2006.

[3] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proceedings of the National Academy of Sciences*, vol. 96, no. 12, pp. 6745–6750, 1999.

[4] S. K. Shevade and S. S. Keerthi, "A simple and efficient algorithm for gene selection using sparse logistic regression," *Bioinformatics*, vol. 19, no. 17, pp. 2246–2253, 2003.

[5] J. Gui and H. Li, "Penalized cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data," *Bioinformatics*, vol. 21, no. 13, pp. 3001–3008, 2005.

[6] T. Jirapech-Umpai and S. Aitken, "Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes," *BMC bioinformatics*, vol. 6, no. 1, p. 148, 2005.

[7] J. Xu, B. Huang, X. Liu, Y. Zhang, Y. Liu, L. Chen, Y. Luan, N. Li, and X. Chu, "Poor prognosis in acute myeloid leukemia patients with monosomal karyotypes," *Turkish Journal of Hematology*, vol. 34, no. 2, p. 126, 2017.

[8] S. A. Armstrong, J. E. Staunton, L. B. Silverman, R. Pieters, M. L. den Boer, M. D. Minden, S. E. Sallan, E. S. Lander, T. R. Golub, and S. J. Korsmeyer, "Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature genetics*, vol. 30, no. 1, pp. 41–47, 2002.

[9] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, *et al.*, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *science*, vol. 286, no. 5439, pp. 531–537, 1999.

[10] G. et al., "Gene Expression Dataset from Golub et al. (1999)." http://web.mit.edu/r/current/arch/i386_linux26/lib/R/library/multtest/html/golub.html.

[11] M. Lukk, M. Kapushesky, J. Nikkilä, H. Parkinson, A. Goncalves, W. Huber, E. Ukkonen, and A. Brazma, "A global map of human gene expression," *Nature biotechnology*, vol. 28, no. 4, pp. 322–324, 2010.

[12] M. Lukk, "E-TABM-185 - Transcription profiling by array of integrated human experiments involving the hgu133a platform to investigate a global map of human gene expression." https://www.ebi.ac.uk/arrayexpress/experiments/E-TABM-185/.

[13] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.

[14] Wright and Ma, *Sparse and Low-Dimensional Models for High-Dimensional Data: Theory, Algorithms and Applications*. Cambridge, 2020.

[15] A. Antoniadis and J. Fan, "Regularization of wavelet approximations," *Journal of the American Statistical Association*, vol. 96, no. 455, pp. 939–967, 2001.

[16] K. Tian, L. Jing, and N. Du, "Sparse representation-based gene selection for cancer prediction," in *2011 4th International Conference on Biomedical Engineering and Informatics (BMEI)*, vol. 4, pp. 1789–1793, IEEE, 2011.