An attempt at Leukemia Classification

Andrew Wei  
Paul G. Allen School of   
Computer Science & Engineering  
nowei@cs.washington.edu

# Abstract

We classify genes associated Leukemia and specific sub-types of Leukemia to find gene expression patterns that exist within Leukemia patients and specific sub-types of Leukemia by examining each sub-type of Leukemia and comparing significant gene expressions. We also attempt to classify specific subtypes of Leukemia with the intent of analyzing learned patterns from the results.

# Introduction

Leukemia is a type of blood cancer that produces excess, abnormal white blood cells that inhibit the production of normal blood cells. There are 17 sub-types of Leukemia and each sub-type is characterized by a different set of regulatory genes. Instead of identifying the gene regulatory networks, we analyze the effects of these regulatory networks as we believe they can give us important clues about the impacts and causes of certain sub-types of Leukemia. With this information, we may have a better idea of the genes involved in the specific regulatory networks of certain sub-types of Leukemia, allowing for better understanding of these gene regulatory networks and the impactful genes involved in the disease.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Count** | **Name** | **Count** |
| MDS | 207 | Non-Leukemia and healthy bone marrow | 73 |
| CLL | 448 | c-ALL/Pre-B-ALL with t(9;22) | 122 |
| AML complex aberrant karyotype | 52 | AML with t(8;21) | 40 |
| AML with normal karyotype + other abnormalities | 347 | ALL with hyperdiploid karyotype | 40 |
| c-ALL/Pre-B-ALL without t(9;22) | 237 | ALL with t(1;19) | 36 |
| T-ALL | 174 | Pro-B-ALL with t(11q23)/MLL | 70 |
| CML | 76 | AML with t(15;17) | 37 |
| AML with t(11q23)/MLL | 38 | AML with inv(16)/t(16;16) | 28 |
| ALL with t(12;21) | 58 | mature B-ALL with t(8;14) | 13 |

Table 1. Distribution of data

We use array-normalized data from the Microarray Innovations in Leukemia study (MILE) [1]. This dataset contains data from 2096 blood and bone marrow samples from patients and has 17,788 genes with 18 classes, 17-subtypes of Leukemia and 1 group of healthy patients.

There is a data imbalance that we will not address in this project, but we note that this may make it harder to learn the important features for classifications of smaller sized subtypes. The small number of samples for some subtypes may cause the model to fit more of the noise than the significant genes involved in the disease. We also note that with the large number of genes, it will be hard to determine the impact of specific genes and verify the results. This problem is challenging because with the rise of machine learning and deep learning, numerous different methods will likely lead to reasonable results, so it is hard to determine which method to use and how to interpret the results. Thus, we use relatively simpler methods to maintain the interpretability of the results.

# Background

Cancer classification on gene expression data using machine learning techniques is not new. There are many published papers with new techniques, and applications of old techniques to new datasets that come out every year. We surveyed a handful of them to get a better idea of approaches we could try when tackling the problem with our gene expression data.

Cascianelli et. al used PCA on the PAM50 dataset to examine whether the subtypes of breast cancer were separable from each other [2]. They found that they could not separate the subtypes by the first two principle components and hypothesized that due to this non-separability, the boundary between subtypes changes depending on the mixed traits. Using Average of Within Class Averages (AWCA), they were able to get a single number that represents a subtype. This method produced relatively decent results with around 90% accuracy. They then went on to try different machine learning approaches and found that regularized multiclass logistic regression performed the best.

Danaee et. al used stacked denoising autoencoders to deal with the high dimensions and noisy inputs of gene expression data [3]. The purpose of using an encoder is denoise the inputs and to try to find the genes that play an important role in breast cancer. They multiply the learned autoencoder weights together to get a rough sense of how the gene expression data interacts, saying that the most heavily weighted ones have the most significant impact and were the most predictive genes. By using the heavily weighted genes as the inputs to a deep learning models, they were able to achieve slightly less accuracy than with the features found using stacked denoising autoencoders but argue that it is more readily interpretable.

Mei et. al trained a stacked autoencoder to provide input to a neural network trained on a subset of the TCGA dataset to identify survival duration for patients with Acute Myeloid Leukemia [4]. They include features like age, cytogenetics, and mutations and achieve results of around 81% accuracy for predicting whether patients survived for more of less than 730 days. They mention that the main limitation to their model was the amount of training data available.

Castillo et. al try to extract differentially expressed genes that help identify different forms of Leukemia [5]. Use the minimum-Redundancy Maximum-Relevance feature selection algorithm, they train SVM, Random Forests, K-Nearest Neighbors, and Naive Bayes classifiers. Afterwards, they extract Differentially Expressed Genes (genes that help differentiate between classes) by testing p-values, log-fold change (measures change in gene expression level), and coverage (whether it helps identify between some number of classes). They conclude that K-Nearest Neighbors with a subset of gene expression data worked the best, followed by SVM and Naive Bayes, with the worst being Random Forests.

Krivtsov et. al outlines procedures on how to extract the gene expression data and utilize some form of either hierarchical or K-means clustering and use permutation analysis to determine significance for the purpose of profiling Leukemia Stem Cells [6]. They use the GenePattern software to distinguish marker genes that would distinguish the two groups, comparing means, using a signal-to-noise statistic, and evaluating significance. This was the only study read that seemed to have extracted and recorded the data themselves instead of taking the data from a database.

These papers showed that there were many ways to approach the problem of cancer classification using gene expression data and have inspired us to take a somewhat similar approaches, with an emphasis on finding DEGs or significant features.

# Results and Methods

## Methods

We take a multifaceted approach to this problem of identifying DEGs and classifying Leukemia sub-types. We use a 90/10 train/test split to leave some data for evaluation purposes.

To look for significant features within each Leukemia sub-type, we perform Welch’s t-test using the training data and compare each sub-type against the “Non-Leukemia and healthy bone marrow” patients. Then we apply the Bonferroni correction with and to correct for multiple hypotheses. Afterwards, we determine the similarity of significant features between Leukemia sub-types to get a better idea of how similar the selected genes were between Leukemia sub-types, so we looked at the Jaccard similarity (a set similarity metric) between the selected significant features between Leukemia sub-types. We then look up some of the top shared features to confirm the pattern was also observed by others.

Afterwards, we attempt to classify the Leukemia sub-types by using multi-class Logistic Regression, which uses a 1 vs. all classification scheme and has a model with learned weights for each classification. We focus on using L1 regularization because we believe that not all gene expressions are relevant to determining whether someone has a specific sub-type of Leukemia or not. We varied our training by using all of the features and just using the combined significant features we found, along with three normalization schemes: not normalizing, normalizing by the training data, and normalizing by the healthy patients in the training data. The same normalization schemes are applied when evaluating the model. We use 5-fold cross-validation to get a general idea of the generalizability of our different models. We then look at the similarity between learned weights of the learned models as well as the learned weights themselves to examine the most impactful genes found by the model. We conclude by comparing the classification accuracy of logistic regression with K-Nearest Neighbors. We vary the number of neighbors and whether we use all features or just the significant features.

We believe that our approach will work well on this problem because it allows us to tackle the problem of identifying the significant genes from multiple directions to arrive at a better understanding of how the different genes interact and what are significant indicators of each sub-type of Leukemia. Our initial methods differ from existing methods in that we perform our analysis on the 17 sub-types of Leukemia, while previous research we have seen only does this analysis among 4 sub-types of Leukemia [5] and do not focus on the similarities of the significant features. Our latter methods do not differ much from existing methods in that it focuses on using a machine learning algorithm as a means of obtaining determining significant features.

We evaluate our ability to locate significant features by checking some of the genes against existing research to confirm that these patterns were also observed by other researchers. This leaves us with many genes that could be involved in the gene regulatory networks that may or may not have been observed by other researchers that could be a focus on future research on specific gene regulatory networks for specific sub-types of Leukemia. We evaluate our classification results on the top-1 and top-5 accuracy because there are 18 total classes, and we want to see how close our model is at predicting the correct class.

## Results

After selecting out the significant features for each sub-type of Leukemia, we found a total of 1,408 significant features. We also examined the [Jaccard similarity](https://github.com/nowei/leukemia-classification/blob/main/features/jaccard.png) the found significant features and learned that *MDS* seemed to be the most different sub-type of Leukemia from the other sub-types. This may require some more investigation in the future. The number of labels that share a particular significant feature range from 1 to 16, so we can look at a handful of the top shared and least shared labels for significant genes and look for them in existing research.

|  |  |  |  |
| --- | --- | --- | --- |
| **Gene** | **Description** | **# of shared labels** | **Observed in** |
| 10487\_at | CAP1 | 16 | [7] |
| 1116\_at | CHI3L1 | 15 | [8] |
| 116362\_at | RBP7 | 15 | [9] |
| 10123\_at | ARL4C | 15 | [10] |
| 1118\_at | CHIT1 | 15 | [11] |
| 114880\_at | OSBPL6 | 1 (MDS) | [12] |
| 100128907\_at | hypothetical protein LOC100128907 | 1 (MDS) | Not found |
| 10006\_at | ABI1 | 1 (CLL) | [13] |
| 100128309\_at | hypothetical protein LOC100128309 | 1 (CLL) | Not found |
| 100129015\_at | hypothetical protein LOC100129015 | 1 (CLL) | Not found |

Table 2. Selected results from the search for significant features

We observe that some hypothetical proteins were marked as significant genes, so further studies could be carried out. There are likely other genes that have not been observed in studies, studying these genes might give us a better idea of how to distinguish some subtypes of Leukemia.

Then we tried multi-class Logistic Regression. Our [5-fold cross-validation results](https://github.com/nowei/leukemia-classification/blob/main/features/cross_validation_results.png) showed that not normalizing had the best top-5 accuracy with all the features and just with significant features, while normalizing by healthy patient data in the training dataset had the best top-1 accuracy. We decided to train with all the normalization schemes because training turned out to be quick. When training the models on the full data, we notice that each model is able to [learn the training data completely](https://github.com/nowei/leukemia-classification/blob/main/features/train_results.png). We present the test results below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Normalization Scheme** | **w/ all features** | | **w/ significant features** | |
| **Top 1 acc.** | **Top 5 acc.** | **Top 1 acc.** | **Top 5 acc.** |
| Don’t normalize | **0.919** | **1.0** | **0.881** | **0.990** |
| Normalize across entire training dataset | 0.919 | 0.995 | 0.857 | 0.981 |
| Normalize by healthy patient data in training dataset | 0.900 | 0.995 | 0.843 | 0.971 |

Table 3. Accuracy of multi-class Logistic Regression models on test data

It is reasonable that the accuracy would be higher with all the features, since significant features don’t include features that indicate that it isn’t some class, which forces the model to find indications that it isn’t some class from the remaining significant features for other classes. Since we used L1 regularization, we have seen that [many of the features in the models are zero-weighted](https://github.com/nowei/leukemia-classification/blob/main/features/num_zeros.png). We notice that on the [confusion matrix with using all the features](https://github.com/nowei/leukemia-classification/blob/main/graphs/cm_logreg_l1_test_.png) and no normalization, the errors we make are pretty sparse with the only readily visible pattern being that we misclassify *c-ALL/Pre-B-ALL without t(9;22)* relatively frequently. These errors might give us some clue as to the feature space that these classifications live in, e.g. how close *c-ALL/Pre-B-ALL without t(9;22)* is to *ALL with hyperdiploid karyotype*.

We can also check the heavily weighted weights our models found. These weights can give us a clue as to what features were most significant in determining whether a patient had a specific sub-type of Leukemia or not. To do this, we would have to look at the data and see whether patients generally had positive or negative records for the heavily weighted features within each subtype’s classification model. If the product is generally positive for a sub-type, then we know that it is a strong indicator of the specific sub-type and if it is generally negative, we know that the feature was a strong indicator that it was not that sub-type. As an example, we look at the most heavily weighted weights of *mature B-ALL with t(8;14).*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Weight | Name | Description | Values generally positive/negative | Product sign | Strong indicator |
| 0.275562 | 151126\_at | ZNF385B - zinc finger protein 385B | Positive | Positive | For |
| 0.204056 | 284013\_at | VMO1 - vitelline membrane outer layer 1 homolog (chicken) | Positive | Positive | For |
| -0.11895 | 7006\_at | TEC - tec protein tyrosine kinase | Positive | Negative | Against |
| -0.12564 | 100131601\_at | similar to hCG1980470 | Positive | Negative | Against |

Table 4. Heavily weighted values of learned weights and whether they are strong indicators for or against *mature B-ALL with t(8;14)* classification.

We also looked at the [cosine similarity between learned weights](https://github.com/nowei/leukemia-classification/blob/main/features/cosine_sim.png) for the best-performing model. We do this to get an idea of how similar the learned weights are. We observe that the majority of the learned weights are generally not in similar directions, with *Non-leukemia and healthy bone marrow* being the most opposed to *MDS* and *c-ALL/Pre-B-ALL with t(9;22)* being the most opposed to *c-ALL/Pre-B-ALL without t(9;22)*. Even though the learned weights were mostly dissimilar, it could give us some context for the errors that our model made. For example, most of the misclassifications our model made are between classes with relatively more opposing weights, which indicates that the model cannot correctly distinguish between these classes using the learned weights due to their sharing of weighted features. Since we learn the training set completely, this error can also be attributed to the variance between the training and test sets, making the model fail to generalize.

Then when trying K-Nearest Neighbors, as a point of comparison, we notice that our results were not as impressive as the ones we obtained through multi-class Logistic Regression.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # of neighbors | w/ all features | | w/ significant features | |
| Top 1 acc. | Top 5 acc. | Top 1 acc. | Top 5 acc. |
| 3 | 0.767 | 0.881 | 0.790 | 0.895 |
| 5 | 0.771 | 0.914 | 0.790 | 0.929 |
| 10 | **0.795** | 0.962 | **0.819** | **0.962** |
| 15 | 0.786 | 0.967 | **0.819** | 0.952 |
| 20 | 0.790 | **0.971** | 0.786 | 0.952 |

Table 5. Performance of K-Nearest Neighbors.

We observe the best top-5 performance using all the features and with 20 neighbors and the best top-1 performance using just the significant features with either 10 or 15 neighbors. When we observe the errors in [the confusion matrix](https://github.com/nowei/leukemia-classification/blob/main/graphs/cm_knn_test_20.png) for 10 neighbors, we see some of the same errors that the multi-class Logistic Regression model made, along with some new errors. This is likely due to the complex shape

## Interpretation

* + Why do you think that your method(s) performed in certain ways?
  + How does your method work compared to alternative approaches?
  + What is the implication of your results with respect to solving your problem?

# Conclusion

We address the computational challenge of finding significant features by…

In the future, we plan on…

We conclude by noting…

* + How is the computational challenge addressed?
  + Future direction (possible next steps)

# Bibliography

1. Kohlmann, Alexander et al. “An international standardization programme towards the application of gene expression profiling in routine leukaemia diagnostics: the Microarray Innovations in Leukemia study prephase.” British journal of haematology vol. 142,5 (2008): 802-7. doi:10.1111/j.1365-2141.2008.07261.x
2. Cascianelli, S., Molineris, I., Isella, C. *et al.* Machine learning for RNA sequencing-based intrinsic subtyping of breast cancer. *Sci Rep* **10,**14071 (2020). <https://doi.org/10.1038/s41598-020-70832-2>
3. Danaee, P., Ghaeini, R., Hendrix, D. A Deep Learning Approach for Cancer Detection and Relevant Gene Identification. PSB (2017).
4. Mei Lin, Vanya Jaitly, Iris Wang, Zhihong Hu, Lei Chen, Md. Amer Wahed, Zeyad Kanaan, Adan Rios, & Andy N. D. Nguyen. (2018). Application of Deep Learning on Predicting Prognosis of Acute Myeloid Leukemia with Cytogenetics, Age, and Mutations.
5. Castillo, Daniel et al. “Leukemia multiclass assessment and classification from Microarray and RNA-seq technologies integration at gene expression level.” *PloS one* vol. 14,2 e0212127. 12 Feb. 2019, doi:10.1371/journal.pone.0212127
6. Krivtsov, Andrei V et al. “Gene expression profiling of Leukemia stem cells.” *Methods in molecular biology (Clifton, N.J.)* vol. 538 (2009): 231-46. doi:10.1007/978-1-59745-418-6\_11
7. Xie, Shuanshuan et al. “Systematic analysis of gene expression alterations and clinical outcomes of adenylate cyclase-associated protein in cancer.” Oncotarget vol. 8,16 (2017): 27216-27239. doi:10.18632/oncotarget.16111
8. Bergmann OJ, et al. High serum concentration of YKL-40 is associated with short survival in patients with acute myeloid leukemia. Clin Cancer Res. 2005;11(24 Pt 1):8644–8652.
9. Qu, Ying, et al. “Novel Gene Signature Reveals Prognostic Model in Acute Myeloid Leukemia.” Frontiers in Genetics, vol. 11, Oct. 2020, p. 566024. DOI.org (Crossref), doi:10.3389/fgene.2020.566024.
10. Engel, T. et al. <https://www.wikigenes.org/e/gene/e/10123.html?vs=1>
11. Cho, Soo Jung et al. “Chitotriosidase in the Pathogenesis of Inflammation, Interstitial Lung Diseases and COPD.” Allergy, asthma & immunology research vol. 7,1 (2015): 14-21. doi:10.4168/aair.2015.7.1.14
12. Figueroa, Maria E et al. “Genome-wide epigenetic analysis delineates a biologically distinct immature acute leukemia with myeloid/T-lymphoid features.” Blood vol. 113,12 (2009): 2795-804. doi:10.1182/blood-2008-08-172387
13. Taki T, Shibuya N, Taniwaki M, Hanada R, Morishita K, Bessho F, Yanagisawa M, Hayashi Y. ABI-1, a human homolog to mouse Abl-interactor 1, fuses the MLL gene in acute myeloid leukemia with t(10;11)(p11.2;q23). Blood. 1998 Aug 15;92(4):1125-30. PMID: 9694699.