Prediction of Leukemia Subtypes from Gene-Expression Data: A Machine Learning Approach

Harihara Prakash Zubin Roy Wrootchit Mishra

Mellon College of Science
Carnegie Mellon University
Pittsburgh, PA

hprakash@andrew.cmu.edu, zroy@andrew.cmu.edu, wrootchm@andrew.cmu.edu

Abstract

Machine learning techniques have shown great promise in the field of medicine and healthcare in recent times. By utilizing models developed from computational techniques, machine learning allows new frontiers in healthcare by predicting disease onset well in advance. Through data obtained from biological experiments such as gene expression data, researchers may predict disease subtypes using machine learning approaches. While many machine learning methods for multi class prediction are available, the search for the most accurate multiclass classifier model continues. Our study utilizes three commonly used machine learning techniques, logistic regression, k-nearest neighbors and support vector machines, to identify the best model for multiclass leukemia prediction. We take help of gene expression data of Leukemia with 7 unique subtypes, and build a classifier model with these three techniques effective at predicting the subtype of a new set of expression data. We show that all three techniques are capable of acting as a suitable classifier with a near-satisfactory accuracy. Our results indicate that support vector machines produce the highest classification accuracy, outperforming k-nearest neighbours and logistic regression by a fair margin. In addition, we also highlight the Leukemia subtypes most likely to be mis-labeled by each model.

1 Introduction

Leukemia is the term used to highlight cancer of the blood cells. While it is most often seen stemming from white blood cells, there have been cases where it has risen from other blood cells as well. This type of cancer can prevent white blood cells from fighting infections and cause them to multiply uncontrollably, leading to health complications throughout the body. There are 4 primary types of leukemia including Acute Lymphoblastic Leukemia (ALL), Acute Myeloid Leukemia (AML), Chronic Lymphocytic Leukemia (CLL) and Chronic Myeloid Leukemia (CML). While the progression of the disease varies between each type, the severity of leukemia can be influenced by several factors, including the specific subtype, age of the patient, and stage of diagnosis. Naturally, each type of leukemia is accompanied by multiple subtypes for each, where each subtype is capable of causing distress to the body's immune system.

Predicting not only leukemia, but any type of cancer ahead of time is crucial because early detection is the key to successful treatment and improved outcomes. Many cancers are asymptomatic in their early stages, making them difficult to diagnose until they have already advanced. Naturally, there is a need to be able to detect cancer early, which will improve the chance of it being treated before it

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

has spread to other parts of the body. This in turn will significantly improve a patient's prognosis. In short, there is a need to predict cancer subtypes much before they manifest, to better treat a patient well in advance. With the advances in computational techniques and their integration with healthcare, machine learning can play a significant role in predicting cancer subtypes. Machine learning models can be trained on biological data in the form of either continuous data (Ex. Gene expression levels) or discrete data (Ex. Number of affected patients with a certain condition) to build classifiers capable of accurately classifying a new set of data into its corresponding cancer subtype. Machine learning can play a critical role in predicting leukemia by analyzing large amounts of gene expression data to identify patterns and predict risk factors for developing cancer. As such, our project aims to use popular multiclass machine learning models to predict leukemia subtypes based on gene expression data, identifying the model which works best in the process.

In our project, we implemented three multiclass machine learning models, inspired by previous work in Leukemia subtype classification [1] [2]. In a paper published by Xiao et al. in 2017 [3] , commonly used models such as k-Nearest Neighbours (kNN), Support Vector Machines (SVM), Decision Trees, Random Forests and Gradient Boosting were used to predict leukemia. The authors utilized an ensemble approach to incorporate all the said models to develop a multi-model based classifier. They were able to obtain an efficient and robust model with an average accuracy of nearly 98%. The first model we built was the Logistic Regression model. This supervised learning algorithm is commonly used for classification tasks. Logistic regression can handle high-dimensional datasets, and it is relatively easy to interpret the results. For the purpose of classifying multiple leukemia subtypes, logistic regression can handle multi-class classification problems using several approaches such as one-vs-rest (OvR), multinomial logistic regression, or softmax regression. Our second model was k-nearest neighbours, an unsupervised learning algorithm that can be used for clustering tasks. It finds clusters of data points that are similar to each other based on a selected distance metric (Ex. Euclidean Distance), while being dissimilar to the data points in other clusters. k-nearest neighbours can handle high-dimensional datasets, such as the one in question, and it is relatively fast and simple to implement. It is a good choice for datasets with various subtypes since it identifies clusters of data points that are similar to each other, which can be useful in identifying and predicting different subtypes of leukemia. Our final method is SVM which is a supervised learning algorithm that can be used for classification tasks. Like logistic regression, It is capable of handling high-dimensional datasets with relatively small sample sizes. SVM tries to find the best hyperplane that separates the data points into different classes while maximizing the margin between the hyperplane and the data points. The separation of classes may be carried out by either a linear, polynomial or even sigmoid kernel. In short, SVM tries to find the best boundary between the different classes that can generalize well to new, unseen data.

To test our models, we obtained a gene expression dataset from the Curated Microarray Database (CuMiDa)[4]. This database contains a wide array of gene expression datasets for multiple cancer types such as heart, brain, breast and more. In addition to containing datasets for each of these cancer types, the datasets have subtype information with each cancer type having nearly 5-7 subtypes. The database presents datasets with background correction as well as feature normalization carried out beforehand, and each dataset is carefully curated. Our objective was to utilize the data from the gene expression dataset for leukemia to train and test the machine learning classifier models we implemented in our project. On splitting the data into training and test sets, we would be enabled to both train our model and test it on "new" data. The testing would provide a brief overview of the performance of the models, further validated by various parameters such as recall, precision, F1-score and accuracy.

2 Data

The data obtained from the Curated Microarray database was a dataset for Leukemia gene expression. The dataset was created from gene expression experiments of Acute Lymphoid Leukemia, where the authors attempted to and successfully identified new markers to detect the cancer [5]. This dataset was well curated, with background correction and normalization already done, enabling us to skip any pre-processing steps that may have been necessary (However, feature selection may provide new insights to the model). The dataset contained 281 samples with nearly 22284 genes each. As such, it is a dataset with a fairly low sample to feature ratio. The dataset also contains 7 classes or subtypes of Leukemia, where each subtype represents a different form of B-Cell Acute Lymphoid Leukemia

(Table 1). Each gene is represented by a continuous data point indicating its expression level in the dataset. The class distribution in the dataset varies, with some classes possessing some more samples as opposed to other classes.

Table 1: Leukemia subtypes

Subtype name	Subtype index	Number of samples
B-CELL_ALL	1	74
B-CELL_ALL_TCF3-PBX1	2	22
B-CELL_ALL_HYPERDIP	3	51
B-CELL_ALL_HYPO	4	18
B-CELL_ALL_MLL	5	17
B-CELL_ALL_T-ALL	6	46
B-CELL_ALL_ETV6-RUNX1	7	53

3 Methods

3.1 Logistic Regression

For logistic regression , we developed a one vs rest logistic regression classifier to accurately classify the seven subtypes of leukemia comprising the high-dimensional dataset. A one vs rest logistic regression classifier is a binary classification method used when there are multiple classes in a dataset. In this method, the classifier creates a separate binary logistic regression model for each class, with one class considered as the positive class and the rest of the classes considered as the negative class. The output of each model represents the probability that the given sample belongs to the positive class. The class with the highest probability is then predicted as the final class for the given sample. The one vs rest logistic regression classifier addresses this problem by breaking down the classification task into several binary logistic regression problems, which are simpler and easier to solve. Furthermore, it allows for the modeling of the relationship between each class and the remaining classes, which can improve the accuracy of the classification. The classifier utilized a binary cross-entropy loss function and was implemented from scratch using only the NumPy library, while scikit-learn was used for the generation of performance metrics and 10-fold cross-validation.

The one vs rest logistic regression classifier utilized 50,000 epochs with a learning rate of 0.15 and a regularization strength of alpha=0.003 for L2 (ridge) regression. The data was split on a 75-25 split for training and testing respectively. Ridge regression was chosen to mitigate the effect of overfitting due to the high dimensionality of the dataset. The binary cross-entropy loss function was chosen because it is well-suited for binary classification problems and it penalizes incorrect predictions with a high loss value. Furthermore, the classifier was implemented from scratch using only the NumPy library to ensure maximum control over the implementation and to avoid any potential issues with pre-built models. Scikit learn was used only for the generation of performance metrics such as precision, recall, and F1-score and for the 10-fold cross-validation to assess the generalization of the model.

3.2 k-Nearest Neighbours

To implement the k-nearest neighbors model for building a Leukemia subtype classifier, we first chose to determine the best value of neighbors for which the model would give highest accuracy. Like logistic regression, we built the code using Python as our programming language of choice within a Jupyter notebook. Our general approach was to first build a kNN class, which would hold three functions. These functions are the initialisation function, responsible for setting the value of k, followed by the fit function, responsible for fitting the training data to their corresponding labels. The last and most important function within the class is the predict function which is responsible for predicting the labels of the new data from the test set. The function works by calculating the Euclidean distance of each point from the test data points, and identifying the k nearest neighbors from the point. The number of neighbors in each class is then identified, with the new data point assigned to the class with the highest number of neighbors. This model works with many different distance metrics, and our model utilizes the Euclidean distance between points.

Before testing the model, we trained it on training data by splitting the dataset into an 80-20 split of training vs test data. In addition to this, we also used a 10-fold cross validation approach to estimate the performance of the model across 10 folds. This was done with the intention of discovering the overall performance of the model based on the repeated training of the model. In our implementation, we first identified for what number of neighbors the highest test accuracy was obtained. Once this was determined, we proceeded to calculate the performance metrics of each of the 7 classes (subtypes) in the dataset for that value of k. We built functions from scratch to determine the accuracy, precision, recall and F1-score of all the classes. For our results, we calculated the performance metrics for each, and also built a confusion matrix for easier interpretation of the results.

3.3 Support Vector Machines

Our implementation of support vector machines was carried out with the open source scikit-learn package available for machine learning in Python. We used the in-built SVC() class to carry out the construction and testing of the model. While the scikit package provides inbuilt functions to carry out accuracy and performance metrics, we built our own functions to calculate these parameters in addition to using scikits. To train the model, we used an 80-20 split, where 80% of the original dataset was used to train the SVM model. The parameters set for our implementation were specific to the kernel type and the penalty parameter denoted by C. In our model, the kernel was set to linear, since we wanted to build a linear boundary between classes and test the subsequent accuracy of the model. Our penalty parameter, which indicates the minimization of training error to increase the margin was set to 1. The final parameter set within the class was the decision function parameter, Since we have multiple classes in the form of subtypes, we set the decision function parameter to a ovr (One-versus-rest) approach. We also implemented 10-fold cross validation for the model, where we utilised the in-built cross_val_score() class. We averaged the accuracy from each of the 10 folds and represented the average accuracy as our 10-fold cross validation accuracy.

4 Results

4.1 Logistic Regression

The results of the one vs rest logistic regression classifier for the leukemia subtype dataset with 7 subtypes are presented in Figure 1, Table 2. The results show that the classifier achieved an average accuracy of 83.67%, which indicates that the model can correctly classify most of the samples. The confusion matrix reveals the number of true positive and false positive predictions for each subtype. From the confusion matrix, we can see that the model is able to correctly classify most of the samples for each subtype. However, there are some false positives and false negatives that stand out. For example, in subtype 1 , there were 3 false positive predictions and especially for subtype 4 (B-cell_ALL_HYPO), which compared to the other classes has such a low number of correctly predicted cases compared to the incorrect ones. To evaluate the performance of the model for each subtype, we computed precision, recall and F1-score. Precision represents the ratio of true positive predictions to the total number of positive predictions, while recall represents the ratio of true positive predictions to the total number of positive samples. F1-score is the harmonic mean of precision and recall, and it gives us an overall measure of the model's performance.

Table 2: Performance metrics for logistic regression

Subtype name	Precision	Recall	F1-score
B-CELL_ALL	0.5357	1	0.6977
B-CELL_ALL_TCF3-PBX1	0.5455	1	0.7059
B-CELL_ALL_HYPERDIP	0.6364	1	0.7778
B-CELL_ALL_HYPO	0.2353	0.6667	0.3478
B-CELL_ALL_MLL	0.24	1	0.3871
B-CELL_ALL_T-ALL	0.9091	1	0.9524
B-CELL_ALL_ETV6-RUNX1	0.3333	1	0.50

The results show that the performance of the model varies for each subtype, with some subtypes having higher precision and recall values than others. For instance, subtype 3 and 6 have a precision of 63.64% and 90.91%, respectively, while subtype 4 has a precision of only 23.53% (Table 2). Except subtype 4, all subtypes have a perfect recall value of 100%, indicating that the model is able to correctly identify all positive samples for these subtypes.

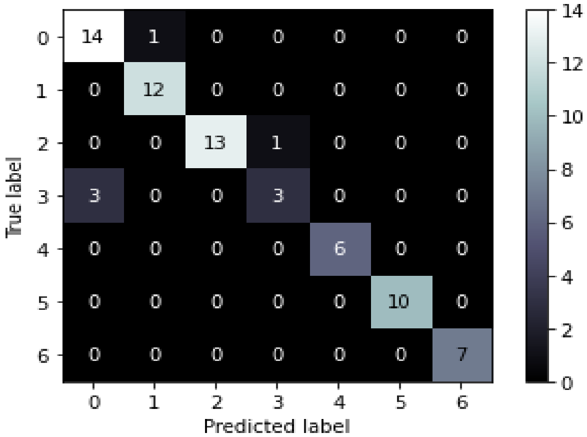

Figure 1: Confusion matrix for logistic regression. It is observed that subtype 3 (Actual subtype 4 via 1-based indexing) has been incorrectly predicted as subtype 0 (Actual subtype 1 via 1-based indexing) three times out of six total predictions for the said class. This implies a 50% accuracy in correctly classifying subtype 3 as subtype 0.

Overall, the results suggest that the one-vs-rest logistic regression classifier is effective in classifying leukemia subtypes based on gene expression data, achieving a high level of accuracy and good performance metrics for most subtypes. However, there is room for improvement, especially for subtypes with lower precision and recall values. Further optimization of the model and exploration of other machine learning algorithms may improve the performance of the classifier.

AUC score is the area under the ROC curve, which measures the model's ability to distinguish between positive and negative samples. We have plotted the ROC curve for said classifier and the same can be viewed in the supplementary .ipynb notebook file for logistic regression. In the present study, the one vs rest logistic regression classifier yielded promising results, with an average accuracy of 83.67%. These results further validate that the one vs rest logistic regression classifier is a viable method for accurately classifying the seven subtypes of leukemia in the high-dimensional dataset.

4.2 k-Nearest Neighbours

Our results indicate that the k-nearest nearest neighbors technique is a viable model to use as a Leukemia subtype classifier. On determining the best number of neighbors for this dataset to build a k nearest neighbors classifier, we identified that 7 nearest neighbors produces the highest accuracy of 85.96% (Figure 2). This indicates that the model can accurately classify new samples taking into consideration 7 of their nearest neighbors. As such, our model was selected to be centered around 7 nearest neighbors for estimating the validation parameters of precision, recall and F1-score. 10-fold cross validation of the model indicated an average accuracy of 82.87%, in line with the obtained accuracy of the model for 7 neighbors. We determined the precision, recall and F1-score for each of the 7 distinct subtypes of Leukemia, taking into consideration 7 nearest neighbors (Table 3).

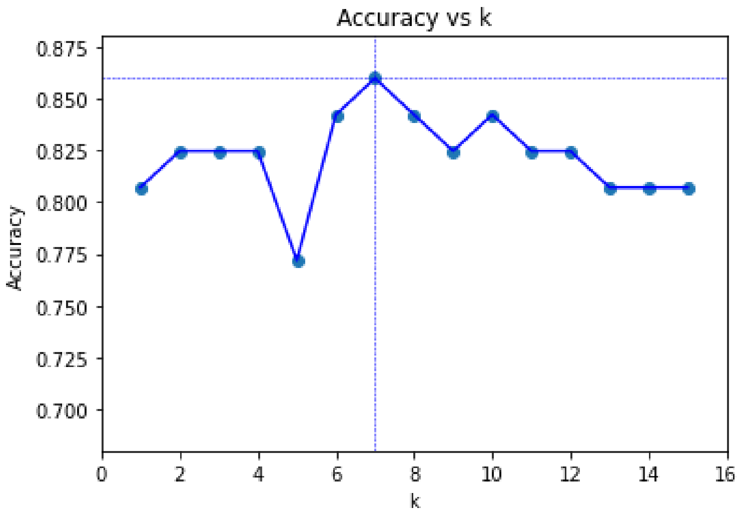

Figure 2: Accuracy vs number of nearest neighbors (k). It can be seen that the highest accuracy is observed for a k value of 7 nearest neighbours. For 7 nearest neighbours, the testing accuracy reaches 85.96%

We found that across all Leukemia subtypes, a high precision, recall as well as F1-score was observed. While varying across subtypes, nearly every performance metric was still greater than 75%, with the exception of subtype 4. Subtypes 2, 6 and 7 possessed a recall of 100%, indicating that perfect classification of samples into that subtype. Interestingly, as implied earlier, we observed a 0% precision, recall and subsequently F1-score for subtype 4. This implies that none of the test samples from the testing dataset were correctly classified as subtype 4 for 7 nearest neighbors.. From the confusion matrix, we can observe that a similar trend to the logistic regression results can be seen, where subtype 4 is most often misclassified as subtype 1. We predict that the reason for this misclassification arises from a distinct similarity between the subtype 4 and subtype 1 gene expression data.

Table 3: Performance metrics for k-nn (k=7)

Subtype name	Precision	Recall	F1-score
B-CELL_ALL	0.8	0.8	0.8
B-CELL_ALL_TCF3-PBX1	1	1	1
B-CELL_ALL_HYPERDIP	0.6923	0.9	0.7826
B-CELL_ALL_HYPO	0	0	0
B-CELL_ALL_MLL	1	0.75	0.8571
B-CELL_ALL_T-ALL	1	1	1
B-CELL_ALL_ETV6-RUNX1	0.9333	1	0.9655

The results of the k-nearest neighbors classifier for the leukemia subtype dataset with 7 subtypes are presented in Figure 2,3, Table 3, and suggest that the k-nearest neighbors model for 7 neighbors is a solid classifier capable of correctly classifying Leukemia subtypes given gene expression data. While the performance metrics are high across all subtype classifications, something which cannot be overlooked is that no instances of subtype 4 were correctly classified. This outlier may be a consequence of few samples, or having its gene expression data similar to subtype 1 gene expression data causing a difficulty in classification.

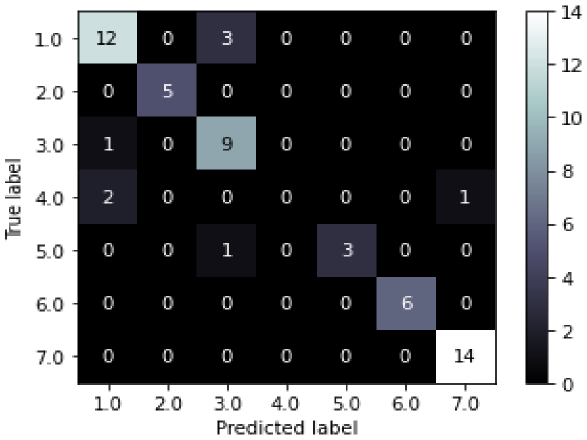

Figure 3: Confusion matrix for k-NN (k=7). Similar to Figure 1, we observe that subtype 4 (via 1-based indexing) is most often mislabelled as subtype 1 (via 1-based indexing), with no correct predictions in the kNN model for said subtype prediction.

4.3 Support Vector Machines

Our implementation for Support Vector Machines provided the highest accuracy across all three classifiers tested. This shows that it behaves as the best model for classifying Leukemia subtypes based on gene expression data. We observed that a linear kernel with a one-vs-rest approach provided an accuracy of 89.47%. This, coupled with a 10-fold cross validation accuracy of 88.60% indicated that the model built was consistent in correctly predicting the subtype of samples in the testing data. Again, following a trend from both previous machine learning models, we saw that the performance metrics of every subtype for SVM was above 80%, with the sole exception of subtype 4. Subtype 4 was seen to have a recall of only 20%, with an F1-score of only 33.34% (Table 4). This indicates that the models overall performance in correctly classifying subtype 4 is very poor.

Table 4: Performance metrics for SVM

Subtype name	Precision	Recall	F1-score
B-CELL_ALL	0.8235	0.875	0.8484
B-CELL_ALL_TCF3-PBX1	1	1	1
B-CELL_ALL_HYPERDIP	1	1	1
B-CELL_ALL_HYPO	0	0.2	0.334
B-CELL_ALL_MLL	1	1	1
B-CELL_ALL_T-ALL	0.8888	1	0.9411
B-CELL_ALL_ETV6-RUNX1	0.8182	1	0.9

This observation is further supported by the confusion matrix for the model (Figure 4), where it is observed that similar to logistic regression and k -nearest neighbors, subtype 4 is most often misclassified as subtype 1. This trend is observed throughout every model we implemented, proving that the gene expression data of subtype 4 is most likely very similar to subtype 1, and that the models were unable to determine the difference well enough to classify them separately.

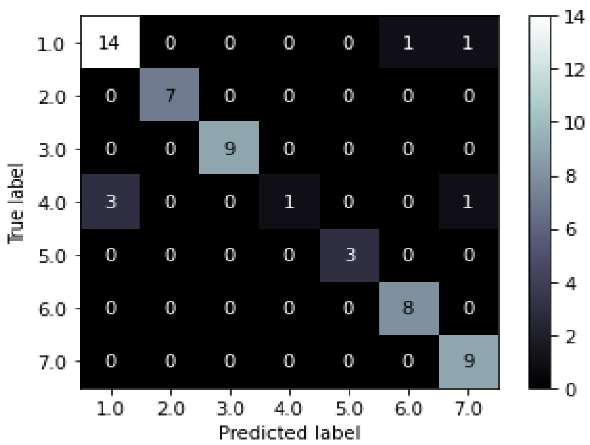

Figure 4: Confusion matrix for SVM. Again, similar to Figure 1 and Figure 3, we observe that subtype 4 (via 1-based indexing) is most often misclassified as subtype 1 (via 1-based indexing). Here, only one prediction of said class is correct, while the remaining are incorrect.

4.4 Comparative Analysis of the 3 models

Overall, every model we tested and trained shows a sufficiently good accuracy, with the scikit-implemented SVM model having the upper hand with an accuracy of 89.47% . This is in contrast to the models built from scratch, i.e, k-NN and logistic regression. We observed that across the three models, logistic regression produces the lowest model accuracy, but highest cross validation accuracy. While k-NN model test accuracy is relatively higher than the respective logistic regression accuracy value, its cross fold validation accuracy is the lowest at 82.87%. Across the three models, the SVM model can be seen the have the best accuracy in correctly classifying Leukemia subtypes.

On comparing the accuracy of 10-fold cross validation as well as test accuracy values of the three models we built, we noticed that the scikit-implemented SVM model possessed the highest accuracy value (Table 5). With a test accuracy of 89.47% accompanied with a 10-fold cross validation accuracy of 88.60%, we can confirm that the SVM model with a linear kernel is likely best equipped to handle multiclass Leukemia subtype prediction using gene expression data. Our implementations of logistic regression and k-NN from scratch give slightly lower test accuracy values of 83.67% and 85.96% respectively. While these test accuracy values are not as high as the SVM model, they still prove to be relatively suitable models to carry out subtype prediction of Leukemia.

Table 5: Accuracy of machine learning models

Accuracy type	Logistic Regression	k-NN (k=7)	SVM
Model Accuracy	83.67%	85.96%	89.47%
10-fold Cross Validation Accuracy	90.89%	82.87%	88.60%

5 Conclusions

In conclusion, we successfully implemented three popular machine learning models for multiclass prediction and classification tasks. Using the techniques of logistic regression, k-nearest neighbours and support vector machines, we were able to build three Leukemia subtype classifiers each with testing accuracy values of over 80%. 10-fold cross validation of the three models indicated a similar testing accuracy as that given by the model. We observed that SVM possessed the highest testing accuracy of 89.47%, with k-NN following with a testing accuracy of 85.96%. Of the three models, we observed that logistic regression produced the least testing accuracy of 83.67%. Universally, all three models show desirable testing accuracy values, with each model fairly equally equipped to carry out the classification of new Leukemia gene expression data into its corresponding subtype.

Interestingly, we observed that across all three models, subtype 4, which corresponds to the B-CELL_HYPO, is most commonly misclassified as subtype 1, which corresponds to B-CELL_ALL. The confusion matrix for all three models shows that the true label of subtype 4 is almost always misclassified as subtype 1. In the logistic regression model, we noticed that out of the total instances of subtype 4 classification, only 50% were correctly classified, with the remaining half classified as subtype 1. In the k-nn model however, we saw that no instances of subtype 4 were correctly classified at all, with most instances classified as subtype 1. This trend was observed to remain constant in all the models we built for the gene expression data which we used in our study. Our belief is that the gene expression data of the two subtypes is highly similar, causing an increases degree of misclassification between these subtypes. Naturally, it would be useful to visualize the data on a plot to identify the degree of similarity between the subtypes and make the necessary improvements for better classification.

We used performance metrics along with a confusion matrix to evaluate the performance of the three models. To further validate performance of the logistic regression classifier, we plotted the Receiver Operating Characteristic (ROC) curve. A high Area Under the Curve (AUC) score on the ROC curve indicates that the classifier has a high degree of separability between the positive and negative classes. We saw that the AUC values for all seven subtypes were also relatively high, with values ranging from 83.85% to 100%. Subtypes 1, 4, 5, and 6 exhibited an AUC value of 100%, indicating that the classifier was able to perfectly distinguish between these subtypes and the rest of the samples.

Thus, we have built three models which can act as multiclass predictors for Leukemia gene expression data. We believe that this work opens up endless possibilities in the field of cancer prediction. One possibility is to investigate and further understand the misclassification of subtype 4 as subtype 1 across all three models. This could involve exploring different techniques for feature selection, as well as visualizing the gene expression data in a more detailed manner. We could also explore other machine learning algorithms or even deep learning models for Leukemia subtype classification, as these may be able to capture more complex relationships within the data. Additionally, it may be beneficial to expand the dataset used in this study to include more samples or other types of cancer, which could help to further validate the efficacy of these models in predicting cancer subtypes. Our models may be used to predict not just Leukemia subtypes, but different forms of cancer with multiple subtypes. Given gene expression data, our models are able to predict subtypes of new data with a good accuracy, and can immensely benefit the medical field in the early detection of cancer.

References

[1] Vaibhavi Rupapara, Faizan Rustam, Wejdan Aljedaani, Hafiz Faheem Shahzad, Eunbyul Lee, and Imran Ashraf. Blood cancer prediction using leukemia microarray gene data and hybrid logistic vector trees model. *Scientific Reports*, 12(1):1000, 2022.

[2] Emrah Simsek, Hüseyin Badem, and Ibrahim Tolga Okumus. Leukemia sub-type classification by using machine learning techniques on gene expression. In *Proceedings of Sixth International Congress on Information and Communication Technology*. Springer, 2022.

[3] Yawen Xiao, Jun Wu, Zongli Lin, and Xiaodong Zhao. A deep learning-based multi-model ensemble method for cancer prediction. *Computer Methods and Programs in Biomedicine*, 152:113–122, 2017.

[4] Bruno C Feltes, Eloísa B Chandelier, Bruna I Grisci, and Márcio Dorn. Cumida: An extensively curated microarray database for benchmarking and testing of machine learning approaches in cancer research. *Journal of Computational Biology*, 26(3):223–233, 2019.

[5] Elaine Coustan-Smith, Guangchun Song, Charlotte Clark, Laura Key, Peng Liu, Maryam Mehrpooya, Patricia Stow, Xiaoping Su, Sheila Shurtleff, Ching-Hon Pui, et al. New markers for minimal residual disease detection in acute lymphoblastic leukemia. *Blood*, 117(23):6267–6276, 2011.