Machine Learning Bioscience
Early Term Project

Team Members

Kai Tong
Jayant Prakash
Aditya Vadhavkar

Features used in analysis

We divided all the available files in groups of 8 that belonged to each patient. For each patient, we computed the mean, standard deviation, height of the highest peak in histogram and x-coordinate corresponding to the highest peak in histogram for each of the columns (7 columns per file) present in the 8 files. Each peak in the histogram corresponds to a cell population and highest peak would be the major cell population. Height and location of the highest peak are good estimates of the proportion of cells and biological signatures of the cell population corresponding to that peak and we expect such properties to be different in normal and cancer patients. We derived 56 (mean) + 56 (standard deviation) + 56 (height of highest peak in the histogram) + 56 (x coordinate corresponding to highest peak in histogram) = 224 features.

Algorithms used

We experimented with Random Forest Classifier, Support Vector Machine, Gradient Boosting Classifier and Multi Layered Perceptron. We observed that Multi Layered Perceptron gave the best results in terms of F1 score. Thus, we decided to use Multi layered Perceptron and used stratified k fold cross validation with 5 splits for tuning the hyperparameters. We tuned activation_function, learning_rate, learning_rate_init and hidden layer sizes of the Multi Layered Perceptron with max_iterations set to 10,000. The scoring metric used for tuning the hyperparameters is area under the receiver operating characteristics curve.

Potential Mistakes

Out of the 180 samples that we predicted as either normal or aml we think that there might be 2 to 4 mistakes. According to the confidence values generated by our code, two samples that are predicted as aml have confidence values around 0.55 and 0.64 and two samples that are predicted as normal have a confidence value around 0.72 which we believe are somewhat low confidence scores and thus could be potential mistakes.

Confidence Values

We are training and testing our Multi Layered Perceptron on datasets with different number of aml and normal samples in training and testing sets. For getting different distributions of the number of aml and normal samples we are using 11 different random_states in the train_test_split function. For each of those 11 distributions we are selecting the model with best F1 score and using that model to predict the 180 samples and storing the predictions of each of those models. Then, from those 11 predictions we are creating the final prediction by voting. For each of the predictions, we count the number of models that predicted it to be as aml and the number of models that predicted it to be as normal, whichever has higher votes is our final prediction and the count of higher votes divided by the total number of votes is the confidence value for that particular prediction.