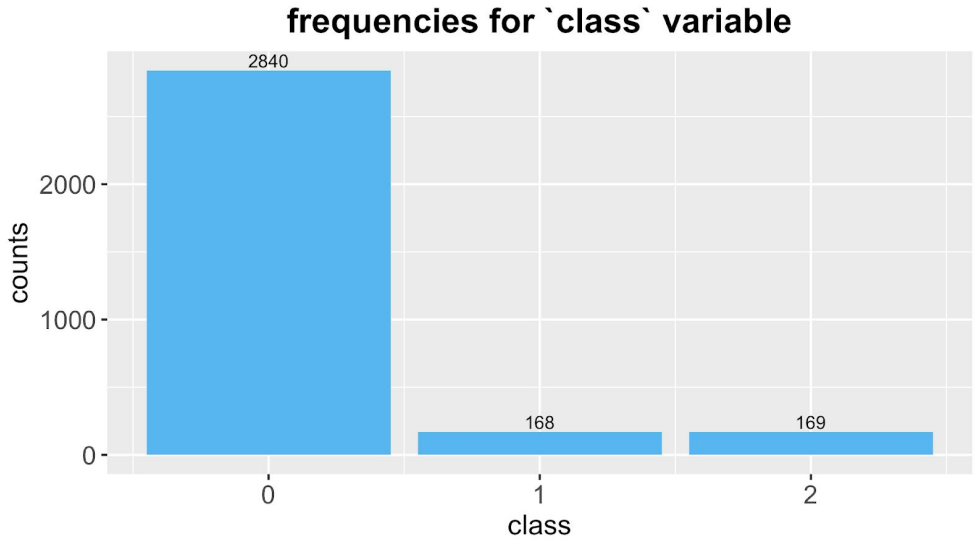
Classification of Cancer Causing Genes

Eustina Kim, Kevin Chen, Tiffany Feng, Jonathan Martinez

Introduction

In this project, we aimed to create a statistical learning model using a dataset of 3177 gene observations and 97 gene related predictors to predict cancer driver genes—Oncogenes (OGs) and Tumor Suppressor Genes (TSGs)—in a test data set of 1363 gene observations. The genes are encoded as the following in the dataset: neutral gene (NG) as 0, OG as 1, and TSG as 2. We used statistical learning methods discussed in chapters 1-5 in "*An Introduction to Statistical Learning with Applications in R"* to create our models.

For our exploratory data analysis, we graphed the frequencies of the categories within the `class` variable. It is apparent that there is class imbalance in that there are very few 1s and 2s compared to 0s. We discuss how we dealt with this issue in the methodology section.



Methodology

a. Preprocessing our data

First, we made sure to remove the ID column from our predictors, since ID is an identifier and is not related to `class`. We then combined variables that may be highly correlated into one variable by standardizing them and summing them up. For instance, we standardized variables `Broad_H3K4me2_percentage`, `H3K4me2_height`,

`H3K4me2_width` and then added the standardized values across rows into one column with the name “H3K4me2”. We repeated this procedure for other predictors that had a similar name pattern and were likely to be very correlated with one another. Through this process we created combined variables like `H3K4me3`, `H3K4me1`, etc. This technique ensures we are addressing the issue of multicollinearity by combining variables that may be highly correlated with one another.

To further narrow down the number of predictors, we constructed an initial linear model with `class` as our response and all of our remaining variables as our predictors. Using the summary output from this model, we then chose the predictors with a p-value less than 0.1. We opted for a more generous p-value cutoff that is greater than the 0.05 standard in order to include more predictors in our model and to prevent underfitting.

b. Statistical Model

For our model, we opted to use the multinomial logistic model, which applies logistic regression when there are more than 2 classes to predict. This was the case for our project, as our response variable `class` could have 3 different values: 0, 1, or 2. We used k-fold cross validation with k = 10 to compare model performance, choosing the model with the highest cross validation accuracy score as our final model to predict on test data. Next, we changed the probability threshold for predicting class ‘0’ to the mean of the vector of probabilities of predicting ‘0’ when we applied the final logistic model on the test data,which was 0.897209. It was necessary to make the threshold for predicting ‘0’ higher because around 90% of the training data is class of ‘0’. Therefore, we knew that the predicted probabilities for class ‘0’ would be a lot higher than 0.5 and increasing the threshold would be one way to combat the imbalance of categories in our data.

Results

Using our model, we obtained a Kaggle score of 0.82606 We trained the given data using KNN, Multinomial logistic, LDA but chose to use multinomial logistic regression for our final model due to the method having the highest cross validation accuracy of 0.9527816. We did not use QDA as a model because we encountered a rank deficiency error. After researching online, it seems like having too little data--especially on classes ‘1’ and ‘2’--may be the reason for the error.

General Conclusions

For this classification project, we combined our knowledge of data processing and statistical models to create cancer-gene predictions. There were two key steps for our data processing part. We found out that the model performance increased when we combined correlated variables into a single predictor to eliminate multicollinearity. We also chose predictors that are statistically significant but not underfit by setting the p value to be 0.1. Multinomial logistics regression worked the best among our models. We further increased its performance by tuning the probability thresholds. It's also important to note that we compared model performances using 10-fold cross validation instead of the ROC-AUC curves. The reason is because we had little data to work with, especially of classes 1s or 2s in comparison to the 0s. If we could obtain more training data, we could also split it into training and validation sets and use ROC-AUC curves to check model performances.

Our model does have space for improvements. We could change the probability threshold for each class and possibly get better predictions. We could also remove the outliers for each predictor. To deal with the unequal numbers of 0s, 1s, and 2s in the class variable, we could try methods like SMOTE algorithm, penalized model, and decision trees. However, as our model achieved a score of 0.82606, we believe it is a useful model for predicting cancer genes.