BOOSTING TOBI PERFORMANCE
Amanda Zong
Summer 2018

Motivation

The motivation of this project was to investigate and improve the accuracy and robustness of TOBI through machine learning and added features.

Background

The Tumor-Only Boosting Identification model (TOBI) was first developed by Chioma Madubata in the Rabadan lab to classify mutations in the DNA of cancer patients as somatic or non-somatic. The motivation for this classification is that oncogenic mutations occur in somatic cells, and so the identification of somatic mutations are targets for new gene-targeting cancer treatments. While TOBI performed extremely well on certain datasets such as bladder urothelial carcinoma (BLCA), lung adenocarcinoma (LUAD), skin cutaneous melanoma (SKCM), and stomach adenocarcinoma (STAD), it performed less well on datasets such as pediatric TALL (pediatric T-cell acute lymphoblastic leukemia). This poor performance is most likely due to the most severe imbalance between the somatic and non-somatic classes, with a ratio of 1:99. The goal of this project was to investigate and improve the performance of TOBI through machine learning and added features.

Methods

The original published model was first analyzed on the TALL (pediatric) dataset, as this dataset was readily available. The correlation matrix of features and the histogram distributions of feature values between the two classes was computed and analyzed. To improve the performance of the machine learning model, the majority class was undersampled to match the size of the minority class and then random forest and logistic regression were implemented. The default settings of these machine learning packages in R were used.

Gender and tumor stage were incorporated as new features in the datasets provided by TCGA (LUAD, SKCM, STAD). Information on gender for each patient case was queried using the TCGA API. Information on tumor stage was obtained by querying and downloading the XML file corresponding to each patient file and extracting tumor stage from the XML file. For most of the datasets, the tumor stage and substage were both given. In order to decrease the number of categories for this feature and increase the number of examples per category, the substage information was omitted. For example, "Stage IIIa" and "Stage IIIb" would be treated equally as "Stage III." For each dataset, after these features were incorporated with the original set of features, the majority class was undersampled to match the size of the minority class and then random forest was implemented. When including the new features, observations that did not contain these features were thrown away. The performance of the model with and without the

new features was evaluated, as well as the contribution of each feature to increasing node purity at each split of random forest.

When running the model, the training size was selected to be 160 for the TALL dataset, indicating that 160 unique patient cases were to be chosen. The training size was kept at 100 for all of the other datasets. The reason for the higher training size for TALL was to confront the greater non-somatic/somatic imbalance problem in the dataset, which causes undersampling to significantly decrease the size of the final training set.

Results and Discussion

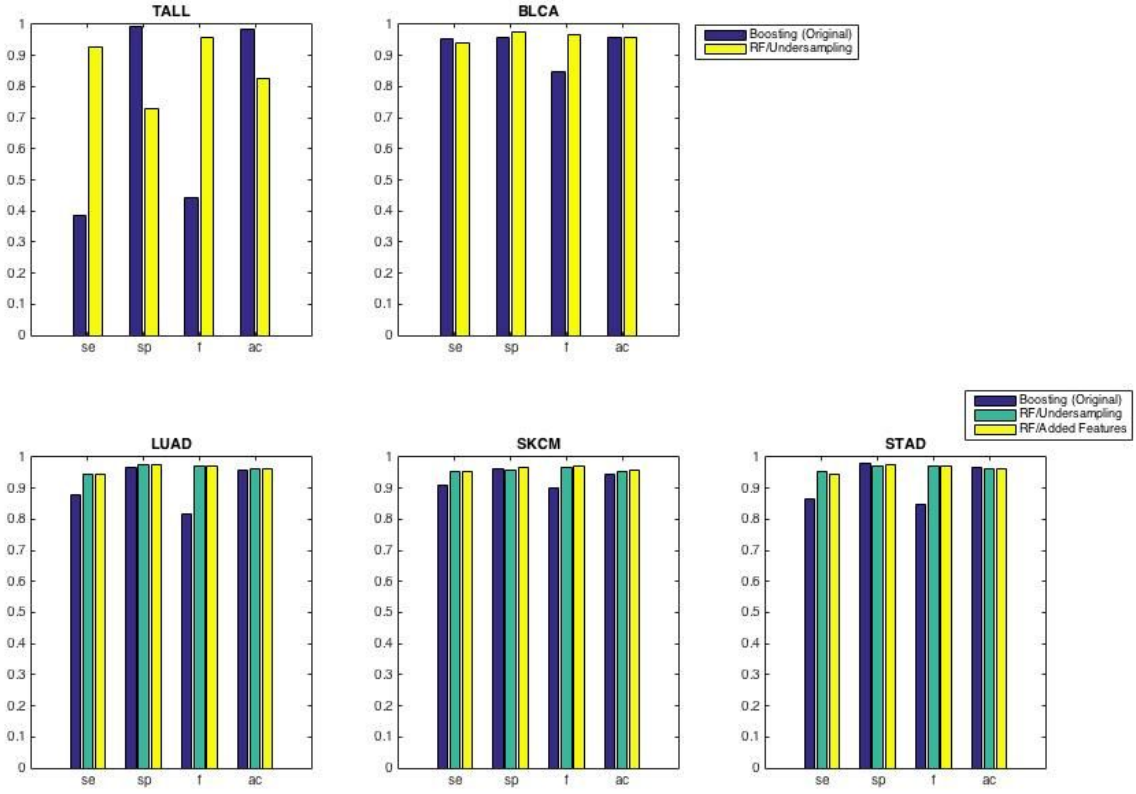

Figure 1: TOBI Performance Results
Each plot shows TOBI performance results for a unique dataset. The categories on the x axis denote standards of performance: sensitivity (se), specificity (sp), f-score (f), and accuracy (ac). The different colored bars correspond to different versions of the model: boosting (original), random forest with undersampling (RF/Undersampling), and random forest with added features (RF/Added Features).

By implementing undersampling and random forest, the f-score of the classification model for the TALL dataset increased more than two-fold from the original boosting model, from .44 to

.96. The sensitivity, a measure of the model's ability to accurately identify somatic mutations, dramatically increased from .39 to .93. This significant increase in sensitivity is coupled with a slight decrease in specificity, from .99 to .73.

When analyzing the features of TALL, it was found that num_mut_gene (total variants per gene in a dataset) and case_mut_gene (total variants per gene in one sample) have a high correlation of .95. Since num_mut_gene is ranked as a more important feature in random forest, it is likely to reduce dependencies in the model by removing case_mut_gene. Furthermore, dbNSFP_CADD_phred and effect_impact have a high correlation of .64, which is explained by the fact that both are scores of variant deleteriousness, and may have been computed from similar metrics.

Undersampling also slightly improved the performance of the model on the other datasets, increasing the f-score by .1 on average and increasing specificity while maintaining or increasing sensitivity. Since class imbalance is not as severe of an problem in the other datasets compared to TALL, undersampling only had a small positive effect on the performance of the model.

The addition of gender and tumor stage as features did not seem to affect the performance of the model, as the measures of model performance stayed relatively the same for all of the datasets. Tumor stage showed some contribution to increasing node purity in random forest, while gender had little contribution to increasing node purity.

Supplementary Graphs/Data
https://docs.google.com/document/d/1LMVVD8U082knW0MFqM6o1cfPPjZwSmhZ-kgEdzhZXe8/edit

Future Work

To further improve the performance of the model on the TALL dataset, it is necessary to extract new features from the original data to gain more information about the differences between the somatic and non-somatic classes. Unfortunately, the original pre-processed dataset could not have been obtained and annotated during the time span of this summer, so any improvements to the performance of TOBI on the TALL dataset was limited to machine learning enhancements such as undersampling.

It may be useful to add tumor stage as a feature for other cancers marked by more pronounced tumor stages, or to include tumor substages rather than the overall tumor stage, to see if they would enhance the performance of TOBI. Furthermore, other patient information such as age could be incorporated as well.

Finally, while the default settings for random forest in R were used, it may improve model performance to fine-tune algorithm parameters through systematic grid search.

Code

NAME: ML_amanda_zong_061418.R
DESCRIPTION: A modified version of machine_learning.bam_input.ROC.R (written by Chioma Madubata) that includes gender and tumor stage as added features and implements random forest instead of boosting. The other features created by Chioma are kept. As in the original script, it calls on helper functions in the supplementary script ml_helpers.R.
TO RUN: There are two settings: the first allows the script to be run in console and debug by setting the parameters in the script; the second allows the script to be run in terminal.
Example of command for running script in terminal on the TCGA-SKCM dataset: Rscript ML_amanda_zong_061418.R TCGA-SKCM-added.xlsx out/ som.180501.626_ALL_VAF_15.8_Foa.57_CUMC.20_TARGET _rel.539_TARGET_WES.txt test 100 TOBI/

NAME: ml_helpers.R
DESCRIPTION: A modified version of ml_helpers.R (written by Chioma Madubata), storing helpful supplementary functions to the main ML script. The following functions were added:
analyze_distribution: Creates and saves a text file called trainTestDistribution.txt that stores information on the class distribution in the training and test sets.
format_features: Formats the added features gender and tumor_stage, and calls on remove_tumor_substage.
remove_tumor_substage: Omits tumor substage information, keeping the most general stage information. For example, "Stage IIIa" and "Stage IIIb" would both be modified to "Stage III."
undersample: Undersamples the majority class to match the size of the minority class.

NAME: analysis_amanda_zong_061218.R
DESCRIPTION: Computes the correlation matrix of a feature set.
TO RUN: Manually change the dataset name, then run in console.

NAME: getPatientInfo.py
DESCRIPTION: Get gender/race information for a dataset by querying the TCGA API and creates a new Excel sheet named "[dataset name]-patient-data.xlsx" where the first column contains the case ID, the second column contains the submitter ID, and the third column contains the corresponding data. For each feature, a new sheet will be added.
TO RUN: Run in terminal with the TCGA category as the first input parameter, then the different responses expected.
Example: getPatientInfo.py cases.demographic.gender male female

NAME: downloadxmldata.py
DESCRIPTION: Obtains tumor stage information for a dataset by querying the clinical XML files for each patient file and saving the extracted information to a new sheet in "[dataset name]-patient-data.xlsx." If no information found, the error code is "no data."

TO RUN: Run in terminal with the dataset name as the first input parameter.
Example: downloadxmldata.py TCGA-SKCM

NAME: addPatientInfoColumnsDictionary.py
DESCRIPTION: Matches the patient information stored in "[dataset name]-patient-data.xlsx" with the raw downloaded data in "[dataset name]-raw.xlsx" and creates a new Excel sheet containing all of the information called "[dataset name]-added.xlsx."
TO RUN: Run in terminal with the dataset name as the first input parameter.