Machine Learning Manuscript Outline

Introduction

General Context of the work

Advances in sequencing technology and decreasing costs of generating 16S rRNA gene sequences allowed rapid exploration and taxonomic characterization of the human associated microbiome. Currently, the microbiome field is growing at an unprecedented rate and as a result, there is an ever-increasing demand for reproducible methods for identifying associations between members of the microbiome and human health. Human associated microbial communities are remarkably uneven and it is unlikely that a single species can explain a disease state comprehensively. It is more likely that subsets of those communities, in relation to one another and to their host, account for the unevenness and can be used as biomarkers of a specific disease state. Thus, researchers have started to explore the utility of machine learning (ML) techniques to identify the biomarkers associated with healthy and diseased individuals. However, currently the field's use of ML lacks clarity and consistency on which methods are used, how these methods are implemented and if these methods are reproducible. Moreover, there is a lack of consideration on why a particular method is utilized. Often, there is a trade-off between the interpretibility and accuracy of ML methods. As researchers, we have to find the right balance to accomplish complex knowledge tasks such as identifying a disease state based on microbiome-associated biomarkers while also understanding how these tasks are accomplished.

Narrower research area and statement of its importance

One application of machine learning to microbiome data has been to classify patients as having colorectal tumors based on microbiota-associated biomarkers. However, the efforts to combine ML methods with relative abundances of bacterial poplations in stool to predict CRC tumors have been negatively effected by the aforementioned flawed ML practices. As a result, the predictive performance of these models varies greatly with areas under the receiver operating characteristic curve (AUROC) of 0.7-0.9.

- What we see in previous studies:
  - Differences in task definition (what is it that we want to predict?)
  - Use of random forest only without discussion of interpretibility.
  - Use of random forest without proper pipeline.
- What we do here:
  - To shed light on how much differences in modeling can affect the results, we performed an empirical analysis comparing several different modeling pipelines.

Summary of appraoch and findings

- Modeling pipelines were established for L2-regularized logistic regression, L1 and L2 support vector machines (SVM) with linear and radial basis function kernels, a decision tree, random forest and XGBoost.

- These methods increase in complexity while decrease in interpretibility.

- We established ML pipeline with held-out test data and performed 100 data-splits to evaluate generalization and predicton performance of the ML method.

- Applied to held-out test data, the mean AUROC varied from 0.68 (std ± 0.04) to 0.82 (std ± 0.04).

- Random Forest had the highest mean AUROC for detecting SRN and was less susceptible to overfitting compared to other methods.

- Despite the lower mean AUROC value, the L1-regularized linear kernel SVM offered the greatest interpretability and stability.

- In terms of computational efficiency, x trained the fastest, while y took the longest.

- We found that cross-validation and testing AUROC could vary by as much as 0.06, highlighting the importance of a separate held-out test set for evaluation.

- Aside from evaluating generalization and classification performance for each of these models, this study established standards for modeling pipelines of microbiome-associated machine learning models.

Results

AUROC results of 7 modeling approaches. (FIT + OTUs)

Comparisons among the 7 modeling approaches. (FIT + OTUs)

- Compare prediction performance, generalization performance and susceptibility to overfitting.

Hyper-parameter tuning budgets and corresponding AUROC values.

- This will show that we have used the right budgets to allow the model to pick the right hyper-parameter.

AUC results of models with just (FIT)

- This could be in supplemental. We want to make sure that using just FIT as a feature does worse than FIT+OTUs.

Discussion

Interpretation of modeling results in terms of reproducibility, robustness, actionability, interpretibility and susceptibility

- What are the metrics to talk about these concepts?

- Emphasize trade-off results we observe in this study.

Consideration of possible weaknesses for each model

- The interactions between the biomarkers may be nonlinear. Obviously, the linear models will not incorporate this because they are linear. Tools like linear models (e.g. metastats, lefse, wilcoxon, etc) are likely worthless. How many OTUs would you find with these methods?

Consideration of possible weaknesses for our approach and chosen dataset

- Why did we use only one dataset? Becuase we are comparing several methods on 1 dataset.

Relationship of results to previous literature and broader implications of this work

Prospects of future progress

Methods

Brief explanation of study design/patient sampling and 16S rRNA gene sequencing/curation

- Baxter et al, 2016

Analysis of data

- What are the features and what are the labels?

  1. Features: Fecal hemoglobin concentration and 16S rRNA gene sequences from stool samples

2. Labels: 490 patients as having advanced tumors (advanced adenoma or carinoma) or not (non-advanced adenoma or normal colon).

- What is the data (temporal or not)? Assumptions we make when we use a dataset. What will the future data look like?
- Pre-proccessing of the data
- Machine Learning pipeline backbone. How do I split, train, validate and test?
  - A diagram to explain the modeling pipeline.
- Which methods are linear/non-linear? Talk about interpretibility.
- Cross-validation and hyper-parameter tuning methods for each modeling method
- Programming languages and packages/modules utilized
- Statistical methods for comparison of model performance
- Code/data availability