TITLE

Running title: INSERT RUNNING TITLE HERE

Courtney R. Armour[1], William L. Close[1,*], Begüm D. Topçuoğlu[1,#], Patrick D. Schloss [1,†]

[1] Department of Microbiology and Immunology, University of Michigan, Ann Arbor MI.

[#] Current Affiliation: Bristol Myers Squibb, Summit, New Jersey, USA

[*] Current Affiliation:

† To whom correspondence should be addressed: pschloss@umich.edu

observation format (max 1200 words, 2 figures, 25 ref)

Abstract (250 word max)

Importance (150 word max)

Introduction (~250 words)

- Gut microbiome community composition has proven useful as a resource for machine learning prediction of various diseases {examples}.
- Amplicon sequencing of the 16S rRNA gene is a reliable tool for assessing the taxonomic composition of microbial communities.
- Analysis of 16S rRNA sequence data generally relies on clustering of sequences based on similarity into operational taxonomic units (OTUs).
- However OTU clustering depends on the data in the dataset and the addition of new data may change the overall OTU clusters.
- The unstable nature of OTU clustering complicates machine learning. When building a classification model, changes to the OTU clusters means you have to re-create the model. This can change the underlying model and be time consuming and resource intensive.
- The ability to integrate new data into an existing model could allow for deployment of a single model that new data can be continually added to and predicted on.
- Recently Sovacool *et al* described a new method for fitting new data into existing OTU clusters {Kelly optifit 2022}.
- While OptiFit works well to fit new sequence data and provide high quality OTU clusters, it is unknown if the use of OptiFit will have an impact on machine learning predictions.
- Here, we use OptiFit with a 16S rRNA sequence dataset consisting of normal and SRN samples to test how well new data integrated with OptiFit performs for prediction of SRN.

Results (~700 words)

- Utilize public dataset with normal and SRN samples to compare prediction between OptiFit and OptiClust
- randomly split data into 80% training 20% test sets 100 times
- Processed the data with both algorithms - Figure 1
    - 1. Used traditional OptiClust method to cluster all data, then split into training and test

set

  - 2. Used OptiClust on the 80% training set, then used OptiFit to fit the remaining 20%

- OptiFit produces similar quality OTU clusters based on MCC (supplement?)
- Used mikropml to train a model on the 80% training set for each data split, then predicted dx on the 20% test set
- Training performance almost identical (OptiClust median CV AUC 0.694, OptiFit median CV AUC 0.693) - Figure 2
- Performance on the test set comparable (OptiClust median CV AUC 0.694, OptiFit median CV AUC 0.693) - Figure 2
- Maybe expand on where optifit did better/worse?

Discussion/Conclusions (~250 words)

- OptiFit works!
- future questions:
  - how much reference do you need?
  - does it work well for other situations?
    - what if we used a new dataset for test set instead of a subset of the full dataset?
    - other diseases?

Materials and Methods

- 16S rRNA amplicon sequence data from 490 subjects {baxter}
  - 261 controls
  - 229 SRN
- created 100 random splits of the data (80% training, 20% test)
- preprocessed data with mothur v1.45

- two pathways

  -opticlust:

  - clustered all data together
  - split the shared file based on the random splits

  -optifit:

  - split the data
  - opticlust on the 80% training set
  - optifit to fit the remaining 20% to the training set OTUs

- ML with mikropml package (version)

  - preprocessed training set and applied preprocessing to test set

    * correlated collapsed, removed nzv

Acknowledgements

- funding

Figures

Figure 1. Workflow description.

Figure 2. Model Performance. A) Mean AUC **B)** Averaged ROC curves

References