Autistic Spectrum Disorder Prediction Based on Human Gut Microbiome

Chris Chen

1. Abstract

Autism Spectrum Disorder (ASD), characterized by persistent deficit in socialization ability (communication and interaction) and repetitive and restricted patterns of behavior, interest, or activities, is considered a severe neurodevelopment disorder that occurr in early childhood development period. In this project, a dataset of a sample of ASD and healthy children that includes their gut microbiome information (encoded as OTUs) was analyzed with machine learning. The goal of the analysis is to 1. predict the subject's ASD outcome based on his/her OTU information; 2. identify the OTU(s) that contribute the most to the outcome; and 3. compare the models constructed and find out the one with the best performance and generalization potential. In the end, we managed to build models that can predict the ASD outcome with 96% accuracy on the test data, and were able to identify the most important few OTUs using traditional machine learning methods including Random Forest, Gradient Boost, XGBoost, and SVM. The model that performed and generalized the best was SVM.

2. Introduction

Autism Spectrum Disorder is a severe neurodevelopmental disorder that typically diagnosed on kids. According to DSM-5, there are 5 criteria for diagnosis:

1. Persistent deficit in socialization ability (communication and interaction).
2. Repetitive and restricted patterns of behavior, interest, or activities.
3. Symptoms occurred in the early developmental stage, but might only become profound until the social demand exceed limited capacity in later stages of life.

4. Severely impairs functioning in social, occupational, or other areas in life.

5. The decifict cannot be attributed to intellectual disability or global developmental delay.

Previous research have shown that ASD outcome can be predicted using human gut microbiome for that very useful information can be extracted from the feces of human. Information is encoded in Operational Taxonomic Units (OTUs), which can be interpreted as “abundance of each microbiota”. Dan et. al managed to construct a prediction method based on OTUs that achieved an accuracy of 91%.

Due to its large impact on well-being of kids and the society’s wellfare system as a whole, we believe it’s important to develop more accurate prediction methods and more efficient treatment or intervention. Therefore, the goals of this project are

1. Predict the subject’s ASD outcome based on his/her OTU information, since accurate prediction could potentially save a lot of medical resources. We aim to push the accuracy to exceed 91%.

2. Identify the OTU(s) contribute the most to the outcome, for that they could bring insight into future ASD treatment or intervention.

3. Compare the different machine learning methods we used and find out the one with the best performance and generalization potential.

3. Data Set Description

3.1 Data Collection

This is a dataset obtained from Kaggle and is about human gut microbiome information of healthy vs Autistic Spectrum Disorder (ASD) patients. We have $n = 254$, and is consisted of 143 ASD subjects and 111 healthy subjects. Response is ASD or not (artificially constructed), encoded as 0 (no) or 1 (yes); covariates are Operational Taxonomic Unit (OTU) abundance, encoded numerically. The OTU information is obtained using High Throughput Sequencing. There are 1322 OTUs in total, and the corresponding taxa to the OTUs are also provided.

All the subjects are Chinese children aged from 2 to 13, recruited from May 2016 to Aug 2017 from hospitals and kindergartens in Southern China. The children with ASD in this study were diagnosed according to DSM-5. Feces were collected at hospital or home according to instruction and delivered immediately (overnight) at low temperatures (using dry ice) to Nanjing Medical University and were analyzed there.

3.2 Descriptive Statistics

Univariate descriptive statistics of only 9 of the OTUs are presented below (See Figure 1) due to two reasons: 1. the distributions of the OTUs are similar in ditribution shapes; and 2. there are 1322 OTUs, it's impossible to include the descriptive statistics for all of them. We can see that all the OTUs have different value ranges but all of them have values greater than or equal to 0. Comparing the medians and maximum gives us the information that all these distributions are right skewed.

3.3 Visualizations

To better visualize these "microscopic" distributions, the histograms of these OTUs are provided in Figure 2. Figure 3 presents the "macroscopic" view of the data–the distributions of the means, standard deviations, medians, and maximum.

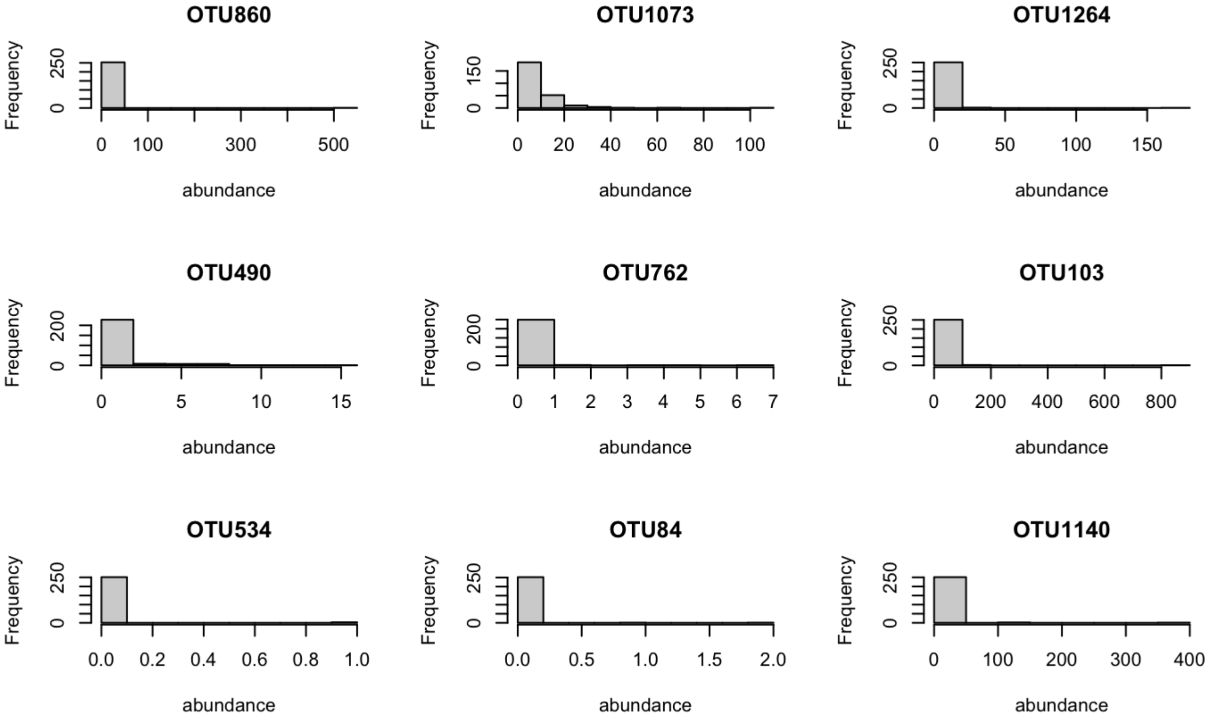
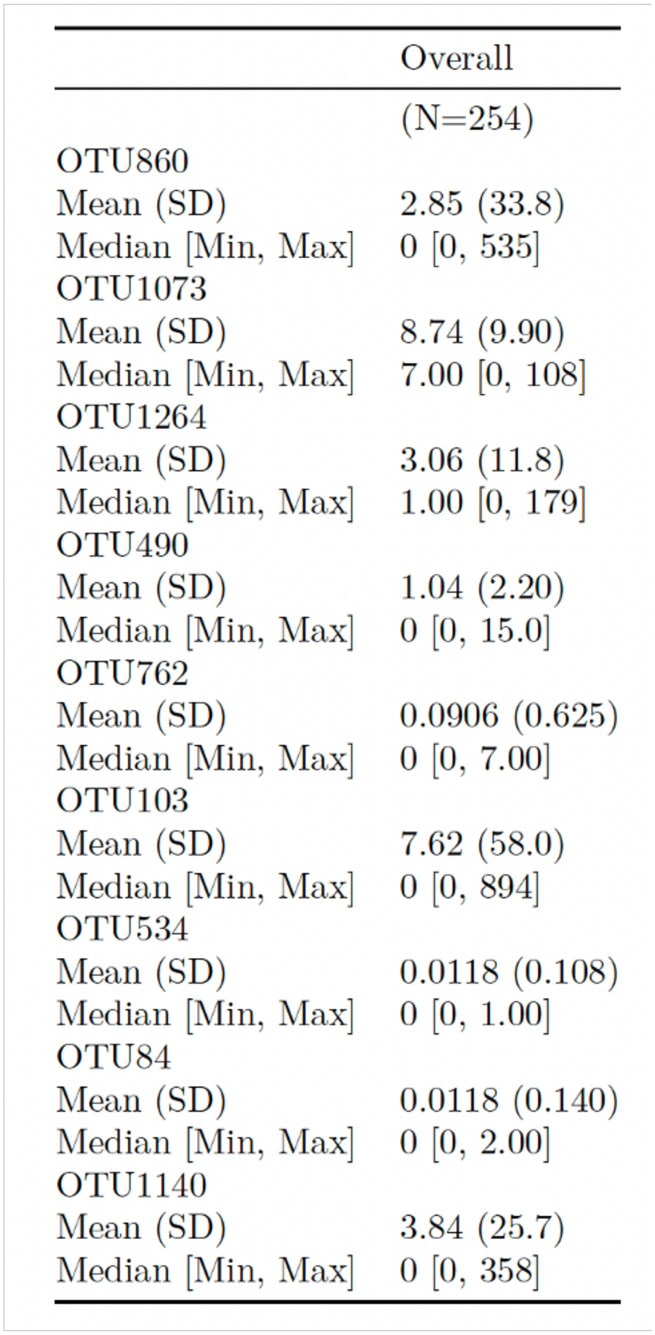


Figure 2: Distributions of the Selected OTUs



	Overall
	(N=254)
OTU860	
Mean (SD)	2.85 (33.8)
Median [Min, Max]	0 [0, 535]
OTU1073	
Mean (SD)	8.74 (9.90)
Median [Min, Max]	7.00 [0, 108]
OTU1264	
Mean (SD)	3.06 (11.8)
Median [Min, Max]	1.00 [0, 179]
OTU490	
Mean (SD)	1.04 (2.20)
Median [Min, Max]	0 [0, 15.0]
OTU762	
Mean (SD)	0.0906 (0.625)
Median [Min, Max]	0 [0, 7.00]
OTU103	
Mean (SD)	7.62 (58.0)
Median [Min, Max]	0 [0, 894]
OTU534	
Mean (SD)	0.0118 (0.108)
Median [Min, Max]	0 [0, 1.00]
OTU84	
Mean (SD)	0.0118 (0.140)
Median [Min, Max]	0 [0, 2.00]
OTU1140	
Mean (SD)	3.84 (25.7)
Median [Min, Max]	0 [0, 358]

Figure 1: Descriptive Statistics for 9 of the OTUs

4. Statistical Methods

4.1 Stage I: Model Fitting and Parameter Tuning

Since we don't know anything about the distribution of the data, it's not preferable to use a parametric model. Hence, we propose $Y = f(X) + \epsilon$, a nonparametric model. We additionally assume that $E[\epsilon|X] = 0$.

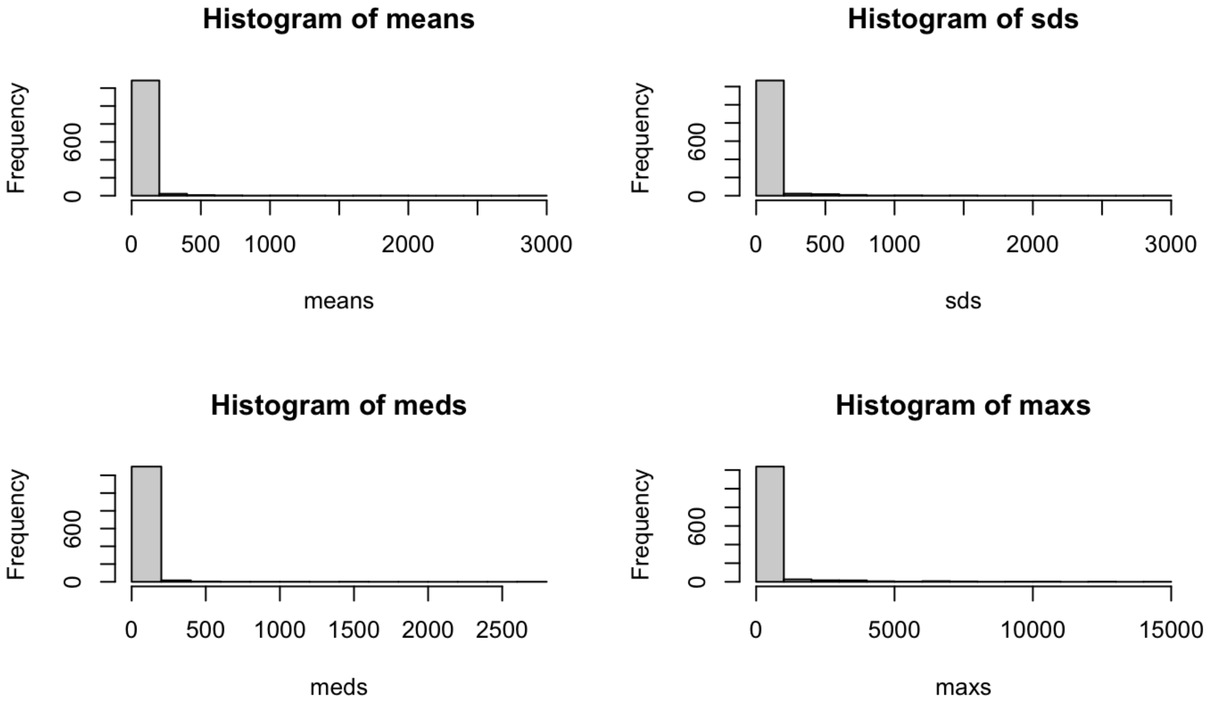


Figure 3: Distributions of the Means, Standard Deviations, Medians, and Maxs

Random Forest, Gradient Boost, Xtreme Gradient Boost, and Support Vector Machine were implemented for the classification task since they are nonparametric methods that work well with smaller datasets and hence are the most widely used ML methods to do disease prediction.

While fitting these models, we took a 80-20 train test split CV approach. Feature re-scaling was done and only done to SVM to improve its performance due to the nature of this algorithm. SVM is a distance-based algorithm–it classifies the data with a hyperplane that cut through the data space at a position that best separate the data into two regions according to their labels. The best position is defined in terms of maximization of the distance between the data to this hyperplane. Therefore, different scales of the data features can seriously mess up the classification task. On the other hand, this is not a problem for tree-based algorithms, because each decision boundary in each decision tree is determined in each feature's own scale.

After each model is trained, parameter was tuned using grid search on a few selected parameters. It was no way near exhaustive due to the limitation imposed by the computation power of my device. For tree-based algorithms, the parameter tuned was number of trees; and for SVM, the parameter tuned was cost. Then, test error rate, precision, recall, and F-1 score were calculated and used to evaluate and compare the models.

Since in our dataset the number of features (1322) largely exceeds the number of observations (254), it's extremely easy and likely to overfit the models. To deal with that, we decided to evaluate whether overfitting occurred and do dimension reduction after the first stage.

In terms of dimension reduction, there are usually two ways to do that–feature extraction and feature selection. Feature extraction is not recommended because of the interpretability problem: since one of our goals is to identify the most important/influential OTUs, it's not desirable to combine the OTUs to form new quantities. We tried PCA, ICA, and t-SNE anyhow, the result produced is extremely bad–the accuracy is around 0.6 for all models. So, there is not a single reason that we should perform feature extraction. On the other hand, feature selection is feasible and meaningful, we just need to come up with a feature selection criteria and model re-fitting procedure. The criteria that makes the most sense is–select the most important ones.

4.2 Stage II: Feature Selection Based on Importance and Model Reconstruction

Feature importance were calculated and ranked via feature importance assessment. We do this only on the training data to avoid information leakage. Models were then re-fitted using the selected most important features (numbers vary) and the convergence in error rates was studied.

We argue that it is crucial to study this convergence behavior because the training error of all models are 0 and we are not sure if it's overfitting or supremacy, yet the answer to this question is very meaningful to us–without knowing this, we cannot be sure if the generalization potential of our models is pushed to the limit. Admittedly, with an exhaustive parameter tuning process, the generalization potential can be maximized, but that is too computationally expensive for our device and software. To answer this "overfitting or supremacy" question, we selected different number of features based on importance, varying from 20 to 1000, trained the models on these features, and observed the training error rate and test error rate. We define supremacy to be satisfaction of these two conditions: 1. the training error rate is always 0; and 2. the test error rate cannot be lower than that of the full model. Otherwise, it's overfitting.

In summary, the procedure of studying the convergence is presented as follows:

- Select the 10, 20, . . . , 500 most important features using each method (RF, GB, XGB, and SVM) from the full model
- Construct a Most-Important-Factor consensus pool (sizes ranged from 27 to 996)
- Fit the model on these features, calculate the train and test error rate
- See if there's any convergence trend and check which models achieved supremacy

Method	Train Error Rate	Test Error Rate	Precision	Recall	F1-Score
Random Forest	0	0.06	0.91	1.00	0.95
Gradient Boost	0	0.08	0.90	0.97	0.93
XGBoost	0	0.06	0.91	1.00	0.95
SVM (raw)	0	0.24	0.84	0.72	0.78
SVM (feature re-sclaed)	0	0.06	0.93	0.97	0.95

Table 1: Evaluation Metrics for Each Model

Lastly, after reconstruction, the model that has the best performance and generalization potential was selected.

5. Results

5.1 Model Performance

The evaluation metrics are summarized in Table 1. We can see that based on test error rate and F-1 score (can be interpreted as a synthesis of precision and recall), Random Forest, XGBoost, and SVM performed the best. Currently our accuracy on test set caps at 94%–it's very promising, because it already exceeded the previous work by Dan et.al yet we might still be able to push one step further. The result is quite good–we see that all of them perform reasonably well, except for the SVM trained on raw data. The reason is because SVM is a distance-based algorithm, and different scales of features can mess up the classfication task.

5.2 Feature Importance

The 10 most important features selected by each model is presented in the Table 2. This number (10) is obtained by taking the maximum of the "number of distinguished influential features" from all four models. "Distinguished" is defined as "very far ahead comparing to the rest of the features in terms of the corresponding metrics". These metrics are: "Mean decreased gini index" for RF, "relative influence" for GB, "gain" for XGB, and "weight" for SVM. In particular, there are 10 distinguished influential features in RF, 2 in GB, 2 in XGB, and none in SVM.

We can see that the tree-based methods give quite similar ranking in terms of feature importance, and they are very different from the ones given by distance-based method. However, this could also be due to the fact that the "weight" function is defined by myself since no available package is available to calculate feature

Model	Most Important OTUs (Decreasing Importance)
Random Forest	625, 1225, 976, 1053, 910, 1301, 970, 390, 628, 793
Gradient Boost	625, 1301, 970, 976, 964, 53, 222, 390, 840, 400
Xtreme Gradient Boost	625, 1301, 976, 390, 1279, 41, 1087, 1225, 75, 222
Support Vector Machine	1020, 713, 1157, 893, 167, 814, 680, 20, 94, 628

Table 2: Most Important Features Ranked by Each Model

importance in SVM. It's defined as follows:

$$W = (X^T\beta)^T(X^T\beta)$$

, where X is a $k \times 1322$ matrix, represents the support vectors correspond to the each feature and β_i is a $k \times 1$ vector, represents the coefficients of the support vectors (k support vectors). Then the weight for the i-th feature is just the i-th entry in W. The weight function is defined this way because "importance" in SVM is equivalent to "influence on determining the position of the decision boundary", and basically my "weight" reflected the latter quantity.

Three models agreed that OTU625, 1301, 976, and 390 are important, two models agreed that OTU1225, 970, 628, and 222 are important.

5.3 Convergence Behavior

Following the procedure in 4.2, the training error rates of all the models (4 method times 50 feature pools = 200 models in total) are 0. The test error rate convergence graph is presented in Figure 4, 5, 6, 7. It's shown that we can indeed push our model's generalization potential a little more without exhaustive parameter tuning–now SVM can achieve a 96% test accuracy. Other than this, a few interesting observations can be made:

1. The test error rate of RF and XGB cannot go below 0.06, the test error rate of full model, which means RF and XGB achieved supremacy from the beginning. However, the test error rate of GB can go below 0.08 (its full model test error rate) the test error rate of SVM can go below 0.06 (its full model test error rate), meaning these two models were overfitting using all features.

2. GB and XGB managed to achieve their best accuracy, i.e. 94% on test set only using about 60 out of 1322 features. That is quite magical.

3. All the tree based algorithm exhibit an "alternating test error rate" pattern. This is because all the trees in the forest only disagree on the classfication result(s) of one or a few points. These problematic

points are called critical points, for that they lie on the decision boundary, and any influx or efflux of information would only move them by a little–either a little positive or a litte negative.

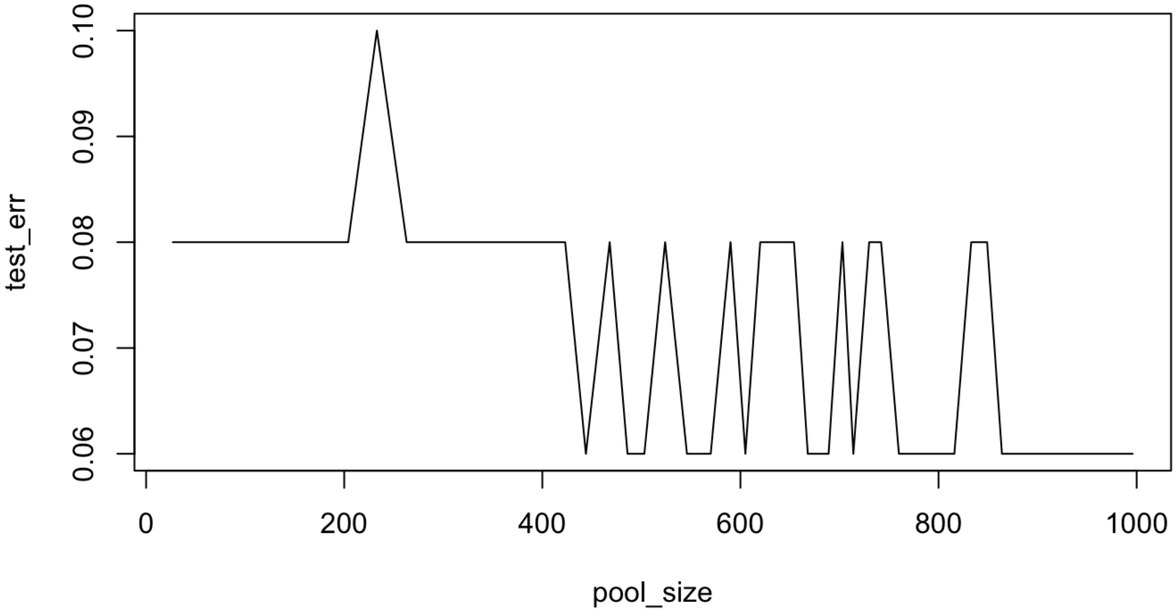


Figure 4: Convergence in Test Error Rate of Random Forest

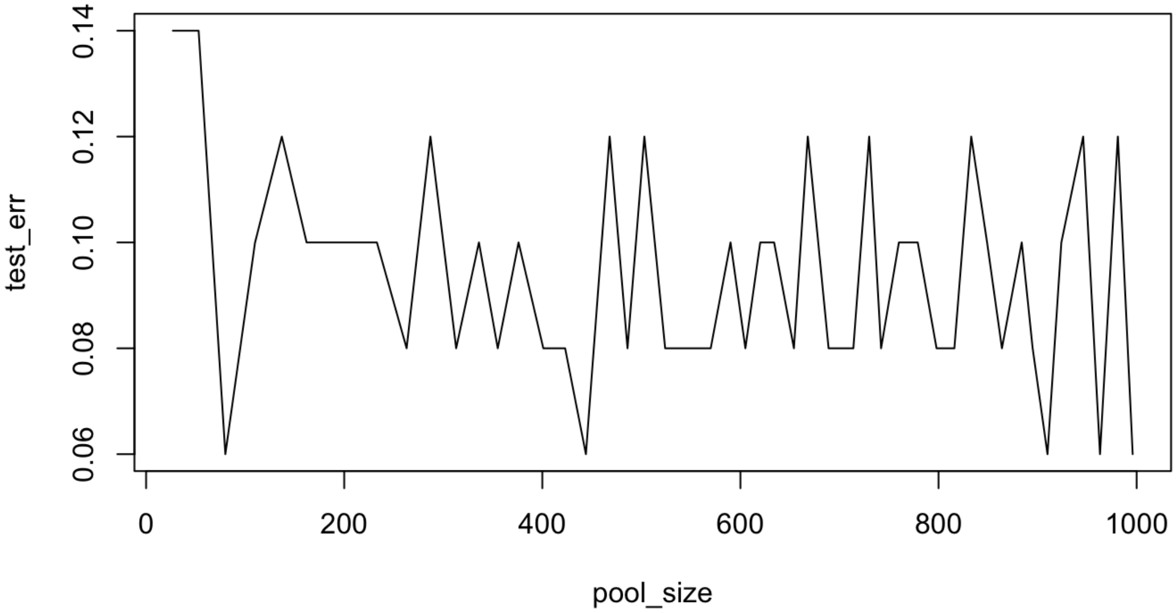


Figure 5: Convergence in Test Error Rate of Gradient Boost

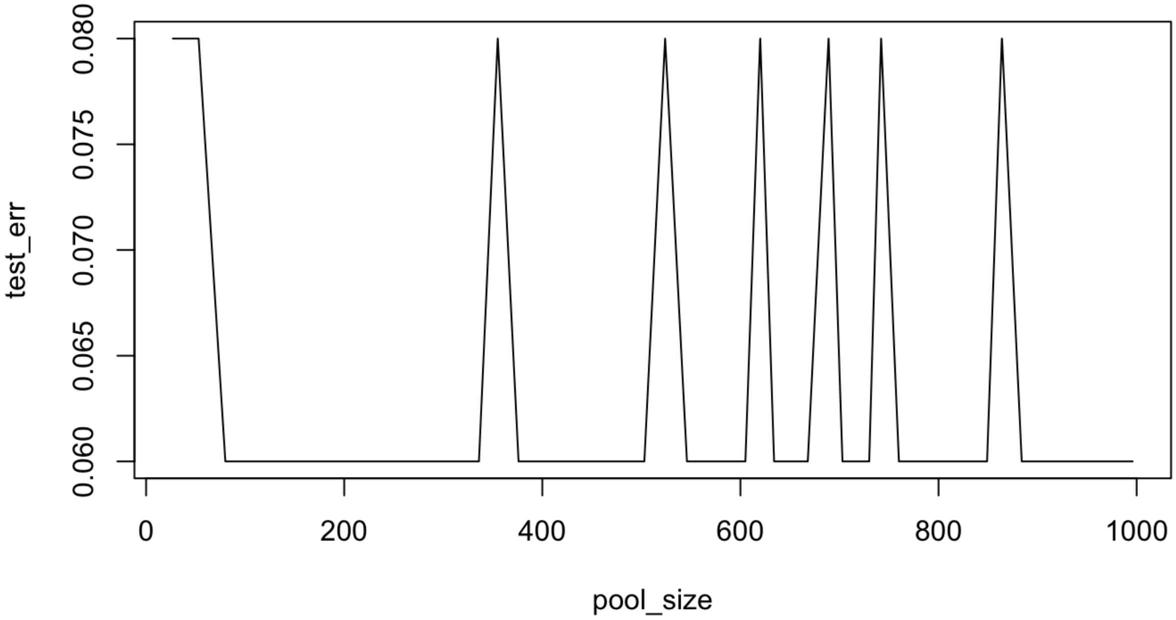


Figure 6: Convergence in Test Error Rate of Xtreme Gradient Boost

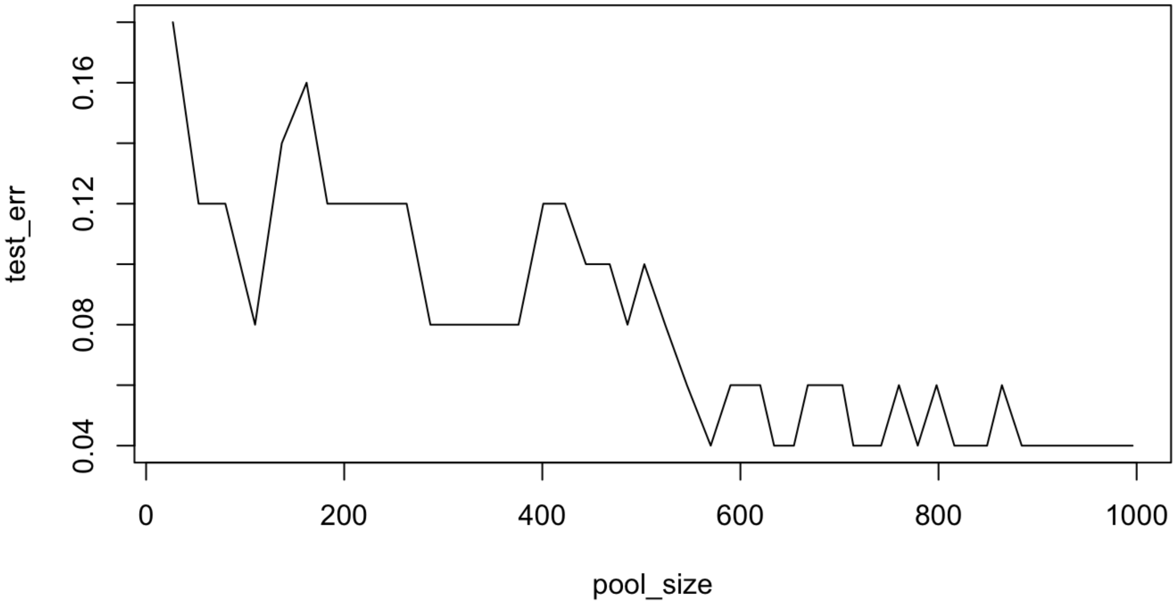


Figure 7: Convergence in Test Error Rate of Support Vector Machine

6. Discussion

6.1 Conclustion

With a slight parameter tuning and correct feature selection, we are able to predict the ASD outcome based on subject's OTU information with 96% accuracy, which surpassed the performance of the models by Dan et. al.

We ranked the most 10 important features using all four models: 3 models agreed that OTU625, OTU976, OTU1301, OTU390 are among the 10 most important OTUs; 2 models agreed that OTU1225, OTU970, OTU628, OTU222 are among the 10 most important OTUs.

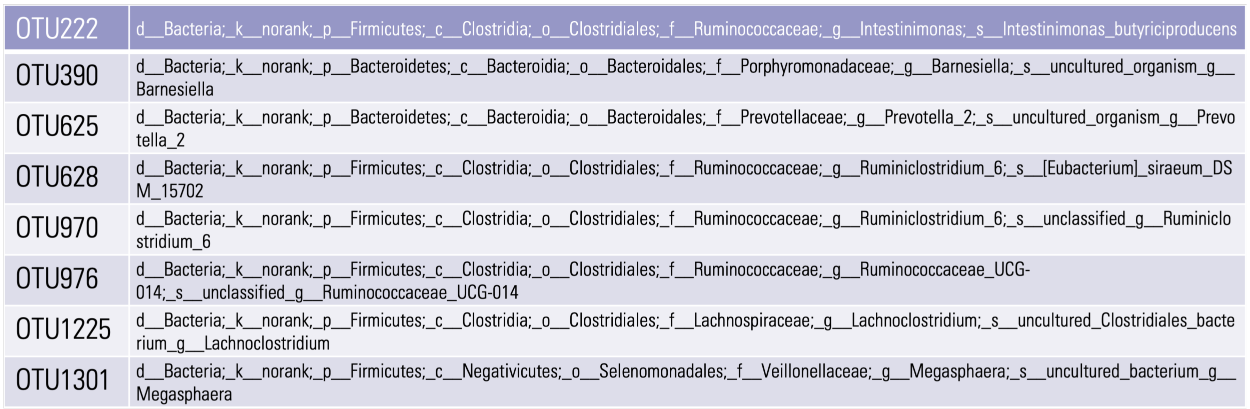
Based on the stage I evaluation metrics, RF, XGB, and SVM perform equally good; but GB and SVM overfitted while others achieved supremacy using all features. After stage II, we see that using the most important k features ($k \approx 600$), SVM outshines the other tree-based methods in terms of test error rate. Therefore, the model using cost-based SVM with $C = 0.5$ and linear kernel, trained on 600 most important features is the best: 100% training accuracy and 96% test accuracy achieved.

6.2 Future Directions

Parameter tuning could be done more exhaustively using k-fold cross validation. We did not have enough time because this process is too computational expensive.

Deep Learning models could be implemented to do the classification since it's becoming a new trend, especially in disease prediction area. We failed to implement such methods due to R problems–Keras (for MLPNN and CNN) and Ruta (for Autoencoders and LSTM) always lead to abortion of R sessions.

Better interpretation can be made on the results–not only on "OTU level", but can go deeper onto microbiome level from a biological perspective. For example, Figure 8 shows the names of the taxa corresponding to some of our "most important features": we can see that there are a lot of similarities in their names, but I'm incapable of decoding the secrets behind.



OTU222	d__Bacteria;_k__norank;_p__Firmicutes;_c__Clostridia;_o__Clostridiales;_f__Ruminococcaceae;_g__Intestinimonas;_s__Intestinimonas_butyriciproducens
OTU390	d__Bacteria;_k__norank;_p__Bacteroidetes;_c__Bacteroidia;_o__Bacteroidales;_f__Porphyromonadaceae;_g__Barnesiella;_s__uncultured_organism_g__Barnesiella
OTU625	d__Bacteria;_k__norank;_p__Bacteroidetes;_c__Bacteroidia;_o__Bacteroidales;_f__Prevotellaceae;_g__Prevotella_2;_s__uncultured_organism_g__Prevotella_2
OTU628	d__Bacteria;_k__norank;_p__Firmicutes;_c__Clostridia;_o__Clostridiales;_f__Ruminococcaceae;_g__Ruminiclostridium_6;_s__[Eubacterium]_siraeum_DSM_15702
OTU970	d__Bacteria;_k__norank;_p__Firmicutes;_c__Clostridia;_o__Clostridiales;_f__Ruminococcaceae;_g__Ruminiclostridium_6;_s__unclassified_g__Ruminiclostridium_6
OTU976	d__Bacteria;_k__norank;_p__Firmicutes;_c__Clostridia;_o__Clostridiales;_f__Ruminococcaceae;_g__Ruminococcaceae_UCG-014;_s__unclassified_g__Ruminococcaceae_UCG-014
OTU1225	d__Bacteria;_k__norank;_p__Firmicutes;_c__Clostridia;_o__Clostridiales;_f__Lachnospiraceae;_g__Lachnoclostridium;_s__uncultured_Clostridiales_bacterium_g__Lachnoclostridium
OTU1301	d__Bacteria;_k__norank;_p__Firmicutes;_c__Negativicutes;_o__Selenomonadales;_f__Veillonellaceae;_g__Megasphaera;_s__uncultured_bacterium_g__Megasphaera

Figure 8: Most Important Features and their Taxa Name

References

1. Dataset: https://www.kaggle.com/datasets/antaresnyc/human-gut-microbiome-with-asd

2. Original publication: https://www.tandfonline.com/doi/full/10.1080/19490976.2020.1747329