



RESEARCH PAPER/REPORT



# Best practices for developing microbiome-based disease diagnostic classifiers through machine learning

Peikun Li <sup>a,\*</sup>, Min Li <sup>a,\*</sup>, and Wei-Hua Chen <sup>a,b</sup>

<sup>a</sup>Key Laboratory of Molecular Biophysics of the Ministry of Education, Hubei Key Laboratory of Bioinformatics and Molecular Imaging, Center for Artificial Intelligence Biology, Department of Bioinformatics and Systems Biology, College of Life Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei, China; <sup>b</sup>School of Biological Science, Jining Medical University, Rizhao, China

## ABSTRACT

The human gut microbiome, crucial in various diseases, can be utilized to develop diagnostic models through machine learning (ML). The specific tools and parameters used in model construction such as data preprocessing, batch effect removal and modeling algorithms can impact model performance and generalizability. To establish an generally applicable workflow, we divided the ML process into three above-mentioned steps and optimized each sequentially using 83 gut microbiome cohorts across 20 diseases. We tested a total of 156 tool-parameter-algorithm combinations and benchmarked them according to internal- and external- AUCs. At the data preprocessing step, we identified four data preprocessing methods that performed well for regression-type algorithms and one method that excelled for non-regression-type algorithms. At the batch effect removal step, we identified the “ComBat” function from the *sva* R package as an effective batch effect removal method and compared the performance of various algorithms. Finally, at the ML algorithm selection step, we found that Ridge and Random Forest ranked the best. Our optimized work flow performed similarly comparing with previous exhaustive methods for disease-specific optimizations, thus is generally applicable and can provide a comprehensive guideline for constructing diagnostic models for a range of diseases, potentially serving as a powerful tool for future medical diagnostics.

## ARTICLE HISTORY

Received 5 December 2024  
Revised 13 March 2025  
Accepted 28 March 2025



## KEYWORDS

Gut microbiome; machine learning; patient stratification; disease diagnostic models; optimal model construction workflow


## Introduction

The human gut microbiome plays crucial roles in various functions such as food digestion,<sup>1</sup> immune system regulation,<sup>2</sup> antiviral and antibacterial protection, and even impacts on the central nervous system.<sup>3</sup> The human gut bacterial community is a complex microbial ecosystem closely linked to overall health. Over the past decades, researchers have highlighted the pivotal role of the gut microbiome in the onset and progression of various diseases,<sup>4</sup> including gastrointestinal diseases like Colorectal Cancer (CRC), Crohn's Disease (CD),<sup>5</sup> and Ulcerative Colitis (UC);<sup>6,7</sup> autoimmune diseases such as Rheumatoid Arthritis (RA)<sup>8</sup> and Systemic Lupus Erythematosus (SLE);<sup>9</sup> metabolic disorders like Obesity<sup>10</sup> and Type 2 Diabetes Mellitus (T2D);<sup>11</sup> and neurological conditions

such as Parkinson's Disease (PD);<sup>12</sup> Alzheimer's Disease (AD)<sup>13</sup> and Autism Spectrum Disorder (ASD).<sup>14–16</sup> Additionally, gut microbiome has been identified as a potential biomarker for distinguishing between patients with CRC, UC, and CD from control groups.<sup>17</sup> Meanwhile, growing evidence supports the feasibility of constructing disease diagnostic models based on gut microbiome using machine learning. These models integrate the composition of gut microbiome with sample meta-data for training, enabling differentiation between patient and control groups and evaluation of predictive performance through area under the receiver operating characteristic curve (AUC) within cohorts (i.e., internal validation). Multicohort diagnostic models for colorectal cancer based on meta-genomic data have demonstrated consistently high

**CONTACT** Wei-Hua Chen  [weihuachen@hust.edu.cn](mailto:weihuachen@hust.edu.cn)  Key Laboratory of Molecular Biophysics of the Ministry of Education, Hubei Key Laboratory of Bioinformatics and Molecular Imaging, Center for Artificial Intelligence Biology, Department of Bioinformatics and Systems Biology, College of Life Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China

\*Equal contribution.

 Supplemental data for this article can be accessed online at <https://doi.org/10.1080/19490976.2025.2489074>

© 2025 Huazhong University of Science and Technology. Published with license by Taylor & Francis Group, LLC.  
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

accuracy across different cohort test sets, outperforming models constructed from data such as metabolic pathways, while also elucidating the relationship between CRC and gut microbiome.<sup>18–20</sup> For metabolic disorders, gut microbiome features can also be used to build machine learning models. Gou W et al. linked input features to Type 2 Diabetes Mellitus using Gradient Boosting Decision Trees (GBDT) and constructed a microbiome risk score (MRS), revealing a set of microbial features consistently associated with Type 2 Diabetes Mellitus.<sup>21</sup> In addition, gut microbiome, through the gut-microbiome-brain axis and its bidirectional interactions with the nervous system via multiple pathways, is also associated with the onset and progression of neurological diseases. The construction of ASD diagnostic models further underscored the immense potential of predicting disease occurrence and progression through gut microbiome.<sup>22</sup> In summary, machine learning models constructed using gut microbiome have demonstrated exceptional efficacy in the identification of disease biomarkers and diagnostic prediction across a range of diseases. This underscores the feasibility and substantial potential for the development of microbiome-based diagnostic models.

However, achieving accurate predictions through microbiome-based ML models still faces several challenges. Firstly, the composition of the gut microbiome is highly complex, exhibiting substantial inter-individual variation. Moreover, a multitude of factors, including environmental, dietary, and genetic influences, contribute to the modulation of the gut microbiome. Furthermore, the performance of microbiome-based ML models could be significantly affected by sample preprocessing processes<sup>23</sup> and choice of machine learning algorithms.<sup>23,24</sup> P. J. McMurdie et al. found that filtering low-abundance microbes can help reduce noise in the data, thereby enhancing the stability and reliability of the model.<sup>25</sup> Rescaling the data ensures that it maintains a consistent scale, facilitating more accurate comparisons and analysis.<sup>26</sup> Normalizing the data is crucial for ensuring comparability and reproducibility of results.<sup>26</sup> Confounding factors, which are external variables that may influence the relationship between the dependent variable (e.g., disease status) and the

independent variable (e.g., gut microbiome), can lead to spurious associations.<sup>27</sup> Additionally, numerous other steps in model construction can affect performance,<sup>28</sup> with each step offering various execution parameters. For example, different filtering thresholds can be applied to remove low-abundance microbes, and multiple methods are available for data normalization.<sup>28</sup> Also, there are numerous algorithms selectable.<sup>29</sup> For instance, among machine learning algorithms, methods based on Random Forest and Least Absolute Shrinkage and Selection Operator (Lasso) logistic regression are particularly popular due to their advantages, which include high performance with small sample sizes (e.g., fewer than 50), robustness with complex and heterogeneous data (e.g., high-dimensional compositional data), clear ranking of feature importance, and reduced risk of overfitting through feature selection.<sup>30</sup> Therefore, to improve model accuracy and applicability, it's crucial to compare data preprocessing methods and explore various machine learning models.

To address the aforementioned challenges, several solutions have been proposed. J. Wirbel et al., when constructing a model on the CRC dataset, found that avoiding feature selection could effectively prevent overfitting and improve the model's external validation AUC.<sup>28</sup> Recently, S. Lee et al. employed an exhaustive approach to compare 5,184 pipeline combinations across four dimensions – profile modality, batch correction, normalization, and algorithm – to identify the most suitable disease-specific pipelines for CRC and CD.<sup>31</sup> However, a comprehensive and universally applicable machine learning model optimization strategy for multiple diseases remains lacking.

To establish a generally applicable workflow, we divided the machine learning process into three steps (including data preprocessing methods, batch effect removal methods, and model algorithms) and optimized each step sequentially using 83 gut microbiome cohorts across 20 diseases. The data preprocessing involved four components: removal of low-abundance taxa (using four different threshold values), data rescaling, normalization (with six different methods), and correction of confounding factors. Thus, we evaluated 156 different combinations of tools, parameters, and algorithms, and assessed their

performance using both internal AUC and external validation. Finally, we identified five high-performing methods. Our optimized workflow performed comparably to previous exhaustive methods for disease-specific optimizations and is generally applicable to a wider range of diseases.

## Results

### *Data collection and experimental workflow*

To explore the optimal construction process for disease diagnostic models based on gut microbiome, we utilized case-control cohort data previously collected and analyzed by M. Li *et al.*,<sup>32</sup> comprising 83 cohorts associated with 20 distinct diseases. Among these, there were 46 cohorts with 16S rRNA sequencing data and 37 with metagenomic sequencing data, encompassing 5,988 disease samples and 4,411 healthy samples (Figure 1b). Of the 20 diseases, 8 were exclusive to the 16S group: Irritable Bowel Syndrome (IBS), Clostridium Difficile Infection (CDI), Alzheimer's Disease (AD), Mild Cognitive Impairment (MCI), Chronic Fatigue Syndrome (CFS), Multiple Sclerosis (MS), Juvenile Arthritis (JA), and Non-alcoholic Fatty Liver Disease (NAFLD). Five diseases were exclusive to the WGS group: Adenoma, Inflammatory Bowel Disease (IBD), Obesity, Overweight, and Ankylosing Spondylitis (AS). Seven diseases were represented in both sequencing types: Colorectal Cancer (CRC), Crohn's Disease (CD), Ulcerative Colitis (UC), Type 2 Diabetes Mellitus (T2D), Parkinson's Disease (PD), Autism Spectrum Disorder (ASD), and Rheumatoid Arthritis (RA) (Figure 1b). According to the Medical Subject Headings (MeSH, <https://meshb.nlm.nih.gov/>) database and the Human Disease Ontology (DO) database,<sup>33</sup> these 20 diseases were categorized into five groups: seven Intestinal, three Metabolic, four Autoimmune, five Mental/nervous system diseases (Mental for short), and one Liver disease (Figure 1b).

To comprehensively compare the various modeling steps and parameters, we divided the model construction process into three stages. First, we investigated the optimal data preprocessing methods for building disease diagnostic models based

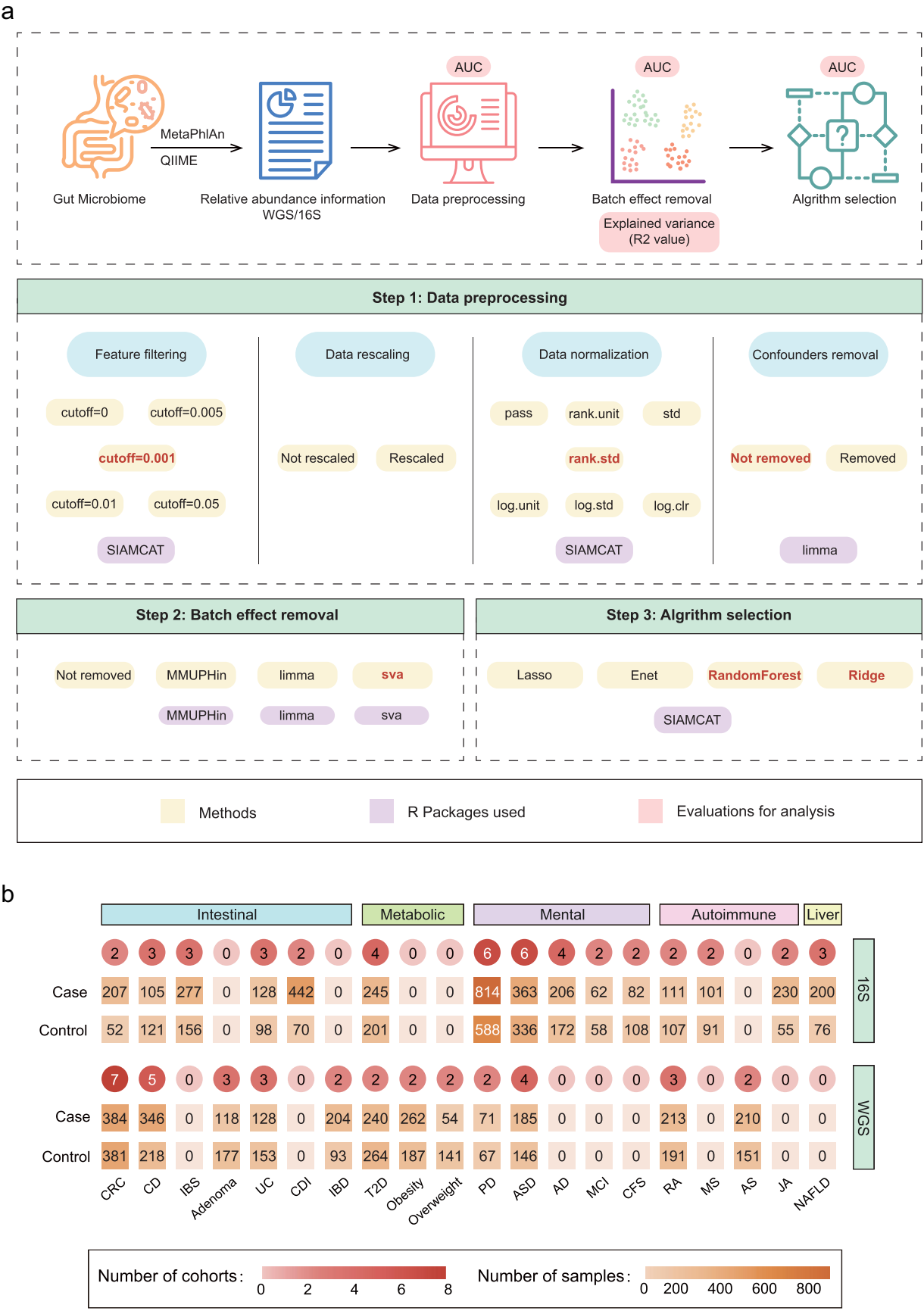
on gut microbiome. The data preprocessing process included four components: filtering low-abundance taxa, data rescaling, data normalization, and confounding factors correction. Data rescaling and confounders correction were binary decisions (i.e., either performed or not), while there were multiple options for the threshold of low-abundance filtering and data normalization methods. Therefore, we first determined the optimal threshold for low-abundance filtering and the best data standardization method, then compared combinations of these with or without rescaling and confounders correction. Next, we compared three methods for removing batch effects to identify the most effective one. Finally, we compared four algorithms to find the one that yielded the best model performance (Figure 1a).

In this study, we constructed models for all datasets corresponding to each disease, performing both internal and external validations. Each dataset underwent five-fold cross-validation repeated three times. In total, 156 tool-parameter-algorithm combinations were developed, with 12,948 internal and external validation processes conducted. From these comprehensive analyses, we identified an optimal model construction workflow that demonstrates robust generalizability across multiple diseases.

### *Appropriate filtering thresholds and normalization methods improve model performance*

Before determining the optimal data preprocessing method, we first compared the performance of models under different thresholds for filtering low-abundance taxa and varying data normalization methods. All models in this section were constructed using the Lasso algorithm. To identify the optimal threshold for filtering low-abundance taxa, we kept all parameters consistent across models except for the threshold itself (i.e., no rescaling of data, no confounders adjustment, and no batch effect correction). Both 16S and WGS data were modeled for each study, and internal and external validations were conducted. The same approach was applied to explore the best data normalization method.

For the selection of the optimal threshold, we compared the performance of models with



**Figure 1.** Research workflow and data collection statistics. (a) This flowchart illustrates the main stages of the research, including data collection, analysis, and the exploration of the optimal model construction process. The first section focuses on comparing data preprocessing methods, including filtering low-abundance taxa (using five different thresholds), data rescaling, data normalization (involving seven methods), and confounding factors adjustment. The second section compares batch effect removal methods

thresholds set at 0.001%, 0.005%, 0.01%, and 0.05% against models where no low-abundance taxa were removed. Taxa with maximum abundance below these thresholds were removed. We calculated the AUCs for each parameter setting and computed the median AUCs for each group. The significance of the differences between the models using these four thresholds and the control group (with no taxa removal) was assessed using the Wilcoxon rank-sum test and paired tests. The comparison of AUCs across models with thresholds of 0.001%, 0.005%, 0.01%, and 0.05% against the no-removal control group showed no significant improvement in AUCs (Figure 2a). However, since the removal of low-abundance taxa was conducted within each cohort, we focused on the internal validation results. For internal validation, the model with 0.001% threshold yielded the highest median AUC, and it also performed well in external validation (Figure 2a). Therefore, in subsequent modeling, we adopted the 0.001% threshold for filtering low-abundance taxa.

In the selection of the data normalization method, we used an approach similar to the threshold selection process. We compared the AUCs of models with data normalized using six different methods against models with data non-normalization and explored the significance of their differences. These six normalization methods are all parameters within the “normalize.features” function of the *SIAMCAT* R package.<sup>28</sup> Specifically, “rank.unit” converts features into ranks and then normalizes each column by the square root of the rank sum (where the number of columns equals the number of samples). “rank.std” converts features into ranks and applies z-score normalization. “log.clr” refers to centered log-ratio transformation (with pseudocounts added). “log.std” involves log-transforming features (after adding pseudocounts) and subsequently applying z-score normalization. “log.unit” applies a logarithmic transformation to features (after adding pseudocounts) and then

normalizes the features or samples using various standardization methods. Finally, “std” refers to the application of z-score normalization only. During internal validation, models with data normalized using any of the five methods, except for std, exhibited higher median AUCs than models with data non-normalization. Specifically, models with data normalized using “rank.std” and “rank.unit” showed the highest median AUCs, both reaching 0.79, significantly higher than the 0.71 AUC observed non-normalization (Figure 2b). Notably, the model with data normalized using the “rank.std” method showed a more significant inter-group difference when compared to the data non-normalization model ( $p < 0.01$ ). Additionally, in external validation, the model with data normalized using the “rank.std” method also had a high median AUC and significant inter-group differences ( $p < 0.001$ ) (Figure 2b). Therefore, in subsequent modeling, we chose to use the “rank.std” method for data normalization.

### **Specific data preprocessing methods enhance model performance**

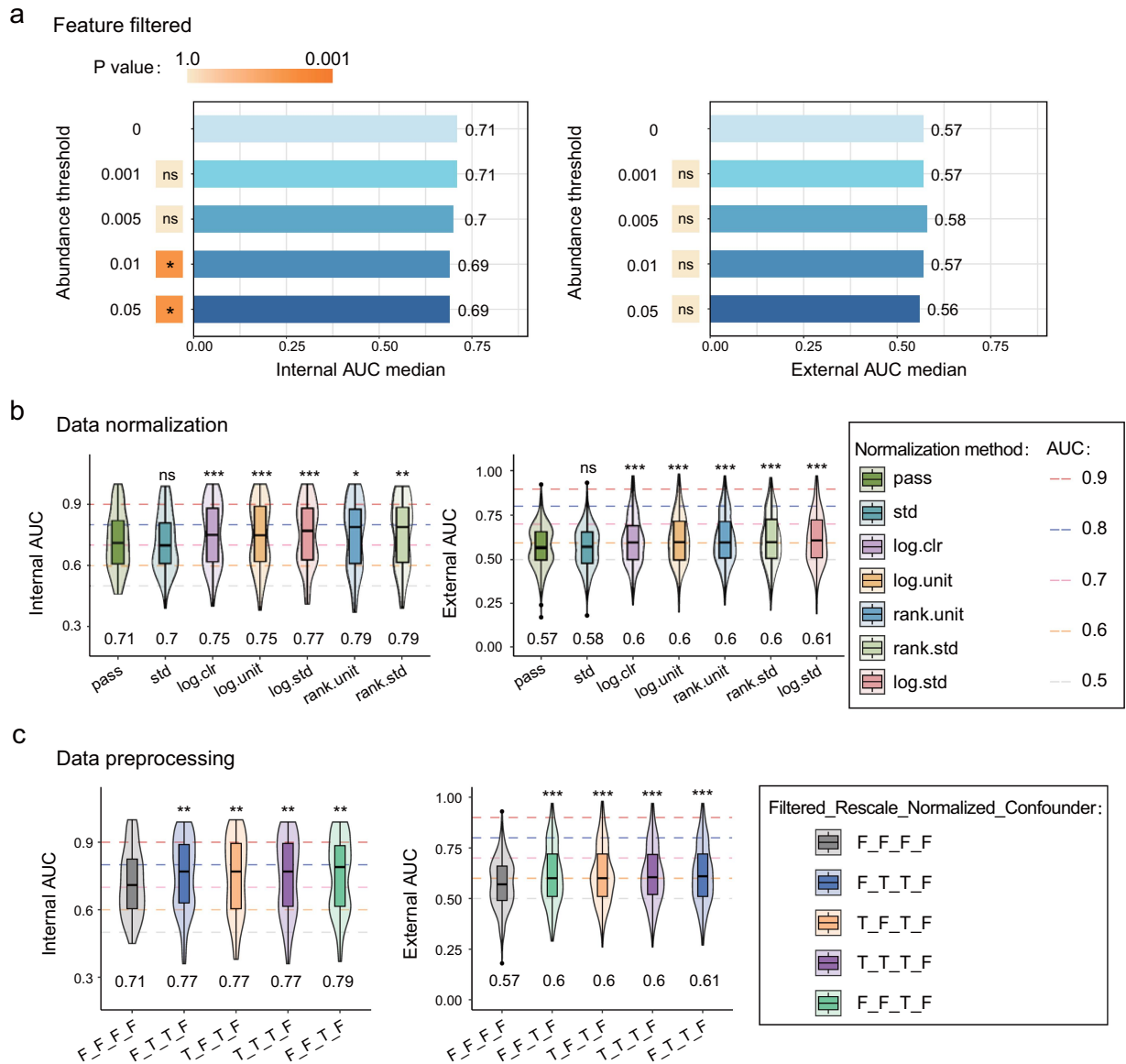
We next explored additional data preprocessing steps including data rescaling and confounding factors correction into our overall comparison.

Firstly, the removal of low-abundance taxa was considered in two scenarios: without removal and with a threshold of 0.001%. Secondly, data rescaling was evaluated as two scenarios: performed or not performed. Thirdly, data normalization was examined in two scenarios: without normalization and with “rank.std” method. Lastly, confounding factors adjustment was also considered in two scenarios: performed (the “removeBatchEffect” function from the *limma* R package) or not performed. We also summarized the specific confounders adjusted for in each cohort (Table S1). These considerations resulted in 16 possible combinations of data preprocessing methods. We constructed models using

---

including three techniques. The third section evaluates algorithm performance involving four algorithms. Yellow indicates selectable methods, purple denotes the R packages used, and pink represents the evaluations used to assess model performance. b. The heatmap shows the composition of samples in the study, categorized into two data types (16S and WGS) and covering 20 diseases. The heatmap also presents the number of cohorts per disease and the count of case and control samples in each cohort. Diseases are classified into six categories based on the Medical Subject Headings (MeSH, <https://meshb.nlm.nih.gov/>) database and the Human Disease Ontology (DO) database.<sup>33</sup>





**Figure 2.** Exploration of optimal data preprocessing methods. a. The bar graph compares model performance between models that did not filter low-abundance taxa and those applied four different thresholds for filtering. The graph displays the median AUCs for both internal and external validations. The left color square highlights the differences in AUC between models using the four thresholds and models without filtering, with  $p$  values calculated using the Two sides Wilcoxon rank-sum test. b. The violin plots compares the model performance between models without data normalization and those with six different normalization methods. The plot annotates the median AUCs for both internal and external validations, and highlights the differences in AUCs between models using the six normalization methods and models without normalization. The  $p$  values are calculated using the two-sided Wilcoxon rank-sum test. c. The violin plot compares model performance between the top four data preprocessing methods and models without any preprocessing. The plot annotates the median AUCs for both internal and external validations and highlights the differences in AUC between models using these four preprocessing methods and those without any preprocessing. The legend indicates whether the four steps – filtering low-abundance taxa, data rescaling, data normalization, and confounding factor correction – were applied (T for applied, F for not applied). The  $p$  values are calculated using the Two sides Wilcoxon rank-sum test. Two sides Wilcoxon rank sum test was used for pairwise group comparisons. The colored horizontal lines represent different AUC levels. The numbers marked in the bottom of d, e, f and g represent the mean of the corresponding AUCs. \* $p < 0.05$ , \*\* $p < 0.01$ , \*\*\* $p < 0.001$ , \*\*\*\* $p < 0.0001$ .

each of these 16 preprocessing methods, including one control group (i.e., no removal of low-abundance taxa, no rescaling, no normalization, and no correction for confounding factors) and 15

experimental groups. We compared the AUCs of these 16 models and the differences between the 15 experimental groups and the control group (Figure S1A).

In internal validation, the models built with the data preprocessing methods F\_T\_T\_F (i.e., no removal of low-abundance taxa, rescaling performed, “rank.std” normalization, no confounding factor correction), T\_F\_T\_F (i.e., removal of low-abundance taxa at 0.001% threshold, no rescaling, “rank.std” normalization, no confounding factor correction), T\_T\_T\_F (i.e., removal of low-abundance taxa at 0.001% threshold, rescaling performed, “rank.std” normalization, no confounding factor correction), and F\_F\_T\_F (i.e., no removal of low-abundance taxa, no rescaling, rank.std normalization, no confounding factor correction) showed higher median AUCs of 0.77, 0.77, 0.77, and 0.79, respectively, compared to the control group’s AUC of 0.71, with significant differences (all  $p$ -values  $<0.01$ ) (Figure 2c). In external validation, these four preprocessing methods also demonstrated good performance, with median AUCs of 0.61, 0.60, 0.60, and 0.60, respectively, notably higher than the control group’s 0.57, and significant differences (all  $p$ -values  $<0.001$ ) (Figure 2c). Besides AUC, we also compared the models using AUPRC (Figure S2A), F1score (Figure S3A), and unitMCC (Figure S4A). The four methods that performed best in terms of AUC also demonstrated consistent advantages across these three additional metrics.

In summary, the models constructed using the F\_T\_T\_F, T\_F\_T\_F, T\_T\_T\_F, and F\_F\_T\_F data preprocessing methods performed best, showing consistency in both internal and external validation. Therefore, for subsequent modeling steps, we adopted these four combinations as effective data preprocessing strategies.

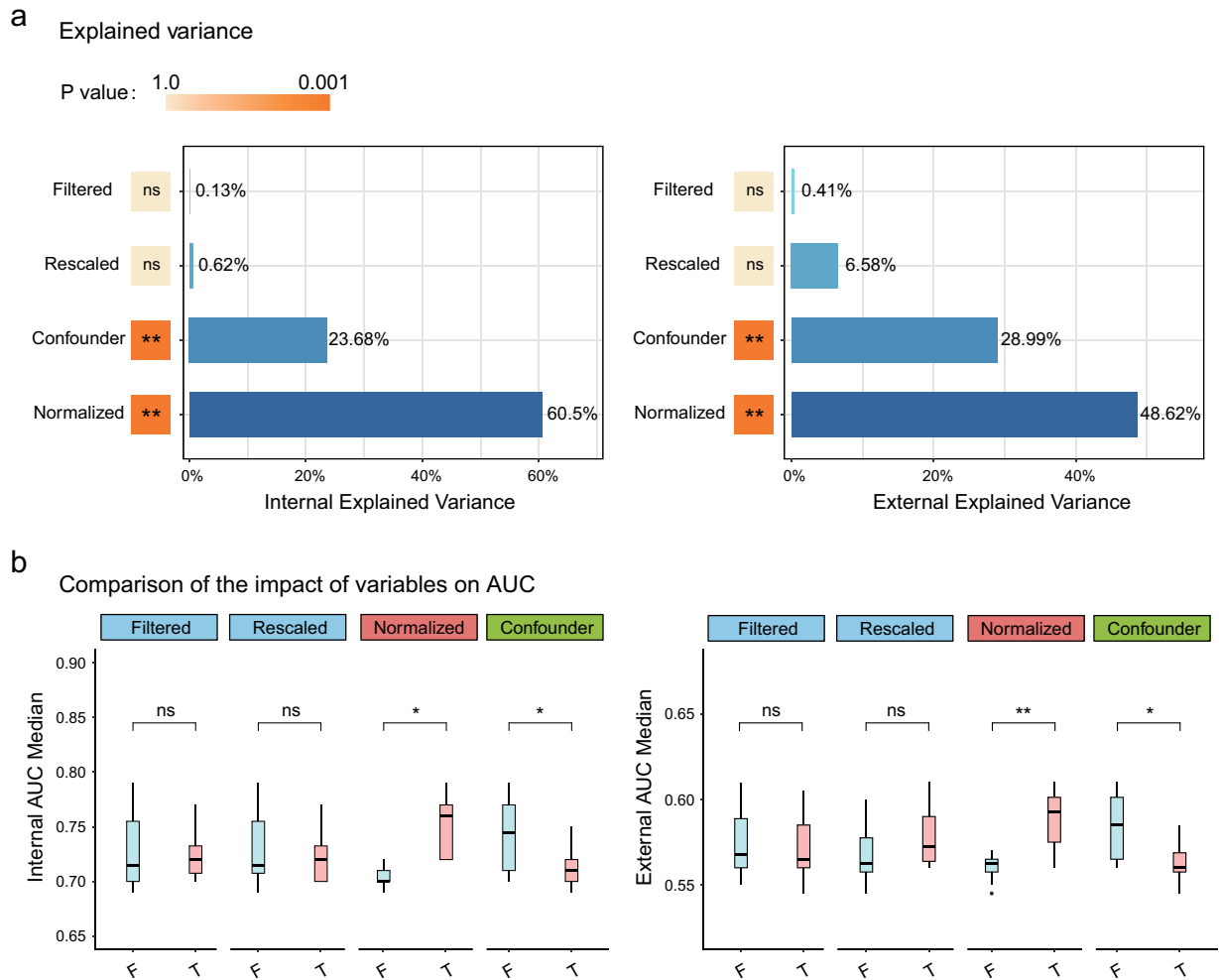
### **Data normalization improves model performance while confounders adjustment lowers it**

To investigate the contribution effects for improving performance AUCs of these four data preprocessing steps, we applied Adonis analysis (a type of multivariate analysis of variance)<sup>34</sup> to assess the contribution and significance of each step. From the internal validation results, we found that data normalization and confounders adjustment were the two primary contributors to model performance differences, both with significant results (Figure 3a,  $p < 0.01$ ). In contrast, removing low-

abundance taxa and data rescaling had minimal contributions to these differences. The external validation results corroborated these findings, further indicating that data normalization and confounders adjustment are the two most impactful factors affecting model performance (Figure 3a).

To determine whether these preprocessing steps positively or negatively impacted model performance, we analyzed and compared the median AUCs of the 16 models derived from the 16 preprocessing methods. In internal validation, we observed no significant impact from either removing low-abundance taxa or data rescaling, whereas both data normalization and confounders adjustment caused significant differences in model performance, which aligns with our earlier conclusions. Notably, data normalization significantly improved AUCs ( $p < 0.05$ ), while confounders adjustment significantly decreased AUCs ( $p < 0.05$ ) (Figure 3b). We observed similar results in external validation that removing low-abundance taxa and data rescaling had no significant effect on AUCs, whereas data normalization significantly improved AUCs ( $p < 0.01$ ), and confounders adjustment significantly reduced AUCs ( $p < 0.05$ ).

Regarding the confounders adjustment, in addition to the methods mentioned above, we also attempted confounders adjustment using a matching approach (the “matchit” function from the MatchIt R package). Internal validation results showed that applying the matchit function from the MatchIt R package for confounders adjustment led to a significantly lower AUCs compared to the removeBatchEffect function from the limma package (Figure S5A). The decrease in AUCs observed with MatchIt correction may be due to the reduction in sample size after matching, as only matched pairs were kept. The smaller dataset could limit the model’s capacity to learn effectively, resulting in lower performance. In contrast, limma correction retains all samples, providing a larger dataset for model training, which likely contributes to the higher AUCs observed. Furthermore, when compared to models without confounders adjustment, this approach also caused a significant reduction in AUCs, consistent with previous findings in the literature.<sup>28</sup> Therefore, the confounder adjustment method applied in subsequent analyses will remain consistent with the previous approach.



**Figure 3.** Analysis of the impact of four data preprocessing steps on model AUC. a. The bar graph illustrates the contribution of the four preprocessing steps (filtering low-abundance taxa, data rescaling, data normalization, and confounding factor correction) to the variance in median AUCs, as determined by Adonis analysis. The left color square highlights the significance of these contributions, with higher values indicating a greater impact of the step on model performance. b. The boxplot shows the differences in median AUCs across 16 different preprocessing methods, as analyzed by the four preprocessing steps. The  $p$  values are calculated using the Two sides Wilcoxon rank-sum test. The legend indicates whether each step was applied (T) or not applied (F).

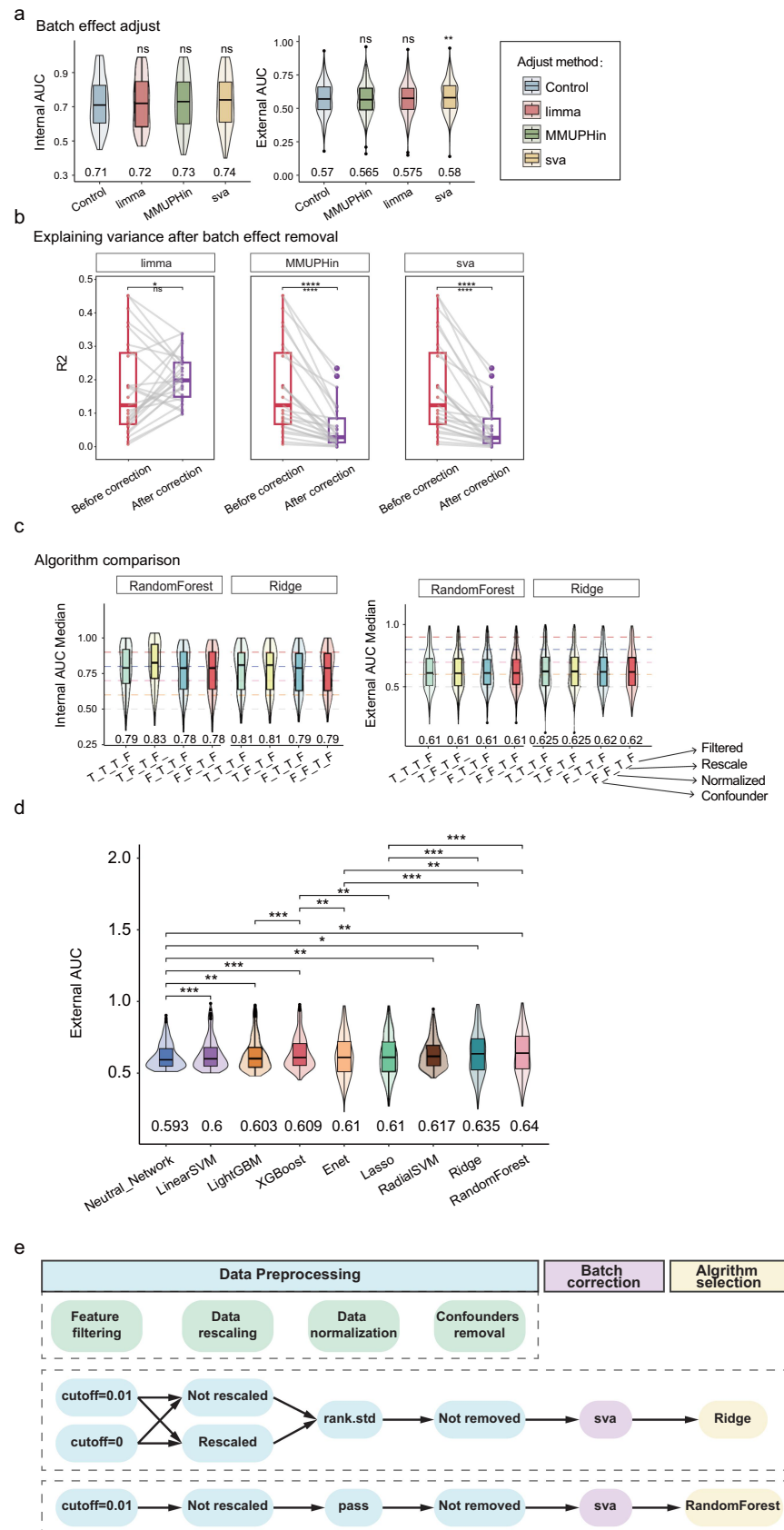
These findings are consistent with the four effective preprocessing methods identified earlier that data preprocessing involved data normalization without confounders adjustment, leading to better model performance. As a result, these four preprocessing methods will be used in subsequent model performance comparisons.

ComBat effectively removes batch effects while preserving strong model performance. In this study, we compared three methods for batch effect removal: the “removeBatchEffect” function from the *limma* R package,<sup>35</sup> the “adjust\_batch” function from the *MMUPHin* R package,<sup>36</sup> and the “ComBat” function from the *sva* R package.<sup>37</sup> First, we compared the

AUCs of models using these three batch effect removal methods with those of models without batch effect removal. In terms of median AUCs, none of the three methods significantly improved the AUCs (Figure 4a). While in the external validation results, the models with batch effect removed by the “ComBat” function from the *sva* R package had a slightly higher median AUC (0.58) compared to 0.57 for the models without batch effect removal, with a significant difference between the groups ( $p < 0.01$ ) (Figure 4a).

However, we believe that evaluating batch effect removal methods solely based on AUCs is insufficient and lacks comprehensive justification. Therefore, we conducted Adonis analysis





**Figure 4.** Comparison of batch effect removal methods and algorithms and summary of optimal machine learning methods for gut microbiome-based model construction. **a.** The violin plot compares the performance of models generated using three batch effect removal methods with those generated without batch effect removal. The plot annotates the median AUCs for both internal and external validations, with  $p$  values calculated using the Two sides Wilcoxon rank-sum test. **b.** The boxplot shows the change in R-squared ( $R^2$ ) values before and after batch effect removal using the *limma*, *MMUPHin*, and *sva* packages, indicating the extent to which batch effects were reduced. P-values are calculated using the Two sides Wilcoxon rank-sum test, with smaller post-removal  $R^2$

to calculate the  $R^2$  values (variance explained by batch effect) before and after batch effect removal using these three methods. A higher  $R^2$  value indicates that batch information explains more of the group differences, while a lower  $R^2$  value indicates the opposite. Thus, if the  $R^2$  value after batch effect removal is significantly lower than before, it suggests that the method effectively removes the batch effect. After batch effect removal using the “removeBatchEffect” function from the *limma* R package, we observed an increase in  $R^2$  values for some models, particularly those constructed with 16S data (Figure S5B), indicating that this method might not be suitable for 16S data. Consequently, we do not consider the “removeBatchEffect” function from the *limma* R package to be the optimal batch effect removal tool. In contrast, both the “adjust\_batch” function from the *MMUPHin* R package and the “ComBat” function from the *sva* R package showed significantly reduced  $R^2$  values after batch effect removal ( $p < 0.0001$ ) (Figure 4b), indicating that these two methods effectively reduce the contribution of batch information to group differences.

Considering the previous AUCs performance, we concluded that the “ComBat” function from the *sva* package was the best batch effect removal method among the three and will be used in subsequent modeling processes.

### **Random forest ranks the best, although different ML algorithm require specific data preprocessing methods**

We selected four commonly used machine learning algorithms in *SIAMCAT*<sup>28</sup> for constructing disease diagnostic models for comparison: Lasso regression,<sup>38</sup> Ridge regression,<sup>39</sup> Elastic Net (Enet),<sup>40</sup> and Random Forest.<sup>41</sup> The relative abundance data used for model construction was

divided into two data types including 16S and WGS, and models were built separately for each type. The data preprocessing methods used were the four well-performing methods previously recommended, namely T\_T\_T\_F, T\_F\_T\_F, F\_T\_T\_F, and F\_F\_T\_F, and the batch effect removal was performed using the “ComBat” function from the *sva* R package. All model parameters were kept identical except for the algorithm. The AUCs of the four groups of models generated by the four different algorithms were compared.

When we combined the models constructed using 16S and WGS data for comparison, we found no significant differences in AUCs among the four algorithms. Therefore, we separately compared the models built with 16S data and WGS data (Figure 4c, Figure S6A). For models built with 16S data, there was no significant difference in the median AUCs among the four algorithms in the internal validation results, as confirmed by the Kruskal-Wallis test (Figure S6B,  $p = 0.67$ ). However, in the external validation results, the Random Forest algorithm showed a slightly higher median AUC compared to the other three algorithms (with AUCs of 0.58, 0.58, 0.57, and 0.58 for the models using the four different preprocessing methods), and significant intergroup differences were observed compared to the other three algorithms ( $p$ -values all  $< 0.05$ ) (Figure S6B). When comparing algorithms for models built with WGS data, the internal validation results again showed no significant difference in median AUCs among the four groups (Figure S6C,  $p = 0.89$ ). However, in the external validation results, the Ridge algorithm resulted in a slightly higher median AUC compared to the other three algorithms (all models using the four different preprocessing methods had an AUC of 0.73), with significant differences observed compared to the Random Forest and Enet groups ( $p$ -values all  $< 0.05$ ) (Figure S6C).

values indicating better batch effect removal. c. The violin plot compares the performance of two better-performing algorithm models constructed using the previously identified optimal preprocessing methods and the best batch effect removal method. The plot annotates the median AUCs for both internal and external validations, with  $p$ -values calculated using the Two sides Wilcoxon rank-sum test. d. The violin plot shows the comparison of external validation AUC results between the models built using the most suitable data preprocessing methods for each of the five additional algorithms, Lasso, Enet and the two best methods from our original conclusion. The plot annotates the median AUCs for external validations, with  $p$ -values calculated using the Two sides Wilcoxon rank-sum test. e. The roadmap illustrates five optimal workflows for constructing diagnostic models, identified through three key steps: data preprocessing, batch effect removal, and algorithm selection.

Since the algorithm used for model construction when exploring the best preprocessing methods earlier was Lasso, we compared the performance of models constructed using the four different algorithms without any data preprocessing. It was observed that the median AUCs of models built with Lasso, Ridge, and Enet was significantly lower than that of models constructed with the previously recommended preprocessing methods, while the median AUC of the Random Forest model was significantly higher than the earlier models (Figure S7A). Therefore, we concluded that the previously recommended best data preprocessing methods may not be suitable for models constructed using the Random Forest algorithm.

To determine the best data preprocessing method for models using the Random Forest algorithm, we re-compared 16 different data preprocessing methods. In internal validation, the T\_F\_F\_F method (i.e., only removing low-abundance taxa) resulted in the highest median AUC of 0.84, significantly higher than the control group's 0.79 (Figure S7B,  $p < 0.05$ ). In external validation, the T\_F\_F\_F method still performed well, with the highest median AUC of 0.64, although it did not show a significant difference compared to the control group (Figure S7B).

Additionally, we used Adonis analysis (multi-factor variance analysis) to calculate the contribution and significance of the four data preprocessing steps to model performance differences. Internal validation results showed that the step of data rescaling contributed significantly to model performance differences ( $p < 0.05$ ). External validation results indicated that confounding factors adjustment was a significant contributor to model performance differences ( $p < 0.01$ ) (Figure S8A). Further analysis of the median AUCs of models built using the 16 different data preprocessing methods revealed that data rescaling significantly lowered the AUCs in internal validation ( $p < 0.05$ ), while confounding factors adjustment significantly lowered the AUCs in external validation ( $p < 0.05$ ) (Figure S8B). These findings are consistent with the conclusions drawn from the Adonis analysis and align with the T\_F\_F\_F method's lack of data rescaling and confounding factor adjustment steps.

Given that the optimal low-abundance taxa removal threshold and data normalization method were initially determined using the

Lasso model, in order to strengthen the robustness of our identified optimal preprocessing methods, we extended our investigation to evaluate these two optimal parameters for both Ridge and Random Forest models which performed better in our previous analyses. For the removal of low-abundance taxa, we assessed four threshold values and discovered that the threshold of 0.001% consistently yielded the highest median AUC in both internal and external validations for both the Ridge and Random Forest models (Figure S9A, S9B). Notably, for the Random Forest model, the internal validation results were significantly improved compared to the models which were not performed the removal of low-abundance taxa ( $p < 0.05$ ) (Figure S9B). In terms of data normalization, we tested six different methods. We found that the "rank.std" method was the most effective for the Ridge model, achieving the highest median AUC (0.78) in internal validation and maintaining robust performance in external validation. Importantly, the performance of the Ridge model with "rank.std" was significantly superior in both internal and external validation compared to the models which were not performed any data normalization method (Figure S9A). Conversely, for the Random Forest model, the highest median AUC was observed when no data normalization was employed, both in internal and external validations (Figure S9B). These results corroborated the optimal data preprocessing methods we concluded above: for the Random Forest model, the optimal method involved removing low-abundance taxa with a 0.001% threshold without applying any data normalization; for the Ridge model, the optimal methods included employing "rank.std" as the normalization method.

To make the conclusions more rigorous, we further investigated whether the most suitable data preprocessing methods for the Ridge and Enet algorithms were the same as the four best methods identified before. We also compared the performance of the 16 data preprocessing methods when using the Ridge and Enet algorithms. As expected, we found that the best preprocessing methods for both the Ridge and Enet algorithms were indeed the same four methods previously identified (Figure S10A, S10B).

In conclusion, when using the Random Forest algorithm, the best model construction process involves the T\_F\_F\_F data preprocessing method and batch effect removal using the “ComBat” function. When using Lasso, Ridge, or Enet regression algorithms, as the external validation results showed no significant differences among the three algorithms for 16S data models (Figure S6B), and Ridge performed better than the other two for WGS data models (Figure S6C), the best model construction process includes the four data preprocessing methods T\_T\_T\_F, T\_F\_T\_F, F\_T\_T\_F, and F\_F\_T\_F, batch effect removal using the “ComBat” function, and model training with the Ridge algorithm (Figure 4e).

To further validate the robustness of our approach, we expanded our comparison to include several widely-used algorithms for constructing gut microbiome-based disease diagnostic models, such as Linear support vector machine (Linear SVM),<sup>42</sup> Radial basis function support vector machine (Radial SVM),<sup>42</sup> Extreme gradient boosting (XGBoost),<sup>43,44</sup> Light gradient boosting machine (LightGBM),<sup>45,46</sup> and Neural Networks,<sup>47</sup> in addition to the four algorithms we initially examined. In order to ensure a fair comparison, we systematically evaluated the performance of the 16 data preprocessing methods for each of these five algorithms in order to identify their most suitable preprocessing methods. Based on AUC results from external validation, the optimal preprocessing methods for each algorithm were determined as follows: T\_T\_T\_F for Linear SVM (Figure S11A), T\_T\_T\_F for Radial SVM (Figure S12A), F\_T\_F\_F for XGBoost (Figure S13A), T\_F\_F\_F for LightGBM (Figure S14A), and T\_T\_T\_F for Neural Networks (Figure S15A).

Subsequently, we built models using the best data preprocessing methods for each of these five algorithms and compared their performance against the models generated by the optimal methods identified for the Lasso, Enet, Ridge and Random Forest. Our analysis revealed that the median AUC for the Ridge and Random Forest (0.635 and 0.64, respectively) outperformed the best models from the seven other algorithms (Figure 4d). This suggests that the optimal modeling workflow we proposed maintains a distinct

advantage over the best workflows of these commonly used algorithms.

Ultimately, given that our optimal method is derived from a macro-level analysis of datasets from multiple diseases, we further validated whether our method improves model performance for each disease individually. We compared the AUC of models constructed using the optimal method with those built using non-optimal methods across 20 diseases. Among the 20 diseases, 9 showed a significant improvement in AUC when the optimal method was used. The AUCs for the other 11 diseases also demonstrated an upward trend (Figure S16A). This indicates that the optimal methods we proposed can be applied to a wide range of diseases, demonstrating a certain level of generalizability.

### **Comparing our pipeline with methods reported in the literature**

We next compared our pipeline with the optimal methods recently reported by S. Lee et al., which were identified through exhaustive search on CRC and CD datasets.<sup>31</sup> Since the profiling methods, as reported by S. Lee et al., also affected significantly the final model performance, we adopted the same data procedure. Specifically, we first used Kraken2<sup>48</sup> and Bracken<sup>49</sup> with the HRGM human gut-specific microbial genome database<sup>50</sup> to obtain the relative abundance profiles of gut microbiome. Using the methods summarized above, we constructed a leave-one-dataset-out (LODO) model, where the training datasets included all datasets analyzed in S. Lee et al.’s study, and the validation datasets were kept consistent with theirs. We then calculated the AUC and unitMCC to evaluate model performance and compared the results with those of S. Lee et al. For the CD dataset, our model demonstrated comparable performance to S. Lee et al.’s approach, with no significant differences observed in AUC or unitMCC (Figure S17B). Importantly, while S. Lee et al.’s methods were specifically optimized for CRC and CD datasets, our methodology was derived from a systematic evaluation across 20 disease datasets. This highlights the broader applicability of our approach, which achieves performance on par with S. Lee et al.’s methods while demonstrating greater generalizability across diverse datasets.

## Discussion

Due to the close relationship between gut microbiome and the onset and progression of human diseases, gut microbiome can be used as biomarkers to distinguish between diseased and healthy samples,<sup>51</sup> making them valuable for constructing diagnostic models. Diagnostic models based on gut microbiome for various diseases have already demonstrated excellent performance.<sup>29</sup> However, the model construction process involves multiple steps and various methods, so it is crucial to compare different data processing and modeling techniques, evaluate various parameters, and establish a comprehensive guide for constructing diagnostic models based on gut microbiome that can be applied across multiple diseases.<sup>52</sup> This guide would provide a valuable reference for related research, greatly enhancing research efficiency.

In this study, we compared three critical aspects of model construction: the selection of data preprocessing methods, the choice of batch effect removal methods, and the selection of algorithms. We conducted modeling analysis using data from 83 case-control cohorts across 20 different diseases, including both 16S and WGS data. Models were constructed using the relative abundance of gut microbiome at the species or genus level, and both within-cohort (internal validation) and cross-cohort (external validation) performances were evaluated. We compared the internal and external validation performances of models constructed using different data preprocessing methods, batch effect removal techniques, and algorithms.

We evaluated multiple thresholds for filtering low-abundance microbes and various data normalization methods. By comparing 16 data preprocessing methods, we identified the four optimal methods, which interestingly shared common features: all included data normalization, and none involved confounding factor correction. The consistent finding that data normalization improves model performance aligns with previous research, as normalization ensures the comparability and reproducibility of results. However, the observation that confounding factors adjustment led to a decrease in AUCs contrasts with some previous reports. Gut microbiome is significantly influenced by

various factors beyond disease, such as diet,<sup>53,54</sup> drugs,<sup>55,56</sup> seasonal variations,<sup>57,58</sup> regional differences,<sup>58,59</sup> and genetic background.<sup>60,61</sup> Confounding factors adjustment is generally thought to more accurately assess the relationship between gut microbiome and disease, thereby enhancing the model's reliability and generalizability. However, our results suggest that confounding factors adjustment may reduce the AUCs because disease characteristics might be driven by potential confounders. Thus, correcting for these confounders could diminish the differences between disease and healthy groups, leading to a lower AUC.

Furthermore, we analyzed the AUC differences among 20 diseases across five disease categories (Intestinal, Autoimmune, Metabolic, Mental and Liver disease types). In internal validation, Intestinal diseases generally exhibited high median AUCs, such as CDI (0.985), IBD (0.97), and CD (0.91). Most other diseases also showed relatively high AUCs, with 13 out of 20 diseases having a median AUC above 0.7. In external validation, however, only CDI, CD, IBD, and CRC had a median AUC above 0.7 (0.975, 0.855, 0.785, and 0.745, respectively), while the others did not exceed 0.7 (Figure S18A). Therefore, it remains necessary to explore more advanced data processing methods, batch effect removal techniques, and more sophisticated algorithms to improve predictive performance for a broader range of diseases.

## Conclusion

We conducted a comprehensive comparison of various data preprocessing methods, batch effect removal techniques, algorithms, and different parameters at each step, ultimately developing an optimal, generalizable guide for constructing disease diagnostic models based on gut microbiome. This guide is applicable to multiple diseases and both 16S and WGS data types. Our workflow enhances the accuracy and applicability of the models, deepening the understanding of the relationship between gut microbiome and disease, and holds promise for providing more reliable tools for future medical diagnostics.



## Methods

### Data collection

In this study, we leveraged 83 case-control cohorts spanning 20 distinct diseases, previously curated and analyzed by M. Li *et al.*,<sup>32</sup> to systematically evaluate and refine workflows for constructing predictive models. These datasets were sourced from public databases, including MGnify (<https://www.ebi.ac.uk/metagenomics/>),<sup>62</sup> the NCBI Sequence Read Archive (SRA) (<https://www.ncbi.nlm.nih.gov/sra>),<sup>63</sup> and GMrepo(<https://gmrepo.humangut.info>).<sup>64</sup> a database focused on the human gut microbiome. These cohorts were selected for providing complete disease phenotype metadata and ensuring robust and reproducible results by including more than 15 samples in both case and control groups.

The eligible data were then categorized into two subtypes based on their sequencing strategies: 16S ribosomal RNA gene amplicon sequencing (16S) and shotgun metagenomic next-generation sequencing (WGS). Within each subtype, diseases represented by more than one cohort were included to facilitate cross-cohort comparisons. Ultimately, our analysis encompassed 83 cohorts spanning 20 diseases: 46 16S studies covering 15 diseases (3,573 cases and 2,242 controls) and 37 WGS studies covering 12 diseases (2,415 cases and 2,169 controls). Notably, 7 diseases were represented by multiple cohorts in both 16S and WGS datasets. These 20 diseases included Colorectal Cancer (CRC), Crohn Disease(CD), Irritable Bowel Syndrome(IBM), Adenoma, Ulcerative Colitis (UC), Clostridium Difficile Infection (CDI), Inflammatory Bowel Disease (IBD), Type 2 Diabetes Mellitus (T2D), Obesity, Overweight, Parkinson's Disease (PD), Autism Spectrum Disorder (ASD), Alzheimer's Disease (AD), Mild Cognitive Impairment (MCI), Chronic Fatigue Syndrome (CFS), Rheumatoid Arthritis (RA), Multiple Sclerosis (MS), Ankylosing Spondylitis (AS), Juvenile Arthritis (JA), and Non-alcoholic Fatty Liver Disease (NAFLD). Finally, we classified these 20 diseases into five categories: Intestinal, Autoimmune, Metabolic, Mental and Liver disease types, using the MeSH database and the Human Disease Ontology (DO) database,<sup>33</sup> to facilitate subsequent comparisons of model performance across different disease categories.

For final validation and methodological comparisons, we further extended our CRC and CD datasets, incorporating five additional cohorts for each disease, as guided by the work of S. Lee *et al.*,<sup>31</sup> to ensure a more rigorous and comprehensive evaluation.

### Sequencing data processing and taxonomic annotation

Raw sequencing data in this study were downloaded from the NCBI SRA<sup>63</sup> or the European Nucleotide Archive (ENA).<sup>65</sup> Trimmomatic<sup>66</sup> was employed to trim the reads, removing sequencing vectors and low-quality bases; reads shorter than 50 bp after trimming were also removed. Following these steps, we obtained the clean reads.

For WGS data, Bowtie2<sup>67</sup> was used to align the clean reads to the human reference genome (hg19), with potential human reads being removed from subsequent analysis using default parameters. Multiple sequencing runs corresponding to the same sample were merged. For the taxonomic profiling of all CRC and CD cohorts during the final validation phase, we calculated the relative abundance of taxa from the phylum to species level using MetaPhlAn3,<sup>68</sup> MetaPhlAn4,<sup>69</sup> and Kraken2<sup>48</sup> & Bracken,<sup>49</sup> with the HRGM human gut-specific microbial genome database,<sup>50</sup> all employing default settings.

### Feature filtering

We selected four thresholds for filtering low-abundance taxa for comparison: 0.001%, 0.005%, 0.01%, and 0.05%. Taxa with maximum abundance below these thresholds were removed. We then compared the AUCs of models constructed using these thresholds with the AUC of the control model, which did not filter low-abundance taxa. All other parameters in these models were kept identical to the control model, except for the threshold used for filtering. We performed the Wilcoxon rank sum test and paired tests to assess the differences between groups and observed which threshold resulted in superior model performance.

## Data rescaling

After filtering the low-abundance taxa, whether rescaling the relative abundance data affects model performance is a potential factor. We compared the AUCs of models with and without rescaling (with all other parameters held constant). We used Adonis analysis to assess whether rescaling impacted the performance of models. The normalization method we employed was min-max rescaling, which, as the name suggests, involves rescaling using the maximum and minimum values in the data column. The rescaled values are scaled to the range [0,1], calculated by subtracting the minimum value of the column from the data and then dividing by the range.

## Data normalization

We selected six data normalization methods, all available in the “normalize.features” function of the *SIAMCAT* R package:<sup>28</sup> rank.unit, rank.std, log.clr, log.std, log.unit, and std. Specifically:

- (1) rank.unit: converts features to ranks and normalizes each column by the square root of the rank sum (where the number of columns equals the number of samples).
- (2) rank.std: converts features to ranks and applies z-score normalization.
- (3) log.clr: refers to centered log-ratio transformation (with pseudocounts added).
- (4) log.std: involves log-transforming features (after adding pseudocounts) and performing z-score normalization.
- (5) log.unit: applies log transformation to features (after adding pseudocounts) and then normalizes the features or samples using different methods.
- (6) std: applies only z-score normalization.

## Identification and removal of confounding factors

The presence of confounding factors within each cohort may affect the performance of the models, such as diet,<sup>70</sup> drugs<sup>71,72</sup> and regional differences.<sup>73</sup> To investigate whether correcting for these confounding factors results in a better-performing model, we constructed both models with and

without confounding factor correction and compared their performance.

In this study, we compared two methods for correcting confounding factors: the “removeBatchEffect” function from the *limma* R package (version 3.46.0)<sup>35</sup> and the “matchit” function from the *MatchIt* R package (version 4.7.0).<sup>74</sup> For disease prediction modeling with confounding factor correction, we identified confounders within each cohort by examining all available metadata factors, such as age, gender, body mass index (BMI), disease stage, and geographical location. When correcting for confounding factors using the *limma* package, we tested for significant differences between the disease group and the control group. Fisher’s exact test was applied to qualitative variables (including gender, disease stage, and geographical location), while non-parametric Wilcoxon rank-sum tests were used for quantitative variables (including age and BMI). The “removeBatchEffect” function was used to adjust for confounding factors, which having a p-value <0.05 (with significant categorical and continuous variables included as covariates and batch parameters in the function, respectively, while other settings remained at their defaults). When using the *MatchIt* package, we employed the “matchit” function to match samples between the control and disease groups based on metadata factors such as age, gender, body mass index (BMI), disease stage, and geographical location. Confounding factor adjustment was performed for cohorts with 15 or more matched pairs.

## Batch effect removal

We also compared different methods for removing batch effects. Specifically, we evaluated the performance of the “removeBatchEffect” function from the *limma* R package,<sup>35</sup> the “adjust\_batch” function from the *MMUPHin* R package (version 1.8.2),<sup>36</sup> and the “ComBat” function from the *sva* R package (version 3.42.0).<sup>37</sup> We constructed models where the only varying parameter was the method used for batch effect removal, keeping all other parameters consistent. To quantify the effectiveness of each method in removing batch effects, we compared the AUC and calculated the proportion of variance explained ( $R^2$ )

using ADONIS analysis. Batch effect removal methods that achieve higher AUC and lower  $R^2$  was used as the optimal approach for subsequent modeling.

### Algorithm selection

In this study, we compared nine algorithms: Lasso regression,<sup>38</sup> Ridge regression,<sup>39</sup> Elastic Net,<sup>40</sup> Random Forest,<sup>41</sup> Linear support vector machine (Linear SVM),<sup>75</sup> Radial basis function support vector machine (Radial SVM),<sup>75</sup> Extreme gradient boosting (XGBoost),<sup>76</sup> Light gradient boosting machine (LightGBM)<sup>77</sup> and Neural Networks. The first four algorithms are all selectable parameters within the “train.model” function of the *SIAMCAT* R package.<sup>28</sup> The Linear SVM and Radial SVM are implemented using the “svm” function from the *e1071* R package (<https://CRAN.R-project.org/package=e1071>), XGBoost with the “xgboost” function from the *xgboost* R package (<https://CRAN.R-project.org/package=xgboost>), LightGBM with the “lgb.train” function from the *lightgbm* R package (<https://CRAN.R-project.org/package=lightgbm>), and Neural Networks with the “nnet” function from the *nnet* R package (<https://CRAN.R-project.org/package=nnet>).

- (1) Lasso regression is a linear regression method that fits data by minimizing the loss function while adding an L1 regularization term. This encourages sparsity in the model parameters, allowing it to automatically select features that significantly impact the model. This method is effective for feature selection and reduces the risk of overfitting, making it suitable for datasets with a large number of features.
- (2) Ridge regression is another linear regression method, similar to Lasso, but it uses an L2 regularization term to control the size of the model parameters rather than inducing sparsity. This helps in reducing the risk of overfitting.
- (3) Elastic Net combines both Lasso and Ridge regression by employing both L1 and L2 regularization terms, thus taking advantage of the strengths of both methods. This

approach allows for simultaneous feature selection and parameter shrinkage.

- (4) Random Forest is an ensemble learning method that constructs multiple decision trees, introducing randomness during the construction of each tree. The results of all trees are then aggregated to improve the model’s generalization ability and stability. Random Forest is known for its robustness and predictive performance, being less prone to overfitting and particularly effective for datasets with non-linear relationships.
- (5) Linear SVM is a supervised learning algorithm that aims to find the optimal hyperplane that best separates data into different classes. It does this by maximizing the margin between the classes while minimizing classification error. Linear SVM is effective for linearly separable datasets and is robust to high-dimensional feature spaces, making it a good choice for text classification and other high-dimensional problems.
- (6) Radial SVM is an extension of the Linear SVM that uses the Radial Basis Function (RBF) kernel to map data into higher-dimensional spaces. This allows it to handle non-linear relationships between features by transforming the input into a space where a linear separator can be found. Radial SVM is particularly effective for datasets with complex, non-linear decision boundaries.
- (7) XGBoost is an ensemble learning technique based on gradient boosting. It builds decision trees in a sequential manner, where each tree corrects the errors made by the previous one. XGBoost is known for its high predictive accuracy, scalability, and regularization capabilities, making it highly effective for both regression and classification tasks, particularly in large datasets with complex relationships.
- (8) LightGBM is another gradient boosting algorithm, similar to XGBoost, but optimized for faster training and lower memory usage. It uses a histogram-based approach to bin continuous features, improving efficiency in large-scale datasets. LightGBM is highly effective for large datasets and is particularly well-suited for high-dimensional data with a large number of features.

- (9) Neural Networks is a class of machine learning models inspired by the structure of the human brain. They consist of layers of interconnected neurons, where each neuron performs a simple mathematical operation. Neural Networks is particularly powerful for modeling complex non-linear relationships and can automatically learn feature representations from raw data. They are widely used in tasks such as image recognition, natural language processing, and time-series forecasting.

When partitioning the data and constructing the models, we used the parameters `num.folds = 5` and `num.resample = 3`. By comparing the performance of models built using these nine different algorithms, we aimed to identify the algorithm that yields the best model performance.

### ***The procedure for comparing the efficacy of parameters and the criteria for evaluating model performance***

In exploring the optimal workflow, our study was divided into three main components: the selection of data preprocessing methods, the choice of batch effect removal methods, and the selection of algorithms.

First, we investigated the optimal data preprocessing strategy, which encompassed four key steps: filtering low-abundance taxa, data rescaling, data normalization, and correction for confounding factors. By systematically evaluating all possible combinations of these steps and their respective parameters, we identified a robust data preprocessing workflow that was applicable across multiple diseases.

Next, in the batch effect removal step, we compared three different methods to determine the most effective approach for removing batch effects.

Finally, we assessed algorithm selection, where we combined the nine algorithms with the selected data preprocessing and batch effect removal method to identify a universally effective algorithm and workflow that yielded the best model performance.

When comparing model performance, we primarily focused on the AUCs and the significance of inter-group differences. We generated boxplots of both internal and external validation AUCs, observing trends in the boxplots, the median AUCs, and the presence of significant differences between groups. In order to assess the imbalance in sample sizes between cases and controls, we also computed AUPRC, F1score, and unitMCC to evaluate model performance: AUPRC measures the performance of binary classifiers, particularly in imbalanced datasets, focusing on precision and recall; the F1 score combines precision and recall into a single value, balancing both metrics; the unitMCC standardizes the MCC (a measure taking into account true positives, true negatives, false positives, and false negatives) by adding 1 and dividing by 2, giving a result between 0 and 1. Additionally, to examine the impact of multiple parameters on model performance and to evaluate the effectiveness of different batch effect removal methods, we conducted Adonis analysis of variance to assess the explanatory power of different grouping factors on sample variation. We calculated the R-squared ( $R^2$ ) value to evaluate the statistical significance of these effects. A higher  $R^2$  value indicated a greater explanatory power of the group for sample variation.

### ***Statistical and bioinformatics methods***

In this study, all data processing, analysis, and visualization were carried out using R (version 4.2.1, <https://www.r-project.org/>). For comparisons between two groups, we utilized the Wilcoxon rank sum test and paired tests for pairwise data as needed. For multi-group comparisons, the Kruskal-Wallis test was applied using the “`stat_compare_means`” function from the *ggpubr* R package (version 0.6.0, <https://github.com/kasambara/ggpubr>). To quantify the contribution of grouping factors to sample variation ( $R^2$  values), we employed the “`adonis2`” function from the *vegan* R package.<sup>34</sup>

### ***Disclosure statement***

No potential conflict of interest was reported by the author(s).



## Funding

This research is supported by National Key Research and Development Program of China [2024YFA0918500 to W.H.C].

## ORCID

Peikun Li  <http://orcid.org/0009-0002-2943-8826>

Wei-Hua Chen  <http://orcid.org/0000-0001-5160-4398>

## Author contributions

Wei-Hua Chen designed the study workflow. Peikun Li and Min Li wrote the code. Peikun Li performed the data analysis, prepared the figures, and drafted the manuscript. Wei-Hua Chen and Min Li revised the manuscript. All authors have read the final manuscript and approved it for publication.

## Data availability statement

These data were derived from the following resources available in the public domain: NCBI (<https://www.ncbi.nlm.nih.gov/sra>), ENA (<https://www.ebi.ac.uk/ena/browser/>), GMrepo v2 (<https://gmrepo.humangut.info>).

## Ethics statement

This study did not receive nor require ethics approval, as it reused the publicly available data.

## References

1. Flint HJ, Scott KP, Duncan SH, Louis P, Forano E. Microbial degradation of complex carbohydrates in the gut. *Gut Microbes*. 2012;3(4):289–306. doi: 10.4161/gmic.19897.
2. Belkaid Y, Hand TW. Role of the microbiota in immunity and inflammation. *Cell*. 2014;157(1):121–141. doi: 10.1016/j.cell.2014.03.011.
3. Cryan JF, Dinan TG. Mind-altering microorganisms: the impact of the gut microbiota on brain and behaviour. *Nat Rev Neurosci*. 2012;13(10):701–712. doi: 10.1038/nrn3346.
4. Marchesi JR, Adams DH, Fava F, Hermes GD, Hirschfield GM, Hold G, Quraishi MN, Kinross J, Smidt H, Tuohy KM, et al. The gut microbiota and host health: a new clinical frontier. *Gut*. 2016;65(2):330–339. doi: 10.1136/gutjnl-2015-309990.
5. Gevers D, Kugathasan S, Denson LA, Vazquez-Baeza Y, Van Treuren W, Ren B, Schwager E, Knights D, Song SJ, Yassour M, et al. The treatment-naïve microbiome in new-onset crohn's disease. *Cell Host & Microbe*. 2014;15(3):382–392. doi: 10.1016/j.chom.2014.02.005.
6. Morgan XC, Tickle TL, Sokol H, Gevers D, Devaney KL, Ward DV, Reyes JA, Shah SA, LeLeiko N, Snapper SB, et al. Dysfunction of the intestinal microbiome in inflammatory bowel disease and treatment. *Genome Biol*. 2012;13(9):R79. doi: 10.1186/gb-2012-13-9-r79.
7. Ott SJ, Musfeldt M, Wenderoth DF, Hampe J, Brant O, Folsch UR, Timmis KN, Schreiber S. Reduction in diversity of the colonic mucosa associated bacterial microflora in patients with active inflammatory bowel disease. *Gut*. 2004;53(5):685–693. doi: 10.1136/gut.2003.025403.
8. Scher JU, Sczesnak A, Longman RS, Segata N, Ubeda C, Bielski C, Rostron T, Cerundolo V, Pamer EG, Abramson SB, et al. Expansion of intestinal prevotella copri correlates with enhanced susceptibility to arthritis. *Elife*. 2013;2:e01202. doi: 10.7554/eLife.01202.
9. Hevia A, Milani C, Lopez P, Cuervo A, Arboleya S, Duranti S, Turrioni F, Gonzalez S, Suarez A, Gueimonde M, et al. Intestinal dysbiosis associated with systemic lupus erythematosus. *mBio*. 2014;5(5):e01548–01514. doi: 10.1128/mBio.01548-14.
10. Turnbaugh PJ, Hamady M, Yatsunenko T, Cantarel BL, Duncan A, Ley RE, Sogin ML, Jones WJ, Roe BA, Affourtit JP, et al. A core gut microbiome in obese and lean twins. *Nature*. 2009;457(7228):480–484. doi: 10.1038/nature07540.
11. Qin J, Li Y, Cai Z, Li S, Zhu J, Zhang F, Liang S, Zhang W, Guan Y, Shen D, et al. A metagenome-wide association study of gut microbiota in type 2 diabetes. *Nature*. 2012;490(7418):55–60. doi: 10.1038/nature11450.
12. Bedarf JR, Hildebrand F, Coelho LP, Sunagawa S, Bahram M, Goeser F, Bork P, Wullner U. Functional implications of microbial and viral gut metagenome changes in early stage L-DOPA-naïve Parkinson's disease patients. *Genome Med*. 2017;9(1):39. doi: 10.1186/s13073-017-0428-y.
13. Vogt NM, Kerby RL, Dill-McFarland KA, Harding SJ, Merluzzi AP, Johnson SC, Carlsson CM, Asthana S, Zetterberg H, Blennow K, et al. Gut microbiome alterations in alzheimer's disease. *Sci Rep*. 2017;7(1):13537. doi: 10.1038/s41598-017-13601-y.
14. Kortenien J, Karlsson L, Aatsinki A. Systematic review: autism spectrum disorder and the gut microbiota. *Acta Psychiatr Scand*. 2023;148(3):242–254. doi: 10.1111/acps.13587.
15. Patel M, Atluri LM, Gonzalez NA, Sakhamuri N, Athiyaman S, Randhi B, Gutlapalli SD, Pu J, Zaidi MF, Khan S. A systematic review of mixed studies exploring the effects of probiotics on gut-microbiome to modulate therapy in children with autism spectrum disorder. *Cureus*. 2022;14(12):e32313. doi: 10.7759/cureus.32313.
16. Lewandowska-Pietruszka Z, Figlerowicz M, Mazur-Melewska K. Microbiota in autism spectrum disorder:



- a systematic review. *Int J Mol Sci.* **2023**;24(23). doi: [10.3390/ijms242316660](https://doi.org/10.3390/ijms242316660).
17. Jiang P, Wu S, Luo Q, Zhao XM, Chen WH, Bucci V. Metagenomic analysis of common intestinal diseases reveals relationships among microbial signatures and powers multidisease diagnostic models. *mSystems.* **2021**;6(3). doi: [10.1128/mSystems.00112-21](https://doi.org/10.1128/mSystems.00112-21).
  18. Thomas AM, Manghi P, Asnicar F, Pasolli E, Armanini F, Zolfo M, Beghini F, Manara S, Karcher N, Pozzi C, et al. Metagenomic analysis of colorectal cancer datasets identifies cross-cohort microbial diagnostic signatures and a link with choline degradation. *Nat Med.* **2019**;25(4):667–678. doi: [10.1038/s41591-019-0405-7](https://doi.org/10.1038/s41591-019-0405-7).
  19. Wirbel J, Pyl PT, Kartal E, Zych K, Kashani A, Milanese A, Fleck JS, Voigt AY, Palleja A, Ponnudurai R, et al. Meta-analysis of fecal metagenomes reveals global microbial signatures that are specific for colorectal cancer. *Nat Med.* **2019**;25(4):679–689. doi: [10.1038/s41591-019-0406-6](https://doi.org/10.1038/s41591-019-0406-6).
  20. Zeller G, Tap J, Voigt AY, Sunagawa S, Kultima JR, Costea PI, Amiot A, Böhm J, Brunetti F, Habermann N, et al. Potential of fecal microbiota for early-stage detection of colorectal cancer. *Mol Syst Biol.* **2014**;10(11). doi: [10.15252/msb.20145645](https://doi.org/10.15252/msb.20145645).
  21. Gou W, Ling CW, He Y, Jiang Z, Fu Y, Xu F, Miao Z, Sun TY, Lin JS, Zhu HL, et al. Interpretable machine learning framework reveals robust gut microbiome features associated with type 2 diabetes. *Diabetes Care.* **2021**;44(2):358–366. doi: [10.2337/dc20-1536](https://doi.org/10.2337/dc20-1536).
  22. Dan Z, Mao X, Liu Q, Guo M, Zhuang Y, Liu Z, Chen K, Chen J, Xu R, Tang J, et al. Altered gut microbial profile is associated with abnormal metabolism activity of autism spectrum disorder. *Gut Microbes.* **2020**;11(5):1246–1267. doi: [10.1080/19490976.2020.1747329](https://doi.org/10.1080/19490976.2020.1747329).
  23. Costea PI, Zeller G, Sunagawa S, Pelletier E, Alberti A, Levenez F, Tramontano M, Driessen M, Hercog R, Jung FE, et al. Towards standards for human fecal sample processing in metagenomic studies. *Nat Biotechnol.* **2017**;35(11):1069–1076. doi: [10.1038/nbt.3960](https://doi.org/10.1038/nbt.3960).
  24. Ling W, Zhao N, Plantinga AM, Launer LJ, Fodor AA, Meyer KA, Wu MC. Powerful and robust non-parametric association testing for microbiome data via a zero-inflated quantile approach (ZINQ). *Microbiome.* **2021**;9(1):181. doi: [10.1186/s40168-021-01129-3](https://doi.org/10.1186/s40168-021-01129-3).
  25. McMurdie PJ, Holmes S, McHardy AC. Waste not, want not: why rarefying microbiome data is inadmissible. *PLOS Comput Biol.* **2014**;10(4):e1003531. doi: [10.1371/journal.pcbi.1003531](https://doi.org/10.1371/journal.pcbi.1003531).
  26. Weiss S, Xu ZZ, Peddada S, Amir A, Bittinger K, Gonzalez A, Lozupone C, Zaneveld JR, Vazquez-Baeza Y, Birmingham A, et al. Normalization and microbial differential abundance strategies depend upon data characteristics. *Microbiome.* **2017**;5(1):27. doi: [10.1186/s40168-017-0237-y](https://doi.org/10.1186/s40168-017-0237-y).
  27. Vittinghoff E, McCulloch CE. Relaxing the rule of ten events per variable in logistic and cox regression. *Am J Epidemiol.* **2007**;165(6):710–718. doi: [10.1093/aje/kwk052](https://doi.org/10.1093/aje/kwk052).
  28. Wirbel J, Zych K, Essex M, Karcher N, Kartal E, Salazar G, Bork P, Sunagawa S, Zeller G. Microbiome meta-analysis and cross-disease comparison enabled by the SIAMCAT machine learning toolbox. *Genome Biol.* **2021**;22(1). doi: [10.1186/s13059-021-02306-1](https://doi.org/10.1186/s13059-021-02306-1).
  29. Pasolli E, Truong DT, Malik F, Waldron L, Segata N. Machine learning meta-analysis of large metagenomic datasets: tools and biological insights. *PLOS Comput Biol.* **2016**;12(7):e1004977. doi: [10.1371/journal.pcbi.1004977](https://doi.org/10.1371/journal.pcbi.1004977).
  30. Ghannam RB, Techtman SM. Machine learning applications in microbial ecology, human microbiome studies, and environmental monitoring. *Comput Struct Biotechnol J.* **2021**;19:1092–1107. doi: [10.1016/j.csbj.2021.01.028](https://doi.org/10.1016/j.csbj.2021.01.028).
  31. Lee S, Lee I. Comprehensive assessment of machine learning methods for diagnosing gastrointestinal diseases through whole metagenome sequencing data. *Gut Microbes.* **2024**;16(1):2375679. doi: [10.1080/19490976.2024.2375679](https://doi.org/10.1080/19490976.2024.2375679).
  32. Li M, Liu J, Zhu J, Wang H, Sun C, Gao NL, Zhao X-M, Chen W-H. Performance of gut microbiome as an independent diagnostic tool for 20 diseases: cross-cohort validation of machine-learning classifiers. *Gut Microbes.* **2023**;15(1). doi: [10.1080/19490976.2023.2205386](https://doi.org/10.1080/19490976.2023.2205386).
  33. Schriml LM, Munro JB, Schor M, Olley D, McCracken C, Felix V, Baron JA, Jackson R, Bello Susan S, Bearer C, et al. The human disease ontology 2022 update. *Nucleic Acids Res.* **2022**;50(D1):D1255–D1261. doi: [10.1093/nar/gkab1063](https://doi.org/10.1093/nar/gkab1063).
  34. Oksanen J, Guillaume Blanchet F, Kindt R, Legendre P, O'Hara RG, Simpson G, Solymos P, Stevens H, Wagner H. Multivariate analysis of ecological communities in R: vegan tutorial R package version 1.7 1–43 <https://cran.r-project.org/package=vegan>. **2013**
  35. Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK. Limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.* **2015**;43(7):e47–e47. doi: [10.1093/nar/gkv007](https://doi.org/10.1093/nar/gkv007).
  36. Ma S, Shungin D, Mallick H, Schirmer M, Nguyen LH, Kolde R, Franzosa E, Vlamakis H, Xavier R, Huttenhower C. Population structure discovery in meta-analyzed microbial communities and inflammatory bowel disease using MMUPHin. *Genome Biol.* **2022**;23(1). doi: [10.1186/s13059-022-02753-4](https://doi.org/10.1186/s13059-022-02753-4).
  37. Leek JT, Johnson WE, Parker HS, Jaffe AE, Storey JD. The sva package for removing batch effects and other unwanted variation in high-throughput experiments.

- Bioinformatics. 2012;28(6):882–883. doi: [10.1093/bioinformatics/bts034](https://doi.org/10.1093/bioinformatics/bts034).
38. Tibshirani R. Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B: Stat Methodol.* 1996;58(1):267–288. doi: [10.1111/j.2517-6161.1996.tb02080.x](https://doi.org/10.1111/j.2517-6161.1996.tb02080.x).
  39. Goldstein M, Smith AFM. Ridge-type estimators for regression analysis. *J R Stat Soc Ser B: Stat Methodol.* 1974;36(2):284–291. doi: [10.1111/j.2517-6161.1974.tb01006.x](https://doi.org/10.1111/j.2517-6161.1974.tb01006.x).
  40. Zou H, Hastie T. Regularization and variable selection via the elastic net. *J R Stat Soc Ser B: Stat Methodol.* 2005;67(2):301–320. doi: [10.1111/j.1467-9868.2005.00503.x](https://doi.org/10.1111/j.1467-9868.2005.00503.x).
  41. Tk H. Random decision forests. *Proc 3rd Int Conf Doc Anal Recognit* 1 278–282 doi:[10.1109/ICDAR.1995.598994](https://doi.org/10.1109/ICDAR.1995.598994). 1995.
  42. Topcuoglu BD, Lesniak NA, Ruffin MT, Wiens J, Schloss PD, Blaser MJ. A framework for effective application of machine learning to microbiome-based classification problems. *mBio.* 2020;11(3). doi: [10.1128/mBio.00434-20](https://doi.org/10.1128/mBio.00434-20).
  43. Bakir-Gungor B, Hacilar H, Jabeer A, Nalbantoglu OU, Aran O, Yousef M. Inflammatory bowel disease biomarkers of human gut microbiota selected via different feature selection methods. *PeerJ.* 2022;10:e13205. doi: [10.7717/peerj.13205](https://doi.org/10.7717/peerj.13205).
  44. Chang CC, Liu TC, Lu CJ, Chiu HC, Lin WN. Machine learning strategy for identifying altered gut microbiomes for diagnostic screening in myasthenia gravis. *Front Microbiol.* 2023;14:1227300. doi: [10.3389/fmicb.2023.1227300](https://doi.org/10.3389/fmicb.2023.1227300).
  45. Zeng F, Su X, Liang X, Liao M, Zhong H, Xu J, Gou W, Zhang X, Shen L, Zheng JS, et al. Gut microbiome features and metabolites in non-alcoholic fatty liver disease among community-dwelling middle-aged and older adults. *BMC Med.* 2024;22(1):104. doi: [10.1186/s12916-024-03317-y](https://doi.org/10.1186/s12916-024-03317-y).
  46. Liu L, Liang L, Luo Y, Han J, Lu D, Cai R, Sethi G, Mai S. Unveiling the power of gut microbiome in predicting neoadjuvant immunochemotherapy responses in esophageal squamous cell carcinoma. *Res (Wash DC).* 2024;7:0529. doi: [10.34133/research.0529](https://doi.org/10.34133/research.0529).
  47. Wu X, Zhang T, Zhang T, Park S. The impact of gut microbiome enterotypes on ulcerative colitis: identifying key bacterial species and revealing species co-occurrence networks using machine learning. *Gut Microbes.* 2024;16(1):2292254. doi: [10.1080/19490976.2023.2292254](https://doi.org/10.1080/19490976.2023.2292254).
  48. Wood DE, Lu J, Langmead B. Improved metagenomic analysis with Kraken 2. *Genome Biol.* 2019;20(1):257. doi: [10.1186/s13059-019-1891-0](https://doi.org/10.1186/s13059-019-1891-0).
  49. Lu J, Breitwieser FP, Thielen P, Salzberg SL. Bracken: estimating species abundance in metagenomics data. *PeerJ Comput Sci.* 2017; 3. doi: [10.7717/peerj-cs.104](https://doi.org/10.7717/peerj-cs.104).
  50. Kim CY, Lee M, Yang S, Kim K, Yong D, Kim HR, Lee I. Human reference gut microbiome catalog including newly assembled genomes from under-represented Asian metagenomes. *Genome Med.* 2021;13(1):134. doi: [10.1186/s13073-021-00950-7](https://doi.org/10.1186/s13073-021-00950-7).
  51. Zhernakova A, Kurilshikov A, Bonder MJ, Tigchelaar EF, Schirmer M, Vatanen T, Mujagic Z, Vila AV, Falony G, Vieira-Silva S, et al. Population-based metagenomics analysis reveals markers for gut microbiome composition and diversity. *Science.* 2016;352(6285):565–569. doi: [10.1126/science.aad3369](https://doi.org/10.1126/science.aad3369).
  52. Knight R, Vrbanac A, Taylor BC, Aksenov A, Callewaert C, Debelius J, Gonzalez A, Kosciolek T, McCall L-I, McDonald D, et al. Best practices for analysing microbiomes. *Nat Rev Microbiol.* 2018;16(7):410–422. doi: [10.1038/s41579-018-0029-9](https://doi.org/10.1038/s41579-018-0029-9).
  53. David LA, Maurice CF, Carmody RN, Gootenberg DB, Button JE, Wolfe BE, Ling AV, Devlin AS, Varma Y, Fischbach MA, et al. Diet rapidly and reproducibly alters the human gut microbiome. *Nature.* 2014;505(7484):559–563. doi: [10.1038/nature12820](https://doi.org/10.1038/nature12820).
  54. De Filippo C, Cavalieri D, Di Paola M, Ramazzotti M, Poullet JB, Massart S, Collini S, Pieraccini G, Lionetti P. Impact of diet in shaping gut microbiota revealed by a comparative study in children from Europe and rural Africa. *Proc Natl Acad Sci USA.* 2010;107(33):14691–14696. doi: [10.1073/pnas.1005963107](https://doi.org/10.1073/pnas.1005963107).
  55. Maurice CF, Haiser HJ, Turnbaugh PJ. Xenobiotics shape the physiology and gene expression of the active human gut microbiome. *Cell.* 2013;152(1–2):39–50. doi: [10.1016/j.cell.2012.10.052](https://doi.org/10.1016/j.cell.2012.10.052).
  56. Forslund K, Hildebrand F, Nielsen T, Falony G, Le Chatelier E, Sunagawa S, Prifti E, Vieira-Silva S, Gudmundsdottir V, Pedersen HK, et al. Disentangling type 2 diabetes and metformin treatment signatures in the human gut microbiota. *Nature.* 2015;528(7581):262–266. doi: [10.1038/nature15766](https://doi.org/10.1038/nature15766).
  57. Davenport ER, Mizrahi-Man O, Michelini K, Barreiro LB, Ober C, Gilad Y, Quintana-Murci L. Seasonal variation in human gut microbiome composition. *PLOS ONE.* 2014;9(3):e90731. doi: [10.1371/journal.pone.0090731](https://doi.org/10.1371/journal.pone.0090731).
  58. Smits SA, Leach J, Sonnenburg ED, Gonzalez CG, Lichtman JS, Reid G, Knight R, Manjurano A, Changalucha J, Elias JE, et al. Seasonal cycling in the gut microbiome of the hadza hunter-gatherers of Tanzania. *Science.* 2017;357(6353):802–806. doi: [10.1126/science.aan4834](https://doi.org/10.1126/science.aan4834).
  59. Yatsunenko T, Rey FE, Manary MJ, Trehan I, Dominguez-Bello MG, Contreras M, Magris M, Hidalgo G, Baldassano RN, Anokhin AP, et al. Human gut microbiome viewed across age and geography. *Nature.* 2012;486(7402):222–227. doi: [10.1038/nature11053](https://doi.org/10.1038/nature11053).
  60. Goodrich JK, Waters JL, Poole AC, Sutter JL, Koren O, Blekhnman R, Beaumont M, Van Treuren W, Knight R, Bell JT, et al. Human genetics shape the gut microbiome. *Cell.* 2014;159(4):789–799. doi: [10.1016/j.cell.2014.09.053](https://doi.org/10.1016/j.cell.2014.09.053).

61. Bonder MJ, Kurilshikov A, Tigchelaar EF, Mujagic Z, Imhann F, Vila AV, Deelen P, Vatanen T, Schirmer M, Smeekens SP, et al. The effect of host genetics on the gut microbiome. *Nat Genet.* 2016;48(11):1407–1412. doi: [10.1038/ng.3663](https://doi.org/10.1038/ng.3663).
62. Mitchell AL, Almeida A, Beracochea M, Boland M, Burgin J, Cochrane G, Crusoe MR, Kale V, Potter SC, Richardson LJ, et al. Mgnify: the microbiome analysis resource in 2020. *Nucleic Acids Res.* 2019; doi: [10.1093/nar/gkz1035](https://doi.org/10.1093/nar/gkz1035).
63. Katz K, Shutov O, Lapoint R, Kimelman M, Brister JR, O'Sullivan C: the sequence read archive: a decade more of explosive growth. *Nucleic Acids Res.* 2022;50(D1): D387–D390. doi: [10.1093/nar/gkab1053](https://doi.org/10.1093/nar/gkab1053).
64. Dai D, Zhu J, Sun C, Li M, Liu J, Wu S, Ning K, He L-J, Zhao X-M, Chen W-H. Gmrepo v2: a curated human gut microbiome database with special focus on disease markers and cross-dataset comparison. *Nucleic Acids Res.* 2022;50(D1):D777–D784. doi: [10.1093/nar/gkab1019](https://doi.org/10.1093/nar/gkab1019).
65. Harrison PW, Ahamed A, Aslam R, Alako BTF, Burgin J, Buso N, Courtot M, Fan J, Gupta D, Haseeb M, et al. The European nucleotide archive in 2020. *Nucleic Acids Res.* 2021;49(D1):D82–D85. doi: [10.1093/nar/gkaa1028](https://doi.org/10.1093/nar/gkaa1028).
66. Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for illumina sequence data. *Bioinformat.* 2014;30(15):2114–2120. doi: [10.1093/bioinformatics/btu170](https://doi.org/10.1093/bioinformatics/btu170).
67. Langmead B, Salzberg SL. Fast gapped-read alignment with bowtie 2. *Nat Methods.* 2012;9(4):357–359. doi: [10.1038/nmeth.1923](https://doi.org/10.1038/nmeth.1923).
68. Beghini F, McIver LJ, Blanco-Míguez A, Dubois L, Asnicar F, Maharjan S, Mailyan A, Manghi P, Scholz M, Thomas AM, et al. Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. *eLife.* 2021; 10. doi: [10.7554/eLife.65088](https://doi.org/10.7554/eLife.65088).
69. Blanco-Míguez A, Beghini F, Cumbo F, McIver LJ, Thompson KN, Zolfo M, Manghi P, Dubois L, Huang KD, Thomas AM, et al. Extending and improving metagenomic taxonomic profiling with uncharacterized species using MetaPhlAn 4. *Nat Biotechnol.* 2023;41(11):1633–1644. doi: [10.1038/s41587-023-01688-w](https://doi.org/10.1038/s41587-023-01688-w).
70. Wu GD, Chen J, Hoffmann C, Bittinger K, Chen YY, Keilbaugh SA, Bewtra M, Knights D, Walters WA, Knight R, et al. Linking long-term dietary patterns with gut microbial enterotypes. *Science.* 2011;334(6052):105–108. doi: [10.1126/science.1208344](https://doi.org/10.1126/science.1208344).
71. Vujkovic-Cvijin I, Sklar J, Jiang L, Natarajan L, Knight R, Belkaid Y. Host variables confound gut microbiota studies of human disease. *Nature.* 2020;587(7834):448–454. doi: [10.1038/s41586-020-2881-9](https://doi.org/10.1038/s41586-020-2881-9).
72. Forslund SK, Chakaroun R, Zimmermann-Kogadeeva M, Marko L, Aron-Wisnewsky J, Nielsen T, Moitinho-Silva L, Schmidt TSB, Falony G, Vieira-Silva S, et al. Combinatorial, additive and dose-dependent drug-microbiome associations. *Nature.* 2021;600(7889):500–505. doi: [10.1038/s41586-021-04177-9](https://doi.org/10.1038/s41586-021-04177-9).
73. He Y, Wu W, Zheng H-M, Li P, McDonald D, Sheng H-F, Chen M-X, Chen Z-H, Ji G-Y, Zheng Z-D-X, et al. Regional variation limits applications of healthy gut microbiome reference ranges and disease models. *Nat Med.* 2018;24(10):1532–1535. doi: [10.1038/s41591-018-0164-x](https://doi.org/10.1038/s41591-018-0164-x).
74. Stuart DE. HaKlaGKaEA: MatchIt: nonparametric pre-processing for parametric causal inference. *J Stat Softw.* 2011;42:1–28. doi: [10.18637/jss.v042.i08](https://doi.org/10.18637/jss.v042.i08).
75. Hearst MA, Dumais ST, Osuna E, Platt J, Scholkopf B. Support vector machines. *IEEE Intell Syst Their Appl.* 1998;13(4):18–28. doi: [10.1109/5254.708428](https://doi.org/10.1109/5254.708428).
76. Guestrin T. Xgboost: a scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* New York, USA: ACM; 2016.
77. Liu G, Ka Q, Ma T, Fa T, Wa W, Ca W, Ma Q, Ya T-Y. LightGBM: a highly efficient gradient boosting decision tree (New York: Curran Associates, Inc.). 2017.