Investigating the Association between Metagenomic Shotgun Data and the Occurrence of IBD Using Machine Learning Models

Ziyu Liu

Mander Portman Woodward, London, The United Kingdom
Email: 15010638483@163.com

***Abstract*—Inflammatory Bowel Disease (IBD) is becoming increasingly common around the world, and changing the gut microbiome has potential to be useful for studying and diagnosing the disease. In this work, we analyzed bacterial abundance data from a previously published shotgun sequencing study [1] with three machine learning classifiers: L2 logistic regression, LASSO logistic regression, and Support Vector Machine (SVM). All the classifiers were trained to distinguish the samples obtained from patients with Ulcerative Colitis (UC) from the healthy population samples. We also studied the 15 bacterial species that contributed the most to each classifier. The three AUROC values were 0.7980 for L2 logistic regression, 0.6970 for LASSO, and 0.8182 for SVM. *Salmonella enterica* serovar Typhimurium was associated with prediction of IBD, whereas members of the *genus Lactobacillus* had a beneficial effect on IBD. These observations are in line with results from previously reported IBD studies. However, SVM produced the highest AUROC out of the three models and was therefore the most accurate model on this dataset. Overall, these results illustrate how machine learning can be used to explore microbiome data for disease prediction and possible biomarker detection. They also show how a nonlinear model like SVM might reveal relationships that might not have been uncovered with a linear model.**

***Keywords*—inflammatory bowel disease, machine learning, disease biomarkers, bacterial abundance**

I. Introduction

Inflammatory Bowel Disease (IBD) comprises both Crohn's disease and ulcerative colitis. Typical symptoms include diarrhoea, cramping and fatigue [1]. The prevalence of IBD worldwide is increasing, with estimates for affected people of approximately 6.9 million [2]. In 2021, an estimated 3,830,119 people had IBD, and nearly 42,423 deaths were attributable to this disease [3]. One possible explanation for the increasing prevalence of IBD is that an industrialized lifestyle and environment leads to changes in gut microbiota composition [4], which cause inflammation, ulceration and disruption of gut homeostasis. Changes in the gut microbiota can lead to gut dysbiosis and have been associated with IBD development [5].

As more people are being diagnosed with IBD, new ways are needed to explore disease and predict it. Statistical and computational methods have already been used to study disease risk and associations with IBD risk factors [6]. For example, in a previous study, consistent gut-microbiome patterns were found in children with ulcerative colitis [6]. Metagenomic sequencing has been used before to identify organisms in polymicrobial infections for which culture was not possible [7]. Shotgun sequencing provides both taxonomic and functional data, facilitating comparison of the healthy and diseased microbiome [8]. Such information can be used by scientists to detect bacteria that are relevant for IBD. Model performance can also be measured with factors such as AUROC. Zuo *et al*. [6] obtained an AUROC of 0.763 for a Random Forest model on shotgun-sequencing data for prediction of IBD, for example.

So what does Zuo *et al*.'s [6] result tell us? It suggests that bacterial abundance data can supply useful information to machine learning models. Other studies have also tried machine learning for clinical and analytical applications in IBD [9]. Cohen-Mekelburg *et al*.' study has discussed how AI can do complex analytical tasks at a high level of accuracy [10]. Nguyen *et al*.'s [11] study has also reported that some tree-based algorithms perform better than traditional statistical methods in a specific risk-prediction task. Taken together, they support further application of machine learning and metagenomic data in IBD research.

One issue with machine-learning models is that they can have different results even if they are trained on the same data. Direct comparisons can therefore be useful to understand the advantages and disadvantages of the models. For this study, we tested several model types using the same species-level bacterial abundance data generated by Zuo *et al*. [6]. We used the processed abundance table because the computational resources we had were not sufficient to process the raw sequencing reads. Although the dataset is small, it was sufficient to compare the models and study their outputs in an accessible way. All analyses were done in Python with freely available packages.

Manuscript received September 10, 2026; accepted October 17, 2026; published November 17, 2026.

II. Methods

A. Data Preparation

Our data were from the 'count_table_shotgun.csv' file published by Zuo *et al.* [6]. The original paper included 19 faecal samples from children with moderate UC and 23 samples from healthy controls. Participants were aged 7–21 years, and the two groups did not differ significantly in age or gender. Both the 16S rRNA gene (v4) of bacterial sequences and the metagenome were sequenced. Faecal DNA was extracted with the QIAamp PowerFecal DNA Kit. The V4 region was then amplified by PCR and sequenced on an Illumina MiSeq Reagent Kit v2 with 150-bp paired-end reads. Metagenome libraries were produced with a Nextera XT DNA Library Preparation Kit and Nextera XT Index v2 Kits A and B. The samples were checked for quality with an Agilent High Sensitivity DNA Bioanalyzer chip.

Trim Galore was used for quality filtering, and KneadData was used to filter out reads derived from the human host. 16S rRNA data were processed in QIIME 2 [12], denoised with DADA2 [13], and were grouped as amplicon sequence variants. The shotgun data were analysed on cleaned reads by Centrifuge and HUMAnN2 using the UniRef90 EC gene database. Mapping reads to microbial genomes and gene families resulted in microbial species abundance and microbial pathway abundance, respectively. For our analyses, we divided the read count for each bacterial genome by the sum of bacterial reads in the same sample. This normalization was necessary as sequencing depth was not the same for each sample. Relative abundances were calculated using the Eq. (1):

$$N_{ij} = \frac{C_{ij}}{\sum_{j=1}^{n} C_{ij}} \quad (1)$$

where i is an individual sample, and j refers to a bacterial species. C_{ij} is the number of raw 150-bp reads from sample (i) mapped to the bacterial genome of species (j), and N_{ij} is the resulting relative abundance.

Sample labels were retrieved from the "metadata.csv" file provided by Zuo *et al.* [6]. The 19 UC samples were labelled 0, and 23 healthy samples were labelled 1. There were 42 samples in total. The 42 samples were divided into the training set and testing set before models were fitted. The training set that is randomly picked includes 10 UC and 12 healthy samples. The remaining 9 UC samples and 11 healthy samples are used for testing. Finally, the numbers of groups are listed in Table I.

TABLE I. Number of Training and Testing Individuals in Each Category

Category	UC	Healthy	Total
Training	10	12	22
Testing	9	11	20
Total	19	23	42

Code Block 1. Data preparation, loading, and standardization.

```
# ========== 1. Feature Standardization ==========
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)  # Use training set parameters


# ========== 2. Load Data ==========
train_info = pd.read_csv('data/selected_samples_UC_Healthy_Training.csv')
test_info = pd.read_csv('data/selected_samples_testing.csv')
abundance_data = pd.read_csv('data/normalised_data_fixed.csv')


# ========== 3. Match Samples ==========
def match_samples(info_df, abundance_data):
    """Match sample IDs and return feature matrix and labels"""
    y = (info_df['SampleType'] == 'UC').astype(int).values
    sample_ids = info_df['shotgun.SampleID'].str.replace('.', '-',
        regex=False).values
    abundance_ids = abundance_data['Sample ID'].values
    X_all = abundance_data.iloc[:, 1:].values
    bacteria_names = abundance_data.columns[1:].tolist()

    X_list = []
    y_list = []
    matched_ids = []

    for i, sample_id in enumerate(sample_ids):
        if sample_id in abundance_ids:
            idx = list(abundance_ids).index(sample_id)
            X_list.append(X_all[idx])
            y_list.append(y[i])
            matched_ids.append(sample_id)
    return np.array(X_list), np.array(y_list), matched_ids,
bacteria_names

# Match training and test sets
X_train, y_train, train_matched, bacteria_names =
match_samples(train_info,
    abundance_data)
X_test, y_test, test_matched, _ = match_samples(test_info,
abundance_data)
```

B. Logistic Regression Model

First, we fitted a logistic regression model to the training data to predict whether each sample belonged to the UC or healthy group. The probability that individual i belongs to the UC group is:

$$P_i(UC) = \left[1 + exp\left(-\beta_0 - \sum_{j=1}^{n} \beta_j N_{ij}\right)\right]^{-1} \quad (2)$$

where n is the number of bacterial genomes considered features, and β_j is the coefficient associated with the relative abundance of species j. $P_i(UC)$ is the probability modelled for the sample i belonging to the UC group. The model was implemented in Python using scikit-learn. The probability expression was transformed into log-odds with the following expression:

$$logit(P_i) = ln\left(\frac{P_i}{1-P_i}\right) = \beta_0 + \sum_{j=1}^{n} \beta_j N_{ij} \quad (3)$$

where $logit(P_i)$ is the log-odds that the sample i belongs to the UC group instead of the healthy one.

First, we tried logistic regression with L2 regularization (also called ridge logistic regression [14]). Its coefficients, β_j, were obtained by minimizing the following objective function [15].

$$L_{log} + \lambda \sum_{j=1}^{n} \beta_j^2 \quad (4)$$

where λ is the regularization parameter. The larger λ, the stronger the regularization, and the closer the coefficients are forced to zero. The term L_{log} is the logistic loss function defined as:

$$L_{log} = -\sum_{i=1}^{m} [y_i \, ln(P_i) + (1 - y_i) ln \, (1 - P_i)] \quad (5)$$

where m is the number of training samples (22 in this work), y_i is the sample class predicted to be sample i.

We also tried LASSO, or L1 regularization, because the abundance table contained a large number of bacterial features, relative to the number of samples. L1 regularization could drive uninformative coefficients to 0. The coefficients were obtained by minimizing the following objective function:

$$L_{log} + \lambda \sum_{j=1}^{n} |\beta_j| \quad (6)$$

Code Block 2. L2 logistic regression applied to the testing dataset.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import roc_auc_score, roc_curve, auc, confusion_matrix,
    classification_report
import warnings
warnings.filterwarnings('ignore')

plt.rcParams['font.sans-serif'] = ['Arial Unicode MS', 'Helvetica', 'Arial']
plt.rcParams['axes.unicode_minus'] = False


# ========== 1. Train Full Model on Training Set ==========
full_model = LogisticRegression(
    penalty='l2',
    C=1.0,
    solver='lbfgs',
    max_iter=1000,
    class_weight='balanced',
    random_state=42
)
full_model.fit(X_train_scaled, y_train)


# ========== 2. Predict on Test Set ==========
# Predict test set
y_test_pred_proba = full_model.predict_proba(X_test_scaled)[:, 1]
y_test_pred = full_model.predict(X_test_scaled)

# Calculate test set AUC
test_auc = roc_auc_score(y_test, y_test_pred_proba)


# ========== 3. Get Top 15 Bacteria from Training Set (for reference)
    ==========
# Get feature importance from full model
train_coefficients = full_model.coef_[0]

# Create DataFrame
train_importance = pd.DataFrame({
    'bacteria': bacteria_names,
    'coefficient': train_coefficients,
    'abs_coefficient': np.abs(train_coefficients),

}).sort_values('abs_coefficient', ascending=False)

# Take top 15 from training set
top_15_train = train_importance.head(15).copy()


# ========== 4. Calculate Each Bacterium's Predictive Power on Test Set
    ==========
# Store each bacterium's information
bacteria_test_performance = []

for i, bacteria in enumerate(bacteria_names):
    # Train model using only this feature
    X_train_single = X_train_scaled[:, i].reshape(-1, 1)
    X_test_single = X_test_scaled[:, i].reshape(-1, 1)

    # Train univariate model
    single_model = LogisticRegression(
        penalty='l2',
        C=1.0,
        solver='lbfgs',
        class_weight='balanced',
        random_state=42
    )

    try:
        single_model.fit(X_train_single, y_train)
        y_test_pred_single = single_model.predict_proba(X_test_single)[:, 1]
        test_auc_single = roc_auc_score(y_test, y_test_pred_single)

        # Get coefficient from training (indicates direction)
        train_coef = single_model.coef_[0][0]

        bacteria_test_performance.append({
            'bacteria': bacteria,
            'test_auc': test_auc_single,
            'train_coef': train_coef,

        })
    except Exception as e:
        # If training fails
        bacteria_test_performance.append({
            'bacteria': bacteria,
            'test_auc': 0.5,
            'train_coef': 0,
            'direction': 'no effect'
        })
    if (i + 1) % 100 == 0:
        print(f"  Processed {i+1}/{len(bacteria_names)} bacteria")


# ========== 5. Find Top 15 Bacteria by Test Set Predictive Power ==========
# Convert to DataFrame and sort by test AUC
test_importance = pd.DataFrame(bacteria_test_performance)
test_importance = test_importance.sort_values('test_auc', ascending=False)

# Take top 15
top_15_test = test_importance.head(15).copy()
```

Code Block 3. LASSO logistic regression with cross-validation.

```
# The rest code is the same as the previous code block.
# ========== 1. Find Best LASSO Parameters with Cross-Validation ==========
param_grid = {'C': [0.001, 0.01, 0.05, 0.1, 0.5, 1, 5, 10]}
lasso_cv = LogisticRegression(
    penalty='l1',
    solver='saga',
```

```
    max_iter=2000,
    class_weight='balanced',
    random_state=42
)

grid_search = GridSearchCV(lasso_cv, param_grid, cv=5,
scoring='roc_auc')
grid_search.fit(X_train_scaled, y_train)

best_C = grid_search.best_params_['C']
best_cv_auc = grid_search.best_score_


# ========== 2. Train Final Model with Best C Value
==========
model = LogisticRegression(
    penalty='l1',
    C=best_C,
    solver='saga',
    max_iter=2000,
    class_weight='balanced',
    random_state=42
)

model.fit(X_train_scaled, y_train)

# Check Lasso feature selection effect
n_zero_coef = sum(np.abs(model.coef_[0]) < 1e-10)
n_nonzero = sum(np.abs(model.coef_[0]) >= 1e-10)
```

C. *Support Vector Machine Model*

Because the bacterial abundance data matrix had more features (bacterial species) than samples, we also applied a Support Vector Machine (SVM). SVM is feature-scaling sensitive and captures non-linear relationships between bacterial species and IBD status. The model maximizes the separation between healthy and UC samples by solving the following optimization problem:

$$y_i(W^T x_i + b) \geq 1, 1 = 1,2, \ldots, m \quad (7)$$

$$\min_{w,b} \frac{1}{2}\|W\|^2 \quad (8)$$

where y_i is the class label (−1 or 1 for UC and healthy, respectively), and x_i is the bacterial abundance vector of sample i, where i = 1, 2, 3,…, m. The vector w is normal to the separating hyperplane, and b is a bias term determining the position of the hyperplane relative to the origin along w.

Because our bacterial samples contain many features that might have non-linear effects on either UC or the healthy classification, we chose an SVM with an RBF (radial basis function) kernel to model non-linear effects and maximize separation between the two classes. The RBF kernel is defined by:

$$K(x, x_i) = exp(-\gamma\|x - x_i\|^2) \quad (9)$$

In the above equation, γ is a kernel parameter that controls the region of influence of each sample on the dividing line. When γ is large, each sample influences the dividing line only in a small region of space, and samples become similar only when they are close in the original feature space. x and x_i are the two feature vectors to be mapped by the kernel into the feature space of higher dimensionality.

Code Block 4. Support vector machine with an RBF kernel.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import roc_auc_score, roc_curve,
confusion_matrix,
    classification_report
from sklearn.model_selection import GridSearchCV
import warnings
warnings.filterwarnings('ignore')

plt.rcParams['font.sans-serif'] = ['Arial Unicode MS',
'Helvetica', 'Arial']
plt.rcParams['axes.unicode_minus'] = False


# ========== 1. GridSearchCV to Find Best Parameters
==========
param_grid = {
'C': [0.01, 0.1, 1, 10, 100],
    'gamma': ['scale', 'auto', 0.01, 0.1, 1]
}


svm = SVC(
    kernel='rbf',
    class_weight='balanced',
    probability=True,
    random_state=42
)

grid_search = GridSearchCV(
    svm,
    param_grid,
cv=5,
scoring='roc_auc',
n_jobs=−1,
verbose=1
)

grid_search.fit(X_train_scaled, y_train)

best_C = grid_search.best_params_['C']
best_gamma = grid_search.best_params_['gamma']
best_cv_auc = grid_search.best_score_

results_df = pd.DataFrame(grid_search.cv_results_)
results_df_sorted = results_df.sort_values('mean_test_score',
    ascending=False).head(10)
for idx, row in results_df_sorted.iterrows():
    AUC={row['mean_test_score']:.4f}                    (+/−
{row['std_test_score']:.4f})")


# ========== 2. Train Final Model with Best Parameters
==========
final_svm = SVC(
    kernel='rbf',
    C=best_C,
    gamma=best_gamma,
    class_weight='balanced',
    probability=True,
    random_state=42
)

final_svm.fit(X_train_scaled, y_train)


# ========== 3. Predict ==========
y_train_pred_proba                                      =
```

```
final_svm.predict_proba(X_train_scaled)[:, 1]
y_test_pred_proba = final_svm.predict_proba(X_test_scaled)[:, 1]
train_auc = roc_auc_score(y_train, y_train_pred_proba)
test_auc = roc_auc_score(y_test, y_test_pred_proba)
```

III. Results

A. Logistic Regression Results

When we applied the L2 logistic regression model to the testing dataset, it achieved an AUROC of 0.7980 with C = 1 (Fig. 1). This means that the L2 model has great predictive ability for IBD status.

We also investigated the 15 bacterial species with the highest impact on the prediction. The green lines at the original model output indicated a positive association with classifying as healthy. All 15 bacterial species have similarly good predictive ability (AUROC ranging from 0.9495 to 1) (Fig. 2). This might suggest that these species are related to intestinal health (also, they were more abundant in healthy subjects).

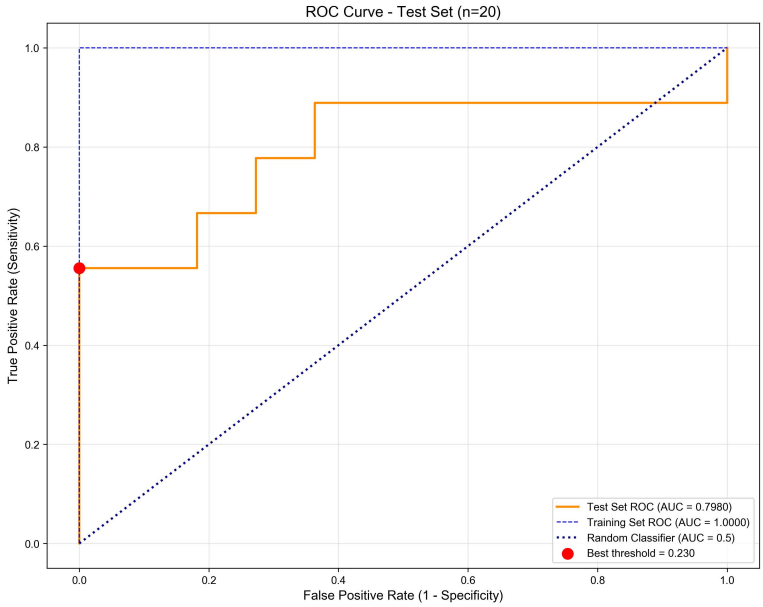


Fig. 1. Receiver operating characteristic curve for the L2 logistic regression model.

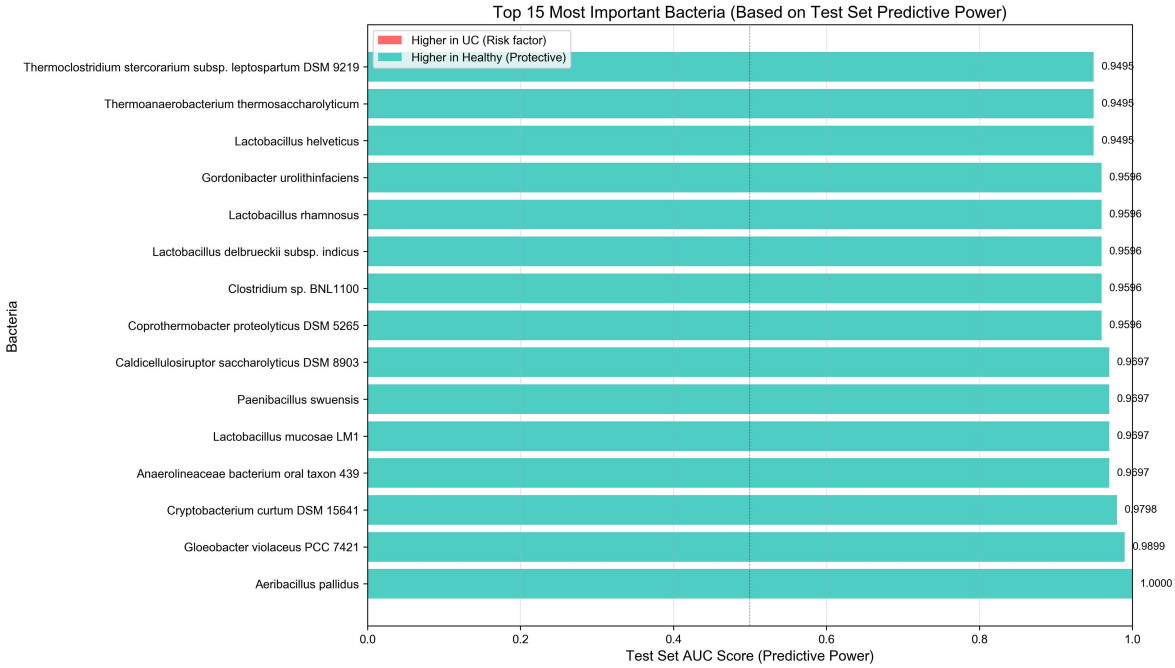


Fig. 2. Top 15 bacterial species identified by the L2 logistic regression model.

In contrast, the AUROC decreased after LASSO regularization was introduced. LASSO may be worse for this dataset because of the small sample size, which made it hard to select the optimal C value, likely leading to overfitting (Fig. 3).

After LASSO regularisation, the top 15 most important bacterial species changed significantly. *Pasteurella*

multocida subsp. *multocida* str (Fig. 4). Pm70 was the only bacterial species with positive correlation with UC prediction. In this model, the C value was selected by GridSearchCV.

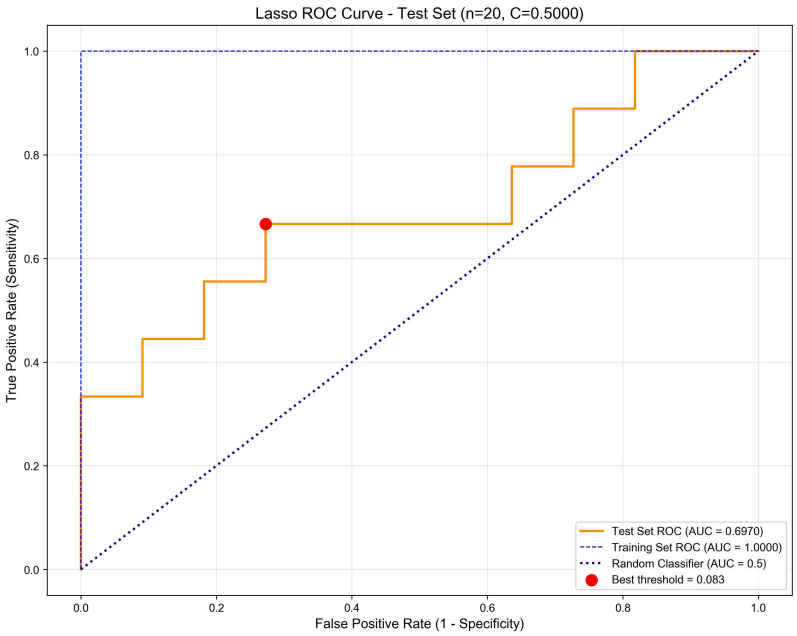


Fig. 3. Receiver operating characteristic curve for the L1 logistic regression (LASSO) model.

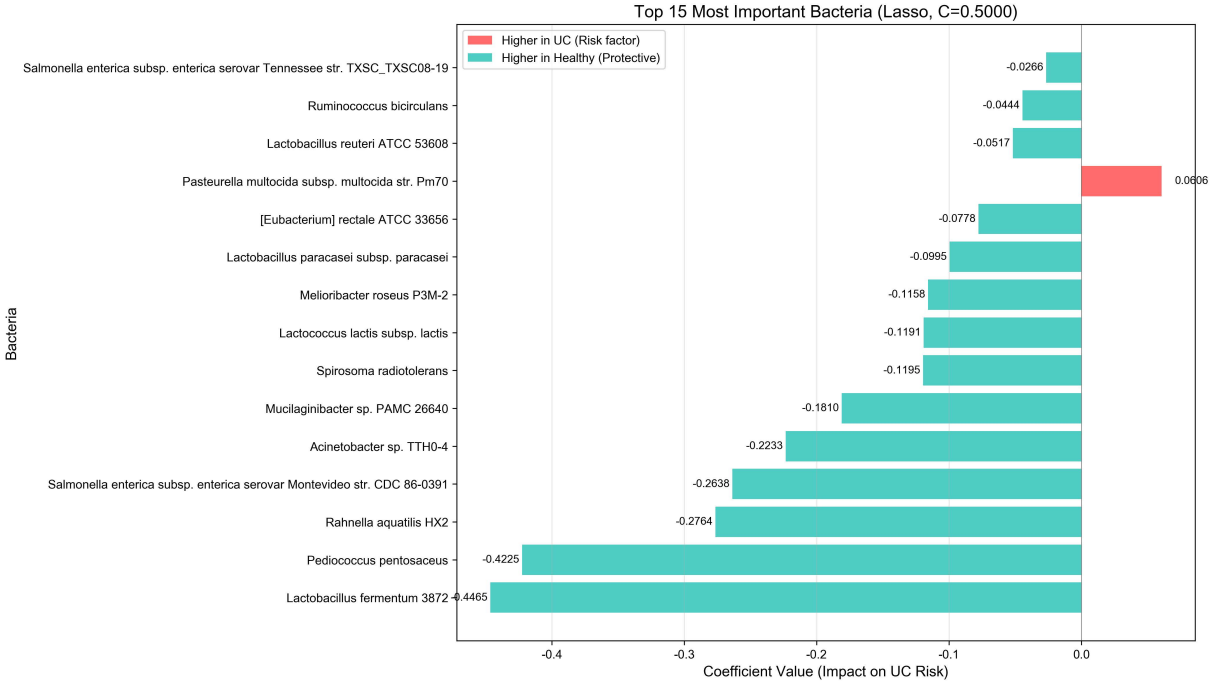


Fig. 4. Top 15 bacterial species identified by the L1 logistic regression (LASSO) model.

B. Support Vector Machine Results

After applying the support vector machine with an RBF kernel to the test dataset, we got: Fig. 5 shows that cross-validation of the training set chose $C = 10$ and gamma = "scale", which corresponds to a gamma value of 0.00626566. The selected parameter values were used to train the final model and predict on the test dataset, and the model then resulted in an AUROC of 0.8182. This means that the SVM classified the samples more accurately than the L1 or L2 logistic regression model. The AUROC shows, therefore, that the SVM works well on this dataset for classifying UC and healthy samples, using bacterial abundance data from Zuo *et al.* [6], when applying the SVM model.

Fig. 6 shows the bacterial species selected by the SVM model's predictions of UC and healthy status. The most significant protective factor was *Pediococcus acidilactici*, followed by *Lactobacillus delbrueckii* subsp. *bulgaricus* ND02. On the other hand, *E. coli* ATCC 8739, *Salmonella enterica* subsp. *enterica* serovar Typhimurium str. U288, *Streptococcus pneumoniae* SPN034156 were the most influential bacteria on UC prediction.

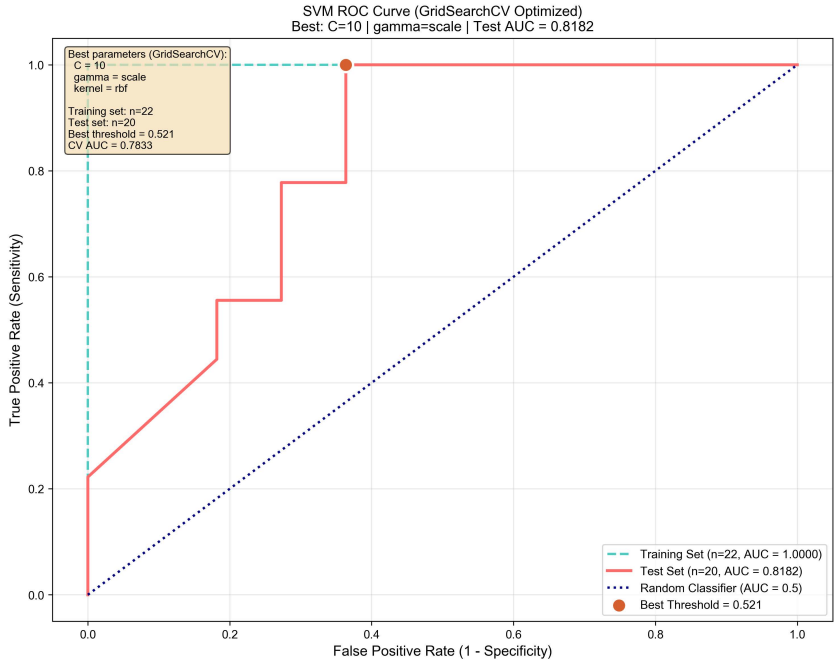


Fig. 5. Receiver operating characteristic curve for the SVM model with an RBF kernel.

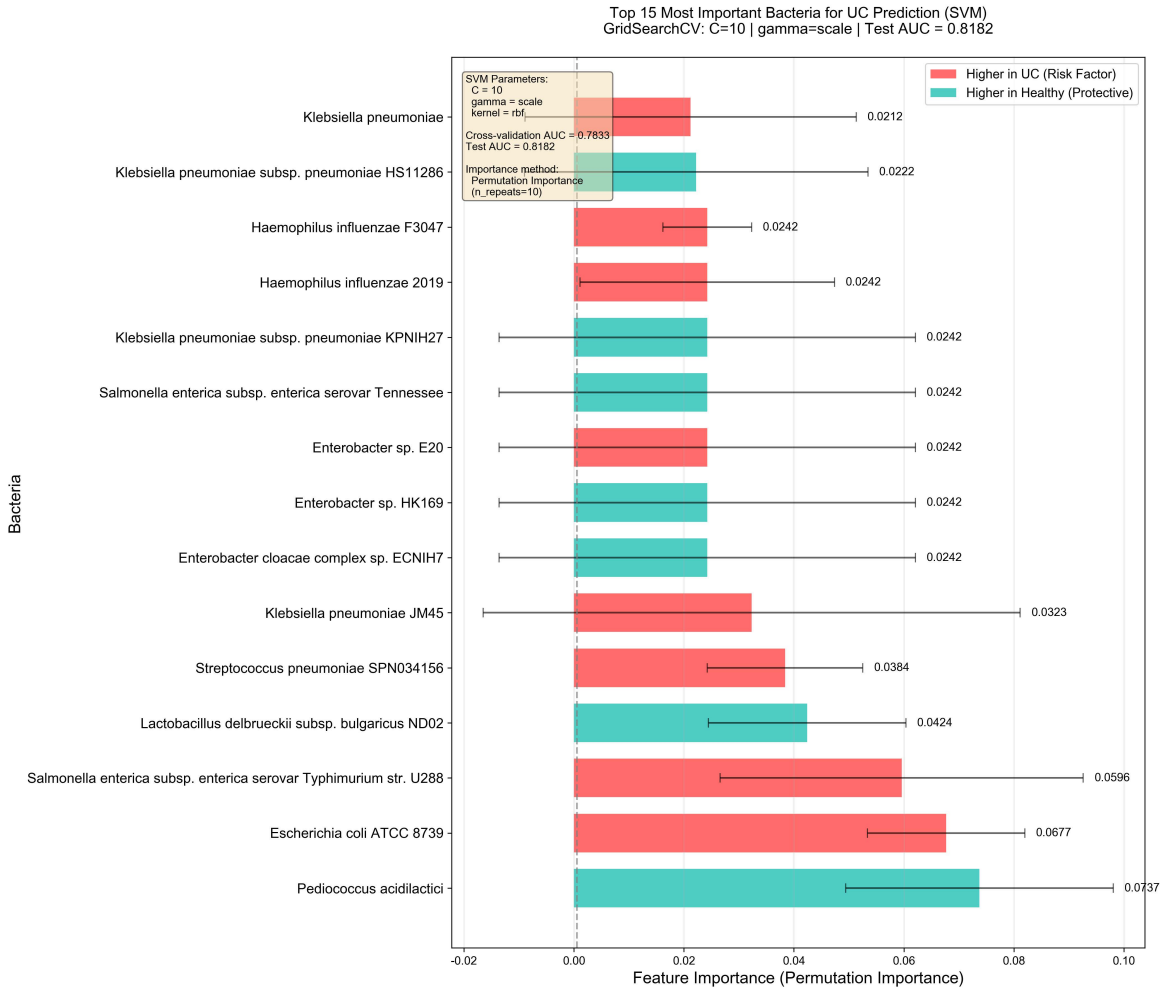


Fig. 6. Top 15 bacterial species selected by the SVM model with an RBF kernel. Red indicates risk factors, and green indicates protective factors.

IV. Discussion

A. Comparison of AUROC Values across Models

Each model gave different AUROC scores for this analysis. The AUROC for the L2 logistic regression model was 0.798, the AUROC for the L1 logistic regression model was 0.697, and the AUROC for the SVM model was 0.818. All models were better than L1 logistic regression (i.e., an AUROC of 0.763 was reported by Zuo *et al.* [6] for shotgun species with the Random

Forest model) and higher than the value of 0.763 from Zuo *et al*. [6] for UC versus healthy samples. On this dataset and test split, therefore, the L2 logistic regression and SVM models both outperformed the Random Forest model in classifying the samples as healthy versus UC.

By contrast, differences in model performances capture the way different machine learning models use the features in bacterial abundance data. For example, Li *et al.* [16] reported AUROC of 0.920 with logistic regression and 0.914 for SVM with 7-mer features. They also reported AUROC of 0.892 and 0.896 with 3-mer features and 0.933 and 0.941 with 5-mer features using logistic regression and SVM, respectively.

These values were much larger than we found in our study or Zuo et al., implying an important caveat of our work: our sample size is much smaller than that in Li *et al.* [16]. Li *et al.* [16] had samples from 1259 people worldwide, whereas our models were trained and tested with 42 samples only. Thus, our performance estimates are probably too high and need further testing. Moreover, Li *et al*. [16] generated k-mer feature tables directly from high-throughput-seq data; by contrast, Zuo *et al*. [6] mapped reads to known microbial genomes to get species abundance at the individual level. Li *et al*. [16] also used a linear SVM, rather than the RBF-kernel SVM, which partly explains the different performances.

Manandhar *et al*. [17] used an SVM with a radial kernel to classify UC versus healthy samples using two sets of features (i.e., the abundances of 50 bacterial taxa, and the 500 OTU features), AUROC values of 0.73 and 0.74 were obtained respectively. These values were lower than the AUROC value 0.8182 of our SVM. However, their sample size of 729 IBD and 700 healthy samples was larger than ours. One reason might be that 1695 bacterial species were used for our analysis, but only 50 bacterial taxa were used for feature set in one of their analyses. The difference may also be due to variations of the evaluation process. We used a single train-test splitting, which is sensitive to randomness. Manandhar et al. performed 50 random runs and reported the average AUROC value. Also, for hyperparameter selection on a very small training set, our SVM may be overfitted.

B. Bacterial Species Associated with IBD

The most important bacterial species found by the three models were largely different from one another, once more highlighting the limitations of a small sample size. But finding different species in the three models also suggests that each model focuses on different features or measures noise. *Salmonella enterica* serovars were found in both SVM and LASSO results, but different strains had different associations with IBD. In the LASSO model, Montevideo and Tennessee strains were labelled as protective, but the *Salmonella enterica* serovar Typhimurium strain was labelled as a risk factor by the SVM model. Experimental evidence also shows that *Salmonella enterica* serovar Typhimurium increases susceptibility of intestinal inflammation in mice [18] and relies on inflammation to outcompete beneficial gut microbiota [19]. The strains Montevideo and Tennessee are less well studied for IBD than Typhimurium.

Available experimental evidence suggests that they might produce weaker inflammatory responses and have a minor effect on damage to intestinal epithelial tissue [20]. These results point to how different models might analyze different properties of bacterial samples and how SVM might be useful in determining bacterial characteristics related to IBD.

However, some related bacterial species were found by more than one model. For example, the species in *genus Lactobacillus* were identified as protective by both the LASSO model (Figs. 3 and 4) and the L2 logistic regression model, suggesting a consistent anti-inflammatory role. There are previous studies indicating that *Lactobacillus reuteri* can tune the intestinal environment, reduce inflammation [21], and improve gut barrier dysfunction [22]. The other species in the *genus Lactobacillus* also have a protective signal in these models, such as *Lactobacillus delbrueckii* in the SVM model and other Lactobacillus species in the LASSO and L2 logistic regression models. These points further support the potential role of Lactobacillus in IBD. Other observations, such as *Pediococcus pentosaceus*, the strongest protective gene for the SVM model, also had a strong protective signal in the LASSO model,

As the evidence on this probiotic species would suggest, this probiotic species can regulate the composition and function of gut microbiota, and it has future clinical potential for IBD [23]. Further experiments demonstrated that *P. pentosaceus* LI05 protects intestinal epithelium integrity and regulates gut microbiota in mice [23]. The bacterial species shared across models can probably be regarded as particularly relevant to intestinal health and demonstrate the potential usefulness of ML to find candidate biomarkers.

Some bacteria were detected by the models but with little evidence that these microorganisms were linked to IBD. For instance, *Pasteurella multocida* subsp. *multocida* str. Pm70 was the only risk-associated species of the 15 species reported to the LASSO model. But it was not found by the other models, and we found no experimental evidence directly associating it with IBD. Lack of evidence to support the association suggests that *P. multocida* might be background noise or a database artefact. Similarly, *E. coli* ATCC 8739 had the most risk-associated value in the SVM model, but we found no evidence that this *E. coli* strain causes IBD. These potential artefactual associations highlight the discrepant feature pick-up between the three models and again exemplify the lack of power due to the small sample size.

C. Comparison of Machine Learning Models

Machine learning models may capture different relationships between bacterial features and IBD. These different behaviors may result from, for example, model type, feature selection method, suitability for the small and high-dimensional data, and the number of hyperparameters. We perform the following comparison between four models: L1 logistic regression model, L2 logistic regression model, SVM with an RBF kernel, and

Random Forest. The two logistic regression models and the SVM model were used in this paper, but the Random Forest model was used by Zuo *et al.* [6].

Both L1 and L2 logistic regression are linear models [24]. They assume that features contribute to the prediction through weighted coefficients, as in $logit(P_i)$, with each feature having an additive contribution to the log-odds. This form of structure makes the results easier to interpret and clinically translate. Non-linear models such as kernel SVM and Random Forest work differently from linear models. With these approaches, a feature's predictive contribution is not independent: it depends on interactions with other features [25]. An SVM kernel does this by transforming the data into higher dimensions to ensure separability [26], whereas a Random Forest combines various thresholds between decision trees [27]. Their popularity in the IBD field is also clear: we reviewed 31 papers and found that Random Forest was the most popular model (32%) and then, behind it, SVM (29%) [28].

Of course, this trade-off is also something we sacrifice in terms of interpretability. Non-linear models are harder to peel open because of their complicated transformations [29]. This is especially so for RBF-kernel SVMs, with their implicit transformation of the data by the kernel, making it hard to link back a model prediction to its original inputs. Of course, that ability to leverage non-linear relationships is very desirable in a clinical context (in which relationships between biomarkers and biomarkers, and genetics, are rarely simple) [30].

Feature selection also distinguishes these models. LASSO zeroes out coefficients of less informative variables, is embedded, and does feature selection that suppresses overfitting [31]. But it risks censoring relevant predictors. L2 regression punishes coefficients but does not eliminate them, and keeps all variables—though this obscures which features are truly relevant. In SVM with an RBF kernel, there is no automatic selection at all, so feature filtering must be applied beforehand; otherwise, irrelevant features may hide relevant patterns [32]. Unlike LASSO, SVM does not suppress the coefficient, and unlike logistic regression, it can be used in a higher-dimensional space [33]. Random Forest randomly selects features at each split [27], with importance being assessed from how often and to what extent features contribute to good prediction. Permutation methods rank features in terms of the decrease in model performance achieved by permuting each feature to scramble (randomising) the feature values—this can help find potentially relevant bacterial species [34].

Our small sample size (much more features than samples) made finding true predictors a challenging problem. On the other hand, our models worked well enough. Logistic regression remains simple with few parameters and is not complex. SVM uses its kernel and its dependence on support vectors to capture nonlinear interactions, and its margin-maximizing approach can also improve generalization [35]. Random Forest can be fiddly to tune for small sets of data, because the individual trees can be unstable on small sets of samples per tree [36]. Hyperparameter count matters too. Both L1 and L2 logistic regression have only one main hyperparameter, C, the regularization penalty. This simplicity means they are easily tuned and are less likely to overfit [37]. RBF-kernel SVM has two hyperparameters (C is the misclassification penalty and gamma is how 'wiggly' the decision boundary is), which makes the method more complex, but also more flexible.

For example, a smaller C allows more misclassifications and, in general, results in a broader, simpler decision boundary; a larger C stresses the training samples in deciding which class an observation should be classified into. Gamma controls the flexibility of the decision boundary and the spatial area of influence of the training samples. An RBF-kernel SVM is more complex to tune than logistic regression, but its two hyperparameters allow more flexibility to model highly complex data sets. Random Forest has more major hyperparameters, such as the number of trees (n_estimators), max tree depth (max_depth), number of features considered for splitting at each node (max_features), minimum number of samples required to make a split in a node (min_samples_split) and minimum number of samples required in a leaf node (min_samples_leaf). These hyperparameters allow more flexibility but also widen the tuning space. Hence, Random Forest is more computationally intensive than the other models.

Random Forest also has a built-in feature-importance measure that helps identify potential biomarkers and bacterial species that are associated with IBD. It also simplifies variations in model performance with bootstrap aggregation by averaging across trees to decrease the variance associated with any individual decision tree [27]. Zuo *et al.* [6] employed a Random Forest classifier with 500 trees and default values for the other hyperparameters for their final model. Zuo *et al.* [6] chose 500 trees rather than 100 or 1000 because this provided stable predictive performance with greater computational efficiency. The above example also demonstrates how a bigger hyperparameter space increases model flexibility and tuning complexity.

D. Potential Use of Machine Learning for IBD Prediction

Overall, machine learning models are useful for predicting IBD, and their use should be studied. Most studies have reported relatively good predictive performance, suggesting that machine learning models can improve IBD diagnosis once more research and validation take place. However, several issues currently limit the application of machine learning models to predict IBD. First, different models might result in different AUROC values and even different feature rankings because of different data-preparation procedures and the limited number of studies focusing on specific bacterial species and their link with IBD. For instance, *Salmonella enterica* serovars Montevideo and Tennessee were recognized by the L1 logistic regression and SVM models, respectively.

Sparse evidence with respect to these species could be an issue when selecting the best-suited model for a microbiome dataset. Thus, more studies with bigger sample sizes and repeated validation are necessary to study model robustness and reproducibility.

Another challenge is the cost and complexity of producing microbiome data [38]. Gut microbial samples must be taken and afterwards processed by Next-Generation Sequencing (NGS) [39], genome mapping, and a plethora of data analyses prior to any application of machine learning techniques. This involves specialist equipment, skills, and computing power and, accordingly, is a time-consuming, costly process. Diagnostic steps of microbiome machine learning are therefore not yet ready for widespread use in the clinical field.

Another aspect of IBD is that it is a dynamic disease that fluctuates with the microbiome of the gut, and the microbiomes of patients with IBD show more temporal instability than healthy people [40]. Time-shifts in the microbiome may slowly contribute to decreasing the performance of machine learning predictions. Another drawback is that a model derived for a population is not always generalizable to another population. These challenges explain why the current clinical application of machine learning for predicting IBD is difficult.

Nevertheless, such challenges can be overcome. To solve model inconsistency, researchers could merge different models to work around the shortcomings of individual models and reduce variation in predictions, or design and develop new models with better performance. The cost of microbiome data collection might also drop with the development of sequencing technologies. Higher availability of electronic health records and clinical data would allow more data for model training. The dynamic nature of IBD may also be addressed by the use of AI methods that are trained on large, longitudinal datasets, thus supporting more personalized diagnosis. Researchers are already working with AI to analyze, amalgamate and decode large datasets in IBD [41]. However, results derived from AI need to be rigorously validated prior to their use in the clinic.

Notwithstanding these difficulties, machine learning models have enormous potential for future use, as technology develops and data become available. Mixing ML methods with advanced machine learning algorithms may eventually enable accurate and efficient IBD diagnosis in daily practice.

V. Conclusion

This study explored the relationship between abundance of gut bacteria and IBD using three ML classifiers. The AUROC values (0.798 for L2 logistic regression, 0.697 for LASSO and 0.818 for SVM) indicate that models can distinguish UC from healthy samples and are able to identify bacterial species relevant for prediction. Differences in performance across the models and in the sets of top species each model identified by feature selection are a reflection of model type, feature selection approach, and flexibility of hyperparameters. We also explored the translational capacity of these approaches. Despite current limitations, there is cause for cautious hope: with continued progress in AI, sequencing technologies, and data availability, machine learning could ultimately be helpful in IBD diagnosis and in our knowledge of the gut microbiome. A. Strengths

This paper has some strengths. We extended the original Random Forest analysis using three other models on the same data, and we compared their results. On our test split, two of our models performed better than the original Random Forest (LASSO being the exception). We highlighted bacterial species with potential biological value for further research, and we provided a qualitative comparison of models by explaining their mathematics and practical realities.

We also point to some limitations. The computational constraints limited us from processing raw sequencing data; instead, we relied on processed abundance tables. The small number of samples may have been responsible for overfitting and also resulted in low confidence in our estimate of model performance. Small overlap between species identified with the different models shows a lack of reproducibility, and our hyperparameter tuning was also inadequate.

Looking ahead, we aim to collect larger datasets, apply other models, and use greater computing power. We also intend to extend our method to other diseases and validate our results against the wider literature. Eventually, we will want to understand the biological processes that link gut bacteria to IBD—from predictions to biology.

Conflict of Interest

The author declares no conflict of interest.

References

[1] R. O'Hare. (2017). Depression link with inflammatory bowel disease remains unclear. *Imperial News, Imperial College London*. [Online]. Available: http://www.imperial.ac.uk/news/179807/depression-link-with-inflammatory-bowel-disease/

[2] M. A. Veral, R. W. Pickup, M. Roy *et al*., "Geographic distribution of inflammatory bowel disease in the UK: A spatially explicit survey," *PLoS One*, vol. 20, e0329317, 2025.

[3] K. Yang, C. Zhang, R. Gong *et al*., "From west to east: Dissecting the global shift in inflammatory bowel disease burden and project future scenarios," *BMC Public Health*, vol. 25, 2696, 2025.

[4] T. Zuo, M. A. Kamm, J. F. Colombel *et al*., "Urbanization and the gut microbiota in health and inflammatory bowel disease," *Nat. Rev. Gastroenterol. Hepatol*., vol. 15, pp. 440–452, 2018.

[5] P. T. Santana, S. L. B. Rosas, B. E. Ribeiro *et al*., "Dysbiosis in inflammatory bowel disease: Pathogenic role and potential therapeutic targets," *Int. J. Mol. Sci*., vol. 23, 3464, 2022.

[6] W. Zuo, B. Wang, X. Bai *et al*., "Consistent pattern of gut microbiome signature by 16S rRNA and metagenomic shotgun sequencing data in children with ulcerative colitis," *Sci. Rep.*, vol. 12, 6421, 2022.

[7] H. Chen, J. Li, S. Yan *et al*., "Identification of pathogen(s) involved in infectious diseases based on shotgun metagenomic sequencing and conventional culture: A comparative study," *PeerJ*, vol. 9, e11699, 2021.

[8] C. R. Armour, S. Nayfach, K. S. Pollard *et al*., "A metagenomic meta-analysis reveals functional signatures of health and disease in the human gut microbiome," *mSystems*, vol. 4, no. 10, e1128, 2019.

[9] A. Javaid, O. Shahab, W. Adorno *et al*., "Machine learning predictive outcomes modelling in inflammatory bowel diseases,"

Inflamm. Bowel Dis., vol. 28, no. 6, pp. 819–829, 2022.
[10] S. Cohen-Mekelburg, S. Berry, R. W. Stidham *et al*., "Clinical applications of artificial intelligence and machine learning-based methods in inflammatory bowel disease," *J. Gastroenterol. Hepatol.*, vol. 36, no. 2, pp. 279–285, 2021.
[11] N. H. Nguyen, D. Picetti, P. S. Dulai *et al.*, "Machine learning-based prediction models for diagnosis and prognosis in inflammatory bowel diseases: A systematic review," *J. Crohn's Colitis*, vol. 16, no. 3, pp. 398–413, 2022.
[12] E. Bolyen, J. R. Rideout, M. R. Dillon *et al.*, "Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2," *Nat. Biotechnol*., vol. 37, no. 8, pp. 852–857, 2019.
[13] B. J. Callahan, P. J. McMurdie, M. J. Rosen *et al*., "DADA2: High-resolution sample inference from Illumina amplicon data," *Nat. Methods*, vol. 13, no. 7, pp. 581–583, 2016.
[14] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
[15] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc., Ser. B (Methodol.)*, vol. 58, no. 1, pp. 267–288, 1996.
[16] L. Li, Z. Liu, J. Qin *et al*., "Construction of disease diagnostic model of inflammatory bowel disease based on k-mer and machine learning," *Front. Microbiol*., vol. 16, 1578005, 2025.
[17] I. Manandhar, A. Alimadadi, S. Aryal *et al*., "Gut microbiome-based supervised machine learning models for clinical diagnosis of inflammatory bowel diseases," *Am. J. Physiol.: Gastrointest. Liver Physiol*., vol. 320, no. 3, pp. G328–G337, 2021.
[18] B. M. Schultz, G. A. Salazar, C. A. Paduro *et al*., "Persistent *Salmonella enterica* serovar Typhimurium infection increases susceptibility of mice to develop intestinal inflammation," *Front. Immunol*., vol. 9, 1166, 2018.
[19] B. Stecher, R. Robbiani, A. W. Walker *et al*., "*Salmonella enterica* serovar typhimurium exploits inflammation to compete with the intestinal microbiota," *PLoS Biol*., vol. 5, no. 10, pp. 2177–2189, 2007.
[20] J. Lalsiamthara and J. H. Lee, "Pathogenic traits of Salmonella Montevideo in experimental infections in vivo and in vitro," *Sci. Rep*., vol. 7, 46232, 2017.
[21] G. Wang, S. Huang, S. Cai *et al.*, "Lactobacillus reuteri ameliorates intestinal inflammation and modulates gut microbiota and metabolic disorders in dextran sulfate sodium-induced colitis in mice," *Nutrients*, vol. 12, no. 8, 2298, 2020.
[22] Y. Wu, X. Zhang, X. Liu *et al*., "Galactooligosaccharide and Limosilactobacillus reuteri synergistically ameliorated gut inflammation and barrier dysfunction by enriching Bacteroides acidifaciens for pentadecanoic acid biosynthesis," *Nat. Commun.*, vol. 15, no. 1, 9291, 2024.
[23] F. Dong, F. Xiao, X. Li *et al*., "The propionate bacterium Pediococcus pentosaceus CECT 8330 protects DSS-induced colitis and regulates intestinal microbial and immune response in mice," *J. Transl. Med.*, vol. 20, no. 1, 33, 2022.
[24] J. K. Harris, "Primer on binary logistic regression," *Fam. Med. Community Health*, vol. 9, no. Suppl 1, e001290, 2021.
[25] S. Ghosh, A. Dasgupta, and A. Swetapadma, "A study on support vector machine based linear and non-linear pattern classification," in *Proc. 2019 Int. Conf. Intell. Sustain. Syst. (ICISS)*, IEEE, Feb. 2019, pp. 24–28.
[26] K. L. Du, B. Jiang, J. Lu *et al*., "Exploring kernel machines and support vector machines: Principles, techniques, and future development," *Mathematics*, vol. 12, no. 24, 3935, 2024.
[27] L. Breiman, "Random forests," *Mach. Learn*., vol. 45, pp. 5–32, 2001.
[28] J. Huang, X. Zhu, Y. Ma *et al*., "Machine learning in differential diagnosis of ulcerative colitis and Crohn's disease: A systematic review," *Transl. Gastroenterol. Hepatol*., vol. 10, 56, 2025.
[29] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable AI: A survey of machine learning interpretability methods," *Entropy*, vol. 23, 18, 2020.
[30] J. H. Moore, "Detecting, characterizing and interpreting nonlinear gene-gene interactions using multifactor dimensionality reduction," *Adv. Genet*., vol. 72, pp. 101–116, 2010.
[31] J. Ranstam and J. A. Cook, "LASSO regression," *J. Br. Surg*., vol. 105, no. 10, pp. 1348–1348, 2018.
[32] I. Guyon, J. Weston, S. Barnhill *et al*., "Gene selection for cancer classification using support vector machines," *Mach. Learn*., vol. 46, no. 1, pp. 389–422, 2002.
[33] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[34] S. Janitza, C. Strobl, and A. L. Boulesteix, "An AUC-based permutation variable importance measure for random forests," *BMC Bioinf*., vol. 14, no. 1, 119, 2013.
[35] A. Grønlund, L. Kamma, and K. G. Larsen, "Near-tight margin-based generalization bounds for support vector machines," in *Proc. Int. Conf. Mach. Learn. (ICML)*, PMLR, Nov. 2020, pp. 1–10.
[36] S. Han, B. D. Williamson, and Y. Fong, "Improving random forest predictions in small datasets from two-phase sampling designs," *BMC Med. Inform. Decis. Mak*., vol. 21, no. 1, 322, 2021.
[37] T. Hastie, "Ridge regularisation: An absolutely essential idea in data science," *Technometrics*, vol. 62, no. 4, pp. 426–433, 2020.
[38] J. Osei Sekyere, "Next generation sequencing in the infectious disease diagnostic laboratory: Economic, regulatory and clinical pathways to adoption," *MicrobiologyOpen*, vol. 14, no. 6, e70104, 2025.
[39] S. A. Boers, R. Jansen, and J. P. Hays, "Understanding and overcoming the pitfalls and biases of Next Generation Sequencing (NGS) methods for use in the routine clinical microbiological diagnostic laboratory," *Eur. J. Clin. Microbiol. Infect. Dis*., vol. 38, no. 6, pp. 1059–1070, 2019.
[40] J. Halfvarson, C. J. Brislawn, R. Lamendella *et al*., "Dynamics of the human gut microbiome in inflammatory bowel disease," *Nat. Microbiol.*, vol. 2, no. 5, 17004, 2017.
[41] J. Gubatan, S. Levitte, A. Patel *et al*., "Artificial intelligence applications in inflammatory bowel disease: Emerging technologies and future horizons," *World J. Gastroenterol.*, vol. 27, 1920, 2021.

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).