# A Comprehensive Study on Predicting Functional Role of Metagenomes Using Machine Learning Methods

Jyotsna Talreja Wassan, *Student Member, IEEE*, Haiying Wang, Fiona Browne, *Member, IEEE and* Huiru Zheng*, *Senior Member, IEEE*

**Abstract**— "*Metagenomics*" is the study of genomic sequences obtained directly from environmental microbial communities with the aim to linking their structures with functional roles. The field has been aided in the unprecedented advancement through high-throughput omics data sequencing. The outcome of sequencing are biologically rich data sets. Metagenomic data consisting of microbial species which outnumber microbial samples, lead to the "curse of dimensionality" in datasets. Hence the focus in metagenomics studies has moved towards developing efficient computational models using Machine Learning (ML), reducing the computational cost. In this paper, we comprehensively assessed various ML approaches to classifying high-dimensional human microbiota effectively into their functional phenotypes. We propose the application of embedded feature selection methods, namely, Extreme Gradient Boosting and Penalized Logistic Regression to determine important microbial species. The resultant feature set enhanced the performance of one of the most popular state-of-the-art methods, Random Forest (RF) over metagenomic studies. Experimental results indicate that the proposed method achieved best results in terms of accuracy, area under the Receiver Operating Characteristic curve (ROC-AUC) and major improvement in processing time. It outperformed other feature selection methods of filters or wrappers over RF and classifiers such as Support Vector Machine (SVM), Extreme Learning Machine (ELM), and *k*- Nearest Neighbors (*k*-NN).

**Index Terms**—Metagenomics, Microbiota, Embedded Feature Selection, Operational Taxonomic Units (OTUs), Classification

## 1 Introduction

METAGENOMICS, is one of the emerging "omics" fields which involves investigation of genomic sequences obtained directly from whole microbial communities present in an environment (such as water, soil, human body, and cattle, etc.), following a culture-independent approach [1]. In recent years, this field has gained attention due to crucial projects such as the Human Microbiome Project [2], American Gut [3], Earth Microbiome [4], and to the unprecedented advances in low-cost high-throughput Next-Generation Sequencing (NGS) such as the 454 pyrosequencing [5], over the traditional Sanger approach [6] for DNA isolation and sequencing from environmental communities. The two primary sequencing profiles are: i) whole metagenome sequencing (WGS), and ii) marker gene (16S rRNA/18S rRNA/ITS) sequencing [7]. The most commonly used taxonomic profiling for microbial analysis is 16S rRNA. The sequence variants of 16S rRNA are clustered at a similarity threshold (usually 97 %) to generate Operational Taxonomic Units (OTUs)/taxas [7].

The in-depth analysis of metagenomic sequencing data with computational models and related experiments provide deeper insights into the complex microbiome ecosystem. Machine learning (ML) techniques learn from data to make future predictions [8], [9]. Hence, they are useful for integrating high-throughput metagenomic data to predict functional roles. The meta-analysis (i.e. the categorization of metagenomes into their functional roles), is achieved through the application of classification, a supervised ML technique; which models the distribution of functional classes in terms of predictors (input features) [9], [10]. The classifier maps microbiome data, such as quantitative abundance count profiles of microbes, to their related meta-data [10]. Functional analysis of metagenomic environments [11] forms a three-step process as listed below:

i. Input: A set of metagenomics sequences binned to Operational Taxonomic Units (OTUs) [7] and their abundance count matrix, $X$ (as shown in (1)), with $m$ metagenomic samples and $n$ OTUs; and set of functional labels $Y$.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots\, x_{i,j} & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix} \tag{1}$$

ii. Fitting an ML model to the input matrix $X$ by providing a functional mapping from the row of $X$ (representing a microbial sample) to a functional label $y \in Y$

iii. Output: Labeled sequences.

Metagenomic OTU data is high-dimensional, sparse,

*Jyotsna Talreja Wassan is a researcher with the School of Computing, Ulster University, Co. Antrim, BT37 0QB, U.K. E-mail: wassan-jt@ulster.ac.uk.*
*Haiying Wang is a Reader with the School of Computing, Ulster University, Co. Antrim, BT37 0QB, U.K. E-mail: hy.wang@ulster.ac.uk*
*Fiona Browne is a Lecturer with the School of Computing, Ulster University, Co. Antrim, BT37 0QB. U.K. E-mail: f.browne@ulster.ac.uk*
*Huiru Zheng* is a Professor in Computer Science with the School of Computing, Ulster University, Co. Antrim, BT37 0QB, U.K. E-mail: h.zheng@ulster.ac.uk*

and skewed with a non-normal or an unknown distribution and it has interdependencies [12]. These key characteristics of OTUs present computational and statistical challenges [12]. Hence, the current research demands development of better algorithmic tools to compute biological phenotypes efficiently.

The motivation of this paper is to provide an intensive assessment on the use of various ML models for analyzing functional metagenomes. We proposed a method for selecting important OTUs from a high-dimensional feature space for classifying them into functional profiles. The objective of OTU feature selection is two-fold: - i) improving the predictive performance of the ML models and ii) providing faster, cost-effective predictions. In the current context, we focused on three feature selection strategies based on the filter, wrapper, and embedded methods for comparative analysis [13], [14]. These methods are helpful in finding a list of candidate microbial species with more informative gene sequences, leading to unbiased estimates and enhanced performance.

With this goal, we have designed a comprehensive study analyzing various ML models tuning hyper-parameters over *k*- folds cross-validation (where $k = 10$) and devised a new integrative method of combining embedded feature selection with the Random Forest (RF) classifier. By applying the afore-mentioned combination in functional analysis of human microbiota, we obtained improved results in comparison to applying RF to metagenomic data alone. In this study, we processed metagenomic samples from three Use Cases related to human microbiome, to assess the predictive performance of models built on metagenomic data and to compare the results. We investigated how the selection of discriminative OTUs impacts the predictive performance in metagenomic case studies involving a large number and variety of microbial species.

The rest of this paper is organized as follows. Section 2 presents relevant related work in the field of functional metagenomics. Materials and methods are detailed in Section 3. Experimental results following evaluations are presented in Section 4. Finally, conclusion and future work are highlighted in Section 5.

## 2 RELATED WORK

Metagenomic studies are being influenced by compositional abundance of OTUs and their functional capabilities across samples. An important research question gathering attraction by the research community is: - "Which OTUs/species/taxas are important to characterize the environmental roles?". The study of the relationships between environment and microbiome has its roots in microbiology, as was first suggested by studies undertaken by Leeuwenhoek [7].

Microbiome research has been extensively applied to the humans. The human body is treated as an ensemble of microorganisms which play an important role in the sustainability of human health [15]. The Human Microbiome Project (HMP) [16] has opened various avenues for relating microbial samples to various human diseases, their interference in medication and in regulating gene expressions. Studies [17-21] have shown an association of the human microbiome to various chronic diseases, such as diabetes (Type 1 and Type 2), Inflammatory Bowel Disease (Crohn's disease or healthy), Obesity (obese, lean, overweight), rheumatoid arthritis, fatty liver disease, Alzheimer's disease, and cancer. The American Gut project [20], a crowd-funded citizen science project, has recently examined linkages between microbial samples and functional factors such as health (diseases like diabetes, autism etc.), lifestyle (sleep patterns, stress levels) and diet data (vegan, carnivore).

Research has also been performed on other varied phenotype hosts such as the ocean, soil, and cattle where microbe-host interactions have great potential in uncovering the influence on an environmental condition being studied [21-28]. Wang et al. [26] provided an integrated metagenomic analysis of rumen microbiome responsible for methane emissions and biomass degradation. Walsh et al. [27] proposed a pipeline for metagenomics analysis for rumen microbiome. Toyama et al. [28] proposed an analysis of microbial communities in freshwater lakes of Amazon Basin. The Earth Microbiome project [4] has emerged to characterize the curation and analysis of microbial species across the globe. Supervised learning using OTU (microbiome) feature space to train models has been used to classify microbiomes into functional classes as indicated in [8], [9], [23-29]. The studies related to ML models for discriminating OTUs or taxa to predict the functional class of a new sample have been conducted [23-32].

Commonly used supervised classification methods in metagenomics, are described in studies by Knights et al. [24] and Statnikov et al. [29] over the benchmark human microbiome data sets. Knights. et al. [29] reviewed supervised classifiers of RF, Nearest Shrunken Centroids, Elastic Net (ENet), Support Vector Machine (SVM) with filters of bi-normal separation and recursive backward feature elimination) over five benchmark data sets (Costello et al. Body Habitats (CBH), Skin Sites (CSS), Subject (CS), Fierer et al. Subject (FS) and Fierer et al. Subject X Hand (FSH)) originated from human microbiome studies [24]. Statnikov et al. [29] extended the comprehensive evaluation of 18 classification methods including RF, Logistic Regression, SVMs and *k*- Nearest Neighbors (*k*-NN) with 5 feature selection methods and 2 accuracy metrics using 8 functional tasks on human microbiome (1802 samples), to be classified into subject categories of body sites. The publications reported RF, as the best supervised learning technique for analyzing microbiome and linking it to functional roles [24], [29]. Yang et al. [30] classified soil samples using SVMs and *k*-NN [30]. Wingfield et al. [31] proposed a hybrid classifier consisting of SVMs and neural networks for efficiently characterizing Pediatric Inflammatory Bowel Disease (IBD) in humans. A comparative study by Pasolli et al. [32] proposed a computational tool for functional meta-analysis using SVM, RF, Least Absolute Shrinkage and Selection Operator (Lasso), and ENet as ML models. Huttenhower et al. [33] discussed advances in microbiome research community and the role of diverse microbial communities in the spectrum of functional phenotypes. Deng et al. proposed the construction and use of gene co-expression networks to predict cancerous genes as phenotypes [34], [35].

In our previous study, we experimented with various classifiers and parameter tunings (e.g. seed, kernel, and number of iterations) over cattle (MetaPlat Project[1]) and human microbiomes [18], at different taxonomical levels [25]. We applied Naïve Bayes (NB); Neural Network (NN); RF; SVM; Logistic Regression (LR) with Ridge estimation; *k*-NN; Adaptive Boosting (AdaB) on trees and an ensemble of classifiers with filter and wrapper-based feature selection procedures [25]. Four dominant classifiers providing overall good accuracy were reported: SVM, LR, NN and RF. LR with penalized Ridge regularization over the features obtained by wrapper based on the LR itself, providing the best results in our study [25].

The method we proposed in our previous work is suitable for smaller data sets but is computationally intensive for large-scale data. Therefore, in this paper, we extend beyond the traditional approaches listed above to effectively classify the large-scale metagenomes.

## 3 Material And Methods

### 3.1 Materials

In this paper, we used three publicly available human metagenomics data sets as Use Cases. These data sets are summarised in Table 1 and described as follows.

**(1)** The curated version of the human gut microbiome data set used in the study by Turnbaugh et al. [36], to analyze the effect of diet on human microbiome (http://www.exploredata.net/Downloads/Microbiome-Data-Set). The data set consists of 658 samples and 6696 OTUs and is mapped to a functional label of diet with two majority class values as: i) LF/PP diet (i.e. standard low-fat, plant polysaccharide–rich diet) and ii) Western diet (i.e. high fat, high sugar diet).

**(2)** The curated data set from the study by Koren et al. 2013 [37] is related to enterotypes across the human body. It is downloadable from http://www.knightslab.org/data. It consists of 1654 samples and 3534 OTUs with two functional labels of HMP and non-HMP feces.

**(3)** The third dataset is obtained from Halfvarson et al. [38] who studied the dynamics of human intestinal microbiota in Inflammatory Bowel Disease (IBD) subjects compared to healthy subjects. This data is downloadable from Qiita (https://qiita. ucsd.edu/) under study ID 1629. The data is also downloadable from *R* platform with command *data(DynamicsIBD)* under *library (microbiome)*[2]. The data set consists of 673 microbial samples (i.e. fecal samples from the population set in Sweden) and 10996 OTUs. The four related functional labels used in our research are: ulcerative colitis (UC), Healthy controls (HC), Crohn's disease (CD), collagenous colitis (CC).

TABLE 1
Summary of Use Cases: - Human Metagenomics

| **Data Set** | **OTUs** | **Samples** | **# of Classes** | **Functional Phenotypes** | **Ref.** |
|---|---|---|---|---|---|
| Diet | 6696 | 658 | 02 | Low-Fat Diet<br>High-Fat Diet | [36] |
| Enterotypes | 3534 | 1654 | 02 | HMP Feces<br>Non-HMP Feces | [37] |
| IBD | 10996 | 673 | 04 | UC, HC, CD, CC | [38] |

### 3.2 Methods

In this section, we provide a description of ML models used in this study to analyze the metagenomics data and describe the new adapted research methodology. The ML models include various classification and feature selection methods [10], [13].

#### *3.2.1 Description of Classification Methods*

The classification methods facilitate predictive modeling over the metagenomic Use Cases to support a holistic understanding of input data and link it to functional classes. An objective of this study is to identify a ML model, which provides good predictive performance and is fast at classification over the high-dimensional metagenomes. The following methods have been applied to the Use Cases listed in section 3.1.

(1) Boosted Trees (XGBoost)

The method e**X**treme **G**radient **Boost**ing (XGBoost) by Chen et al. [39], serves as a sparsity-aware algorithm that supports scalable tree boosting for classification [39]. It is an improvement over the gradient boosting framework by Friedman et al. [40]. The classification decision trees are grown iteratively by learning decisions from a previously grown tree. The method continuously tries to improve its prediction in subsequent iterations by reducing the misclassification error rate. It works in parallel fashion with its two main components:

a) linear boosting model solver
b) and tree ensemble learning algorithm

A tree ensemble method uses $N$ additive functions over training input $x_i$ (multiple features) to predict the output target $Y_i$ in (2) [39].

$$Y_i = \sum_{n=1}^{N} f_n(x_i), \; f_n \in F, \qquad (2)$$

where $N$ is the number of trees, $f$ is the function in $F$, *which* serves as a regression tree space and may be regularized to penalize the complexity of model [39].

(2) Penalized Logistic Regression (Glmnet)

The method tries to fit a generalized logistic model (3) for classification by choosing the parameters that maximize the log-likelihood of observed sample values with associated regularization penalty (λ) (4a, 4b, 4c) [41].

[1] http://www.metaplat.eu/

$$P(Y = 1/x) = 1/(1 + e\char`^(-h(x))\,), \qquad (3)$$

where $h(x) = w_0 + \sum_{j=1}^{d} w_j \; x_j$, and d represent the number of dimensions/features; $w_0$ , $w_{j's}$ are the regression coefficients and $x_j's$ are input features.

Regularization methods of Lasso, Ridge and ENet, constraint the regression coefficients ($w_j$'s) by imposing a penalty $\lambda$ on their values (4a, 4b and 4c) [41], [42].

$$\text{Lasso} = \arg\max_w \sum_{k=1}^{n} \big(y_k(h(x) - \log\big(1 + h(x)\big)\ \big) - \lambda \sum_{j=1}^{d} \mid w_j \mid \qquad (4a)$$

$$\text{Ridge} = \arg\max_w \sum_{k=1}^{n} \big(y_k(h(x) - \log\big(1 + h(x)\big)\ \big) - \lambda \sum_{j=1}^{d} w_j^2 \qquad (4b)$$

$$\text{ENet} = \arg\max_w \sum_{k=1}^{n} \big(y_k(h(x) - \log\big(1 + h(x)\big)\ \big) - \lambda \sum_{j=1}^{d} (1 - \alpha) \mid w_j \mid + \ \alpha\ (w_j^2\ )) \quad \text{where } (\alpha \in [0,1]) \qquad (4c)$$

where $\lambda$ controls the overall strength of the penalty and penalizes large coefficients to zero to avoid overfitting.

ENet is a generalization of both models Lasso and Ridge [42]. The penalties maintain the balance between the measure of fit of the logistic model and the measure of coefficients. Hence, this approach provides an effective predictive modeling especially in the case where input microbiome features overwhelm each other.

(3) RF

RF is widely used in metagenomic studies and is considered a state-of-the-art approach for classifying metagenomes [43], [44]. RF works by constructing a set of decision trees on training samples [43]. The optimal number of decision trees aims to minimize the classification error on the test sample. The method applies majority voting amongst individual trees to derive the functional class [44].

(4) SVM

SVM is one of the robust supervised learning methods that serve as non-probabilistic linear classifiers over the training data [45]. The method selects hyperplanes in context of linearly separable data, which tends to separate functional classes by maximizing the distance (margin) between them, forming an optimization problem.

(5) ELM

Extreme Learning Machine (ELM), proposed by Huang et al. [46], is a classifier method which is based on the principle of single hidden layer feed-forward networks with randomly generated weights and supports no gradient-based backpropagation [46]. Hence it is a faster method. The model is represented by (5) [46].

$$Y_i = M_2 \sigma(M_1 x) \qquad (5)$$

where $M_1$ is the matrix representing input-to-hidden-layer weights in a Neural Network, $\sigma$ is some activation function such as sigmoid, sine etc., and $M_2$ is the matrix of hidden-to-output-layer weights. $M_1$ is randomly generated with Gaussian noise and $M_2$ is estimated by Least Squares fit w.r.t to output class label class [46].

(6) $k$-NN

$k$-NN is an instance-based learner method. It considers the closely related instances of input features (i.e. instances having same properties and at minimized distance), within a data set. The class of neighbors determines the class of an individual instance [47].

### *3.2.2 Description of Feature Selection Methods*

Feature selection methods determine an appropriate subset from the full data set that leads to the smallest classification error [48]. The methods used in our study are discussed below.

(1) Embedded Methods

Embedded methods learn which OTUs (input features) contribute best towards attaining higher performance for an ML model, while the model is being constructed (Fig.1. a.) [48]. Commonly used embedded feature selection methods include regularization and boosting methods. Regularization methods tend to introduce additional constraints (penalties) into the predictive model (e.g. LR) as an optimization to lower the complexity of the original model. Examples of regularization algorithms are the Lasso, ENet, and Ridge (listed in the section 3.2.1. above) [41], [42], [48]. Lasso and ENet penalties perform inherent feature selection by setting the coefficients of weak OTU features (i.e. when a feature does not fit to LR), to zero. Increasing the penalty will produce a solution to sparse data by increasingly setting coefficients to zero in case of more features. Ridge produces a more stable effect by setting up similar coefficients for correlated OTU features and spreading the effect of regularization equally. The features with non-zero coefficients are regarded useful [41], [42], [48].

Boosting (XGBoost) constructs boosted trees and calculates an importance score for each feature based on its participation in making key decisions with boosted decision trees and ranking the respective features. The performance score may be calculated by purity index at a splitting node or other error functions [39], [40].

(2) Filter methods

Filter methods evaluate predictive OTUs outside of the predictive learner models (Fig.1. b.), by applying statistical tests to input OTU features and determining which OTUs are more plausibly related with the functional class [48]. It is usually a pre-cursor step for model evaluation. We used the following filters in our integrated approach.

- Mutual Information

The method finds weights of discrete OTU features based on their correlation with functional class [49]. Features that aid in perfectly separating the classes, give maximum information and unrelated features give no information. It measures the impurity in samples, which is also known as, Entropy ($H$) as depicted in (6a). The individual probabilities of the values n $\in$ N, estimate a feature N from the training data. Mutual Information (MI) is calculated on the basis of difference in entropies (related to features). For example, considering the two features N and M, $MI$ is given in (6b) [49].

$$H\ (N) = -\ \sum_{n \in N} p_n \log p_n \qquad (6a)$$

where $p_n$ denotes the class probability (i.e. proportion of samples belonging to class n and N = 1, 2…, n.)

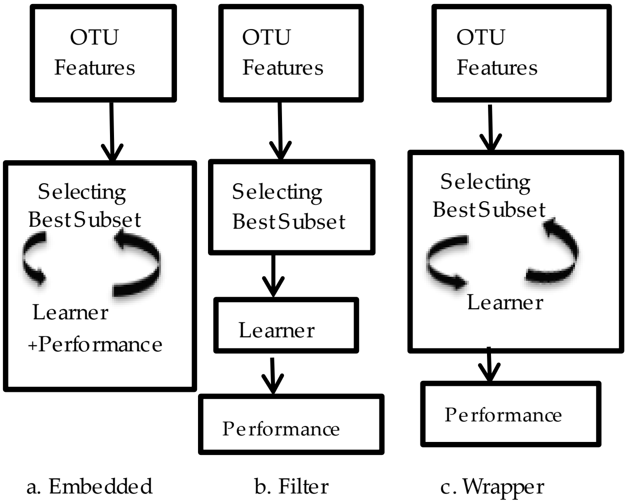
$$\mathrm{MI} = \ H(N\ )\ + H(M) - H(N,M) \qquad (6b)$$

- oneR

The method is proposed by Holte [50] and it finds weights of discrete OTU features by deriving an association rule for each attribute and then calculating the associated error rate. It selects the rule with minimum error rate. This has been used as baseline performance benchmark for other feature selecting methods.

(3) Wrapper Methods

Wrapper method trains a learner (ML) model on different subsets of predictive features by continuously adding or removing features to choose the optimal set of features, that maximizes the prediction performance (Fig.1.c.) [48]. We used Recursive Feature Elimination [52] that serves as a greedy optimization over wrapper which repeatedly creates best and worst feature sets and constructs successive model iterations with best features from the previous model iteration. The process stops when all the features have been processed. It tends to rank features in order of their elimination.



a. Embedded b. Filter c. Wrapper

Fig. 1. Types of feature selection methods

### 3.2.3 *Performance Evaluation of Classification*

The main objective of this task is to perform functional classification and predict its performance in metagenomics. The performance of the functional predictions in the current metagenomic study is primarily driven by statistical assessment strategies, based on a confusion matrix of $c\ X\ c$ w.r.t to "$c$" functional classes (a sample case of binomial class ($c$ = 2), is shown below in Table2) [53].

TABLE 2
Confusion Matrix Representation

| | **Positive (Actual)** | **Negative (Actual)** |
|---|---|---|
| Positive (Predicted) | True Positive (TP) | False Positive (FP) (Type I Error) |
| Negative (Predicted) | False Negative (FN) (Type II Error) | True Negative (TN) |

Assessment strategies for binary classification, making use of TP and TN for predictions (from Table 2), are listed in (7), (8), (9), and (10) [52].

$$\text{Accuracy } (Ac) \ = \ (\mathrm{TP} + \mathrm{TN})/(\mathrm{P} + \mathrm{N}) \qquad (7)$$

$$\text{Precision } (\ Pr) = \ \mathrm{TP}/\ (\mathrm{TP} + \mathrm{FP}) \qquad (8)$$

$$\text{Sensitivity } (Se) = \mathrm{TP}/\ \mathrm{P} \qquad (9)$$

$$\text{Specificity } (\ Sp) = \ \mathrm{TN}/\ \mathrm{N} \qquad (10)$$

where P = TP + FN, and N = TN + FP.

For multinomial classification with $c$ classes, $c\ x\ c$ confusion matrix $M$=$m_{i,j}$ is constructed, where $m_{i,j}$ represents sample numbers predicted as class $j$ but belonging to class $i$. Consider, $S_i = \sum_{1 \le j \le c} m_{i,j}$ be the number of input samples associated with class $i$, and $F_i = \sum_{1 \le j \le c} m_{i,j}$ be the number of input features predicted to be in functional class $j$. For the case of $c > 2$, the assessment metrics are generalized as follows in (11), (12), (13) and (14). We calculated mean values of assessments of all individual classes.

$$\text{Accuracy } (Ac) = \ \sum_{1 \le \mathrm{i} \le \mathrm{c}} \mathrm{m}_{\mathrm{i,i}\,,} / \sum_{1 \le \mathrm{i} \le \mathrm{c}} \mathrm{F}_{\mathrm{i}\,,} \qquad (11)$$

$$\text{Precision} (\ Pr) = (\sum_{1 \le \mathrm{i} \le \mathrm{c}}(\mathrm{m}_{\mathrm{i,i}\,.} \ /\ \mathrm{F_i}))/\mathrm{c} \qquad (12)$$

$$\text{Sensitivity } (Se) \ = (\sum_{1 \le \mathrm{i} \le \mathrm{c}}(\mathrm{m}_{\mathrm{i,i}\,.} \ /\ \mathrm{S_i}))/\mathrm{c} \qquad (13)$$

$$\left.\begin{array}{l} \mathrm{S} = (\sum_{1 \le \mathrm{i} \le \mathrm{c}}(\sum_{\mathrm{k} \ne \mathrm{i}, \mathrm{j} \ne \mathrm{i}} \mathrm{m_{k,j}} / \ \sum_{\mathrm{k} \ne \mathrm{i}, \mathrm{i} \le \mathrm{j} \le \mathrm{c}} \mathrm{m_{k,j}})) \\ \text{and } \ \text{Specificity } Sp = \mathrm{S/c} \end{array}\right] \qquad (14)$$

Receiver operating characteristic curve (ROC) evaluates the performance of a classifier in terms of how good the classifier in is separating positive and negative samples and identifies the best threshold for separating them. It characterizes the tradeoffs between sensitivity and specificity. A low threshold has the capability to produce positive labels more liberally, so it is prone to have more false positives (less specific) but also more true positives (more sensitive).

### *3.2.4 An Integrative Experimental Workflow for Metagenomic Analysis*

The data collected from NGS sequencing are immense and it is important to select the best and most suitable functional metagenomes to differentiate the samples collected from an environmental study. The general experimental workflow which integrates various ML models for functional classification in metagenomics is described as follows (illustrated in Fig.2.).

(1) Input. An OTU table and a label mapping file serve as inputs. The OTU table derived from Biological Observation Matrix (BIOM), consists of raw abundance counts of OTUs in a microbial sample; where rows represent the samples, columns represent the OTUs and the cell entries at the intersection of the sample and OTU are abundance counts. The samples also associate meta-data, describing their association with environmental traits (from label mapping file).

(2) Preprocessing and Feature Selection. The OTU tables containing abundance counts of microbial species/taxas in a sample, are pre-processed and transposed to fit to a ML. We transformed the input data to a suitable form for classification and apply various feature selection techniques (listed in section 3.2.1) to derive important features. OTU tables have a variety of feature attributes. It is important to select relevant features to maximize the performance of our experimental design. Feature selection methods removed irrelevant and redundant features. Consider, a set of m metagenomic samples $x_{i,}$ $y_{i,}$ where i = 1, ...m; consisting of $n$ OTU features $\{x_{i,j}\}$ (j = 1, ...n) and one functional variable $y_i$. Feature selection methodsaim to identify a suitable fitness function $F(i, j)$ that is computed over $\{x_{i,j}\}$ to predict $y_i$, in functional metagenomics.

(3) Cross-Validation. We partitioned the input data set into training and test sets for ten-folds cross-validation.

(4) Model fitting. Applying ML model for categorizing the OTU features, into one of a pre-specified set of functional categories or classes, is the key characteristic step of functional metagenomics. To identify the most suitable model for predicting functional metagenomes, different supervised ML classification algorithms (listed in the section 3.2.1), were evaluated for their fitness in the prediction task against the OTU feature sets.

(5) Performance Evaluation. We predicted the performance of classifier using assessment measures listed in section 3.2.2.

(6) Output. Best Model for classifying metagenomes

Along the integrative experimental workflow (Fig.2.), we proposed an ensemble method for predicting metagenomes effectively. The method first selected the important features from embedded feature selectors and thereafter applied RF to a reduced set of features (Fig.3.). This construction combined boosting of trees or penalized regression as embedded methods with RF to attain better modeling. The construction is shown in Fig.3. Embedded methods (XGBoost and Glmnet) for feature selection and classification are faster, efficient, provides good accuracy and are scalable. Hence, their combination with RF, state of art method, may infuse a good predictive capability in the ensembled approach. Also, we performed experiments with other ML models along the workflow: - XGBoost, Glmnet (with alpha = 0, 0.3, 0.5 and 1), RF, SVM, ELM, $k$-NN, MI+RF, MI+SVM, oneR + RF, RF with rfe wrapper for comparative analysis and validation in metagenomic Use Cases.

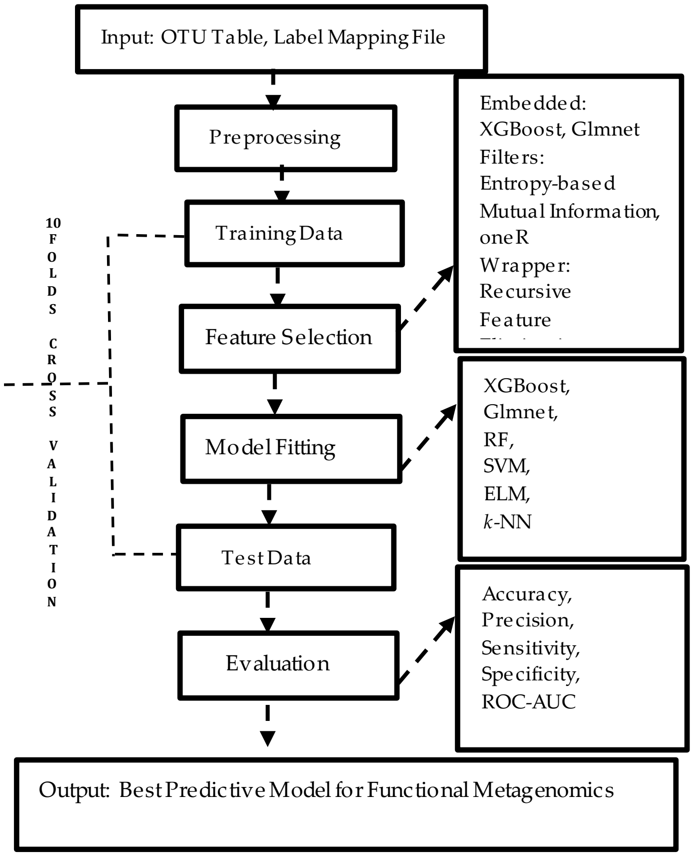

Input: OTU Table, Label Mapping File
Preprocessing
Training Data
Feature Selection
Model Fitting
Test Data
Evaluation
10 FOLDS CROSS VALIDATION
Embedded: XGBoost, Glmnet
Filters: Entropy-based Mutual Information, oneR
Wrapper: Recursive Feature
XGBoost, Glmnet, RF, SVM, ELM, k-NN
Accuracy, Precision, Sensitivity, Specificity, ROC-AUC
Output: Best Predictive Model for Functional Metagenomics

Fig.2. A general experimental workflow for functional analysis of metagenomes

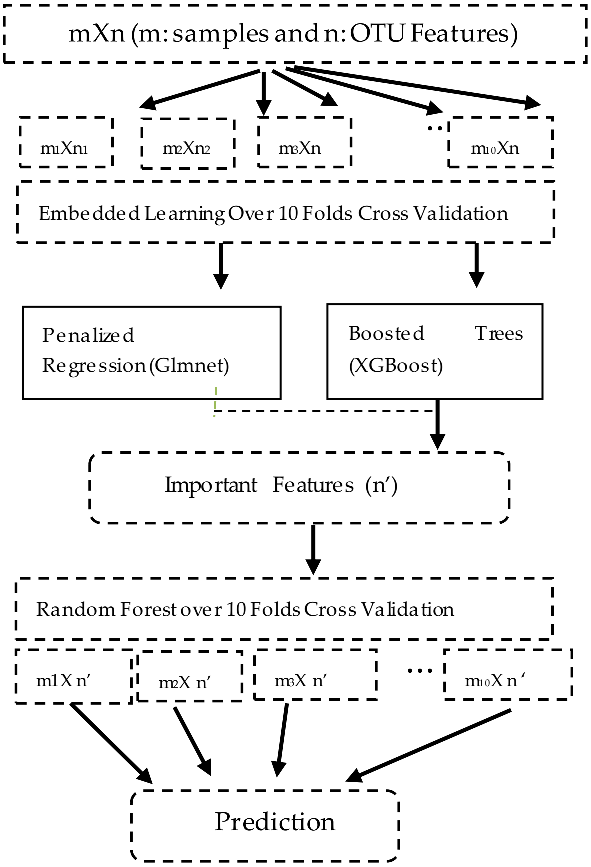

mXn (m: samples and n: OTU Features)
m1Xn1
m2Xn2
m3Xn
..
m10Xn
Embedded Learning Over 10 Folds Cross Validation
Penalized Regression (Glmnet)
Boosted Trees (XGBoost)
Important Features (n′)
Random Forest over 10 Folds Cross Validation
m1X n′
m2X n′
m3X n′
...
m10X n′
Prediction

Fig.3. The Proposed Ensembled Approach

## 4 Experimental Results And Discussion

### 4.1 Experimental Settings

The experiments were conducted to predict metagenomes, using R platform (http://www.R-project.org, version 3.0.4). The various packages[2] related to ML models and the related optimum set of configurational parameters, used in this study are listed below.

a) XGBoost Extreme Gradient boosting machines. Implementation: package XGBoost. *Configurational parameters: objective" = "binary: logistic" for binary classes and objective" = "multi: softmax"  for mutinomial classes, "nthread" = 8, "max_depth" = 3, "gamma loss reduction " = 0*

b) Glmnet Lasso, Ridge, ENet logistic classifier. Implementation: package glmnet. *Configurational parameters: family = "binomial","multinomial", alpha regularization penalty (α)=0,0.3,0.5,1.*

c) RF Random forest. Implementation: package randomForest.*Configurational parameters: ntree=100*

d) svmRadial A SVM with RBF kernel. package e1071. *Configurational parameters: kernel="radial", cost=1, gamma=0.5, scale=TRUE*

e) ELM Extreme learning machines Implementation: package elmNN *Configurational parameters: nhid = 100, actfun= "sig"*

f) *k*-NN k-Nearest Neighbor's classifier. Implementation: package class. *Configurational parameters: k = =10*

g) *rfe* Recursive Feature Elimination. Implementation: package caret. *Configurational parameters: rfeControl = rfFuncs*

h) Entropy-based Mutual Information (information. gain()) and oneR Filters. Implementation: package FSelector. *Configurational parameters: Top 20 Features*

i) Confusion Matrix. Implementation: package caret. *Evaluation metrics over $overall and $byClass associated parameters*

j) ROC. Implementation: package pROC. *Evaluation over binomial and multinomial classes by:* area under curve(ROC-AUC) values.

k) *glmnetRank.* Implementation: package SurvRank, *Rank order of Coefficients in glmnet*

l) *Random forest. Importance.* Implementation: package FSelector. *Configurational parameters: Top 10 Features*

A 10-folds cross-validation was performed for all experiments. Each of the data sets were divided into 10 partitions known as folds. One-fold was used for testing and the remaining 9 were used for training the data. The process was repeated for every fold. The time recorded for ML models is the User (CPU) time charged for the execution of user instructions of the calling process (in seconds). The running environment consisted of a system configured with AMD processor A8-7410 @2.20 GHz, Quad Core, 8 GB RAM.

### 4.2 Performance of Prediction Models

In this work, we investigated the combination of feature subset selection method and classifier models to tackle an important question of metagenomic analysis: - "Which OTUs or functions are important for differentiating the phenotypic classes and attain good prediction performance of classification in metagenomic studies?" We conducted this study to select predominant classifiers for downstream metagenomic analysis and subsequently to evaluate the efficacy of functional predictions, with and without the implementation of feature selection methods, over the high-dimensional metagenomic data.

Predictive modeling over the case studies supports in understanding the input data behavior, and an objective of this study is to identify an ML model, which is quick to train and accurate at classification in functional metagenomics. The RF [43], [44] and SVM [45] are popular state-of-the-art approaches for predicting functions in metagenomics analysis [24], [29]. The classification algorithms used in our experiments were: - XGBoost, Glmnet, RF, SVM, ELM and *k*-NN (described in section 3.2.2.1). We first applied these classifiers and evaluated their performance over the 3 Use Cases (mentioned in Materials section). We tuned Glmnet classifier with a regularization penalty of 0, 0.3, 0.5 and 1.0. The results obtained by ML algorithms (based on an optimum set of parameters) over the 3 Use Cases, ar detailed in Table 3, 4 and 5 respectively. From the obtained results, the dominant classifiers providing overall

[2] https://cran.r-project.org/web/packages/available_packages_by_name.html

good accuracy and ROC-AUC were noted as: - RF (Use Case 1), Glmnet (Use Case 2) and XGBoost (Use Case 3). The results are useful for further comparative analysis. Although the accuracy of RF is good, Glmnet and XGBoost served as scalable and faster models than RF on high-dimensional metagenomes.

TABLE 3
Performance of Classifiers (10-folds CV) over Use Case 1

| **Model** | **Time (secs)** | ***Ac*** | ***Pr*** | ***Se*** | ***Sp*** | **ROC-AUC** |
|---|---|---|---|---|---|---|
| XGBoost | 62 | 0.931 | 0.925 | 0.957 | 0.895 | 0.926 |
| Glmnet-$\alpha = 1$ (Lasso) | 29 | 0.92 4 | 0.884 | 1.000 | 0.816 | 0.908 |
| Glmnet-$\alpha = 0.5$ (ENet) | 28 | 0.950 | 0.924 | 0.995 | 0.888 | 0.942 |
| Glmnet $\alpha = 0$ (Ridge) | 244 | 0.567 | 0.635 | 0.589 | 0.529 | 0.571 |
| Glmnet $\alpha = 0.3$ | **32** | **0.951** | **0.928** | **0.992** | **0.896** | **0.944** |
| RF | **2085** | **0.953** | **0.937** | **0.985** | **0.908** | **0.947** |
| SVM (radial) | 793 | 0.591 | 0.591 | 1.000 | 0.000 | 0.500 |
| ELM (nhid=100) | 258 | 0.944 | 0.930 | 0.977 | 0.895 | 0.936 |
| *k*-NN (*K*=10) | 87 | 0.919 | 0.897 | 0.975 | 0.835 | 0.905 |

TABLE 4
Performance of Classifiers (10-folds CV) over Use Case 2

| **Model** | **Time (secs)** | ***Ac*** | ***Pr*** | ***Se*** | ***Sp*** | **ROC-AUC** |
|---|---|---|---|---|---|---|
| XGBoost | 41 | 0.979 | 0.977 | 0.990 | 0.961 | 0.976 |
| Glmnet-$\alpha = 1$(Lasso) | 34 | 0.936 | 0.911 | 0.995 | 0.842 | 0.919 |
| Glmnet-$\alpha = 0.5$ (ENet) | 37 | 0.968 | 0.955 | 0.996 | 0.920 | 0.958 |
| Glmnet $\alpha = 0$ (Ridge) | **291** | **0.993** | **0.993** | **0.996** | **0.988** | **0.992** |
| Glmnet $\alpha = 0.3$ | 42 | 0.985 | 0.980 | 0.997 | 0.966 | 0.981 |
| RF | **1712** | **0.991** | **0.987** | **0.998** | **0.978** | **0.988** |
| SVM(radial) | 181 | 0.625 | 0.625 | 1.00 | 0.000 | 0.500 |
| ELM (nhid=100) | 25 | 0.898 | 0.903 | 0.938 | 0.832 | 0.885 |
| k-NN (K=10) | 190 | 0.935 | 0.913 | 0.990 | 0.842 | 0.916 |

### 4.3 Performance of Feature Selection

RF was further tuned with feature selection methods with the aim of attaining superior performance and enabling its capability for high-dimensional Use Cases. We selected filter-based (information.gain, oneR), embedded (XGBoost, Glmnet) and wrapper-based on recursive feature elimination (rfe); as the feature selection strategies. Glmnet, SVM and XGBoost methods were also combined with filters. The ML models used in current context for analysis are listed in Table 6. We investigated how the selection of discriminative features impact the performance of prediction.

TABLE 5
Performance of Classifiers (10-folds CV) over Use Case 3

| **Model** | **Time (secs)** | ***Ac*** | ***Pr*** | ***Se*** | ***Sp*** | **ROC-AUC** |
|---|---|---|---|---|---|---|
| XGBoost | **198** | **0.770** | **0.793** | **0.623** | **0.901** | **0.730** |
| Glmnet-$\alpha = 1$(Lasso) | 254 | 0.695 | 0.742 | 0.466 | 0.865 | 0.657 |
| Glmnet-$\alpha = 0.5$ (ENet) | 299 | 0.728 | 0.725 | 0.548 | 0.883 | 0.728 |
| Glmnet $\alpha = 0$ (Ridge) | **5109** | **0.770** | **0.796** | **0.603** | **0.901** | **0.709** |
| Glmnet $\alpha = 0.3$ | 406 | 0.747 | 0.744 | 0.567 | 0.891 | 0.712 |
| RF | **3229** | **0.746** | **0.786** | **0.533** | **0.885** | **0.703** |
| SVM (radial) | 946 | 0.481 | 0.481 | 0.25 | 0.75 | 0.500 |
| ELM (nhid=100) | 45 | 0.350 | 0.437 | 0.355 | 0.796 | 0.614 |
| k-NN (K=10) | 230 | 0.585 | 0.566 | 0.400 | 0.821 | 0.636 |

TABLE 6
Ensemble of ML Models Used in the Current Study

| **ML Model** | **Feature Selection** | **Classifier** |
|---|---|---|
| RF_XB_GMi where i = 0,0.3,0.5 or 1 | XGBoost+Glmnet | RF |
| GMj_XB_GMi where i, j = 0,0.3,0.5 or 1 | XGBoost+Glmnet | Glmnet |
| RF_XB | XGBoost | RF |
| RF_IG | Mutual Information | RF |
| RF_oneR | oneR | RF |
| RF_rfe | Recursive Feature Elimination (rfe) | RF |
| SVM_IG | Mutual Information | SVM |
| XB_XB_GMi where i = 0,0.3,0.5 or 1 | XGBoost+Glmnet | XGBoost |

The results of an afore-mentioned combination of feature selection and the classifiers (listed in Table 6), were obtained on an optimum set of parameters (as listed in section 4.1), and recorded in Tables 7, 8 and 9 for Use Cases 1,2, and 3 respectively. The embedded feature selection with XGBoost and Glmnet (penalized) in combination with RF classifier, potentially provided higher accuracy and ROC-AUC values for predicting functional classes in our metagenomic Use Cases. In Use Case 1, we extended the analysis in designing an optimal feature subset by combining important features of XGBoost (features retrieved based on rank importance with associated R function of XGBoost importance ()[2]) and Top 120 features ranked by glmnetRank[2] associated with Glmnet ($\alpha = 0.3$), i.e. XB_GM0.3. We selected $\alpha = 0.3$, as it gave higher accuracy than Glmnet regularized with other $\alpha$ values of 0, 0.5 and 1 (Table 3), in this Use Case. We used glmnetRank function ()[2] to rank order the OTU features obtained by Glmnet and then selected top 120 features as we verified high ROC-

AUC on this value (Fig.4).

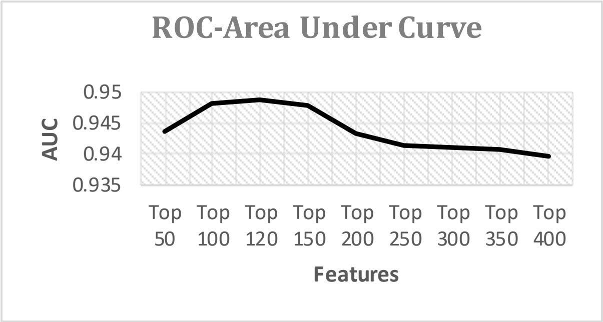


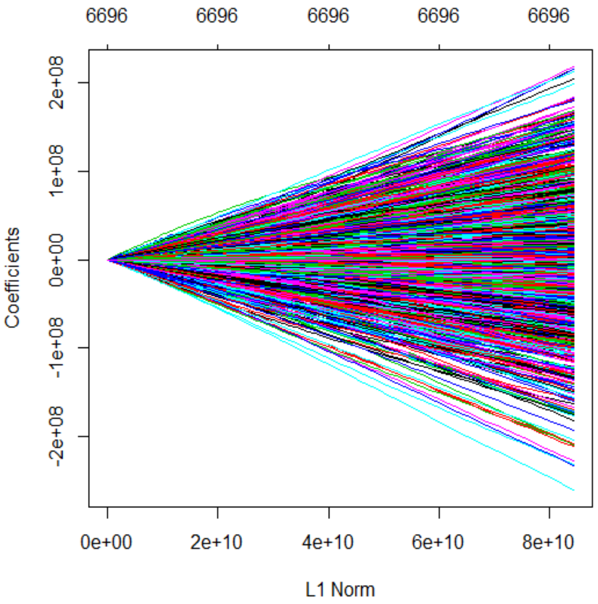
Fig.4. Selecting Top Features from Glmnet Model (α = 0.3) over Use Case 1

XB_GM0.3 enhanced the performance of supervised classification with RF as classifier (i.e. RF_XB_GM0.3). With this ensemble setting, RF performance (Time: 2085 secs, *Ac*: 0.953, ROC-AUC:0.944, Number of Features (NFS): 6697), improved to (Time: 160 secs, *Ac*: 0.960, ROC-AUC:0.952, Number of Features (NFS): 271), over Use Case 1.When XB_GM0.3 (feature selection method) was applied to Glmnet (classifier) with a ridge (α = 0), in Use Case 1, the performance improved from the *Ac* of 0.567 to0.965 (Fig.5., Fig.7. a., Table 3, Table 7). Also, the *Ac* of XGBoost improved slightly from 0.930 to 0.945 when applied to the optimal OTU feature set obtained by XB_GM0.3.

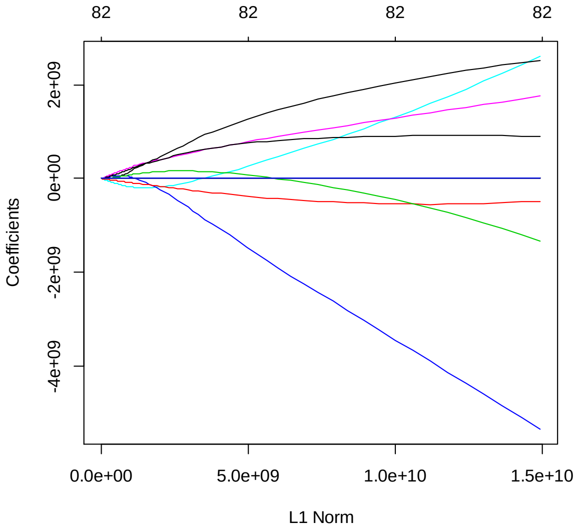
TABLE 7
Performance of Feature Selector and Classifiers (10 Folds CV) over Use Case 1 (NFS: # of Features Selected)

| Models | NFS | Time (Secs) | *Ac* | *Pr* | *Se* | *Sp* | ROC-AUC |
|---|---|---|---|---|---|---|---|
| RF_XB_GM0.3 | **270** | **160** | **0.960** | **0.946** | **0.990** | **0.914** | **0.952** |
| GM0_XB_GM0.3 | **270** | **103** | **0.965** | **0.958** | **0.985** | **0.936** | **0.960** |
| XB_XB_GM0.3 | 270 | 98 | 0.945 | 0.938 | 0.973 | 0.903 | 0.938 |
| RF_XB | 83 | **81** | 0.954 | 0.942 | 0.984 | 0.911 | 0.948 |
| RF_IG | 20 | 290 | 0.936 | 0.928 | 0.967 | 0.889 | 0.928 |
| RF_oneR | 20 | 252 | 0.767 | 0.721 | 0.987 | 0.447 | 0.717 |
| SVM_IG | 20 | 288 | 0.911 | 0.922 | 0.927 | 0.889 | 0.907 |
| RF_rfe | ---- | ~ 80K | 0.940 | 0.936 | 0.962 | 0.907 | 0.934 |

RF_XB_GM0.3 and GM0_XB_GM0.3 proved to be best ML models over Use Case 1. These models provided better performance than filters and wrappers; and proved better than models listed in Table 3. The overall performance of SVMs over this Use Case data was also improved by applying entropy-based mutual information. gain filter method (e.g. *Ac* improves from 0.591 to 0.911).



a) Glmnet Ridge over Original Feature set



b) Glmnet Ridge over Feature set obtained by RF_XB_GM0.3 method

Fig.5. Improvement in Ridge over Use Case 1 with RF_XG_GM0.3: - (a) Original and (b) Improved

In Use Case 2, ensembled combination of XGBoost and top 120 features ranked by glmnetRank ()$^2$ associated with, Glmnet at α = 0, (the hyper-parameter providing highest accuracy of Glmnet (Table 4)) as feature selector and RF as Classifier, i.e. RF_XB_GM0; provided the best performance in terms of highest *Ac* and ROC-AUC of 0.999 (Fig. 7.b). Overall, embedded feature selection strategies improved the performance over classification in this Use Case as well (Table 4, Table 8). The Use Case 3 relates to a study on dynamics of IDB disease in relation to the human microbiome. The case deals with multinomial functional classes unlike previous Use Cases of binomial classes.

TABLE 8
Performance of Feature Selector and Classifiers (10- folds CV) over Use Case 2 (NFS: # of Features Selected)

| Models | NFS | Time (Secs) | *Ac* | *Pr* | *Se* | *Sp* | ROC-AUC |
|---|---|---|---|---|---|---|---|
| RF_XB_GM0 | **250** | **409** | **0.999** | **0.999** | **1.000** | **0.998** | **0.999** |
| XB_XB_GM0 | 250 | 337 | 0.989 | 0.983 | 1.000 | 0.971 | 0.986 |
| RF_XB | 38 | **59** | 0.988 | 0.985 | 0.996 | 0.984 | 0.984 |
| RF_IG | 20 | 169 | 0.979 | 0.980 | 0.987 | 0.965 | 0.976 |
| RF_oneR | 20 | 149 | 0.626 | 0.626 | 1.000 | 0.000 | 0.500 |
| SVM_IG | 20 | 175 | 0.918 | 0.892 | 0.990 | 0.796 | 0.892 |
| RF_rfe | --- | ~ 210K | 0.994 | 0.991 | 1.000 | 0.984 | 0.992 |

Glmnet (α = 0; Ridge), provided the best accuracy in this Use Case, however the time taken by Ridge Regression in this case, is 5109 secs > 3229 secs of RF, so we did not consider the Ridge in embedded feature selection in this study related to multiple classes. We implemented some of other models listed in Table 6 over this
Use Case and attained the results recorded in Table 9.

The ensemble of XGBoost for feature selection (embedded method) and RF for classification i.e. RF_XB produced best results with (Time: 46 secs, average *Ac*: 0.795, average ROC-AUC: 0.746, Number of Features (NFS): 449,), which is an improvement over RF applied on original data set with (Time: 3229 secs, average *Ac*: 0.746, average ROC-AUC: 0.703, Number of Features (NFS): 10997) (Fig.7.c).

The result was also validated by comparing RX_XB with RF statistically (using t-test) over 10 folds cross-validated data. The significant improvement was achieved ($p < 0.05$).

### 4.3 Comparison with Previous Study

Thereafter, we investigated the procedure of selecting a reduced subset (Top-10) by using RF algorithm (RandomForest filter i.e. random.forest.Importance ()[2] as indicated in section 4.1), before the application of classifier RF.

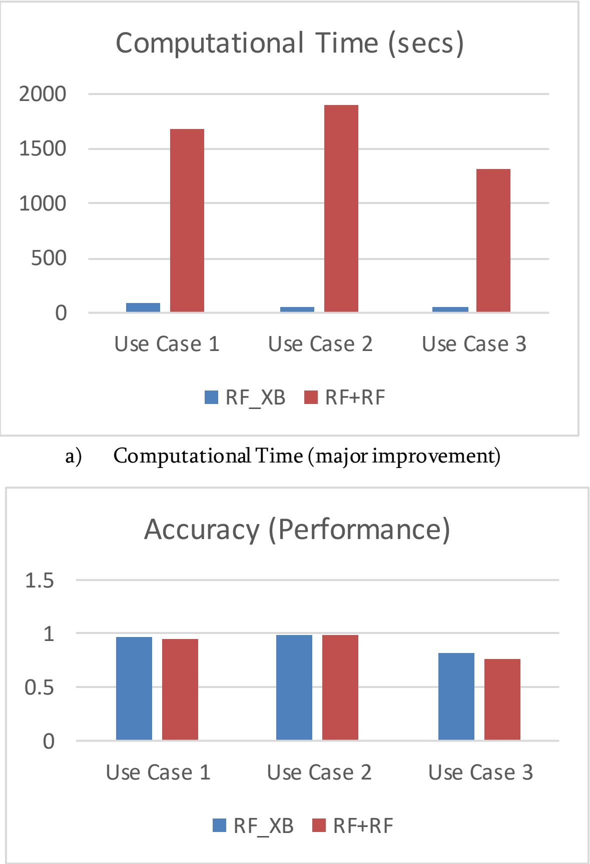
TABLE 9
Performance of Feature Selector and Classifiers (10- folds CV) over Use Case 3 (NFS: # of Features Selected)

| Models | NFS | Time (Secs) | *Ac* | *Pr* | *Se* | *Sp* | ROC-AUC |
|---|---|---|---|---|---|---|---|
| RF_XB | **449** | **46** | **0.795** | **0,846** | **0.618** | **0.907** | **0.746** |
| RF_IG | 20 | 151 | 0.726 | 0.750 | 0.550 | 0.882 | 0.706 |
| RF_oneR | 20 | 154 | 0.600 | 0.550 | 0.894 | 0.344 | 0.626 |
| SVM_IG | 20 | 152 | 0.685 | 0.683 | 0.463 | 0.861 | 0.670 |
| RF_rfe | --- | ~25K | 0.710 | 0.740 | 0.530 | 0.870 | 0.695 |

This embedded RF approach was suggested by Pasolli et al. [32] for meta-analysis of large metagenomic datasets. The results of this ensemble (RF+RF) over our Use casesare listed in Table 10. Comparing the performance of RF+RF with RF_XB with 10-folds cross-validation, our proposed method of RF_XB exhibited better results than RF+RF (Fig.6, Table 7,8,9,10)

TABLE 10
Performance of RF+RF (Top 10 of Features Selected)

| RF + RF Model with 10 folds CV | Time (Secs) | *Ac* | *Pr* | *Se* | *Sp* | ROC-AUC |
|---|---|---|---|---|---|---|
| Use Case 1 | 1680 | 0.947 | 0.938 | 0.972 | 0.911 | 0.941 |
| Use Case 2 | 1900 | 0.978 | 0.978 | 0.988 | 0.962 | 0.975 |
| Use Case 3 | 1315 | 0.762 | 0.772 | 0.582 | 0.898 | 0.730 |



a) Computational Time (major improvement)

b) Accuracy (marginal Improvements)

Fig.6. Comparing RF+RF and RF_XB in terms of (a) Time and (b) Performance

### 4.4 The effect of tuning the hyper-parameters of RF And XGBoost

We further studied the impact of tuning parameters of XGBoost and RF over the Use Cases, to evaluate classification performance of RF_XB. The *Ac* and ROC-AUC of

RF_XB were enhanced to 0.962 and 0.957 respectively with tuning of max_depth parameter of XGBoost to 5 and number of trees in RF to 500 in Use Case 1 (Table 11). Also, we observed that the increasing the max_depth of XGBoost (1 to 5), improves the performance in this Use Case.

The performance of differencing human enterotypes (Use Case 2), was also influenced by parameter tuning of XGBoost and RF (shown in Table 12). But this case suggested that it is not always necessary that with increasing *max_depth* of XGBoost, we may improve on computational performance unlike in Use Case 1.

TABLE 11
Parametric Tuning (XGBoost & RF) and Performance Analysis of RF_XB ((10-folds CV) ) over Use Case 1 (NFS: # of Features Selected)

| **Max Depth (XGBoost)** | **Number of Trees (RF)** | **NFS** | ***Ac*** | ***Pr*** | ***Se*** | ***Sp*** | **ROC-AUC** |
|---|---|---|---|---|---|---|---|
| **1** | RF: 100 | 12 | 0.935 | 0.932 | 0.959 | 0.897 | 0.929 |
| | RF:500 | 12 | 0.935 | 0.932 | 0.959 | 0.899 | 0.930 |
| **3** | RF: 100 | 82 | 0.954 | 0.942 | 0.984 | 0.911 | 0.948 |
| | RF:500 | 82 | 0.950 | 0.935 | 0.984 | 0.900 | 0.942 |
| **5** | RF: 100 | 104 | 0.958 | 0.946 | 0.986 | 0.920 | 0.953 |
| | **RF:500** | **104** | **0.962** | **0.948** | **0.989** | **0.924** | **0.957** |

TABLE 12
Parametric Tuning (XGBoost & RF) and Performance Analysis of RF_XB ((10-folds CV) over Use Case 2 (NFS: # of Features Selected)

| **Max Depth (XGBoost)** | **Number of Trees (RF)** | **NFS** | ***Ac*** | ***Pr*** | ***Se*** | ***Sp*** | **ROC-AUC** |
|---|---|---|---|---|---|---|---|
| 1 | RF: 100 | 9 | 0.975 | 0.973 | 0.986 | 0.955 | 0.970 |
| | RF:500 | 9 | 0.974 | 0.973 | 0.985 | 0.955 | 0.970 |
| **3** | RF: 100 | 37 | 0.988 | 0.985 | 0.996 | 0.984 | 0.984 |
| | RF:500 | 37 | 0.972 | 0.967 | 0.989 | 0.942 | 0.966 |
| **5** | RF: 100 | 47 | 0.993 | 0.993 | 0.996 | 0.988 | 0.992 |
| | RF:500 | 47 | 0.992 | 0.992 | 0.995 | 0.986 | 0.990 |

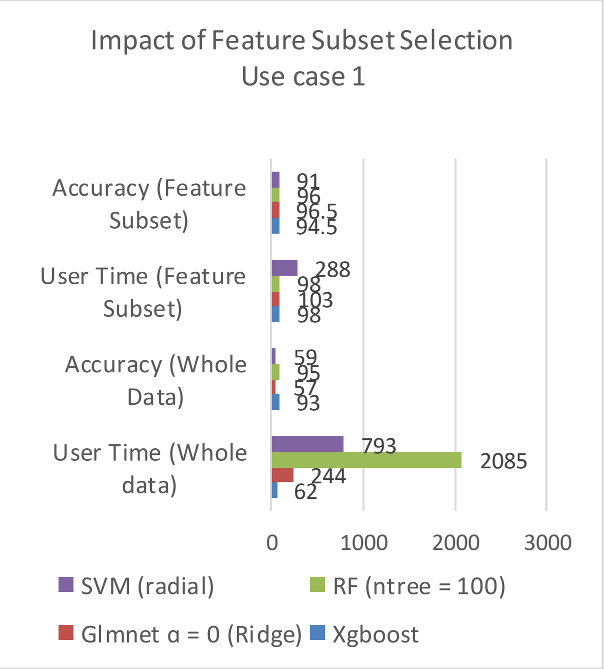
In Use Case 3, RF_XB method provided the best performance, when it was tuned to 500 as the number of trees in RF and *max_depth* in XGBoost as 1. There was a significant improvement in the performance of RF_XB over RF from 0.746 to 0.817.

Overall, it is suggested that the embedded methods of feature selection with parametric tunings (e.g. RF: number of trees: 500 and max depth in XGBoost as 1,3,5), may enhance the performance of analysis.

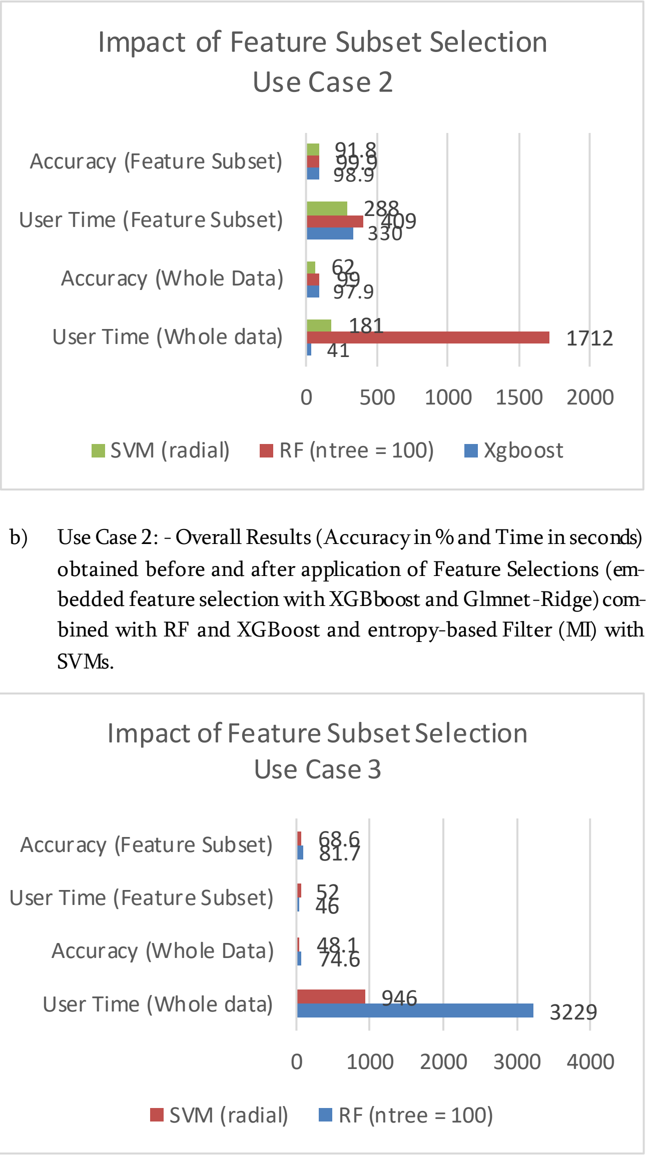
## 4.5 The study of the dynamics of the proposed model

The application of embedded methods over all the Use Cases, indicated a significant improvement in comparison to run time of state-of-the-art RF and other strategies of filters and wrappers ($p<0.01$). To summarize the experimental results, we highlight the following key discussion points:-

1. RF has been established as the best classifier in literature [24], [29], for functional metagenomic analysis. But we proposed Boosted Trees(XGBoost) and Penalized LR (Glmnet with penalty) as alternative and efficient methods for classification of high-dimensional metagenomes.
2. The ensemble of embedded feature selection using XGBoost and/or Glmnet and RF as classifier, potentially provides high predictive performance in metagenomic case studies in comparison to filter and wrapper methods over RF.
3. The proposed embedded ML methods (feature selection + classifier) outperform other classifiers namely, SVM, ELM, and *k*-NN.
4. Tuning the hyper-parameters in the ensemble (embedded) approach, may further improve the performance of analysis.
5. The embedded method (Feature Selection + Classification) of XGBoost + RF may provide a competitive marginal to competitive improvement in accuracy but a major improvement in computational time over



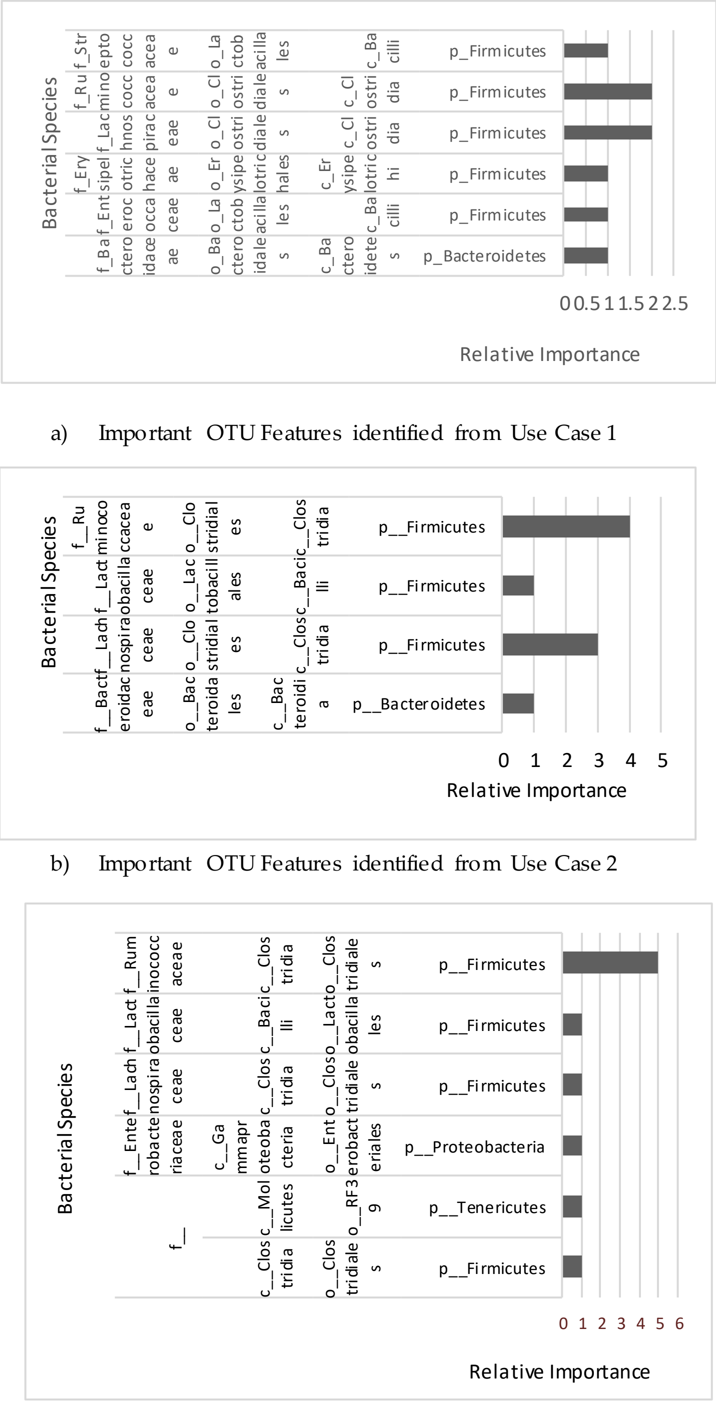
a) Use Case 1: - Overall Results (Accuracy in % and Time in seconds) obtained before and after application of Feature Selections (embedded feature selection with XGBboost and Glmnet-Ridge) combined with RF and XGBoost and entropy-based Filter (MI) with SVMs.



b) Use Case 2: - Overall Results (Accuracy in % and Time in seconds) obtained before and after application of Feature Selections (embedded feature selection with XGBboost and Glmnet-Ridge) combined with RF and XGBoost and entropy-based Filter (MI) with SVMs.

c) Use Case 3: - Overall Results (Accuracy in % and Time in seconds) obtained before and after application of Feature Selections (embedded feature selection obtained from XGBboost) combined with RF; and entropy-based (MI) Filters with SVMs

Fig.7. Improvements over ML models in predictive performance with feature selectors using (a) Use Case1, (b) Use Case 2, (c) Use Case 3



a) Important OTU Features identified from Use Case 1

b) Important OTU Features identified from Use Case 2

c) Important OTU Features identified from Use Case 3

Fig.8. Dominant Predictive OTU features for Classifying Human Microbiome in context of (a) Use Case 1, (b) Use Case 2 and (c) Use case 3

## 5. Conclusion

The current study makes important methodological contributions for use of ML models in the field of functional metagenomics. In this study, we uniformly evaluated metagenomic human microbiome data from 3 studies and used 10 folds cross-validation to evaluate the performance of ML models used for prediction of functions. We recommended some of the best models in general for functional metagenomic analysis. We have shown that embedded feature selection strategies of XGBoost or Glmnet are most effective in dealing with high-dimensional metagenomic data, as they are faster, provide better performance and are scalable with large scale metagenomic data.

Overall, the classification algorithms of RF, Glmnet (Penalized LR) with Ridge and XGBoost, resulted in the higher predictive performance. RF has been established as one of the best models for classifying metagenomes in listed in the literature [8], [24], [29]. However, we propose that in terms of computational cost, embedded feature selection methods outperform RF by scaling well to high dimensions and hence could also be combined with RF to further improve its performance. We proposed an ensemble predictor in which features were selected using scalable boosting of trees (XGBoost) and /or in combination with Penalized LR (Glmnet) and RF acts a classifier. The approach is effective in functional metagenomics.

The method provided marginally better or competitive accuracy in comparison to RF by selecting important features but reduced the computational cost significantly. SVM on a features subset obtained by entropy-based filter performed better than SVM on whole set of features. Hence, this reflects overall feature engineering may play an important role in analyzing the data and improving overall performance in metagenomic Use Cases. We hope that these results would inform future human microbiome studies related to dietary effects on human microbiota, enterotypes and IBD disease, and creates base knowledge for further scientific research.

In future, we would like to experiment with phylogeny-aware ML models to characterize not just the abundance counts of OTUs but also the relationships between various OTUs at different taxonomic levels; and to propose a new framework for metagenomics functional analysis by in corporating such association analysis. Also, we propose to explore other advances in ML such as deep learning with neural nets, sparse graphs and kernel-based similarity and gene expression networks (co-occurrence, inter relations), for increasing the reliability of microbiome analysis in our workflow.

## ACKNOWLEDGMENT

This research is supported by Research Strategy Fund of Ulster University, United Kingdom and the MetaPlat project, (www.metaplat.eu), funded by H2020-MSCA-RISE-2015. Prof. Huiru Zheng ( h.zheng@ulster.ac.uk) is the corresponding author of this research.

## REFERENCES

[1] P. Hugenholtz and G. W. Tyson, "Metagenomics," *Nature*, vol. 455, no. August 2006, pp. 481–483, 2008.

[2] The NIH HMP Working Group, "The NIH Human Microbiome Project," *Genome Res.*, vol. 19, no. 12, pp. 2317–2323, 2009.

[3] American Gut Project. http://americangut.org/about/. Accessed July 2017.

[4] J. A. Gilbert *et al.*, "The Earth Microbiome Project," in *1st EMP meeting on sample selection and acquisition*, 2010.

[5] S. C. Schuster, "Next-generation sequencing transforms today's biology," *Nat. Methods*, vol. 5, no. 1, pp. 16–18, 2008.

[6] F. Sanger, S. Nicklen, and a R. Coulson, "DNA sequencing with chain-terminating inhibitors.," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 74, no. 12, pp. 5463–7, 1977.

[7] J. C. Wooley, A. Godzik, and I. Friedberg, "A primer on metagenomics," *PLoS Computational Biology*, vol. 6, no. 2. 2010.

[8] H. Soueidan and M. Nikolski, "Machine learning for metagenomics: methods and tools," *arXiv*, pp. 1–23, 2015.

[9] J. L. Bouchot, W. L. Trimble, G. Ditzler, Y. Lan, S. Essinger, and G. Rosen, "Advances in Machine Learning for Processing and Comparison of Metagenomic Data," *Comput. Syst. Biol. From Mol. Mech. to Dis. Second Ed.*, pp. 295–329, 2013.

[10] B. Kotsiantis, Sotiris, I. Zaharakis, and P. Pintelas. "Supervised machine learning: A review of classification techniques." ,3-24, 2004.

[11] K. N. Lam, J. Cheng, K. Engel, J. D. Neufeld, and T. C. Charles, "Current and future resources for functional metagenomics," *Front. Microbiol.*, vol. 6, no. OCT, 2015.

[12] H. Li, ""Microbiome, Metagenomics, and High-Dimensional Compositional Data Analysis," Annu. Rev. Stat. Its Appl., vol. 2, no. 1, pp. 73–94, 2015.

[13] D. Asir Antony Gnana Singh, S. Appavu alias Balamurugan KLN, and E. Jebamalar Leavline, "Literature Review on Feature Selection Methods for High-Dimensional Data," *Int. J. Comput. Appl.*, vol. 136, no. 1, pp. 975–8887, 2016.

[14] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 16–28, 2014.

[15] I. Cho and M. J. Blaser, "The human microbiome: at the interface of health and disease.," *Nat. Rev. Genet.*, vol. 13, no. 4, pp. 260–270, 2012.

[16] D. Gevers *et al.*, "The Human Microbiome Project: A Community Resource for the Healthy Human Microbiome," *PLoS Biol.*, vol. 10, no. 8, 2012.

[17] C. Manichanh, N. Borruel, F. Casellas, and F. Guarner, "The gut microbiota in IBD.," *Nat. Rev. Gastroenterol. Hepatol.*, vol. 9, no. October, pp. 599–608, 2012.

[18] P. J. Turnbaugh *et al.*, "A core gut microbiome in obese and lean twins," *Nature*, vol. 457, no. 7228, pp. 480–484, 2009.

[19] J. U. Scher and S. B. Abramson, "The microbiome and rheumatoid arthritis.," *Nat. Rev. Rheumatol.*, vol. 7, no. 10, pp. 569–78, 2011.

[20] D. McDonald, A. Birmingham, and R. Knight, "Context and the human microbiome.," *Microbiome*, vol. 3, no. 1, p. 52, 2015.

[21] N. Arslan, "Obesity, fatty liver disease and intestinal microbiota", World Journal of Gastroenterology, vol. 20, no. 44, p. 16452, 2014.

[22] D. R. Learman *et al.*, "Biogeochemical and microbial variation across 5500 km of Antarctic surface sediment implicates organic matter as a driver of benthic community structure," *Front. Microbiol.*, vol. 7, no. MAR, pp. 1–11, 2016.

[23] S. Hiraoka *et al.*, "Genomic and metagenomic analysis of microbes in a soil environment affected by the 2011 Great East Japan Earthquake tsunami.," *BMC Genomics*, vol. 17, no. 1, p. 53, 2016.

[24] D. Knights, E. Costello and R. Knight, "Supervised classification of human microbiota", FEMS Microbiology Reviews, vol. 35, no. 2, pp. 343-359, 2011.

[25] J. T. Wassan *et al.*, "An Integrative Approach for the Functional Analysis of Metagenomic Studies," International Conference on Intelligent Computing (ICIC), pp. 421–427, Springer, Cham, 2017.

[26] H. Wang, H. Zheng, F. Browne, R. Roehe, R. Dewhurst, F. Engel, M. Hemmje, X. Lu and P. Walsh, "Integrated metagenomic analysis of the rumen microbiome of cattle reveals key biological mechanisms associated with methane traits", Methods, vol. 124, pp. 108-119, 2017.

[27] P. Walsh, C. Palu, B.Kelly, B.Lawor, J.Wassan, H.Zheng, and H.Wang. "A metagenomics analysis of rumen microbiome." In 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 2077-2082. IEEE, 2017.

[28] D. Toyama, L. Kishi, C. Santos-Júnior, A. Soares-Costa, T. de Oliveira, F. de Miranda and F. Henrique-Silva,

"Metagenomics Analysis of Microorganisms in Freshwater Lakes of the Amazon Basin", Genome Announcements, vol. 4, no. 6, pp. e01440-16, 2016.

[29] A. Statnikov *et al.*, "A comprehensive evaluation of multicategory classification methods for microbiomic data.," *Microbiome*, vol. 1, no. 1, p. 11, 2013.

[30] C. Yang *et al.*, "An ecoinformatics tool for microbial community studies: Supervised classification of Amplicon Length Heterogeneity (ALH) profiles of 16S rRNA," *J. Microbiol. Methods*, vol. 65, no. 1, pp. 49–62, 2006.

[31] B. Wingfield and S. Coleman, "A Metagenomic Hybrid Classifier for Paediatric Inflammatory Bowel Disease," pp. 1083–1089, 2016.

[32] E. Pasolli, D. T. Truong, F. Malik, L. Waldron, and N. Segata, "Machine Learning Meta-analysis of Large Metagenomic Datasets: Tools and Biological Insights," *PLoS Comput. Biol.*, vol. 12, no. 7, 2016.

[33] R. Knight, C. Brown, J. Caporaso, J. Clemente, D. Gevers, E. Franzosa, S. Kelley, D. Knights, R. Ley, A. Mahurkar, J. Ravel and O. White, "Advancing the Microbiome Research Community", Cell, vol.159, no. 2, pp. 227-230, 2014

[34] S. Deng, L. Zhu and D. Huang, "Predicting Hub Genes Associated with Cervical Cancer through Gene Co-Expression Networks", IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 13, no. 1, pp. 27-35, 2016.

[35] S. Deng, L. Zhu and D. Huang, "Mining the bladder cancer-associated genes by an integrated strategy for the construction and analysis of differential co-expression networks", BMC Genomics, vol. 16, no.3, p. S4, 2015.

[36] P. J. Turnbaugh, V. K. Ridaura, J. J. Faith, F. E. Rey, R. Knight, and J. I. Gordon, "The Effect of Diet on the Human Gut Microbiome: A Metagenomic Analysis in Humanized Gnotobiotic Mice," vol. 1, no. 6, 2009.

[37] O. Koren *et al.*, "A Guide to Enterotypes across the Human Body : Meta-Analysis of Microbial Community Structures in Human Microbiome Datasets," vol. 9, no. 1, 2013.

[38] H. Jonas, Colin J. Brislawn, R. Lamendella, Y.Vázquez-Baeza, William A. Walters, Lisa M. Bramer, Mauro D'Amato et al. "Dynamics of the human gut microbiome in inflammatory bowel disease." Nature microbiology 2 , 2017.

[39] T. Chen and C. Guestrin, "XGBoost : A Scalable Tree Boosting System," pp. 785–794, 2016.

[40] J. H. Friedman,"Greedy function approximation: a gradient boosting machine. Annals of statistics", pp.1189-1232,2001.

[41] J.Hosmer, D.W., S.Lemeshow, R.Sturdivant, "Applied logistic regression",vol.398,.John Wiley & Sons,2013.

[42] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net - Zou - 2005 - Journal of the Royal Statistical Society: Series B (Statistical Methodology) - Wiley Online Library," *... R. Stat. Soc. Ser. B (Statistical ...*, 2005.

[43] L. Breiman and A. Cutler, "Breiman and Cutler's random forests for classification and regression," *Packag. "randomForest,"* p. 29, 2012.

[44] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[45] L. Saitta, "Support-Vector Networks," vol. 297, pp. 273–297, 1995.

[46] G. Huang, Q. Zhu, and C. Siew, "Extreme Learning Machine : A New Learning Scheme of Feedforward Neural Networks I]," pp. 985–990,2004.

[47] P.Cunningham, and S.J.Delany."k-Nearest neighbour classifiers." Multiple Classifier Systems , vol 34, pp 1-17, 2007.

[48] Y. Saeys, I. Inza, and P. Larra.aga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19. pp. 2507–2517, 2007.

[49] J. Vergara and P. Estévez, "A review of feature selection methods based on mutual information", Neural Computing and Applications, vol. 24, no. 1, pp. 175-186, 2013.

[50] R. Holte, "Very Simple Classification Rules Perform Well on Most Commonly Used Data sets", *Machine Learning*, vol. 11, pp. 63, 1993.

[51] M. Hall, "Correlation-based Feature Selection for Machine Learning," Methodology, vol. 2, no. April, pp. 1–5, 1999.

[52] P. M. Granitto, C. Furlanello, F. Biasioli, and F. Gasperi, "Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products," vol. 83, pp. 83–90, 2006.

[53] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.

**Jyotsna Talreja Wassan** is pursuing Ph.D. from School of Computing, Ulster University, U.K., and her current research area is "Integrative Data Analytics in Metagenomics". She is serving as an Assistant Professor in Dept. of Computer Science, Maitreyi College, University of Delhi, INDIA, since 2010 and currently is on academic leave. She has published papers in international journals and conference proceedings, e-lessons, and book chapters. She worked as a Software Engineer in her early career at ST. Microelectronics Pvt. Ltd., India and received Silver Recognition for FALCON project.

**Haiying Wang (M-'05)** received the Ph.D. degree on artificial intelligence in biomedicine in 2004 and he is currently a Reader in the School of Computing at Ulster University, UK. His research area includes artificial intelligence, complex network analysis, computational biology, and bioinformatics. He has a research interest and expertise in network-based approaches to the field of systems biology and metagenomics. Since 2004, he has published more than130 peer-reviewed research papers in international journals and conference proceedings.

**Fiona Browne** is a Lecturer in Computing Science at the Ulster University since 2013 with over 8 years research experience and 3 years industrial experience. She is a Fellow of the Higher Education Academy. In 2009, she received a PhD on Artificial Intelligence in Bioinformatics from the Ulster University. She worked as a research associate for Ulster University (EU-FP6 funded CARDIOWORKBENCH project) and as a Senior Software Developer at PathXL. She joined Queen's University Belfast as a Research Fellow on an Invest NI START project. She has published various papers in peer reviewed international journals and conference proceedings.

**Huiru Zheng (SM'03)** received the Ph.D. degree on data mining and bioinformatics from Ulster University, UK, in 2003. Her research area lies on the broad area of healthcare informatics, including bioinformatics, medical informatics, data mining and artificial intelligence and their applications on systems biology, telecare, and tele-medicine. She has published over 230 research papers in peer reviewed international journals and conferences. Prof. Zheng is currently a Professor of Computer Science with the School of Computing at Ulster University.