RESEARCH

MetaAnalyst: a user-friendly tool for metagenomic biomarker detection and phenotype classification

Mustafa Alshawaqfeh[1*], Salahelden Rababah[1,2], Abdullah Hayajneh[3], Ammar Gharaibeh[1] and Erchin Serpedin[3]

Abstract

Background: Many metagenomic studies have linked the imbalance in microbial abundance profiles to a wide range of diseases. These studies suggest utilizing the microbial abundance profiles as potential markers for metagenomic-associated conditions. Due to the inevitable importance of biomarkers in understanding the disease progression and the development of possible therapies, various computational tools have been proposed for metagenomic biomarker detection. However, most existing tools require prior scripting knowledge and lack user friendly interfaces, causing considerable time and effort to install, configure, and run these tools. Besides, there is no available all-in-one solution for running and comparing various metagenomic biomarker detection simultaneously. In addition, most of these tools just present the suggested biomarkers without any statistical evaluation for their quality.

Results: To overcome these limitations, this work presents MetaAnalyst, a software package with a simple graphical user interface (GUI) that (i) automates the installation and configuration of 28 state-of-the-art tools, (ii) supports flexible study design to enable studying the dataset under different scenarios smoothly, iii) runs and evaluates several algorithms simultaneously iv) supports different input formats and provides the user with several preprocessing capabilities, v) provides a variety of metrics to evaluate the quality of the suggested markers, and vi) presents the outcomes in the form of publication quality plots with various formatting capabilities as well as Excel sheets.

Conclusions: The utility of this tool has been verified through studying a metagenomic dataset under four scenarios. The executable file for MetaAnalyst along with its user manual are made available at `https://github.com/Rababa-Salahaldeen/MetaAnalystV1`.

Keywords: Metagenomics; Biomarker detection; Phenotype classification; Graphical user interface; Software

Background

Recent advances in high throughput sequencing technologies have opened the door to a new era for genetic studies, called *metagenomics*. In contrast to the conventional cultivation-based approaches, metagenomics enables the characterization of compositional and functional profiles of microbial colonies directly from environmental samples. Increasing number of metagenomic studies have revealed strong associations between the imbalance in microbial abundance profiles and a wide range of diseases such as obesity [1, 2], diabetes [3], inflammatory bowel disease (IBD) [4], and cancer [5, 6]. These results suggest utilizing metagenomic data for identifying potential biomarkers and developing phenotype classification models for microbial-associated diseases.

In addition to the contribution of the biomarkers in understanding the biological process under study, biomarker detection and phenotype predictive models play a central role in translating the embedded information in metagenomic datasets into clinical applications. One potential application is to utilize the detected markers for the development of potential therapies and treatments for microbial related diseases. Another application is to integrate the abundance levels of the suggested biomarkers into a single numeric

[*] Correspondence: mustafa.shawaqfeh@gju.edu.jo
[1] School of Electrical Engineering and Information Technology, German Jordanian University, Amman, Jordan
Full list of author information is available at the end of the article

value, called the *dysbiosis index*, that measures and tracks the disease activity [7, 8].

Therefore, several algorithms and computational tools have been proposed for biomarker detection such as LEfSe [9], RPCA [10], RegLRSD [11], IMG/M [12], MeAtML [13], Fizzy [14], Boruta [15], ENNB [16], MetagenomeSeq [17], MicrobiomeDDA [18], ShotgunFunctionalizeR [19], MetaStats [20], Raida [21], FANTOM [22]. Due to the similarity between metagenomic data sequence-based transcriptomics, tools that were developed originally for analyzing RNA sequencing (RNA-seq) data such as edgeR [23] and DESeq2 [24] can be applied to analyzing metagenomic data. In addition, conventional standard hypothesis testing (e.g., chi-squared, log-t, t-test, Leven Quadrati, Leven absolute, Wilcoxon rank sum, Brown-Forythe, Welch and Kolmogorov-Smirnov) and feature selection techniques (e.g., ReliefF [25], Pearson correlation [26], BSS/WSS [26]) have been suggested for finding differentially abundant microbes in metagenomic data. It is worth to mention that there exist various general purpose (i.e., not dedicated to metagenomic) tools that packed several feature selection techniques to find the important features (markers) such as FeatureSelect [27] and MetaFS [28]. Table 1 summarizes the characteristics of these tools.

However, the majority of these tools lack a user-friendly interface, and more challenging, most of them are of command-line nature, which is less comfortable compared to graphical user interface (GUI)-based software. Besides, these methods were developed using different programming languages (R, Python, C, C++, Matlab, etc), and their operation requires handling version compatibility and package dependencies related issues. Therefore, installing, configuring, and running these tools present sometimes a serious challenge for researchers with limited background in such professional soft skills. This problem becomes more challenging in scenarios where these tools are required to be installed on a large number of workstations (e.g., educational and research laboratories). One further limitation of several existing tools is that they accept one or few file types (e.g., xlsx, txt, biom). Even more, some tools require the data to be arranged in a certain format in the input data file. This decreases the comfort of utilizing these tools, especially when dealing with human-unreadable files such as biom files.

Another challenge is the lack of an *all-in-one* solution that provides a researcher in the field of metagenomics with an easy solution to conduct analysis over multiple tools simultaneously. This feature becomes more demanding if it is combined with the fact that there is no golden method that provides reliable results over all datasets. Specifically, the lists of potential markers that are generated by different algorithms vary significantly [35, 10, 11]. This variation stems from the underlying assumptions behind each method and the characteristics of the input data. Thus, using a variety of biomarker detection algorithms is crucial to enable the researcher to explore the biological problem from different angles (i.e., different algorithms may suggest different markers). Therefore, it is informative to obtain and compare the suggested signatures of several meteganomic biomarker detection techniques simultaneously. To achieve this goal, the current standard approach is to install and run the targeted tools individually. Then, combine the obtained results from these tools manually to generate comparison tables and figures. This involves an additional burden and consumes a significant amount of time and effort.

Besides, the majority of existing tool for metagenomic biomarker discovery lack the flexibility of reformatting the input dataset to enable studying the data under different conditions. In particular, the fundamental step for comparative analysis such as biomarker detection and phenotype classification is to divide the samples into positive and negative classes based on their class labels. Typically, biological samples are annotated with several kinds of information such as health status, body site location, gender. In order to construct the positive and negative cohorts, existing tools with one-level labeling capability enable the user to conduct comparative analysis with respect only to one criterion. To illustrate, assume that the health status (i.e., healthy or diseased) and gender information of the samples are available. With one-level labeling capability, a researcher can directly compare healthy and diseased samples (irrespective of the gender) or to compare male and female subjects (irrespective of the health status). In other words, the user can divide the samples into positive and negative groups based only on one criterion (either health status or gender). However, if the user is interested in creating positive and negative cohorts by combining the two criteria (i.e., health status and gender), then the researcher needs to do this manually. For example, assume that it is required to compare healthy and diseased females (male samples are excluded), then the researcher needs to manually divide the samples into two groups. The first group represents the positive class and it is composed of “diseased and female” subjects, while the second group compromises the “healthy and female” samples and represents the negative class. This one-level labeling becomes more inconvenient to provide flexible study design if the original data includes several levels of labels and the researcher is interested in studying various scenarios (by combining different levels of samples’ labels).

Table 1 The characteristics of existing packages for biomarker detection. Y: Yes, N: No, CL: Command Line.

Tool	Language	Interface	Input Files	Flexible Data Format	Flexible Study Design	Preprocessing	Evaluate Biomarkers	Run Multiple Algorithms
Boruta [15]	-R	-CL	-R frame	N	N	N	N	Y
edgeR [23]	-R	-CL	-R frame	N	N	-Filtering -Normalization	Y	N
DESeq2 [24]	-R	-CL	-R frame	N	N	N	N	N
ENNB [16]	-R	-CL	-R frame	N	N	-Normalization	N	N
MetagenomeSeq [17]	-R	-CL	- R frame	N	N	-Normalization	N	N
MicrobiomeDDA [18]	-R	-CL	-R frame	N	N	-Normalization	N	N
ShotgunFunctionalizeR [19]	-R	-CL	- R frame	N	N	N	N	Y
MetaStats [20]	-R	-CL	-R frame	N	N	-Normalization	N	N
Raida [21]	-R	-CL	-R frame	N	N	N	N	N
LEfSe [9]	-Python	-Web	-Tabular	Y	Y	N	N	N
MetaML [13]	-Python	-CL	-tsv	N	N	N	Y	N
Fizzy [14]	-Python	-CL	-biom -csv	N	N	N	N	N
"stats" package of R [29]	-R	-CL	-R frame	N	N	N	N	N
FANTOM* [22]	-Python	-GUI	-txt	NA	N	N	N	N
STAMP [30]	-Python	-GUI	-tsv	N	N	-Filter unclassiifed reads	N	N
XIPE-TOTEC [31]	-Python	-CL -Web	-tsv	N	N	N	N	N
Microbiome Analyst [32]	-Java -R -Java Script	-Web	-txt -csv -biom	N	N	-Filtering -Scaling -Normalization -Transformation	Y	N
METAREP [27]	-Python -Java -Matlab	-CL -GUI	-txt -xlsx -mat	N	N	N	Y	N
DAME [33]	-R	-Web	-biom -csv -trf -HDF5 -JSON	N	Y	-Filtering	Y	N
ShinyMB [34]	-R	-Web	-csv -tsv	N	N	-Stratification	Y	N
MetaFS [28]	-Web server	-Web	-csv	N	N	-Centering -Transformation -Scaling -Normalization	Y	N
RPCA* [10]	-Matlab -C	-CL	-xls	N	N	N	N	N
RegLRSD [11]	-Matlab -C	-GUI	-xls	N	N	N	N	N
MetaAnalyst	-Matlab	-GUI	-csv -tsv -xls -mat -biom -json	Y	Y	-Scaling -Normalization -Centering	Y	Y

* The implementation of the code is not available.

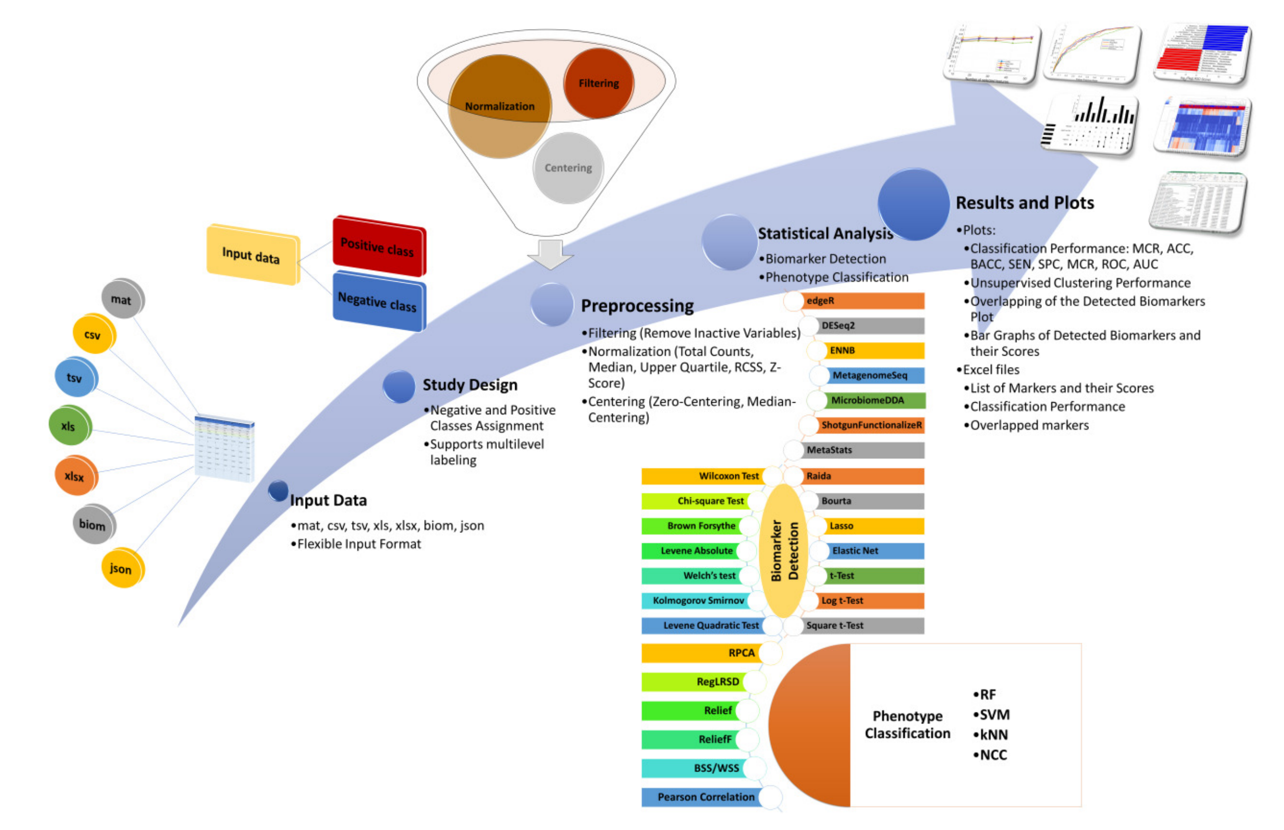


Figure 1 The general workflow of MetaAnalyst is composed of five steps. Step 1: input data. Step 2: study design. Step 3: preprocessing. Step 4: statistical analysis. Step 5: results and plots.

One additional serious limitation of several existing tools is that they provide the researcher with the suggested list of biomarkers without performance assessment. In the field of metaproteomics, researchers have payed special attention for evaluating the suggested markers. For example, the authors in [28] have designed an online tool, named MetaFS, that is designed to evaluate the performance of 13 metaproteomics biomarker detection algorithms using four evaluation criteria (clustering, classification, consistency, and prediction of spiked protein). As an additional example, the authors in [36] have conducted comprehensive assessment to 14 biomarker detection algorithms using two criteria: (i) classification power, and (ii) spiked protein discovery. Therefore, it is crucial to develop a user friendly tool to quantify the quality of the detected metagenomic biomarkers.

In order to provide a remedy for these challenges and to improve the efficiency of metagenomic analysis, this work proposes *MetaAnalyst*, an all-in-one standalone package equipped with a user friendly graphical user interface. MetaAnalyst automatically installs and configures 28 tools designed specifically for metagenomic biomarker detection. In addition, MetaAnalyst package includes 4 classifiers, namely support vector machines (SVM), random forest (RF), nearest centroid (NC), and k-nearest neighbor (kNN). These classification methods enable the researcher to go beyond the basic biomarker detection functionality to build complete phenotype classification models and to evaluate the discrimination power of the detected markers. MetaAnalyst provides a variety of metrics to asses the different aspects of classification performance. As a single criterion is not efficient to capture the overall performance of BD algorithms [37], in addition to the classification power, MetaAnalyst captures the unsupervised clustering performance [38] of the detected markers (visualized as two-way dendrogram plots) and the overlapping of the detected biomarkers across multiple BD algorithms (visualized as upset plot).

Furthermore, MetaAnalyst runs and evaluates, simultaneously, any subset of the 28 packed biomarker detection tools and compare their results directly. In contrast to the one-level labeling strategy, MetaAnalyst supports the *multilevel labeling* feature, through which researchers are able to define the positive and negative classes as any logical combination of up to three levels of labels to enable flexible study design. From input perspective, MetaAnalyst accepts 7 different types of input files. In addition to the publication-quality figures, the obtained results are reported as Excel tables to provide the user with further flexibility to generate other kinds of figures.

It is worth to mention that there are some other computational tools, platforms and projects available in the field of analyzing metagenomic data, such as Bioconda [39], Megan [40], UniFrac [41], CAMERA [42] and Galaxy [43]. However, each introduced work tackles an aspect different than the contribution of this paper. For instance, Bioconda [39] is a repository of bioinformatics packages and CAMERA [42] is a community database project that aims to collect metagenomic data and bioinformatics tools to make them widely available to the research community.

Implementation

The workflow of MetaAnalyst can be divided into five main steps as shown in Fig. 1. To facilitate the analysis, each main step is represented by one tab of the MetaAnalyst software. These tabs are designed to self-guide the user smoothly through the analysis. The following subsections describe these tabs (i.e., steps). Further details with a step-by-step example are available in the software manual `https://github.com/Rababa-Salahaldeen/MetaAnalystV1`.

Step 1: Input data

In general, metagenomic data files are composed of two parts: (1) numerical data, and (2) metadata. The numerical data represents the abundance levels of the operational taxonomic units (OTUs) across all samples. In metagenomics assays, each OTU represents a cluster of similar variants of the 16S rDNA marker gene sequence. Hence, each cluster (i.e., OTU) represents one bacterial species or genus. The second part, which is metadata, contains descriptive information about data such as OTU names, sample IDs, sample labels (e.g., disease/health status, body site location, ethnicity, gender).

The main tasks of this step are (1) to upload the input data file and (2) to extract the numerical data and their associated metadata. Regarding uploading the data, the user needs only to browse existing files on her/his local machine to locate the input file. In order to provide users with higher flexibility, the MetaAnalyst package is designed to support seven different types of input files: mat (Matlab file), csv, tsv, xls, xlsx, biom (Biological Observation Matrix) and json (JavaScript Object Notation). This feature is important to support the all-in-one feature of the MetaAnalyst software by reducing the dependency on other utilities/tools to handle specific input formats such as biom and json files. Upon loading the file, the MetaAnalyst automatically converts it into tabular format to facilitate extracting the abundance levels data and the samples' labels.

To extract these information, the user needs only to specify their location (rows and columns) in the input file. To simplify this task, the MetaAnalyst package automatically displays the content of the input file directly after selecting the input file in a table within the main window of the MetaAnalyst package. Therefore, users can directly specify the required information to extract different parts of the data without the need to open the original input files externally (using other tools). This feature is especially useful when dealing with "biom" files since such files are not human readable, and typically they require special tools to convert them into readable format. Again, this embedded display of the input data enhances the all-in-one experience. Unlike most existing packages that assume input files to follow specific templates (e.g., the data is column-wise and the variable names are listed in the first column), the MetaAnalyst package is flexible to handle different styles for the input files.

Step 2: Study design

The first step in comparative-based analysis, such as biomarker detection and phenotype classification, is to construct the positive and negative cohorts. The majority of existing tools perform this division based only on one criterion, commonly the health status (i.e., negative class represents healthy subjects while positive class represents diseased samples). On the other hand, the MetaAnalyst package supports a multilevel labeling strategy that enables researchers to combine several criteria for classifying the samples into positive and negative groups. In particular, a researcher is able to define the positive and negative classes as any logical combination of up to three levels of labels. This flexibility in forming the negative and positive cohorts enables researchers to easily study the datasets from different angles without the need to prepare a special file for each scenario. Further details on how to utilize the multilevel labeling to construct various scenarios is explained in the "Results and discussion" section.

Step 3: Data pre-processing

MetaAnalyst provides a variety of pre-processing procedures before downstream statistical analysis. These pre-treatment procedures can be categorized into: (i) filtering, (2) centering, and (3) normalization operations. Filtering aims at removing the variables that are not present in the majority of samples. Removing such under-represented (i.e., absent) variables simplifies and accelerates the downstream analysis. Centering operations convert the abundances to be around zero or median instead of the mean of the microbe abundance levels [44]. Normalization seeks converting the samples to be comparable by removing the systematic variability due to differences in sequence depth. In total, users are provided with one filtering (i.e., removing inactive variables), two centering (i.e., median and zero), and five normalization (i.e., total counts, median, upper quartile, reversed cumulative sum scaling (RCSS), z-score) operations to prepare their input data for subsequent analysis. The detailed information of each pre-processing procedure can be found in the software manual.

Step 4: Statistical analysis

MetaAnalyst supports two kinds of analysis: (1) biomarker detection, and (2) phenotype classification. For biomarker detection, the MetaAnalyst packs 28 metagenomic biomarker discovery algorithms, namely, Shotgun-FunctionalizeR [19], Boruta [15], edgeR [23], DESeq2 [24], ENNB [16], MetagenomeSeq [17], MicrobiomeDDA [18], MetaStats [20], Raida [21], LEfSe [9], RPCA [10], RegLRSD [11] , RSPCA [45], Lasso [46], Relief [47], ReliefF [48], and the following hypothesis tests: Wilcoxon Rank Sum Test [49], t-Test [50], log t-Test [50], square t-Test [50], Welch's Test [51], Chi-square Test [52], which are implemented using "stats" package R [29], Kolmogorov Smirnov Test [53], Levene Absolute Test [54], Levene Quadratic Test [54], Brown Forsythe Test [55], BSS/WSS (Between Sum of Squares over Within Sum of Squares) [56], and Pearson Correlation [57], which are implemented using MATLAB. Detailed description of these methods are provided in the User Manual. The biomarker detection phase assigns each variable (i.e., microbe) a score that determines its significance. Then, the top scored variables, according to a predefined number, will be declared as potential markers.

For phenotype classification, the MetaAnalyst package included RF, kNN, four variates of SVM (linear, polynomial, gaussian and radial basis function (RBF)), and two variates of the NCC (namely NCC-1 and NCC-2) classifiers. The difference between NCC-1 and NCC-2 is that the former utilizes the l_1 norm to measure the distance, while the second uses the Euclidean distance. These classifiers can be used for (i) building phenotype classification models, and (ii) evaluating the discrimination power of the detected markers. To achieve this, the data corresponding to the identified markers are extracted and used to train and test the classifier using k-fold cross validation.

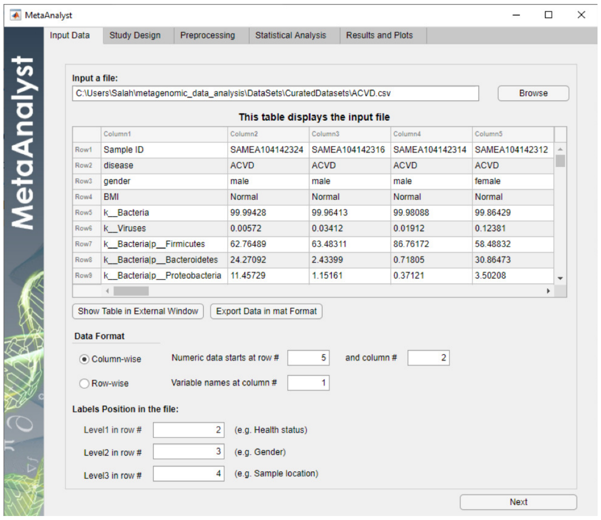
To provide the user with a comprehensive analysis capability, the MetaAnalyst package enables the user to select multiple biomarker detection algorithms to evaluate different numbers of potential markers at once. Besides, the MetaAnalyst package provides the user with the capability of saving the current simulation settings to be used in future analyses. Also, it enables the user to load the previously saved configuration. This feature helps researchers to generate reusable workflows to compare several algorithms under the same settings and conduct the same analysis over multiple datasets.

Further details about the packed algorithms and the classification measures are provided in the software manual.

Step 5: Results and plots

MetaAnalyst software provides several publication-quality interactive plots, as listed below, to present the obtained results:

- **Detected biomarkers:** for each BD algorithm and for each number of top features (i.e., biomarkers), the MetaAnalyst presents the identified markers and their scores as a horizontal bar graph. The blue and red bars represent the markers that are enriched in negative and positive class, respectively.
- **Consensus performance** Consensus performance aims at presenting the agreement among different biomarker detection algorithms as an upset plot. This plot shows the overlap between the suggested markers by the BD algorithms included in the analysis.
- **Clustering performance:** Based on the idea that reliable markers are supposed to enlarge the difference between samples belonging to different groups, the two-way unsupervised hierarchical clustering can be utilized to visualize the discrimination power of the biomarker detection algorithm [38]. In particular, the data corresponding to the detected markers are employed to perform hierarchical clustering of samples and selected microbes. This generates a clustering diagram (visualized as a heatmap and two dendrograms, and hence the name two-way clustering), where the rows and columns of the heatmap represent the microbes and samples, respectively. Under such a setting, a reliable biomarker detection algorithm is expected to generate heatmaps with clear separation between the positive and negative cohorts. For each BD algorithm and for each number of top features, the MetaAnalyst shows the two-way clustering over the significantly identified differential markers as a heatmap and dendrogram.
- **Classification performance:** To evaluate the classification performance, MetaAnalyst computes the overall classification accuracy (ACC),



Input a file: C:\Users\Salah\metagenomic_data_analysis\DataSets\CuratedDatasets\ACVD.csv

This table displays the input file

	Column1	Column2	Column3	Column4	Column5
Row1	Sample ID	SAMEA104142324	SAMEA104142316	SAMEA104142314	SAMEA104142312
Row2	disease	ACVD	ACVD	ACVD	ACVD
Row3	gender	male	male	male	female
Row4	BMI	Normal	Normal	Normal	Normal
Row5	k__Bacteria	99.99428	99.96413	99.98088	99.86429
Row6	k__Viruses	0.00572	0.03412	0.01912	0.12381
Row7	k__Bacteria|p__Firmicutes	62.76489	63.48311	86.76172	58.48832
Row8	k__Bacteria|p__Bacteroidetes	24.27092	2.43399	0.71805	30.86473
Row9	k__Bacteria|p__Proteobacteria	11.45729	1.15161	0.37121	3.50208

Data Format

● Column-wise Numeric data starts at row # 5 and column # 2

○ Row-wise Variable names at column # 1

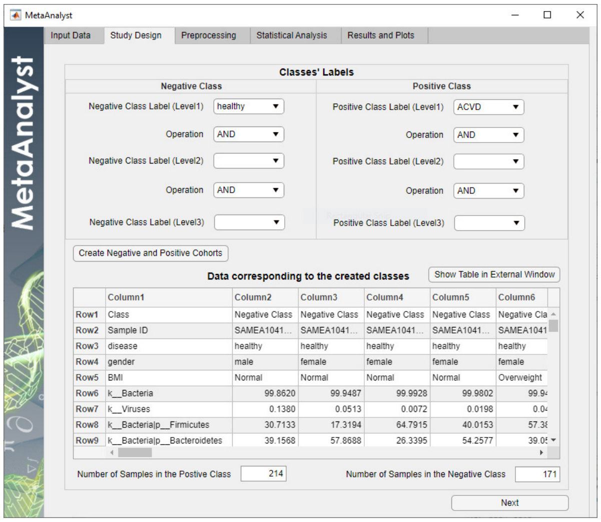
Labels Position in the file:

Level1 in row # 2 (e.g. Health status)

Level2 in row # 3 (e.g. Gender)

Level3 in row # 4 (e.g. Sample location)

Figure 2 Input tab.



Classes' Labels

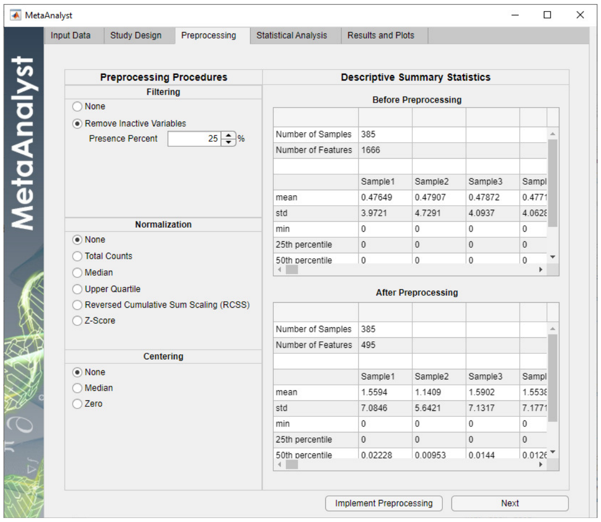
Negative Class		Positive Class	
Negative Class Label (Level1)	healthy	Positive Class Label (Level1)	ACVD
Operation	AND	Operation	AND
Negative Class Label (Level2)		Positive Class Label (Level2)	
Operation	AND	Operation	AND
Negative Class Label (Level3)		Positive Class Label (Level3)	

Data corresponding to the created classes

	Column1	Column2	Column3	Column4	Column5	Column6
Row1	Class	Negative Class	Negative Class	Negative Class	Negative Class	Negative Cla
Row2	Sample ID	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041
Row3	disease	healthy	healthy	healthy	healthy	healthy
Row4	gender	male	female	female	female	female
Row5	BMI	Normal	Normal	Normal	Normal	Overweight
Row6	k__Bacteria	99.8620	99.9487	99.9928	99.9802	99.9
Row7	k__Viruses	0.1380	0.0513	0.0072	0.0198	0.0
Row8	k__Bacteria|p__Firmicutes	30.7133	17.3194	64.7915	40.0153	57.3
Row9	k__Bacteria|p__Bacteroidetes	39.1568	57.8688	26.3395	54.2577	39.0

Number of Samples in the Postive Class 214 Number of Samples in the Negative Class 171

Figure 3 Study design tab for Scenario 1.



Preprocessing Procedures

Filtering

○ None

● Remove Inactive Variables

Presence Percent 25 %

Normalization

● None

○ Total Counts

○ Median

○ Upper Quartile

○ Reversed Cumulative Sum Scaling (RCSS)

○ Z-Score

Centering

● None

○ Median

○ Zero

Descriptive Summary Statistics

Before Preprocessing

Number of Samples	385			
Number of Features	1666			
	Sample1	Sample2	Sample3	Sampl
mean	0.47649	0.47907	0.47872	0.4771
std	3.9721	4.7291	4.0937	4.062
min	0	0	0	0
25th percentile	0	0	0	0
50th percentile	0	0	0	0

After Preprocessing

Number of Samples	385			
Number of Features	495			
	Sample1	Sample2	Sample3	Sampl
mean	1.5594	1.1409	1.5902	1.553
std	7.0846	5.6421	7.1317	7.1771
min	0	0	0	0
25th percentile	0	0	0	0
50th percentile	0.02228	0.00953	0.0144	0.012

Figure 4 Preprocessing tab for Scenario 1.

balanced accuracy (BACC), sensitivity (SEN), specificity (SPC), miss classification rate (MCR), receiver operation curve (ROC), and area under the curve (AUC). These metrics capture various aspects of the classification performance. For example, the accuracy (the ratio of the correctly detected samples in both classes) is biased toward the class with dominant samples. Therefore, for extremely skewed datasets, the accuracy may be misleading, and hence class-specific measures (e.g., sensitivity and specificity) or BACC may be more reliable to account for bias.

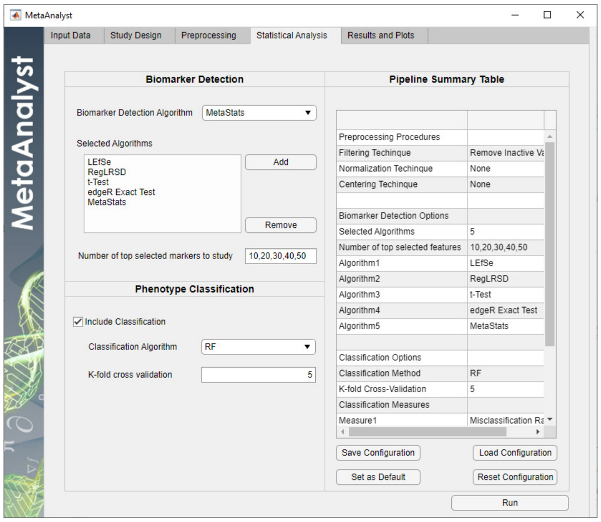
MetaAnalyst displays the seven classification performance metrics (i.e., ACC, BACC, SPC, SEN, ROC, AUC, MCR) for all the included algorithms in the analysis.

To enhance the user's experience, the MetaAnalyst software provides the user with the flexibility to control various settings of the generated plots such as the size of the plots, description of the axis (i.e., x-label and y-label), the title of the figure, the fontsize, etc. After finalizing the figure formatting, the user can save the plots in thirteen different formats: jpg, png, tif, pdf, fig, eps, bmp, emf, pcx, pbm, pgm, ppm, svg. In addition to the generated plots, the user can export all the results as excel sheets.

Results and Discussion

This section demonstrates the flexibility and ease-of-use of MetaAnalyst by analyzing a metagenomic dataset related to acute cardiovascular disease (ACVD) under various scenarios/conditions. This dataset studies the relationship between human gut microbiota and ACVD [58]. The dataset is composed of metagenomic stool samples from 218 ACVD patients and 187 healthy subjects. A snapshot of the loaded dataset as displayed by MetaAnalyst is shown in Fig. 2.

As can be seen in Fig 2, each sample is annotated with three levels of labels: level 1: disease status (in row 2), level 2: gender (in row 2), and level 3: BMI status (in row 3). It is worth to mention that it is not required to fill the information about the location of all levels of labels. It is required only to locate the levels that the user is interested in his/her study. For example, assume that the researcher is interested only on the effect of disease and BMI status, as discussed in the following two subsections, then it is necessary



Biomarker Detection

Biomarker Detection Algorithm: MetaStats

Selected Algorithms:
LEfSe
RegLRSD
t-Test
edgeR Exact Test
MetaStats

Number of top selected markers to study: 10,20,30,40,50

Phenotype Classification

☑ Include Classification

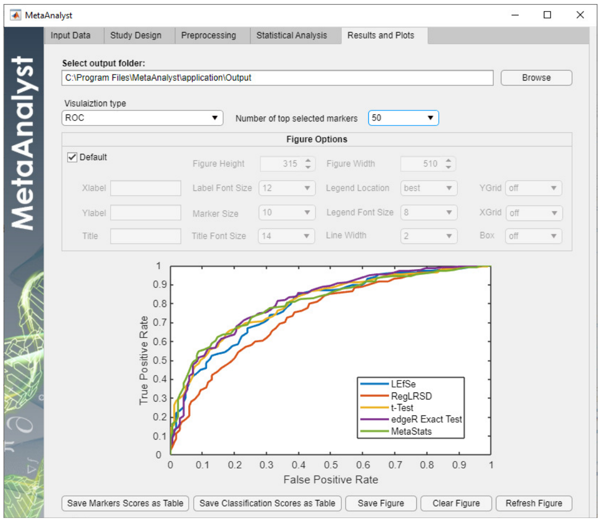
Classification Algorithm: RF

K-fold cross validation: 5

Pipeline Summary Table

Preprocessing Procedures	
Filtering Techinque	Remove Inactive Va
Normalization Techinque	None
Centering Techinque	None
Biomarker Detection Options	
Selected Algorithms	5
Number of top selected features	10,20,30,40,50
Algorithm1	LEfSe
Algorithm2	RegLRSD
Algorithm3	t-Test
Algorithm4	edgeR Exact Test
Algorithm5	MetaStats
Classification Options	
Classification Method	RF
K-fold Cross-Validation	5
Classification Measures	
Measure1	Misclassification Ra

Figure 5 Statistical analysis tab for Scenario 1.



Select output folder: C:\Program Files\MetaAnalyst\application\Output

Visulaiztion type: ROC

Number of top selected markers: 50

☑ Default

Figure Height: 315
Figure Width: 510
Xlabel:
Label Font Size: 12
Legend Location: best
YGrid: off
Ylabel:
Marker Size: 10
Legend Font Size: 8
XGrid: off
Title:
Title Font Size: 14
Line Width: 2
Box: off

Figure 6 Results and plots tab for Scenario 1.

to specify only the rows that store the labels of disease and BMI status. Besides, the loaded data as shown in Fig 2 is in column-wise format (each column represents one sample), the numeric data starts at row 5 and column 2, and the variable names resides at column 1.

To demonstrate the capability of MetaAnalyst to study the dataset from different perspectives, we consider four scenarios as summarized in Table 2.

Healthy versus diseased

Initially, let us consider the first scenario that aims at finding potential markers that discriminate between healthy and diseased subjects irrespective of their gender and obesity status (BMI value). Therefore, in the "Study Design" tab, the negative and positive classes are set, using only level 1, to include "healthy" and "ACVD" subjects, respectively, and left the other two levels (i.e., level 2: gender and level 3: BMI status) empty as shown in Fig. 3.

Table 2 Four possible scenarios to study the ACVD dataset.

	Positive Class	Negative Class
Scenario 1	Diseased	Healthy
Scenario 2	Diseased & Normal	Healthy & Normal
Scenario 3	Diseased & Normal & Female	Healthy & Normal & Female
Scenario 4	Diseased & Normal & Male	Healthy & Normal & Male

Next, as an option, the user preprocesses the data before the downstream analysis using the options in the "Preprocessing" tab shown in Fig. 4. In this study, we chose to only remove all inactive variables that are not present in at least 25% of the samples of either class. This reduces the number of variables from 1666 to only 495. Furthermore, the "Preprocessing" tab displays two tables presenting summary statistics about the number of samples, number of features, mean, standard deviation, minimum, 25^{th} percentile, 50^{th} percentile, 75^{th} percentile and the maximum value for each sample.

Upon preparing the data, the user can design the analysis work-flow by selecting the biomarker detection algorithms from the drop-down list as shown in Fig. 5. As mentioned earlier, the user can select multiple detection algorithms for various number of top features to conduct the analysis over all of them simultaneously. In this study, 5 biomarker detection algorithms (LEfSe, RegLRSD, t-Test, edgeR, and MetaStats) were included in the analysis. Besides, the user can extend the biomarker detection algorithm to include a classifier model. This classifier is used for both (1) evaluating the performance of biomarker detection algorithms and (2) building a phenotype classification model. In this experiment, the RF classifier was employed. The classification performance is estimated using 5-fold cross-validation. MetaAnalyst shows a pipeline summary describing the designed analysis workflow. To enhance the user experience and the reproducibility of the results, the MetaAnalyst software provides the user with the capability to set the current analysis work-flow as a default configuration. Also, the user can save multiple configurations (i.e., analysis workflows) and load the suitable configuration for future analysis.

The user can select the type of the results to be visualized from the "Visualization type" drop list in the "Results and plots" tab as shown in Fig. 6. A sample of these plots is displayed in Fig. 7. For example,

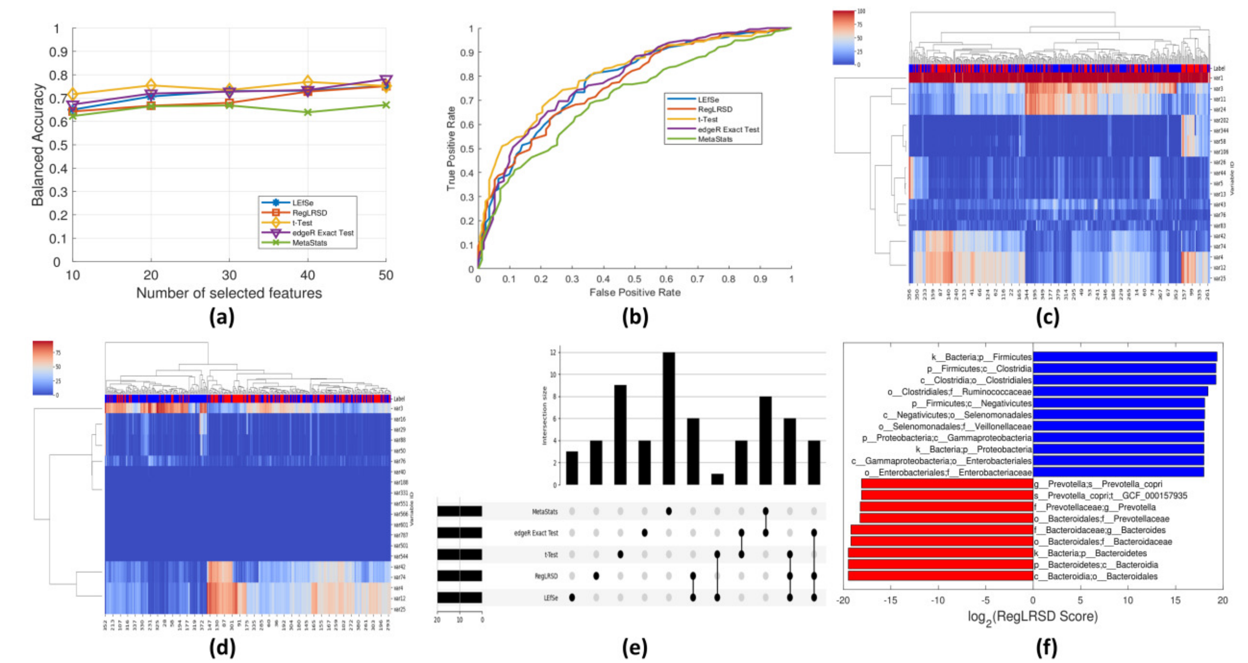
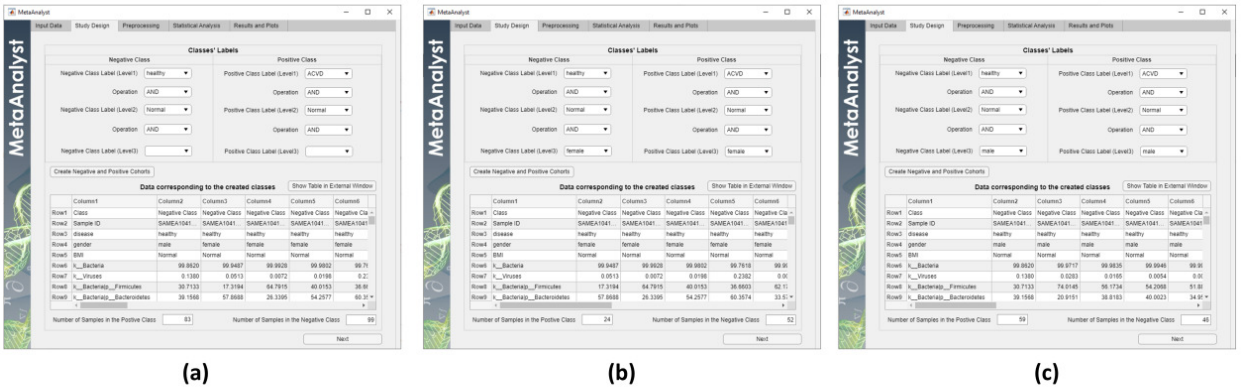


Figure 7 Sample of the obtained results over Scenario 1. (a)&(b) Achieved BACC and ROC for the selected biomarker detection algorithms: LEfSe, RegLRSD, t-Test, Exact-Test (edgeR), and MetaStats, respectively. (c) The unsupervised clustering performance of the top 20 markers as suggested by LEfSe algorithm. (d) The unsupervised clustering performance of the top 20 markers as suggested by RegLRSD algorithm. (e) The number of overlapped potential markers among the five BD algorithms. (f) Suggested markers by the RegLRSD algorithm.



(a)

Classes' Labels

Negative Class		Positive Class	
Negative Class Label (Level1)	healthy	Positive Class Label (Level1)	ACVD
Operation	AND	Operation	AND
Negative Class Label (Level2)	Normal	Positive Class Label (Level2)	Normal
Operation	AND	Operation	AND
Negative Class Label (Level3)		Positive Class Label (Level3)	

Data corresponding to the created classes

	Column1	Column2	Column3	Column4	Column5	Column6
Row1	Class	Negative Class	Negative Class	Negative Class	Negative Class	Negative Cla
Row2	Sample ID	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041
Row3	disease	healthy	healthy	healthy	healthy	healthy
Row4	gender	male	female	female	female	female
Row5	BMI	Normal	Normal	Normal	Normal	Normal
Row6	k__Bacteria	99.8620	99.9487	99.9928	99.9802	99.7
Row7	k__Viruses	0.1380	0.0513	0.0072	0.0198	0.2
Row8	k__Bacteria|p__Firmicutes	30.7133	17.3194	64.7915	40.0153	36.6
Row9	k__Bacteria|p__Bacteroidetes	39.1568	57.8688	26.3395	54.2577	60.3

Number of Samples in the Postive Class 83 Number of Samples in the Negative Class 99

(b)

Classes' Labels

Negative Class		Positive Class	
Negative Class Label (Level1)	healthy	Positive Class Label (Level1)	ACVD
Operation	AND	Operation	AND
Negative Class Label (Level2)	Normal	Positive Class Label (Level2)	Normal
Operation	AND	Operation	AND
Negative Class Label (Level3)	female	Positive Class Label (Level3)	female

Data corresponding to the created classes

	Column1	Column2	Column3	Column4	Column5	Column6
Row1	Class	Negative Class	Negative Class	Negative Class	Negative Class	Negative Cla
Row2	Sample ID	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041
Row3	disease	healthy	healthy	healthy	healthy	healthy
Row4	gender	female	female	female	female	female
Row5	BMI	Normal	Normal	Normal	Normal	Normal
Row6	k__Bacteria	99.9487	99.9928	99.9802	99.7618	99.9
Row7	k__Viruses	0.0513	0.0072	0.0198	0.2382	0.0
Row8	k__Bacteria|p__Firmicutes	17.3194	64.7915	40.0153	36.6603	62.1
Row9	k__Bacteria|p__Bacteroidetes	57.8688	26.3395	54.2577	60.3574	33.5

Number of Samples in the Postive Class 24 Number of Samples in the Negative Class 52

(c)

Classes' Labels

Negative Class		Positive Class	
Negative Class Label (Level1)	healthy	Positive Class Label (Level1)	ACVD
Operation	AND	Operation	AND
Negative Class Label (Level2)	Normal	Positive Class Label (Level2)	Normal
Operation	AND	Operation	AND
Negative Class Label (Level3)	male	Positive Class Label (Level3)	male

Data corresponding to the created classes

	Column1	Column2	Column3	Column4	Column5	Column6
Row1	Class	Negative Class	Negative Class	Negative Class	Negative Class	Negative Cla
Row2	Sample ID	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041...	SAMEA1041
Row3	disease	healthy	healthy	healthy	healthy	healthy
Row4	gender	male	male	male	male	male
Row5	BMI	Normal	Normal	Normal	Normal	Normal
Row6	k__Bacteria	99.8620	99.9717	99.9835	99.9946	99.9
Row7	k__Viruses	0.1380	0.0283	0.0165	0.0054	0.0
Row8	k__Bacteria|p__Firmicutes	30.7133	74.0145	56.1734	54.2068	51.8
Row9	k__Bacteria|p__Bacteroidetes	39.1568	20.9151	38.8183	40.0023	34.9

Number of Samples in the Postive Class 59 Number of Samples in the Negative Class 46

Figure 8 Construct scenarios 2, 3, and 4 in Table 2 using the multi-level labeling property supported in the "Study Design" tab.

the MetaAnalyst shows the detected markers along with their scores in bar graph plots as shown in Fig. 7-a. The unsupervised clustering performance is presented as a tow-way clustering heatmap as shown in 7-b. The achieved classification performance in terms of the BACC is depicted in Fig. 7-c. In addition to the BACC, the MetaAnalyst generates similar plots to ACC, SPC, SEN, ROC, and AUC. The agreement between biomarker detection algorithms is depicted in Fig. 7-d. The algorithm recommending more overlapped markers is expected to be more accurate.

The complete set of results that re generated by the MetaAnalyst software for scenario 1 is depicted in the Additional file 1. In addition to the generated plots, the user can export the classification performance [see Additional file 2], detected markers by each algorithm [see Additional file 3], and the overlapped set of markers [see Additional file 4] as excel sheets.

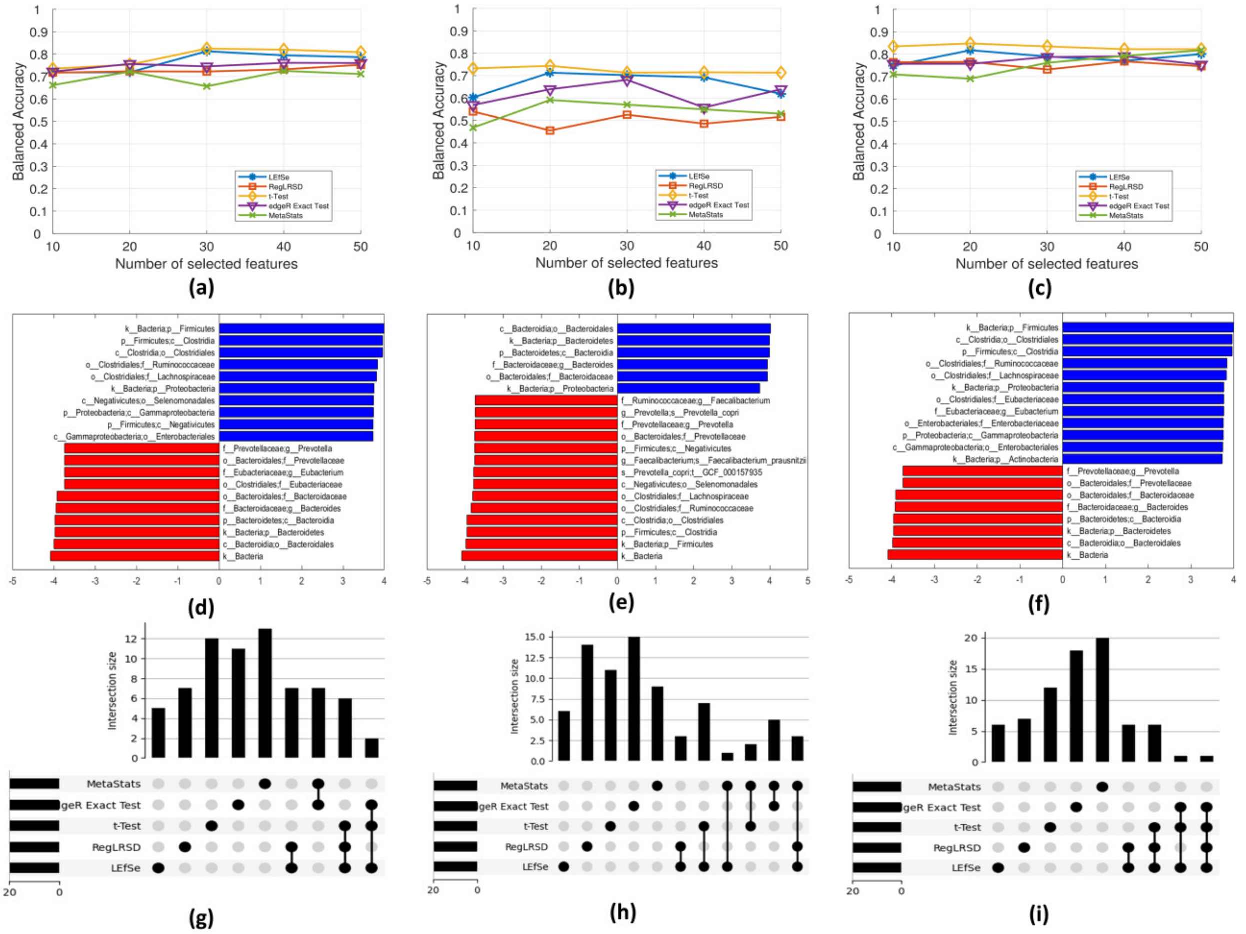


Figure 9 Sample of the obtained results, using LEfSe, RegLRSD, t-Test, edgeR, and MetaStats algorithms, that studies the impact of obesity and gender over the ACVD dataset (i.e., scenarios 2, 3, and 4 in Table 2). The results in the first, second and third column correspond to scenario 2, 3, and 4, respectively. The first row displays the achieved BACC performance over the three scenarios. The second row shows the suggested 20 markers by edgeR. The third row presents the overlapping between the suggested 20 markers by the five BD algorithms.

Impact of obesity and gender

To shed insights on the utility of the multi-level labeling feature of MetaAnalyst in enabling the study of the dataset from different perspectives, this section demonstrates how to extend the previous scenario to study the impact of obesity and gender. To illustrate, assume that the researcher is interested in excluding the impact of obesity from the previous study (i.e., Scenario 2 in Table 2). That is, to compare healthy versus diseased samples over only normal subjects (i.e., subjects with BMI values in the range 18.9-25). Constructing this study can be achieved simply by setting the operation between the first and second levels to "AND" and select the label to be "Normal" for both negative and positive cohorts as shown in Fig. 8-a. Furthermore, assume that the researcher is interested in extending this study to investigate whether the microbial patterns differ between female ((i.e., Scenario 3 in Table 2)) and male ((i.e., Scenario 4 in Table 2)) individuals. Again, these two studies can be constructed easily by proper setting of the study design tab as shown in Figs. 8-b and 8-c, respectively.

A sample of the obtained results under scenarios 2, 3 and 4 is depicted in Fig. 9. As it can be observed from Figs.9-a, 9-b, and 9-c, the achieved BACC performance by the five BD algorithms in male subjects is significantly higher than female subjects. Interestingly, male individuals present higher discrimination power compared to females. This result may indicate that the bacterial composition in males present stronger variation compared to females in response to ACVD. Indeed, this observation needs further investigations to evaluate the gender effect on the interaction between human microbiota and cardiovascular disease.

This suggests that potential treatments may need to be gender-specific to account for the gender association with ACVD risk factors. Thus, the multi-level labeling feature of MetaAnalyst allows such observations to be easily visualized and detectable.

Conclusions

This work proposed MetaAnalyst, a stand-alone software package for metagenomic biomarker detection and phenotype classification. The MetaAnalyst package aims at reducing the programming skills and simplifying the tasks required to analyze metagenomic datasets. The MetaAnalyst package (i) automatically installs and handles all package dependencies-related issues of 28 state-of-the-art biomarker detection algorithms and 4 classification models with several data preprocessing capabilities, (ii) provides a simple graphical user interface that naturally guides the user through the analysis pipeline, (iii) accepts input datasets in several files with flexible data formats, (iv) supports multi-level labeling feature to flexibly cluster the positive and negative cohorts and to study a given dataset under a multitude of scenarios, (v) runs several algorithms simultaneously and evaluates their performance according to three criteria (classification, clustering, and overlapping performance), (vi) reports the results in publishing-quality plots as well as Excel sheets. Due to the similarity between metagenomic data and other omic data and the possibility of applying the packed algorithms in MetaAnalyst to other omic data, we believe that MetaAnalyst will become a popular tool for metagenomics applications and other studies. The executable file for MetaAnalyst along with a detailed user manual are made available at `https://github.com/Rababa-Salahaldeen/MetaAnalystV1`.

Ethics approval and consent to participate

Not applicable.

Consent to Publish

Not applicable

Availability of Data and Materials

- Project name: MetaAnalyst
- Project home page: `https://github.com/Rababa-Salahaldeen/MetaAnalystV1`
- Operating system(s): Microsoft Windows
- Programming language: Matlab, R and Python
- Other requirements: No requirements
- License: MetaAnalyst is made readily available freely to any scientist wishing to use it for non-commercial purposes, without any restriction.
- Any restrictions to use by non-academics: license needed

Competing interests

The authors declare that they have no competing interests.

Funding

This work is supported by (i) Seed Grant (ID: SEEIT 1/2020) from the Deanship of Scientific Research at the German Jordanian University and (ii) TAMUQ Research Initiative.

Author's contributions

MA initiated the research idea, developed the framework, provided the results interpretation and wrote the manuscript. SR conducted the analysis and wrote the user manual. AH contributed to the framework development and analysis. MA, SR, and AH wrote the MetaAnalyst package. AG and ES participated in the interpretation of the results and were involved in revising the manuscript. All authors read and approved the final manuscript.

List of Abbreviations

ACC: Accuracy; ACVD: Acute Cardiovascular Disease; AUC: Area Under the Curve; BACC: Balanced Accuracy; BD: Biomarker Detection; GUI: Graphical User Interface; IBD: Inflammatory Bowel Disease; kNN: k-Nearest Neighbor; MCR: Miss Classification Rate; OTU: Operational Taxonomic Unit; RCSS: Reversed Cumulative Sum Scaling RF: Random Forest; ROC: Receiver Operation Curve; SEN: Sensitivity; SPC: Specificity; SVM: Support Vector Machine;

Additional Files

Additional file 1
The complete set of plots that are generated by the MetaAnalyst software for scenario 1.

Additional file 2 — classification performance
An Excel sheet that is generated by the MetaAnalyst software that contains the obtained classification performance (in terms of MCR, SEN, SPC, ACC, BACC, ROC and AUC) by the five algorithms included in the study (i.e., LEFSe, RegLRSD, t-Test, EdgeR Exact Test, and MetaStats) over Scenario 1 for various number of top features (i.e., 10, 20, 30, 40, and 50).

Additional file 3 — detected markers
An Excel sheet that is generated by the MetaAnalyst software that contains the top 20 detected metagenomic markers by the five algorithms included in the study (i.e., LEFSe, RegLRSD, t-Test, EdgeR Exact Test, and MetaStats) over Scenario 1.

Additional file 4 — the overlapped set of markers
An Excel sheet that is generated by the MetaAnalyst software that contains the overlapped set of metagenomic markers by the five algorithms included in the study (i.e., LEFSe, RegLRSD, t-Test, EdgeR Exact Test, and MetaStats) over Scenario 1.

Author details

[1]School of Electrical Engineering and Information Technology, German Jordanian University, Amman, Jordan. [2]Department of Systems Science and Industrial Engineering, State University of New York at Binghamton, Binghamton, NY, USA. [3]Electrical and Computer Engineering Department, Texas A&M University, College Station, TX, USA.

References

1. Flint HJ. Obesity and the gut microbiota. Journal of Clinical Gastroenterology. 2011;45:S128-32.
2. Ridaura VK, Faith JJ, Rey FE, Cheng J, Duncan AE, Kau AL, et al. Gut microbiota from twins discordant for obesity modulate metabolism in mice. Science. 2013;341(6150):1241214.
3. Larsen N, Vogensen FK, Van Den Berg F, Nielsen DS, Andreasen AS, Pedersen BK, et al. Gut microbiota in human adults with type 2 diabetes differs from non-diabetic adults. PLoS One. 2010;5(2):e9085.
4. Morgan XC, Tickle TL, Sokol H, Gevers D, Devaney KL, Ward DV, et al. Dysfunction of the intestinal microbiome in inflammatory bowel disease and treatment. Genome Biology. 2012;13(9):R79.
5. Moore W, Moore LH. Intestinal floras of populations that have a high risk of colon cancer. Applied and Environmental Microbiology. 1995;61(9):3202-7.
6. Ahn J, Sinha R, Pei Z, Dominianni C, Wu J, Shi J, et al. Human gut microbiome and risk of colorectal cancer. Journal of the National Cancer Institute. 2013:djt300.
7. Alshawaqfeh M, Wajid B, Guard M, Minamoto Y, Lidbury J, Steiner J, et al. A Dysbiosis Index to Assess Microbial Changes in Fecal Samples of Dogs with Chronic Enteropathy. Journal of Veterinary Internal Medicine. 2016;30(4):1536. Available from: `http://dx.doi.org/10.1111/jvim.13963`.

8. AlShawaqfeh M, Wajid B, Minamoto Y, Markel M, Lidbury J, Steiner J, et al. A dysbiosis index to assess microbial changes in fecal samples of dogs with chronic inflammatory enteropathy. FEMS Microbiology Ecology. 2017;93(11):fix136.
9. Segata N, Izard J, Waldron L, Gevers D, Miropolsky L, Garrett WS, et al. Metagenomic biomarker discovery and explanation. Genome Biology. 2011;12(6):R60.
10. Alshawaqfeh M, Bashaireh A, Serpedin E, Suchodolski J. Consistent metagenomic biomarker detection via robust PCA. Biology direct. 2017;12(1):1-16.
11. Alshawaqfeh M, Bashaireh A, Serpedin E, Suchodolski J. Reliable Biomarker discovery from Metagenomic data via RegLRSD algorithm. BMC Bioinformatics. 2017 Jul;18(1):328. Available from: `http://dx.doi.org/10.1186/s12859-017-1738-1`.
12. Chen IMA, Chu K, Palaniappan K, Pillay M, Ratner A, Huang J, et al. IMG/M v. 5.0: an integrated data management and comparative analysis system for microbial genomes and microbiomes. Nucleic Acids Research. 2019;47(D1):D666-77.
13. Pasolli E, Truong DT, Malik F, Waldron L, Segata N. Machine learning meta-analysis of large metagenomic datasets: tools and biological insights. PLoS Computational Biology. 2016;12(7):e1004977.
14. Ditzler G, Morrison JC, Lan Y, Rosen GL. Fizzy: feature subset selection for metagenomics. BMC Bioinformatics. 2015;16(1):358.
15. Kursa MB, Rudnicki WR, et al. Feature selection with the Boruta package. J Stat Softw. 2010;36(11):1-13.
16. Pookhao N, Sohn MB, Li Q, Jenkins I, Du R, Jiang H, et al. A two-stage statistical procedure for feature selection and comparison in functional analysis of metagenomes. Bioinformatics. 2015;31(2):158-65.
17. Paulson JN, Stine OC, Bravo HC, Pop M. Differential abundance analysis for microbial marker-gene surveys. Nature Methods. 2013;10(12):1200-2.
18. Chen J, King E, Deek R, Wei Z, Yu Y, Grill D, et al. An omnibus test for differential distribution analysis of microbiome sequencing data. Bioinformatics. 2018;34(4):643-51.
19. Kristiansson E, Hugenholtz P, Dalevi D. ShotgunFunctionalizeR: an R-package for functional comparison of metagenomes. Bioinformatics. 2009;25(20):2737-8.
20. Paulson JN, Pop M, Bravo HC. Metastats: an improved statistical method for analysis of metagenomic data. Genome Biology. 2011;12(1):1-27.
21. Sohn MB, Du R, An L. A robust approach for identifying differentially abundant features in metagenomic samples. Bioinformatics. 2015;31(14):2269-75.
22. Sanli K, Karlsson FH, Nookaew I, Nielsen J. FANTOM: Functional and taxonomic analysis of metagenomes. BMC bioinformatics. 2013;14(1):38.
23. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2010;26(1):139-40.
24. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology. 2014;15(12):550.
25. Fang W, Chang X, Su X, Xu J, Zhang D, Ning K. A machine learning framework of functional biomarker discovery for different microbial communities based on metagenomic data. In: 2012 IEEE 6th International Conference on Systems Biology (ISB). IEEE; 2012. p. 106-12.
26. Knights D, Costello EK, Knight R. Supervised classification of human microbiota. FEMS Microbiology Reviews. 2011;35(2):343-59.
27. Masoudi-Sobhanzadeh Y, Motieghader H, Masoudi-Nejad A. FeatureSelect: a software for feature selection based on machine learning approaches. BMC Bioinformatics. 2019;20(1):1-17.
28. Tang J, Mou M, Wang Y, Luo Y, Zhu F. MetaFS: performance assessment of biomarker discovery in metaproteomics. Briefings in Bioinformatics. 2021;22(3):bbaa105.
29. Team RC, et al. Package "Stats.". RA Lang Environment Stat Comput Vienna, Austria: R Foundation for Statistical Computing. 2013.
30. Parks DH, Tyson GW, Hugenholtz P, Beiko RG. STAMP: statistical analysis of taxonomic and functional profiles. Bioinformatics. 2014;30(21):3123-4.
31. Rodriguez-Brito B, Rohwer F, Edwards RA. An application of statistics to comparative metagenomics. BMC Bioinformatics. 2006;7(1):162.
32. Dhariwal A, Chong J, Habib S, King IL, Agellon LB, Xia J. MicrobiomeAnalyst: a web-based tool for comprehensive statistical, visual and meta-analysis of microbiome data. Nucleic acids research. 2017;45(W1):W180-8.
33. Piccolo BD, Wankhade UD, Chintapalli SV, Bhattacharyya S, Chunqiao L, Shankar K. Dynamic assessment of microbial ecology (DAME): a web app for interactive analysis and visualization of microbial sequencing data. Bioinformatics. 2018;34(6):1050-2.
34. Mattiello F, Verbist B, Faust K, Raes J, Shannon WD, Bijnens L, et al. A web application for sample size and power calculation in case-control microbiome studies. Bioinformatics. 2016;32(13):2038-40.
35. Alshawaqfeh M, Al Kawam A, Serpedin E. Sparse-low rank matrix decomposition framework for identifying potential biomarkers for inflammatory bowel disease. In: 2017 25th European Signal Processing Conference (EUSIPCO). IEEE; 2017. p. 1882-6.
36. Tang J, Wang Y, Fu J, Zhou Y, Luo Y, Zhang Y, et al. A critical assessment of the feature selection methods used for biomarker discovery in current metaproteomics studies. Briefings in Bioinformatics. 2020;21(4):1378-90.
37. Christin C, Hoefsloot HC, Smilde AK, Hoekman B, Suits F, Bischoff R, et al. A critical assessment of feature selection methods for biomarker discovery in clinical proteomics. Molecular & Cellular Proteomics. 2013;12(1):263-76.
38. Griffin NM, Yu J, Long F, Oh P, Shore S, Li Y, et al. Label-free, normalized quantification of complex mass spectrometry data for proteomic analysis. Nature Biotechnology. 2010;28(1):83-9.
39. Grüning B, Dale R, Sjödin A, Chapman BA, Rowe J, Tomkins-Tinch CH, et al. Bioconda: sustainable and comprehensive software distribution for the life sciences. Nature methods. 2018;15(7):475-6.
40. Huson DH, Auch AF, Qi J, Schuster SC. MEGAN analysis of metagenomic data. Genome research. 2007;17(3):377-86.
41. Lozupone C, Hamady M, Knight R. UniFrac–an online tool for comparing microbial community diversity in a phylogenetic context. BMC bioinformatics. 2006;7(1):1-14.
42. Sun S, Chen J, Li W, Altintas I, Lin A, Peltier S, et al. Community cyberinfrastructure for advanced microbial ecology research and analysis: the CAMERA resource. Nucleic acids research. 2010;39(suppl_1):D546-51.
43. Afgan E, Baker D, Batut B, Van Den Beek M, Bouvier D, C̈ech M, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Research. 2018;46(W1):W537-44.
44. van den Berg RA, Hoefsloot HC, Westerhuis JA, Smilde AK, van der Werf MJ. Centering, scaling, and transformations: improving the biological information content of metabolomics data. BMC Genomics. 2006;7(1):1-15.
45. Shen H, Huang JZ. Sparse principal component analysis via regularized low rank matrix approximation. Journal of multivariate analysis. 2008;99(6):1015-34.
46. Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society Series B (Methodological). 1996:267-88.
47. Kira K, Rendell LA. A practical approach to feature selection. In: Machine learning proceedings 1992. Elsevier; 1992. p. 249-56.
48. Kononenko I. Estimating attributes: Analysis and extensions of RELIEF. In: European conference on machine learning. Springer; 1994. p. 171-82.
49. Mann HB, Whitney DR. On a test of whether one of two random variables is stochastically larger than the other. The annals of mathematical statistics. 1947:50-60.
50. Kim TK. T test as a parametric statistic. Korean journal of anesthesiology. 2015;68(6):540.
51. Welch BL. The generalization of 'STUDENT'S'problem when several different population varlances are involved. Biometrika. 1947;34(1-2):28-35.
52. Plackett RL. Karl Pearson and the chi-squared test. International Statistical Review/Revue Internationale de Statistique. 1983:59-72.
53. Darling DA. The kolmogorov-smirnov, cramer-von mises tests. The Annals of Mathematical Statistics. 1957;28(4):823-38.

54. Levene H. Robust tests for equality of variances. Contributions to probability and statistics Essays in honor of Harold Hotelling. 1961:279-92.
55. Brown MB, Forsythe AB. Robust tests for the equality of variances. Journal of the American Statistical Association. 1974;69(346):364-7.
56. Box GE. Non-normality and tests on variances. Biometrika. 1953;40(3/4):318-35.
57. Benesty J, Chen J, Huang Y, Cohen I. Pearson correlation coefficient. In: Noise reduction in speech processing. Springer; 2009. p. 1-4.
58. Jie Z, Xia H, Zhong SL, Feng Q, Li S, Liang S, et al. The gut microbiome in atherosclerotic cardiovascular disease. Nature Communications. 2017;8(1):1-12.