Host trait prediction from high-resolution microbial features

Giovanni Bacci1\*  
1Department of Biology, University of Florence, Via Madonna del Piano 6, Sesto Fiorentino, FI, I-50019, Italy  
\*Corresponding author: [giovanni.bacci@unifi.it](mailto:giovanni.bacci@unifi.it)

## Running head

Predictions from metagenomes

## Summary (500 characters)

Predicting host traits from metagenomes presents new challanges that can be difficult to overcome for researchers without a strong background in bioinformatics and/or statistics. Profiling bacterial communities usign shotgun metagenomics often leads to the generation of a large amount of data that cannot be used directly for training a model. In this chapter we provide a detailed description of how to build a working machine learning model based on taxonomic and functional features of bacterial communities inhabiting the lungs of cystic fibrosis patients. Models are build in the R environment by using different freely available machine learning algortihms.

## Keywords

machine learning, next generation sequencing, metagenomics, host trait prediciton, community profiling, taxonomic profiling, functional profiling

## 1. Introduction

## 2. Materials

A working installation of R [1] is required for this tutorial to work, along with a set of libraries mainly used for building and validating the model. The workflow here proposed uses data coming from a metagenomic study on cystic fibrosis lung communities along time [2]. A complete description of datasets, hardware and software requirements is given below.

### 2.1 Data files

In this tutorial bacterial features obtained from shotgun metagenomics sequencing are used to build a machine learning model and to make infereces on bacterial community structure. The data consist of three main tables reporting quantitative and qualitative information about taxa detected in the lung of subjects included in the study, genes harbored by those taxa, and clinical characteristics of the subjects. Data can be downloaded from LINK DA INSERIRE. Tables are available in the RDS format and can be easily imported into R using the function readRDS(). Since RDS is the native data file format for R, tables can be directly loaded into the R environment without worrying about additional parameters such as field separator, decimal separator, charcater encoding format, and so on. A description of the data files available is given below:

1. taxa\_ab.rds: taxa abundances in all subjects included in the study. Each row of the table is a different observation whereas each column represent a different taxa detected. The proportion of taxa is reported as relative abundance so that the sum of all taxa abundances in each observation is one.
2. gene\_counts.rds: counts of metagenomic reads mapping to bacterial genes recovered from lung communities. The same standard used for taxa abundance was used here with each reporting a different observation and each olumn reporting a different gene. In metagenomic studies genes are usually more than observations and they can be reported into rows instead of columns so to minimize the number of variables and reduce memory requirement.
3. gene\_meta.rds: characteristics of genes included in the gene count table reported in 2. This table is a slight modifyed version of the output produced by eggNOG mapper [3].

### 2.2 Software requirements

Models are generated using a free software environment for statistical computing called R. R is part of many Linux distributions but it can be freely downloaded and installed from <https://cran.r-project.org/> by choosing the appropriate operation system in the “Download and Install R” window. Additional packages needed are listed below (the version of each package used in this tutorial is reported between brackets):

1. compositions [4] (version 1.40.3): a collection of functions for analysing compositional data (quantitative data, strictly positive, which sum to a constant value).
2. vegan [5] (version 2.5.6): a package developed for studying multivariate data produced during ecological studies. It contains several functions for dimensionality reduction (such as correspondence analysis, non-metric multidimensional scaling, and others) and for diversity analysis (either alpha or beta diversity).
3. DESeq2 [6] (version 1.26.0): a suite for the analysis of count data from many biological assays. The package was developed for analysing RNA-seq data but it is also used for amplicon sequence data (such as 16S rRNA metabarcoding) or ChIP-Seq. It implements a transformation function (called variance stabilizing transformation or VST) useful to prepare count data for many machine learning approaches.
4. caret [7] (version 6.0.85): a collection of functions for training and validation of multiple machine learning algorithms. It contains methods for fine tuning classification and regression alorithms using a unified syntax. It also evaluates the performance of models produced using standard metrics such as root mean squared error, receiver operating characteristic curve, accuracy and others.
5. randomForest [8] (version 4.6.14): implementaion of the algorithm described by Breiman in 2001 [9]. This package is used in combination with caret to build the final model.
6. pROC [10] (version 1.16.1): tool for visualizing receiver operating characteristic (ROC) curves. It contains also function for comparing curves from different models.

### 2.3 Hardware requirements

## 3. Methods

### 3.1 Importing data

# importing gene counts  
genes <- readRDS("./data/gene\_counts.rds")  
  
# importing proportion of taxa  
taxa <- readRDS("./data/taxa\_ab.rds")  
  
# sample data  
meta <- readRDS("./data/sample\_meta.rds")

### 3.2 Data transformation

taxa.clr <- clr(taxa)  
class(taxa.clr) <- "matrix"  
  
genes.vst <- DESeqDataSetFromMatrix(t(genes), colData = meta,   
 design = ~ 1)  
genes.vst <- estimateSizeFactors(genes.vst)  
genes.vst <- vst(genes.vst)  
genes.vst <- t(assay(genes.vst))  
  
alpha <- diversity(taxa, index = "invsimpson")  
alpha <- data.frame(A=(alpha - mean(alpha))/sd(alpha))  
  
taxa.pca <- prcomp(taxa.clr, center = T, scale. = T)  
genes.pca <- prcomp(genes.vst, center = T, scale. = T)  
  
taxa.x <- taxa.pca$x[,taxa.pca$sdev^2 >= 1]  
genes.x <- genes.pca$x[,genes.pca$sdev^2 >= 1]  
  
colnames(taxa.x) <- gsub("PC", "T", colnames(taxa.x))  
colnames(genes.x) <- gsub("PC", "F", colnames(genes.x))  
  
y <- meta$genotype  
full.data <- cbind(y = y, taxa.x, genes.x, alpha)  
full.data <- droplevels(full.data[full.data$y != "other",])

## Notes

## References

1. R Core Team (2019) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria

2. Bacci G, Taccetti G, Dolce D, Armanini F, Segata N, Di Cesare F, Lucidi V, Fiscarelli E, Morelli P, Casciaro R, others (2019) Taxonomic and functional dynamics of lung microbiome in cystic fibrosis patients chronically infected with pseudomonas aeruginosa. bioRxiv 609057

3. Huerta-Cepas J, Forslund K, Coelho LP, Szklarczyk D, Jensen LJ, Mering C von, Bork P (2017) Fast Genome-Wide Functional Annotation through Orthology Assignment by eggNOG-Mapper. Molecular Biology and Evolution 34:2115–2122. doi:[10.1093/molbev/msx148](https://doi.org/10.1093/molbev/msx148)

4. Boogaart KG van den, Tolosana-Delgado R (2008) “Compositions”: A unified r package to analyze compositional data. Computers & Geosciences 34:320–338. doi:[10.1016/j.cageo.2006.11.017](https://doi.org/10.1016/j.cageo.2006.11.017)

5. Oksanen J, Blanchet FG, Friendly M, Kindt R, Legendre P, McGlinn D, Minchin PR, O’Hara RB, Simpson GL, Solymos P, Stevens MHH, Szoecs E, Wagner H (2019) Vegan: Community ecology package

6. Love MI, Huber W, Anders S (2014) Moderated estimation of fold change and dispersion for rna-seq data with deseq2. Genome biology 15:550. doi:[10.1186/s13059-014-0550-8](https://doi.org/10.1186/s13059-014-0550-8)

7. Kuhn M (2015) Caret: Classification and regression training. Astrophysics Source Code Library

8. Liaw A, Wiener M (2002) Classification and regression by randomForest. R News 2:18–22

9. Breiman L (2001) Random forests. Machine learning 45:5–32. doi:[10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324)

10. Robin X, Turck N, Hainard A, Tiberti N, Lisacek F, Sanchez J-C, Müller M (2011) PROC: An open-source package for r and s+ to analyze and compare roc curves. BMC Bioinformatics 12:77. doi:[10.1186/1471-2105-12-77](https://doi.org/10.1186/1471-2105-12-77)