

# INSTALLATION AND USER GUIDE FOR BioMiCo

## 1. Contents of the download

Directory	Contents
BioMiCo_R_Package	Contains code for building BioMiCo.
Files	Function
trainbyalltestbyone.R	R script for training and testing on the data.
BioMiCoScripts.R	R script used by trainbyalltestbyone.R.
BioMiCo_analysis.R	Example R script provided to obtain: posterior probabilities for labels for each sample, assemblages for each label, and OTUs for each assemblage.
test.ix	Example test file.
train.ix	Example training file.
env	Example label file

## 2. Installing BioMiCo R package

### 2.1 Prerequisites:

Boost C++ Libraries  
GNU scientific library (GSL)

### 2.2 Installation instructions:

Before building BioMiCo you need to edit the `Makevars` file which is found in the `src` directory in the `BioMiCo_R_Package` directory. The `Makevars` file must indicate the location of the boost header files *on your system*.

Specifically, you must edit the fourth line of `Makevars` file to point to the path where your boost header files are located. Something like  
`/opt/local/include/.`

To Build BioMiCo you must be in the main directory (BioMiCo) which contains the directory BioMiCo\_R\_Package.

To build and install, type the following at the command line:

```
R CMD build BioMiCo_R_Package
R CMD INSTALL BioMiCo_1.0.1.tar.gz
```

Note: The above commands are case sensitive.

### 3. Data files

BioMiCo requires three data input files. Two files contain the OTU count information, one for the training set (`train.ix`) and one for the test set (`test.ix`). The file format for these is: the first row contains the OTU labels (this must be the same between the test and training data). The first column contains the sample IDs, and the remaining columns contain the counts of OTUs for each sample ID. Examples are included as `train.ix` and `test.ix`. (Note the counts must be whole integers, decimals are not allowed.) The remaining file needed is the file containing the labels for the training data (`env`). This file contains two columns, the first column is the sample ID list from the `train.ix` file (in the same order as it occurs in the `train.ix` file) and the second column is the label for that sample ID. (Note no header line should be included). An example file (`env`) is included.

### 4. Running BioMiCo

Once you have input files, you are ready to run BioMiCo. To do this you will use the `trainbyalltestbyone.R` script. There are two parts to this script, a training phase and a testing phase. Parameters that can be adjusted are:

<code>burnin</code>	number of samples before information is collected
<code>nrestarts</code> ,	number of times to restart the runs
<code>ndraws.per.restart</code> ,	how many times you are going to sample the data
<code>delay</code>	how long between sample points

These are adjusted both for the training phase, (line `train.results` in the script) and the testing phase (line `test.results` in the script).

To run the script type:

```
R CMD BATCH trainbyalltestbyone.R
```

This script will create two R binary images one containing the results from the training run (`trainImage.RData`) and one with the results from the test samples

(`predictionImage.RData`). The training results can be reused for future runs if so desired.

Note this will take some time to run depending on number of OTUs, and samples.

## 5. Analysis

We provide an example R script for analysis of the data `BioMiCo_analysis.R`. This script will create three files (`Predictions.txt`, `Assemblage_pp`, and `OTU_pp`). `Predictions.txt` gives the posterior probabilities of labels for each test sample in a tab-delimited table with the rows being samples and columns being labels. `Assemblage_pp` gives the posterior probabilities of assemblages in each label in a tab-delimited table with rows being assemblages and columns being the labels. `OTU_pp` gives the posterior probabilities of the OTUs in each assemblage in a tab-delimited table with rows being the OTUs and columns being the assemblages.

To run the analysis script type:

```
R CMD BATCH BioMiCo_analysis.R
```

**\*\*Please note all of these scripts provided are currently set to run in the current directory however to change this simply change the `setwd` line to point to the directory you want or give complete path to files read or written to. This is also important to note as subsequent runs will overwrite files that currently exist.**