Examples and Figures from Microbiome Recursive Partitioning 2019

Dake Yang, Jethro Johnson, Xin Zhou, Elena Deych, Berkley Shands,
Blake Hanson, Erica Sodergren, George Weinstock, Bill Shannon

August 29, 2019

First we need to load the `HMP` package and data:

```
library(HMP)

data(dmrp_data)
data(dmrp_covars)
```

The data consists of 128 subjects and 29 taxa at the Genus level. The taxon labeled "Other" is the rarest 139 taxa collapsed into one and combined they make up less than 5 percent of the total reads. This was done by the function Data.filter in the HMP package.

The covariate file consists of the same 128 subjects in the same order and 11 cytokines.

I Figure 1

The figure below is the results of running the DM-RPart analysis on the given data and cytokine covariates. The top number in each box is the node number, n= is the number of subjects in that node and the percentage next to that is the percentage of total subjects in that node.

Below each box is the splitting rule to get to the next level. The left branch are subjects that respond TRUE to the splitting rule. For example all the subjects that have a LEPTIN value less than 1476 go to the left and all the others go to the right.

The nodes at the very bottom are called terminal nodes.

```
# Set splitting parameters for DM-Rpart (see ??DM-Rpart for details)
minBucket <- 6
minSplit <- 18

# Set the number of cross validations
# 20 means the model will run 20 times, each time holding 5% of the data out
numCV <- 20

# Run the DM-RPart function with a seed set
set.seed(2019)
DMRPResults <- DM.Rpart.CV(dmrp_data, dmrp_covars, plot=FALSE, minsplit=minSplit,
                minbucket=minBucket, numCV=numCV)

# Pull out and plot the best tree
bestTree <- DMRPResults$bestTree
rpart.plot::rpart.plot(bestTree, type=2, extra=101, box.palette=NA, branch.lty=3,
                shadow.col="gray", nn=FALSE)
```

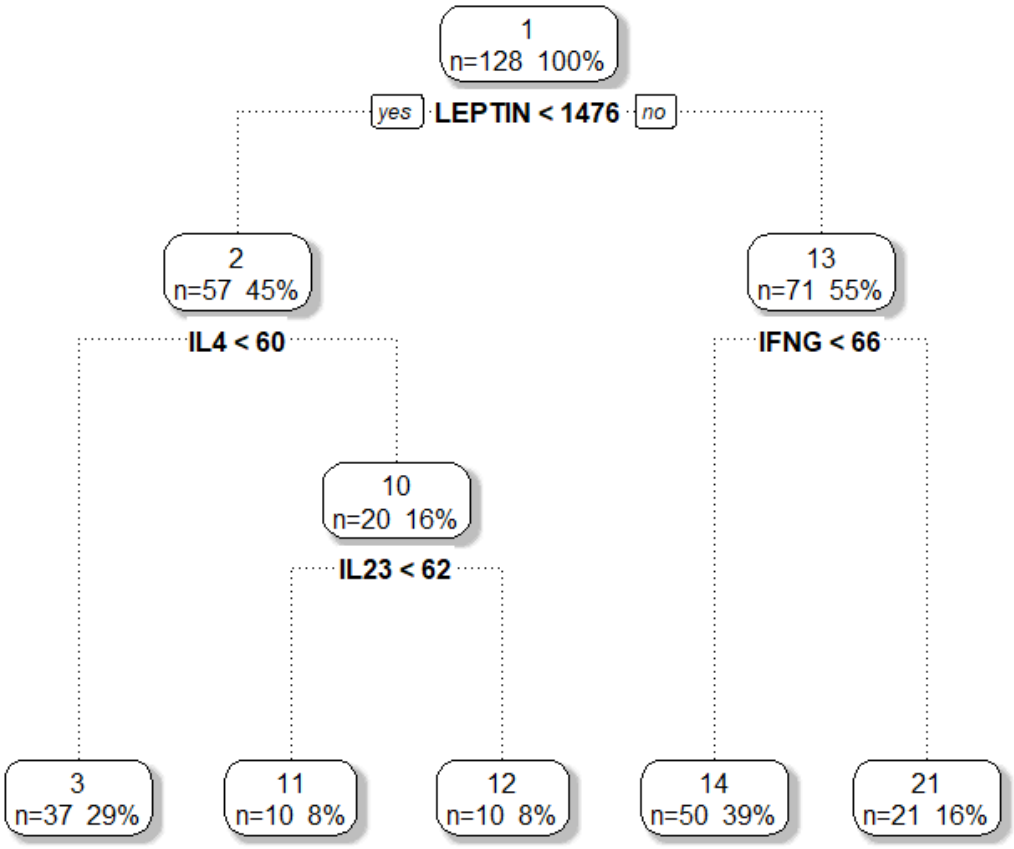

Figure 1: DM-RPart Tree

II Figure 2

The barcharts below show the taxa composition for each terminal node from the above rpart tree plot.

```
# Split the data by terminal nodes
nodeNums <- bestTree$frame$yval[bestTree$frame$var == "<leaf>"]
nodeList <- split(dmrp_data, f=bestTree$where)
names(nodeList) <- paste("Node", nodeNums)

# Get the PI for each terminal node
myEst <- Est.PI(nodeList)
myPI <- myEst$MLE$params

# Plot the PI for each terminal node
myColr <- rainbow(ncol(dmrp_data))
lattice::barchart(PI ~ Group, data=myPI, groups=Taxa, stack=TRUE, col=myColr,
                  ylab="Fractional Abundance", xlab="Terminal Node",
                  auto.key=list(space="top", columns=3, cex=.65, rectangles=FALSE,
                                col=myColr, title="", cex.title=1))
```

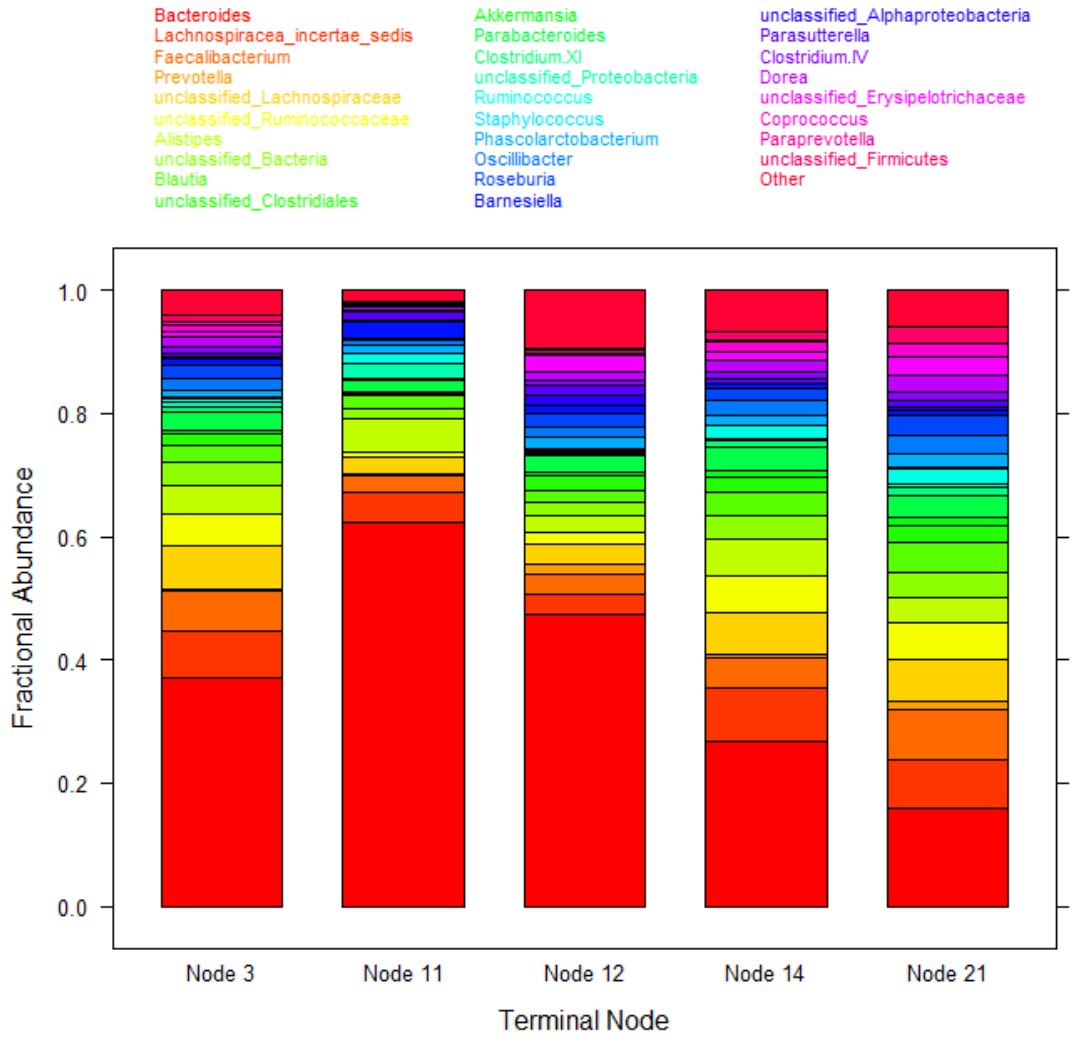

Figure 2: Barchart Comparing Terminal Nodes Taxa Composition