Example 3: Community-level Hypothesis Test using BlockBootID with mvabund, block length selection

Setting Up

First load the data and R packages you need into the R workspace.

```r
load("PlayData_for_Examples.RData") ### load the data for the examples
source("LoadFunctions.R") #### Source the functions required to run the block bootstrap


#Install packages if required
if(!require(mvabund)) { install.packages("mvabund", repos =
"http://cran.us.r-project.org"); require(mvabund) }

library(mvabund)
```

Motivation

```r
set.seed(42)
treatment =rbinom(500,1,0.5)
multispecies_dat = data.frame(multispecies_dat,treatment)
```

Coding StatFunction argument

Define MultSpeciesTestStat a function which will be the *StatFunction* argument to *BlockBootApply*.

```r
MultSpeciesTestStat = function(dat){
responseMultiSpecies = mvabund(dat[,1:20])

null.model = manyglm (responseMultiSpecies~1, data=dat, family="binomial")
alt.model = manyglm (responseMultiSpecies~temperature*treatment, data=dat, family="binomial")
Test.stat = sum(alt.model$two.loglike) - sum(null.model$two.loglike)
;
Test.stat
}

MultSpeciesTestStat (multispecies_dat)
```

```
## [1] 10261.29
```

Run Block Bootstrap

Run Block Bootstrap, using Lahiri method of block size selection. We search over 4 blocks sizes (0- i.e. IID, 0.05, 0.1, 0.2). BlockBootApply uses the size which minimises the Empirical MSE of SE(T), where T is the multispecies likelihood ratio test statistic. Use 0.05 as tuning block length, (tuning_block_length =2).

```
lookuptables.folderpathname = "LookupTables/"
```

```
Results.Example3 = BlockBootApply (x = x ,y = y ,block_Ls = c(0,0.05,0.1,0.2), Grid_space = 0.01 ,
dat = multispecies_dat , Stat.function = MultSpeciesTestStat,  tuning_block_length = 2, NBoot = 2,
method.block.length.select = "Lahiri",  type="SE", lookuptables.folderpath=lookuptables.folderpathname)


L.example3  = Results.Example3$Lahiri_block_size
#The chosen block size parameter
L.example3
```

```
## [1] 0.1
```

Now make a bootID matrix and feed into mvabund::summary.

```
BootID.example3 = BlockBootID(x = x ,
y = y,
block_Ls = L.example3,
NBoot = 500,
Grid_space = 0.01,
lookuptables.folderpath =  lookuptables.folderpathname, shape="disc")

#######

responseMultiSpecies = mvabund(multispecies_dat[,1:20])

mod.full  = manyglm(responseMultiSpecies~temperature*treatment, data = multispecies_dat,
family="binomial")

anova(mod.full, bootID = BootID.example3, resamp="case")
```

```
## Warning in anova.manyglm(mod.full, bootID = BootID.example3, resamp =
## "case"): 'montecarlo' or 'pit.trap' should be used for binomial regression.
```

```
## Warning in anova.manyglm(mod.full, bootID = BootID.example3, resamp =
## "case"): case resampling with score and LR tests is under development. try
## case resampling with wald test.
```

```
## Using <int> bootID matrix from input.
## Time elapsed: 0 hr 1 min 22 sec
```

```
## Analysis of Deviance Table
##
## Model: manyglm(formula = responseMultiSpecies ~ temperature * treatment,
## Model:     family = "binomial", data = multispecies_dat)
##
## Multivariate test:
##                       Res.Df Df.diff   Dev Pr(>Dev)
## (Intercept)              499
## temperature              498       1 10224    0.421
## treatment                497       1    19    0.990
## temperature:treatment    496       1    19    0.982
## Arguments:
##  Test statistics calculated assuming uncorrelated response (for faster computation)
##  P-value calculated using 500 resampling iterations via case resampling (to account for correlation i
```

```
##########
```