SparCC (global gut full data)

Back to Table of Contents

All of the code in this page is meant to be run on the command line unless otherwise specified.

Install SparCC

Download the repository from the downloads page. Unzip the folder, and put it somewhere convenient (like in your course code repository folder). Test that it is working by running the following command on the command line:

```
# Test that SparCC is running.
# Note that you will have to fix the path to be the correct path
# to the "SparCC.py" file on your computer.
python ../../../sparcc/SparCC.py -h
```

```
## Usage: Compute the correlation between components (e.g. OTUs).
## By default uses the SparCC algorithm to account for compositional effects.
## Correlation and covariance (when applies) matrices are written out as txt files.
## Counts file needs to be a tab delimited text file where columns are samples and rows are components
##  See example/fake_data.txt for an example file.
##
## Usage:   python SparCC.py counts_file [options]
## Example: python SparCC.py example/fake_data.txt -i 20 --cor_file=Cor_mat.out
##
## Options:
##   -h, --help            show this help message and exit
##   -c COR_FILE, --cor_file=COR_FILE
##                         File to which correlation matrix will be written.
##   -v COV_FILE, --cov_file=COV_FILE
##                         File to which covariance matrix will be written.
##   -a ALGO, --algo=ALGO  Name of algorithm used to compute correlations (SparCC
##                         (default) | pearson | spearman | kendall)
##   -i ITER, --iter=ITER  Number of inference iterations to average over (20
##                         default).
##   -x XITER, --xiter=XITER
##                         Number of exclusion iterations to remove strongly
##                         correlated pairs (10 default).
##   -t TH, --thershold=TH
##                         Correlation strength exclusion threshold (0.1
##                         default).
```

Prepare data

We will run SparCC on the Global Gut genus data, only including adults living in the USA. We will also choose a subset of the more prevalent genera (present in about 20% of people or more) for testing to keep things running quickly.

```
# RUN ON THE COMMAND LINE

# First, extract only adults living in the USA.
filter_samples_from_otu_table.py -i otu_table.biom -m map.txt -o otu_table_USA_adults.biom -s "AGE_GROU

# Summarize taxa at the genus level
summarize_taxa.py -i otu_table_USA_adults.biom -L 6 -o taxa-USA-adults

# remove genera present in < 60 samples
filter_otus_from_otu_table.py -i taxa-USA-adults/otu_table_USA_adults_L6.biom -s 60 -o taxa-USA-adults/

# create a text version (for SparCC) and a JSON version (for R)
biom convert -i taxa-USA-adults/otu_table_USA_adults_L6_s60.biom --to-json -o taxa-USA-adults/otu_table_
biom convert -i taxa-USA-adults/otu_table_USA_adults_L6_s60.biom --to-tsv -o taxa-USA-adults/otu_table_U

# Now remove the first line of the taxon file. Same would apply to an OTU table.
sed 1d taxa-USA-adults/otu_table_USA_adults_L6_s60.txt > taxa-USA-adults/otu_table_USA_adults_L6_s60_fo
```

Run SparCC

```
python ../../../sparcc/SparCC.py taxa-USA-adults/otu_table_USA_adults_L6_s60_for_sparcc.txt
```

```
## reading data
## computing correlations
##  Running iteration 0
##  Running iteration 1
##  Running iteration 2
##  Running iteration 3
##  Running iteration 4
##  Running iteration 5
##  Running iteration 6
##  Running iteration 7
##  Running iteration 8
##  Running iteration 9
##  Running iteration 10
##  Running iteration 11
##  Running iteration 12
##  Running iteration 13
##  Running iteration 14
##  Running iteration 15
##  Running iteration 16
##  Running iteration 17
##  Running iteration 18
##  Running iteration 19
## writing results
## wrote cor_mat_SparCC.out
## wrote cov_mat_SparCC.out
## Done!
```

The correlation output will be in the file cor_mat_SparCC.out.

Comparison to Spearman correlation

Now we will compare SparCC to Spearman correlation in R.

The following commands will be run in R.

First, load the data into **R**.

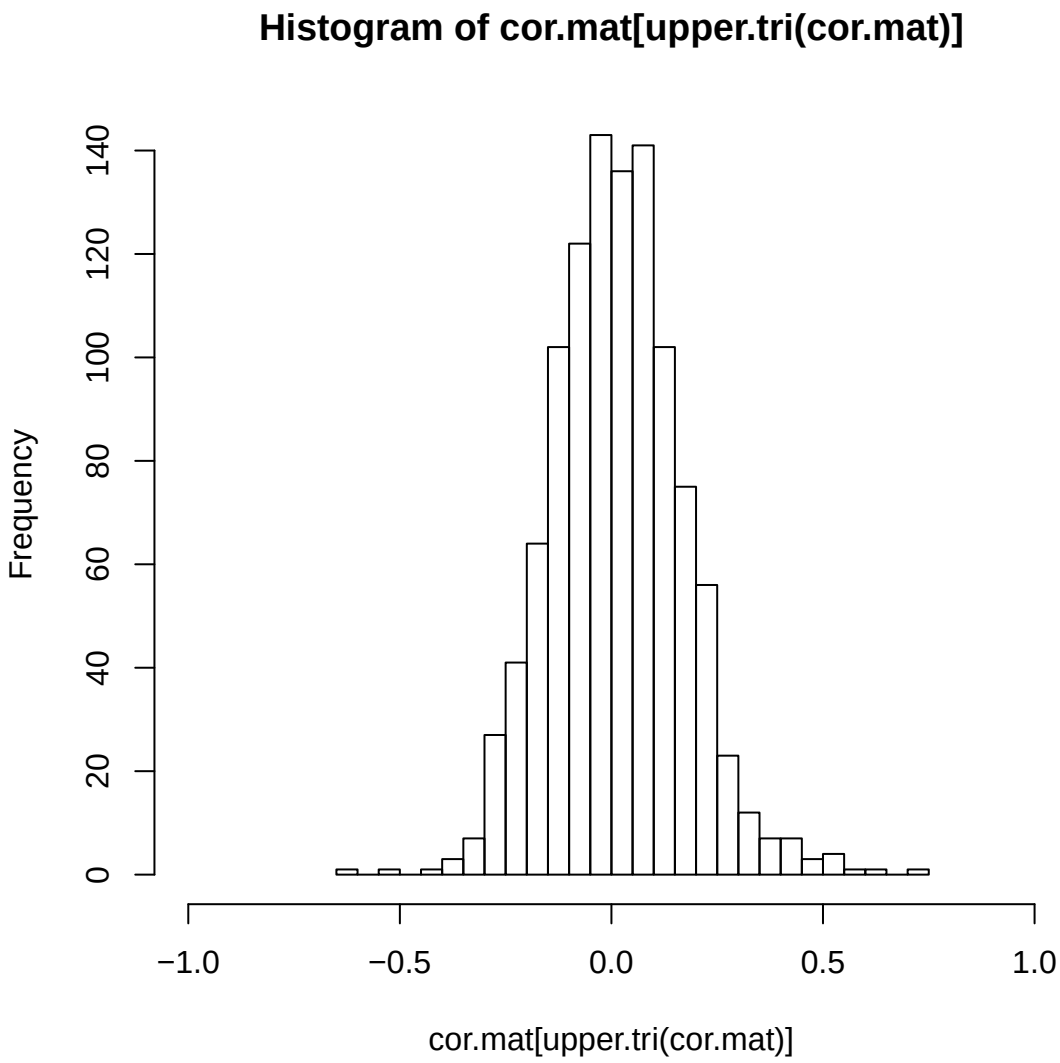
```
# load the biom library
library('biom')

# read in the biom table and extract the data
x <- t(as.matrix(biom_data(read_biom('taxa-USA-adults/otu_table_USA_adults_L6_s60_json.biom'))))

# read in the sparcc results
sparcc.mat <- read.table('cor_mat_SparCC.out',sep='\t',head=T,row=1)
```

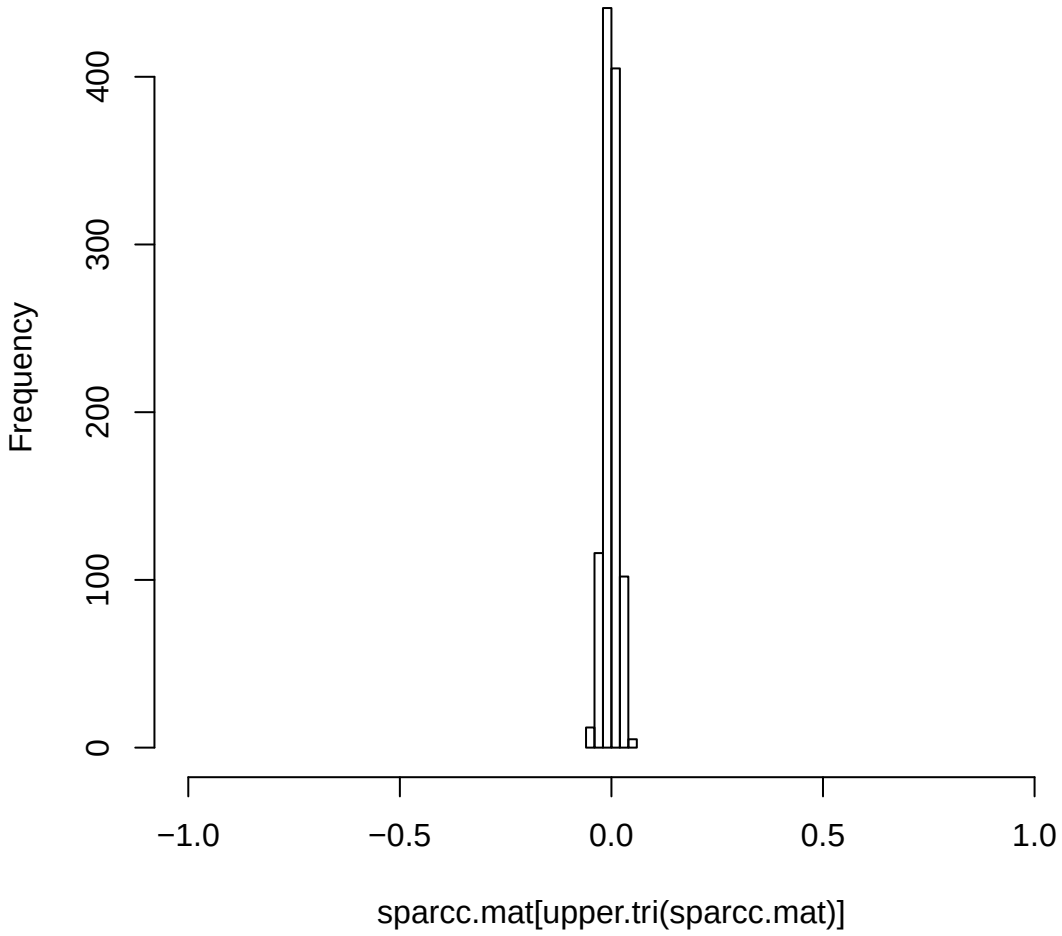
Plot a histogram of correlations inferred by Spearman correlation.

```
cor.mat <- cor(x,method='spear')
hist(cor.mat[upper.tri(cor.mat)], br=30, xlim=c(-1,1))
```



There are a number of large correlations found above 0.5. Let us compare to SparCC.

```
hist(sparcc.mat[upper.tri(sparcc.mat)],br=6,xlim=c(-1,1))
```



SparCC found no large correlations, indicating that the correlations found by Spearman correlation could be explained by compositional artifacts.