Computing local specificity index

Mahendra Mariadassou
mahendra.mariadassou@jouy.inra.fr

August 24, 2015

This vignette shows how to reproduce the analysis and graphics used in Mariadassou et al. (2015). The function used to compute local specificity index make heavy use of phyloseq (McMurdie and Holmes, 2013) and the graphics are all made using ggplot2 (Wickham, 2009) so we load those two packages.

```
library(phyloseq)
packageVersion("phyloseq")

## [1] '1.12.2'

library(ggplot2)
packageVersion("ggplot2")

## [1] '1.0.1'
```

Instructions about how to install phyloseq are available at the author's website: https://joey711.github.io/phyloseq/. ggplot2 is available on CRAN and can be installed with install.packages. We then load the custom functions used to compute local specificity index.

```
source("specificity_methods.R")
```

and illustrate their use on the Gloabl Patterns data set (Caporaso et al., 2011) provided by phyloseq

```
data(GlobalPatterns)
## Filter out mock communities
GP <- subset_samples(GlobalPatterns, SampleType != "Mock")
```

To use your own data, you can use the various data import functions of phyloseq, detailed in the very nice following tutorial: http://joey711.github.io/phyloseq/import-data.html

To speed up computations a bit, we filter out the singletons and rarefy the dataset to 10,000 reads per sample.

```
set.seed(24082015)  ## for reproducibility
GP.down <- prune_taxa(taxa_sums(GP) > 1, GP)  ## remove singletons
GP.down <- rarefy_even_depth(GP.down, sample.size = 10000)  ## rarefaction
```

GP.down is a phyloseq-class object with several components. We only use two here: otu_table, the count table, and sample_data, the metadata associated with the samples (see phyloseq documentation for more details on additional components).

```
print(GP.down)

## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 8172 taxa and 23 samples ]
## sample_data() Sample Data:       [ 23 samples by 7 sample variables ]
## tax_table()   Taxonomy Table:    [ 8172 taxa by 7 taxonomic ranks ]
## phy_tree()    Phylogenetic Tree: [ 8172 tips and 8171 internal nodes ]

head(otu_table(GP.down), n = 2)

## OTU Table:          [2 taxa and 23 samples]
##                      taxa are rows
##        CL3 CC1 SV1 M31Fcsw M11Fcsw M31Plmr M11Plmr F21Plmr M31Tong M11Tong
## 549322   0   0   0       0       0       0       0       0       0       0
## 255340   2   2   0       0       0       0       0       0       0       0
##        LMEpi24M SLEpi20M AQC1cm AQC4cm AQC7cm NP2 NP3 NP5 TRRsed1 TRRsed2
## 549322        0        0      0      0      1   0   0   0       0       0
## 255340        0        0      0      0      0   0   0   0       0       0
##        TRRsed3 TS28 TS29
## 549322       0    0    0
## 255340       0    0    0

head(sample_data(GP.down), n = 2)

## Sample Data:        [2 samples by 7 sample variables]:
##     X.SampleID  Primer Final_Barcode Barcode_truncated_plus_T
## CL3        CL3 ILBC_01        AACGCA                   TGCGTT
## CC1        CC1 ILBC_02        AACTCG                   CGAGTT
##     Barcode_full_length SampleType
## CL3         CTAGCGTGCGT       Soil
## CC1         CATCGACGAGT       Soil
##                                Description
## CL3 Calhoun South Carolina Pine soil, pH 4.9
## CC1 Cedar Creek Minnesota, grassland, pH 6.1
```

We can now compute the local specificity using `SampleType` (read directly from the sample data component) as a grouping factor. Alternatively, you can provide the grouping factor directly. We also use 999 stratified bootstrap replicates to estimate the error of the local specificity coefficient. As stated in Mariadassou et al. (2015), the bootstrap is stratified by levels of the grouping factor to preserve the structure of the data set (with respect to the grouping factor): to create bootstrap Soil samples, we only resample from Soil samples.

```
specOTUS <- estimate_local_specificity(GP.down, group = "SampleType", index = "indspec",
    B = 999)

## Estimating se, may take a few minutes
```

The result is a data frame with one line per taxon and the following columns:

- specificity: observed local specificity
- group: grouping factor level
- abundance: local relative abundance of otu (all samples of a level are weighted equally)

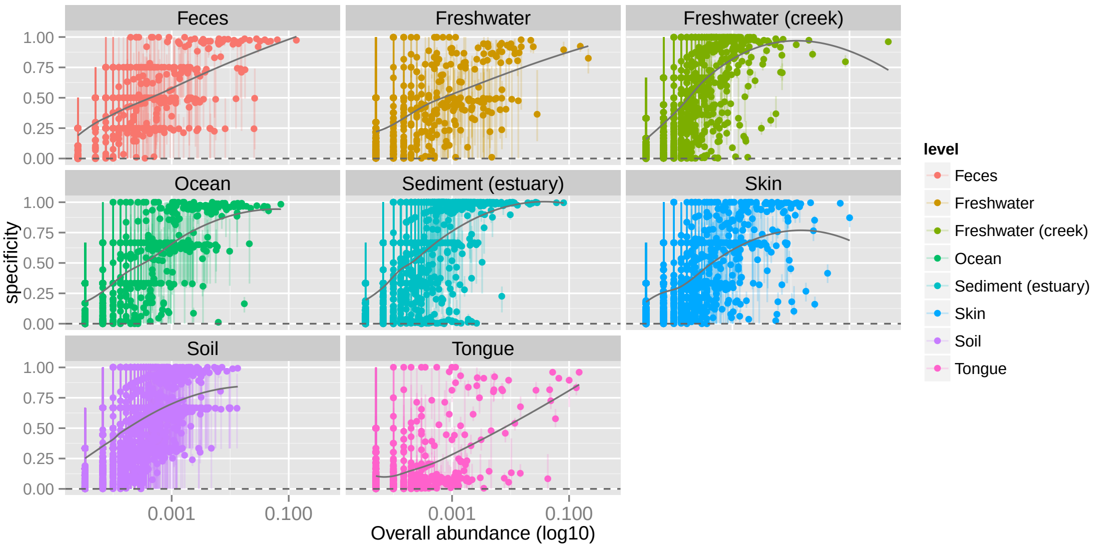

Figure 1: Local specificity. Error bars correspond to the 50% confidence interval for the local specificity values (computed from the stratified bootstrap distribution).

- mean, sd, quantiles 5, 25, 50, 75 and 95% (of the stratified bootstrap distribution) if 'se' is TRUE

```
head(specOTUS, n = 2)

##        otu level abundance specificity mean    sd q5  q25   q50   q75   q95
## 21   54107 Feces   2.5e-05       0.009 0.02 0.033  0 0.00 0.009 0.024 0.079
## 73 227785 Feces   7.5e-04       0.500 0.51 0.252  0 0.25 0.500 0.750 1.000
```

We can then plot the results (Figure 1) using default settings of the `plot_local_specificity` function. You may get some warnings for the loess fit to the data but it is safe to ignore them.

```
plot_local_specificity(specOTUS)
```

The default settings plot the standard error but you can remove them from the plot to make it easier to read (Figure 2) using `se = FALSE` in the call:

```
plot_local_specificity(specOTUS, se = FALSE)
```

Stratified bootstrap computes the variance associated with a limited number of samples in each group. It is clear from Figure 1 that local specificity increases on average with local abundance but this could simply be a fluke. We test whether abundance-specificity relationship derives from the grouping factor by "breaking" the structure imposed by the grouping factor and assessing whether the relationship is preserved. We do so with standard bootstrap replicates, *i.e.* to create bootstrap Soil samples, we resample from all samples. Once again, we use 999 boostrap replicates.

```
specOTUStest <- test_local_specificity(GP.down, group = "SampleType", B = 999,
    replace = TRUE, type = "global", index = "indspec")

## Estimating p-values, may take a few minutes
```

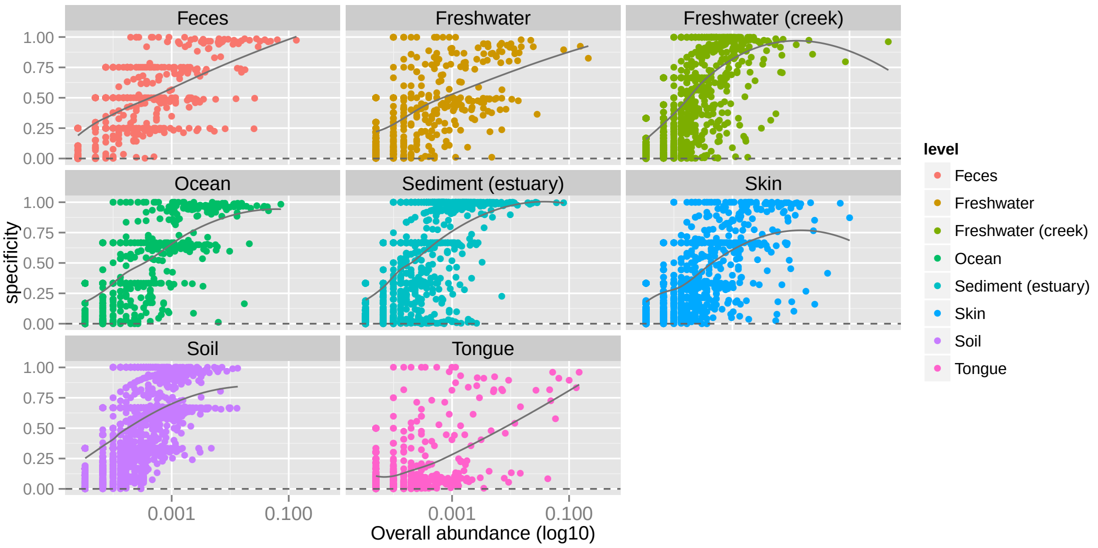

Figure 2: Local specificity without error bars.

The result is a data frame with one line per taxon and the following columns:

- specificity: observed specificity
- level: grouping factor level
- abundance: local relative abundance of otu (all samples of a level are weighted equally)
- rawp: raw p-value
- adjp: adjusted p-value (corrected using "fdr")
- mean, sd, quantiles 50, 75, 90, 95 and 99% (of the standard bootstrap distribution) if 'se' is TRUE

The p-values correspond to the probability that the local specificity is as high as observed under the hypothesis of no structure from the grouping factor. They are computed from the quantiles of the standard bootstrap distribution.

```
head(specOTUStest, n = 3)

##        otu level abundance specificity  rawp adjp bmean  bsd    bmin bq50
## 21   54107 Feces   2.5e-05     0.00904 1.000 1.00  0.29 0.14 0.00904 0.26
## 73  227785 Feces   7.5e-04     0.50000 0.092 0.34  0.25 0.15 0.00000 0.25
## 147  12812 Feces   5.0e-05     0.00051 1.000 1.00  0.45 0.20 0.00051 0.40
##     bq75 bq90 bq95 bq99
## 21  0.36 0.49 0.57 0.74
## 73  0.33 0.46 0.50 0.67
## 147 0.57 0.75 0.86 0.95
```

We can represent the local specificity under the null hypothesis of no structure, taken here to be the mean of the standard bootstrap distribution (Figure 3):

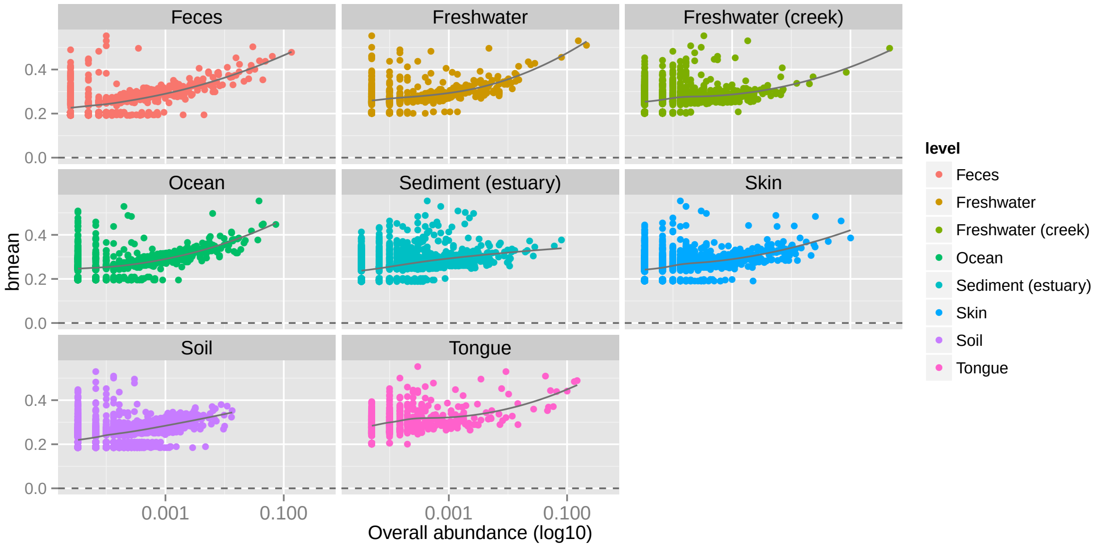

Figure 3: Local specificity without error bars, under the null hypothesis.

```
plot_local_specificity(specOTUStest, y = "bmean", se = FALSE)
```

There is a upward trend, but it is much shallower than for the structured data (note that the y-scale is different)

We can finally plot the observed specificity and its expected value under the null distribution on the same graph but it require a bit more work.

```
specOTUS <- merge(specOTUS, specOTUStest)
attr(specOTUS, "index") <- "indspec"
## Plot local specificty values for SampleType
p <- plot_local_specificity(specOTUS, y = "mean", plot = FALSE)
## Add smoother (loess fit) to highlight the abundance-specificity
## relationship
p <- p + geom_smooth(color = "grey40", method = "loess", se = TRUE)
## Add expected specificity values under the null distribution
p <- p + geom_point(aes(x = abundance, y = bmean), color = "grey60", alpha = 0.5)
## Add error bars for the expected values
p <- p + geom_errorbar(aes(x = abundance, ymin = bmean - bsd, ymax = bmean +
    bsd), color = "grey60", alpha = 0.2)
## Add smoother for the abundance-specificity relationship for the expected
## values
p <- p + geom_smooth(aes(x = abundance, y = bmean), method = "loess", color = "grey40",
    se = TRUE)
## change axes titles, ticks and tick label
p <- p + labs(x = "Local abundance", y = "Local specificity")
p <- p + scale_y_continuous(breaks = c(0:4)/4, limits = c(0, 1))
## remove the legend and use white background
p <- p + theme_bw() + theme(legend.position = "none")
plot(p)
```

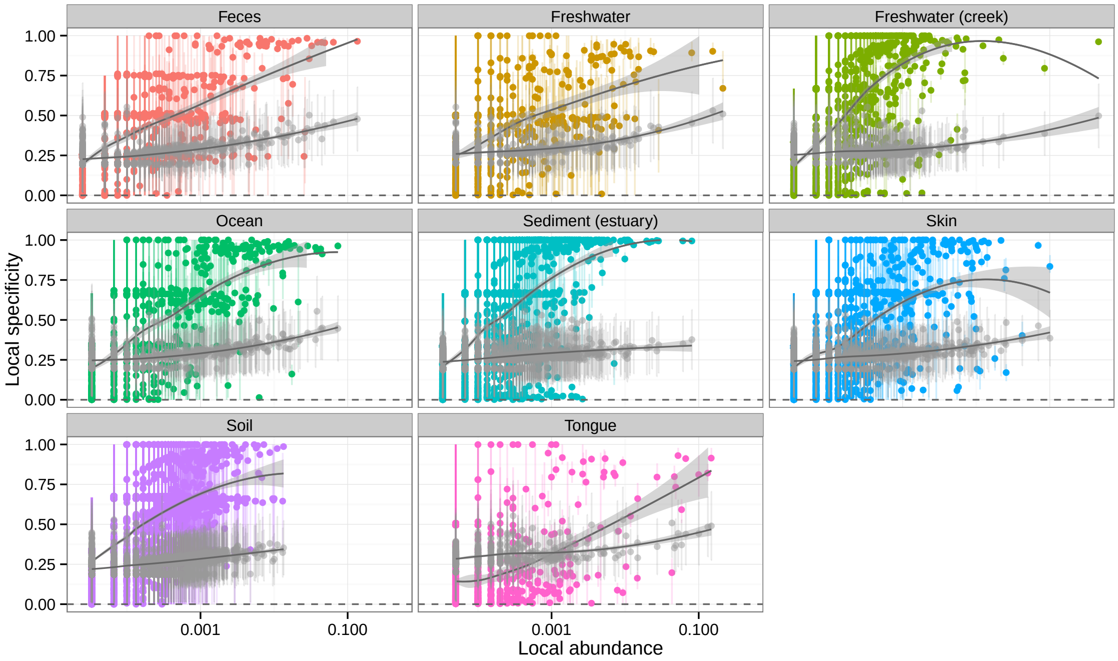

Figure 4: Observed (colored) and expected (grey) abundance-specificity relationships.

```
## Save the figure in your favorite format
ggsave(filename = "Figure.png", plot = p, width = 10, height = 6)
```

References

J. Gregory Caporaso, Christian L Lauber, William A Walters, Donna Berg-Lyons, Catherine A Lozupone, Peter J Turnbaugh, Noah Fierer, and Rob Knight. Global patterns of 16s rrna diversity at a depth of millions of sequences per sample. *Proc Natl Acad Sci U S A*, 108 Suppl 1: 4516–4522, Mar 2011. doi: 10.1073/pnas.1000080107. URL `http://dx.doi.org/10.1073/pnas.1000080107`.

Mahendra Mariadassou, Samuel Pichon, and Dieter Ebert. Microbial ecosystems are dominated by specialist taxa. *Ecology Letters*, 18(9):974–982, 2015. ISSN 1461-0248. doi: 10.1111/ele.12478. URL `http://dx.doi.org/10.1111/ele.12478`.

Paul J. McMurdie and Susan Holmes. phyloseq: An r package for reproducible interactive analysis and graphics of microbiome census data. *PLoS ONE*, 8(4):e61217, 04 2013. doi: 10.1371/journal.pone.0061217. URL `http://dx.doi.org/10.1371%2Fjournal.pone.0061217`.

Hadley Wickham. *ggplot2: elegant graphics for data analysis*. Springer New York, 2009. ISBN 978-0-387-98140-6. URL `http://had.co.nz/ggplot2/book`.