Introduction to coreMicrobiome

Dan Lin

2022-04

1. Introduction

coreMicrobiome is a web-based R Shiny graphical user interface (GUI) with a R package for scientists without/with programming expertise to conduct explore and visualization of core microbial species which comprises four functional modules:

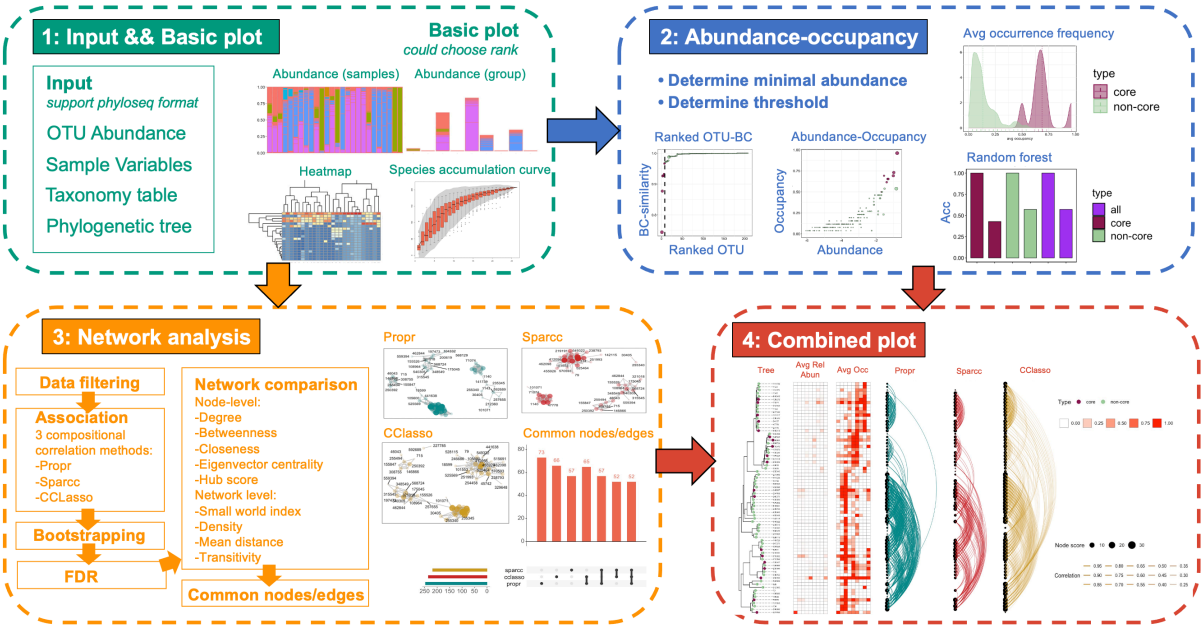
(1) Initial visualization of sampling effort and distribution of dominant bacterial taxa among groups or individual samples at different taxonomic levels;

(2) Analysis of Abundance-occupancy distribution and visualizations;

(3) Co-occurrence network construction, analysis, comparisons and visualizations;

(4) A combined visualization of abundance-occupancy distribution and co-occurrence network for understanding the core species from the common perspective and ecosystem perspective.

2. Overview of the coreMicrobiome analysis



3. Loading packages

Let us first load the package from github.

```
library(devtools)
devtools::install_github('lindan1128/coreMicrobiome', force = TRUE) #
, force = TRUE
```

```
library(coreMicrobiome); packageVersion("coreMicrobiome")
```

```
[1] ‘1.0’
```

Let us then some load necessary libraries.

```
library(phyloseq); packageVersion("phyloseq")
library(microbiome); packageVersion("microbiome")
library(ggplot2); packageVersion("ggplot2")
library(gridExtra); packageVersion("gridExtra")
library(ggtree); packageVersion("ggtree")
library(ggpubr); packageVersion("ggpubr")
library(pheatmap); packageVersion("pheatmap")
library(vegan); packageVersion("vegan")
library(dplyr); packageVersion("dplyr")
library(tidyr); packageVersion("tidyr")
library(randomForest); packageVersion("randomForest")
library(caret); packageVersion("caret")
library(tidygraph); packageVersion("tidygraph")
library(igraph); packageVersion("igraph")
library(qgraph); packageVersion("qgraph")
library(ggraph); packageVersion("ggraph")
library(compositions); packageVersion("compositions")
library(reshape2); packageVersion("reshape2")
library(UpSetR); packageVersion("UpSetR")
library(propr); packageVersion("propr")
```

```
[1] ‘1.36.0’
[1] ‘1.14.0’
[1] ‘3.3.5’
[1] ‘2.3’
[1] ‘3.0.4’
[1] ‘0.4.0’
[1] ‘1.0.12’
[1] ‘2.5.7’
[1] ‘1.0.8’
[1] ‘1.2.0’
[1] ‘4.6.14’
[1] ‘6.0.89’
[1] ‘1.2.0’
[1] ‘1.3.0’
[1] ‘1.9’
[1] ‘2.0.5’
[1] ‘2.0.4’
[1] ‘1.4.4’
[1] ‘1.4.0’
[1] ‘4.3.0’
```

4. Data preparation

We show the GlobalPatterns example workflow as initially outlined in (McMurdie and Holmes 2013).

We retrieve the example data in phyloseq format.

Let us load the data.

```
data(GlobalPatterns, package = "phyloseq")
GlobalPatterns
```

```
phyloseq-class experiment-level object
otu_table()   OTU Table:         [ 19216 taxa and 26 samples ]
sample_data() Sample Data:       [ 26 samples by 7 sample variables ]
tax_table()   Taxonomy Table:    [ 19216 taxa by 7 taxonomic ranks ]
phy_tree()    Phylogenetic Tree: [ 19216 tips and 19215 internal nodes
 ]
```

We only use the archaea data to reduce the data load.

```
data = subset_taxa(GlobalPatterns, Kingdom == "Archaea")
data
```

```
phyloseq-class experiment-level object
otu_table()   OTU Table:         [ 208 taxa and 26 samples ]
sample_data() Sample Data:       [ 26 samples by 7 sample variables ]
tax_table()   Taxonomy Table:    [ 208 taxa by 7 taxonomic ranks ]
phy_tree()    Phylogenetic Tree: [ 208 tips and 207 internal nodes ]
```

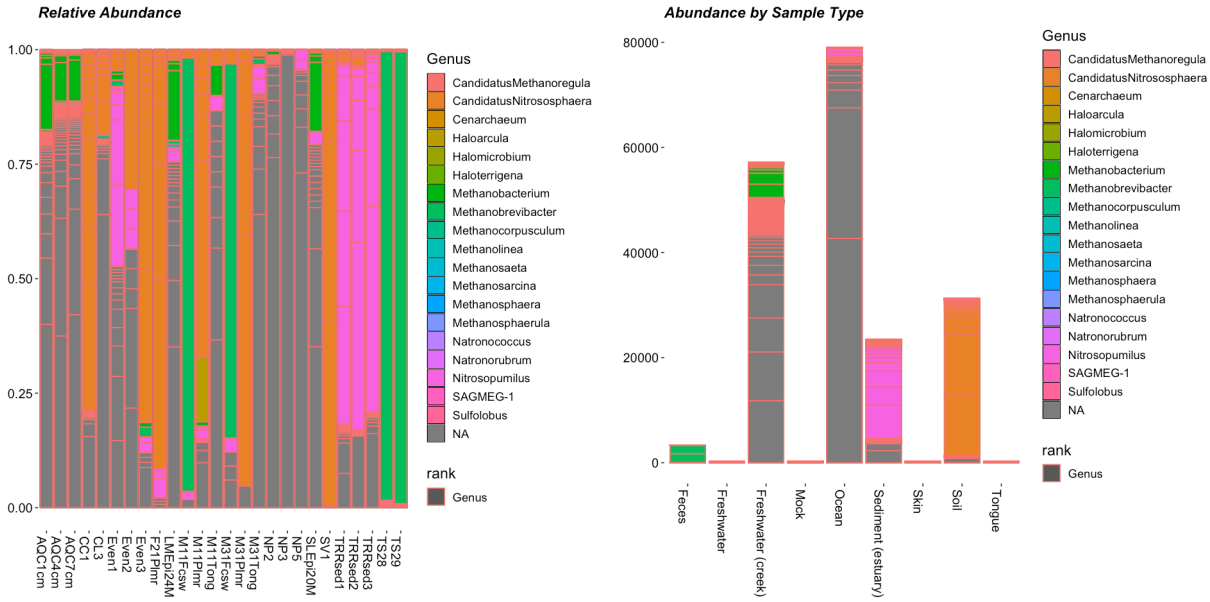
```
otu = otu_table(data)
sample = sample_data(data)
tax = tax_table(data)
tree = phy_tree(data)
```

5. coreMicrobiome

5.1 Basic plot

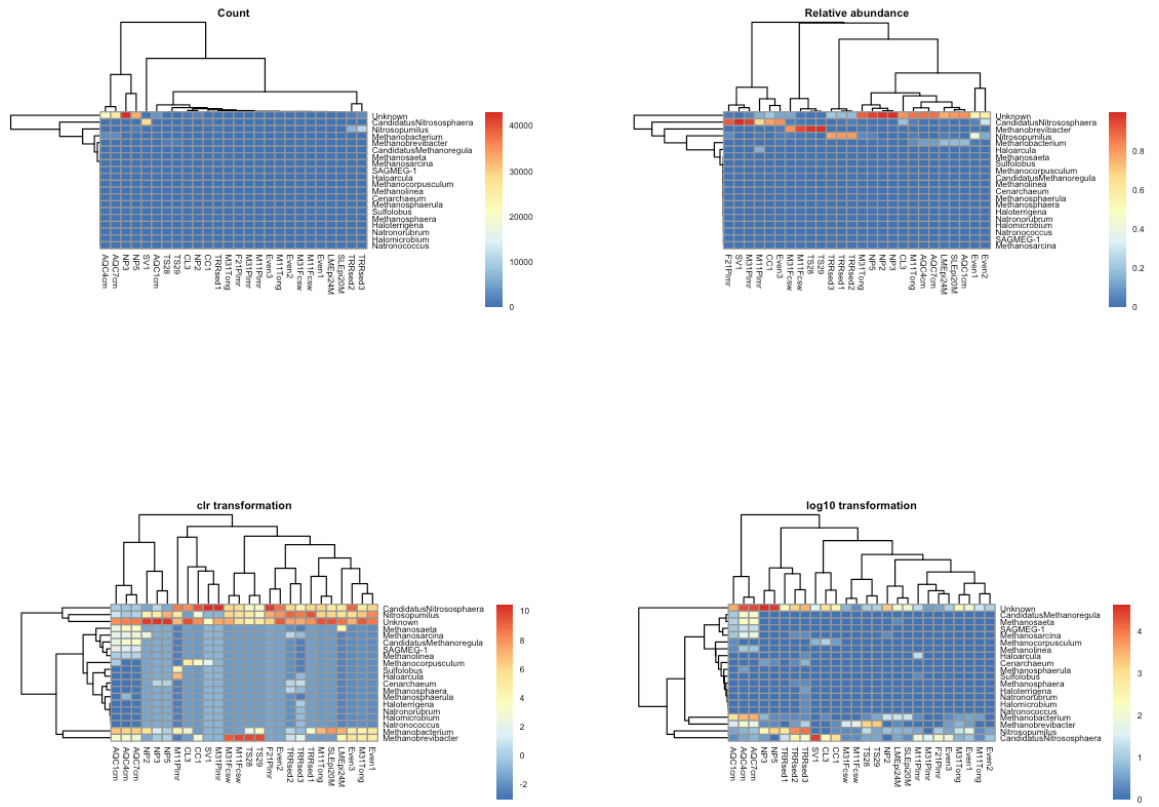
We plots abundance among samples and groups (sample_group) at specific taxonomic level.

```
options(repr.plot.width = 16, repr.plot.height = 8)
abundance_plot(otu, tax, sample, sample_group = 'SampleType', rank = '
Genus')
```



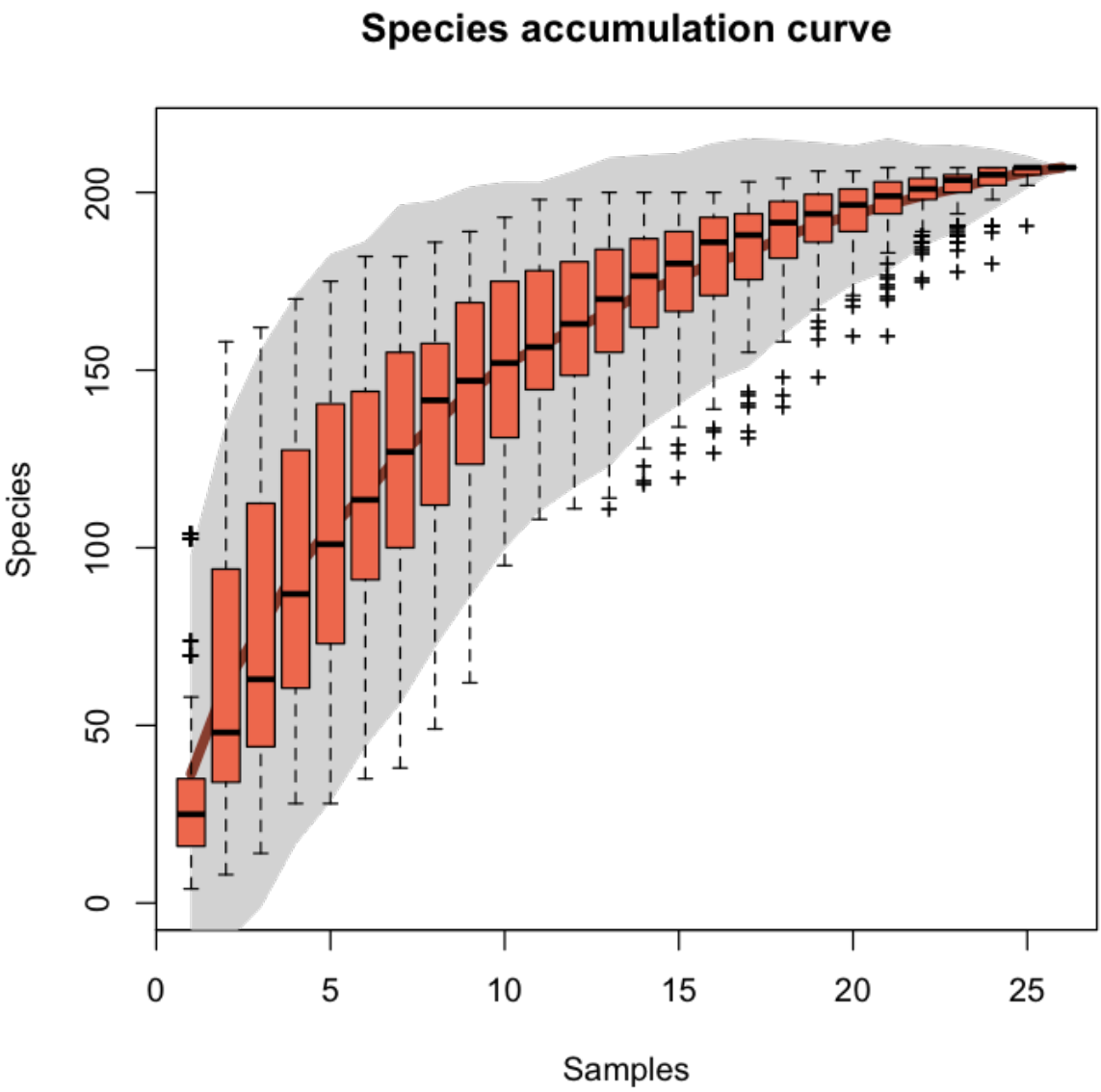
We plots heatmap based on count data, relative abundance data, clr transformation data or log10 transformation data.

```
options(repr.plot.width = 10, repr.plot.height = 8)
heatmap_plot(otu, tax, sample, 'Genus')
```



We plots species accumulation curve with boxplots indicating the 95% CI.

```
options(repr.plot.width = 6, repr.plot.height = 6)
species_acc_plot(otu)
```

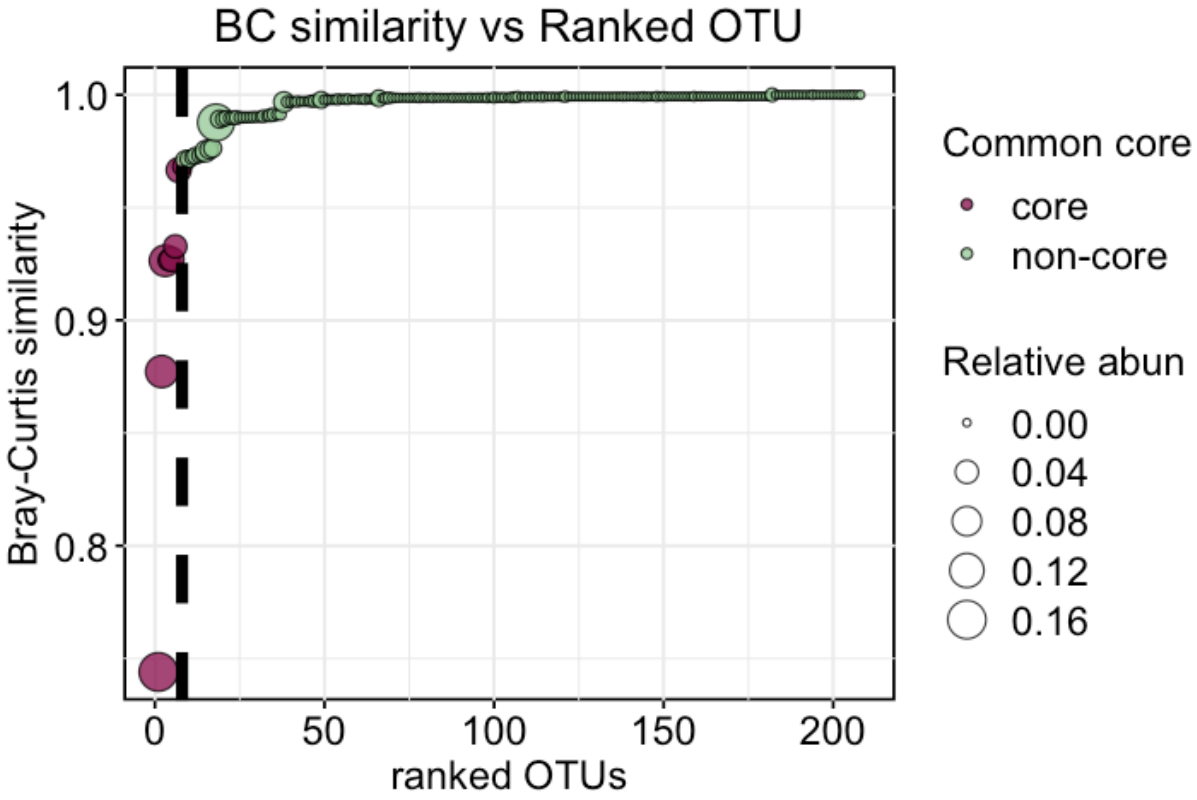


5.2 Abundance-occupancy analysis

We perform the abundanc-occupancy analysis as described in (Shade and Stopnisek 2019).

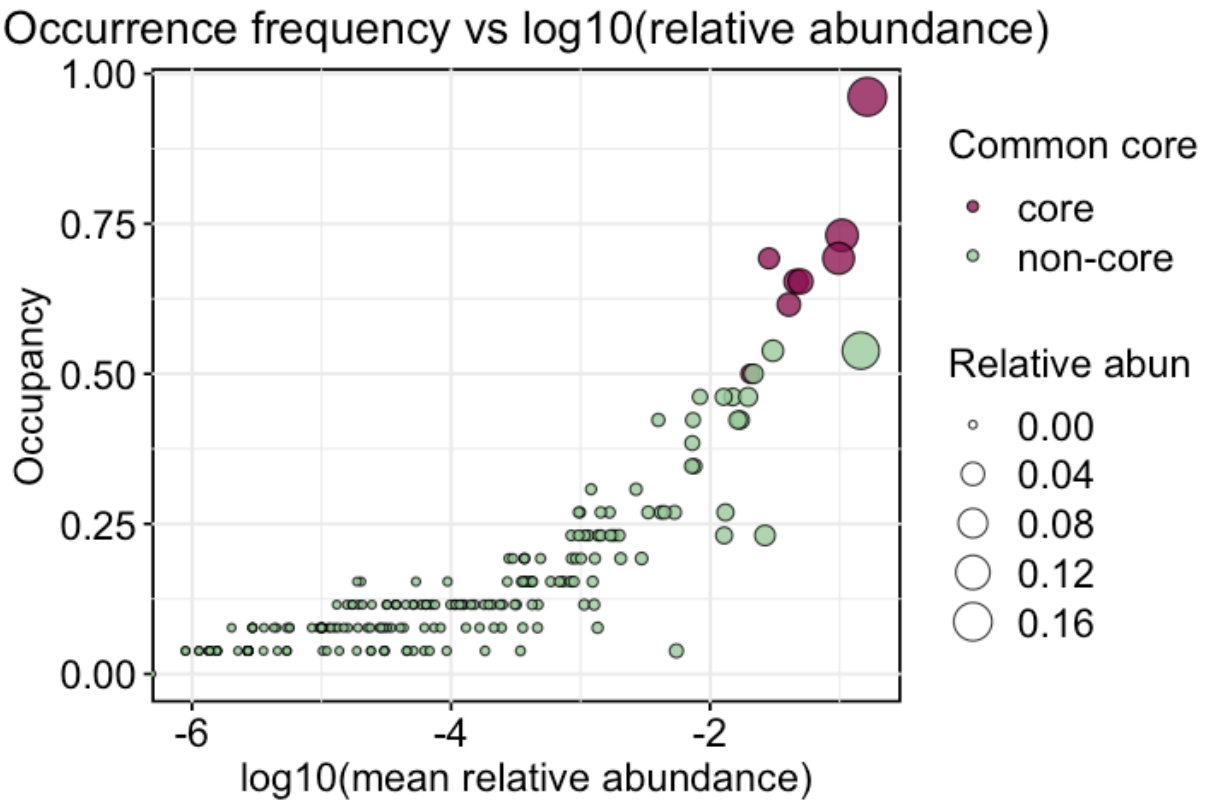
First, we plots BC similarity vs ranked otu.

```
options(repr.plot.width = 6, repr.plot.height = 4)
bc_rank_plot(otu, sample, mini_abun = 0, threshold = 0.02, sample_name
 = 'X.SampleID', sample_group = 'SampleType') +
labs(title = 'BC similarity vs Ranked OTU') +
scale_fill_manual(values = c("deeppink4", "darkseagreen3")) +
theme(plot.title = element_text(size = 16, color = "black", hjust = 0.
5, vjust = 1, lineheight = 0.2))
```



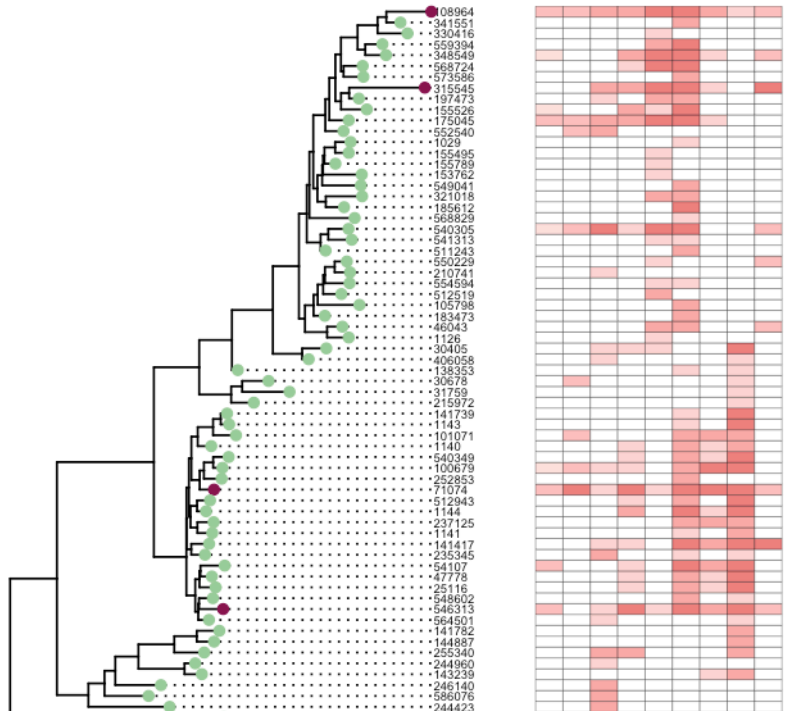
Then, we observe the relationship between occupancy and abundance with color denoting and core or non-core otu.

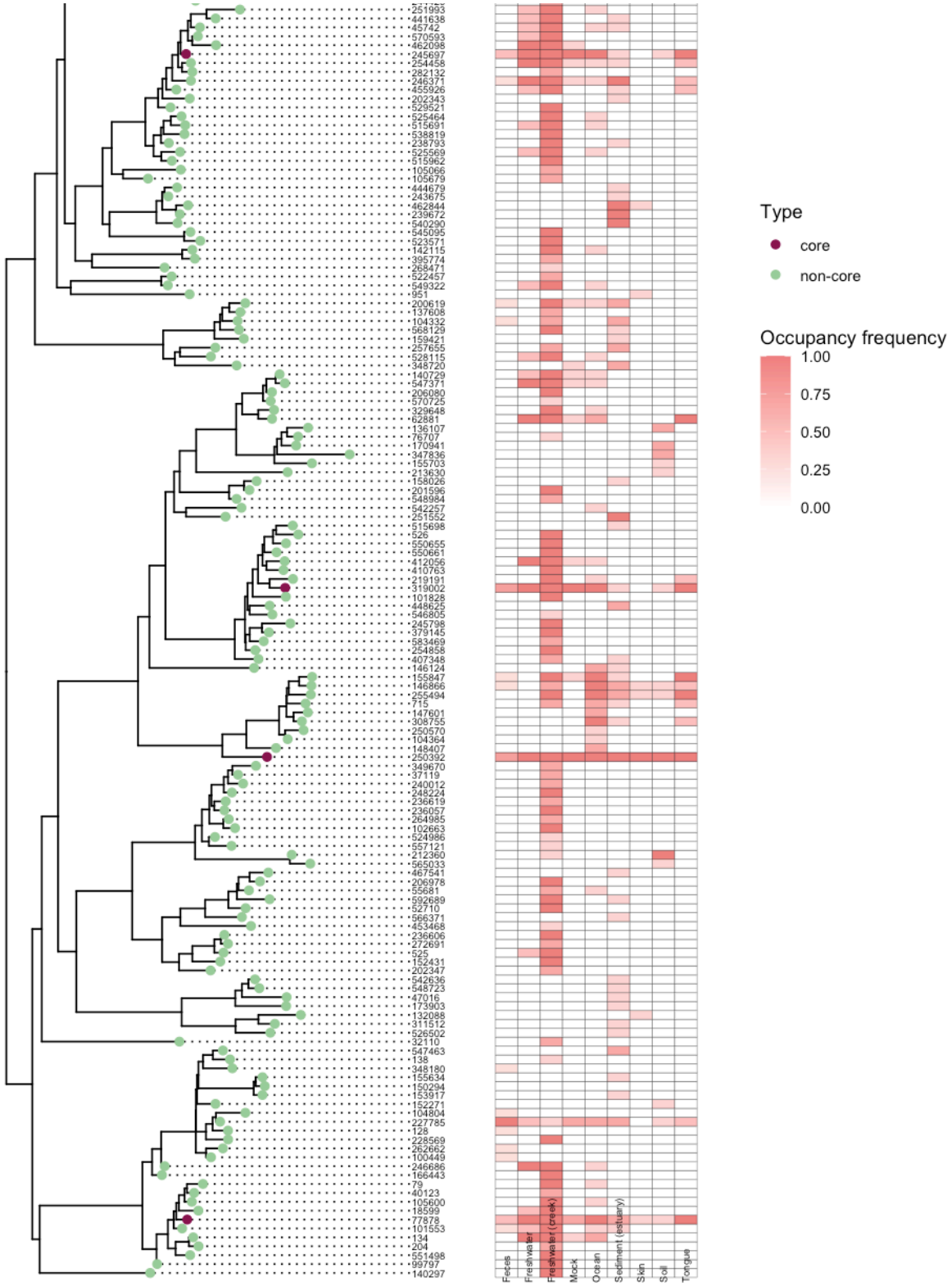
```
options(repr.plot.width = 6, repr.plot.height = 4)
abun_occ_plot(otu, sample, mini_abun = 0, threshold = 0.02, sample_nam
e = 'X.SampleID', sample_group = 'SampleType')+
labs(title = 'Occurrence frequency vs log10(relative abundance)') +
scale_fill_manual(values = c("deeppink4", "darkseagreen3")) +
theme(plot.title = element_text(size = 16, color = "black", hjust = 0.
5, vjust = 1, lineheight = 0.2))
```



Then, we plot the phylogenetic tree with a heatmap denoting the average occurrence frequency of each taxa with color denoting and core or non-core otu.

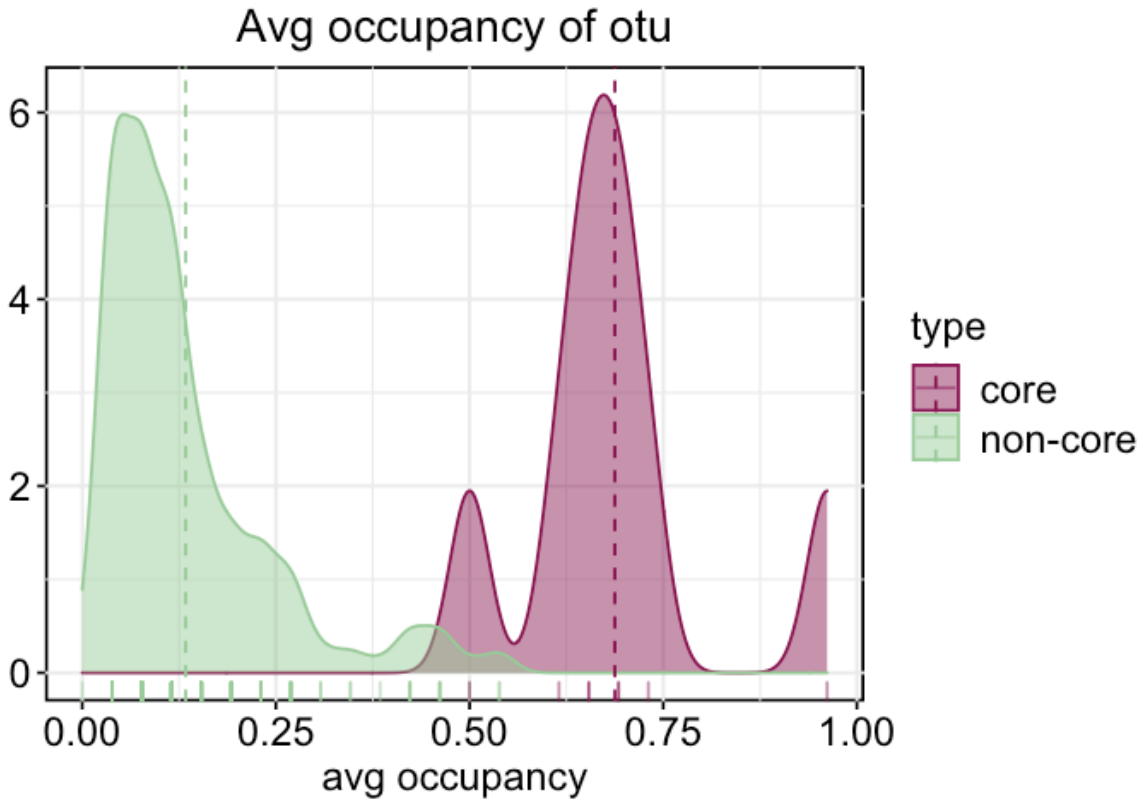
```
options(repr.plot.width = 8, repr.plot.height = 15)
tree_abun_plot(otu, sample, tree, mini_abun = 0, threshold = 0.02, sam
ple_name = 'X.SampleID', sample_group = 'SampleType', offset = 0.1, wi
dth = 0.5, core_col = 'deeppink4', noncore_col = 'darkseagreen3', low_
col = 'white', high_col = 'lightcoral', hlab = 2, ttip = 2, tlab = 2)
```

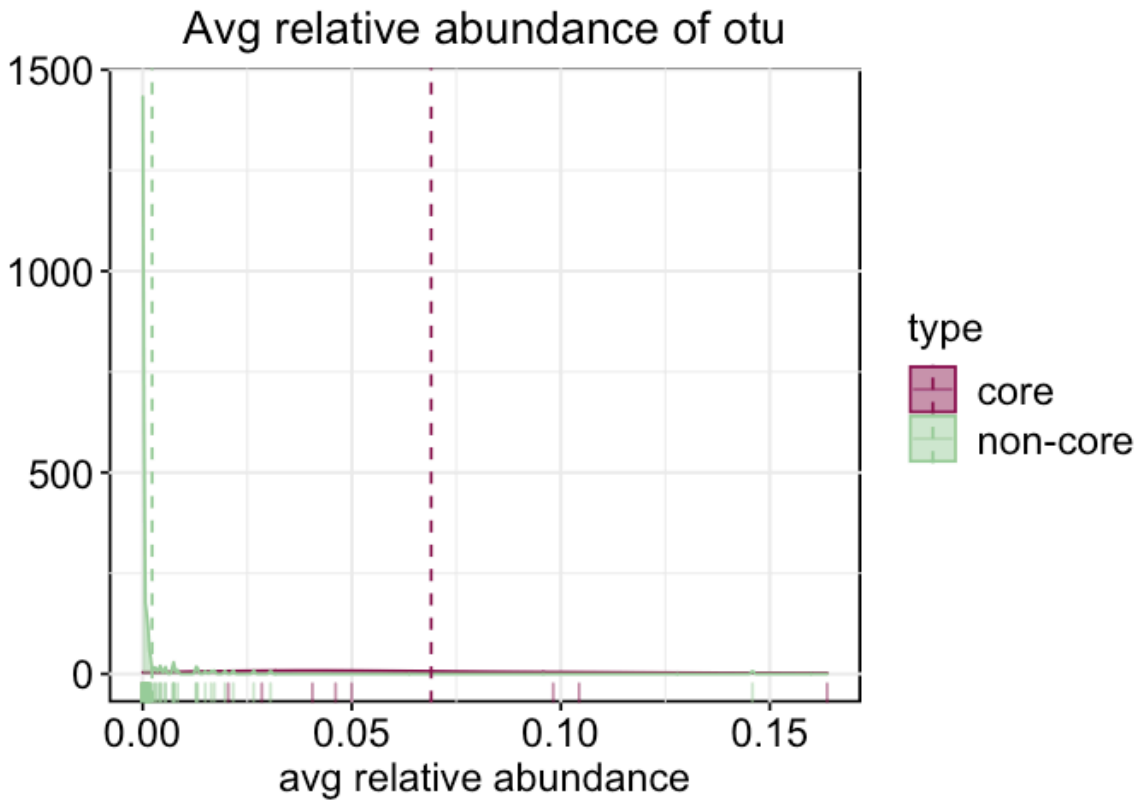




Then, we show the average occupancy and average relative abundance between core and non-core otu.

```
options(repr.plot.width = 6, repr.plot.height = 4)
p <- preve_abun_dis_plot(otu, sample, mini_abun = 0, threshold = 0.02,
 sample_name = 'X.SampleID', sample_group = 'SampleType', core_col = "dee
p[[1]] +
scale_fill_manual(values = c("deeppink4", "darkseagreen3")) +
theme(plot.title = element_text(size = 16, color = "black", hjust = 0.
5, vjust = 1, lineheight = 0.2))
p[[2]] +
scale_fill_manual(values = c("deeppink4", "darkseagreen3")) +
theme(plot.title = element_text(size = 16, color = "black", hjust = 0.
5, vjust = 1, lineheight = 0.2))
```

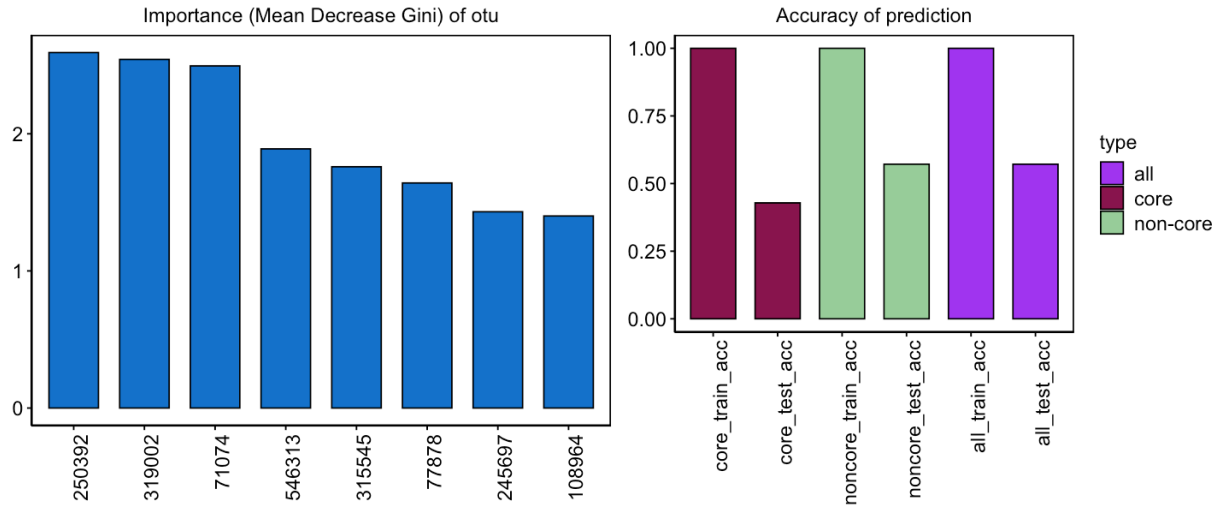




Finally, we use random forest to identify the importance of otu (core otu, non-core otu and all otu) on predicting/classifying the sample_group and display the accuracy.

```
options(repr.plot.width = 12, repr.plot.height = 5)
random_forest_plot(otu, sample, mini_abun = 0, threshold = 0.02, sampl
e_name = 'X.SampleID', sample_group = 'SampleType')
```

```
[1] "importance_rf"
       MeanDecreaseGini     id
250392         2.592395 250392
319002         2.542659 319002
71074          2.495226  71074
546313         1.889925 546313
315545         1.759458 315545
77878          1.641290  77878
245697         1.431425 245697
108964         1.400817 108964
[1] "accuracy"
      type train_test       acc                id
1     core      train 1.0000000    core_train_acc
2     core       test 0.4285714     core_test_acc
3 non-core      train 1.0000000 noncore_train_acc
4 non-core       test 0.5714286  noncore_test_acc
5      all      train 1.0000000     all_train_acc
6      all       test 0.5714286      all_test_acc
```



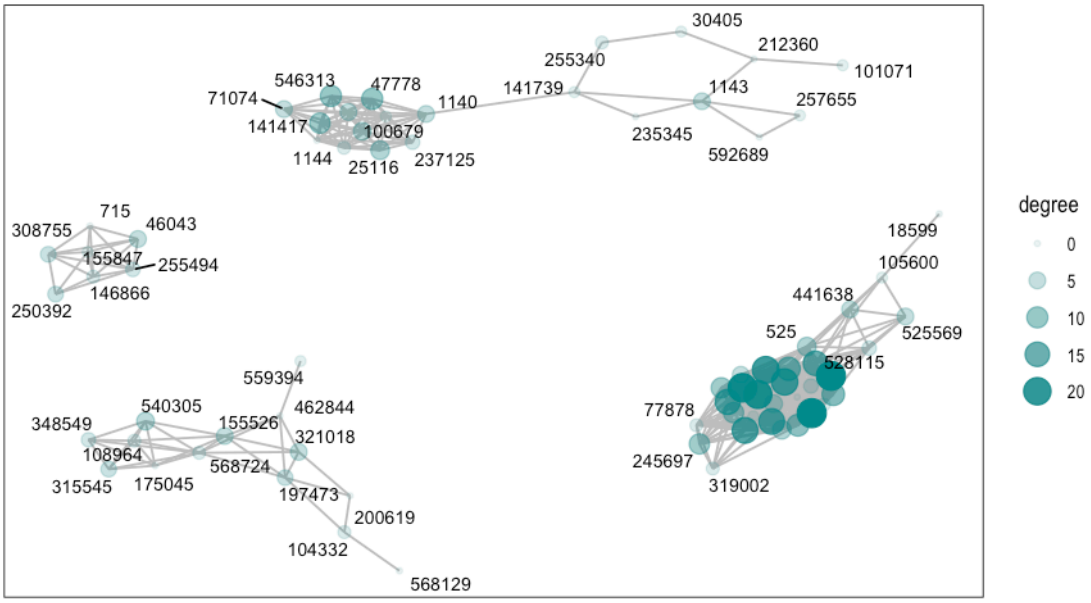
5.3 Network analysis

We construct the co-occurrence networks with 3 optional compositional association methods on the filtered data (low occurrence frequency otu should be removed): propr, sparcc and cclasso, based on certain permutation number, FDR and association threshold.

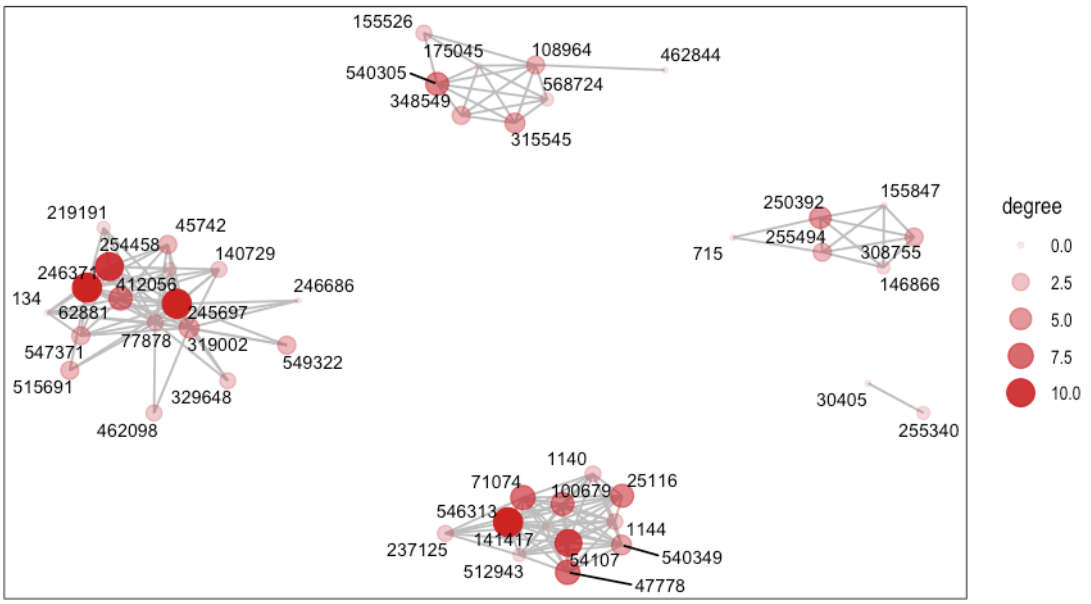
In the example, we only construct the networks by propr and sparcc. Whey using the defined thresholds (FDR = 0.1, cor = 0.6), cclasso identifies no association.

```
options(repr.plot.width = 8, repr.plot.height = 10)
networks_plot(otu, tax, sample, pre_threshold = 0.15, fdr_threshold =
0.1, cor_threshold = 0.6, permutation = 100, propr = TRUE, sparcc = TR
UE, cclasso = FALSE, propr_col = 'darkcyan', sparcc_col = 'firebrick3'
, cclasso_col = 'goldenrod3')
```

Propr

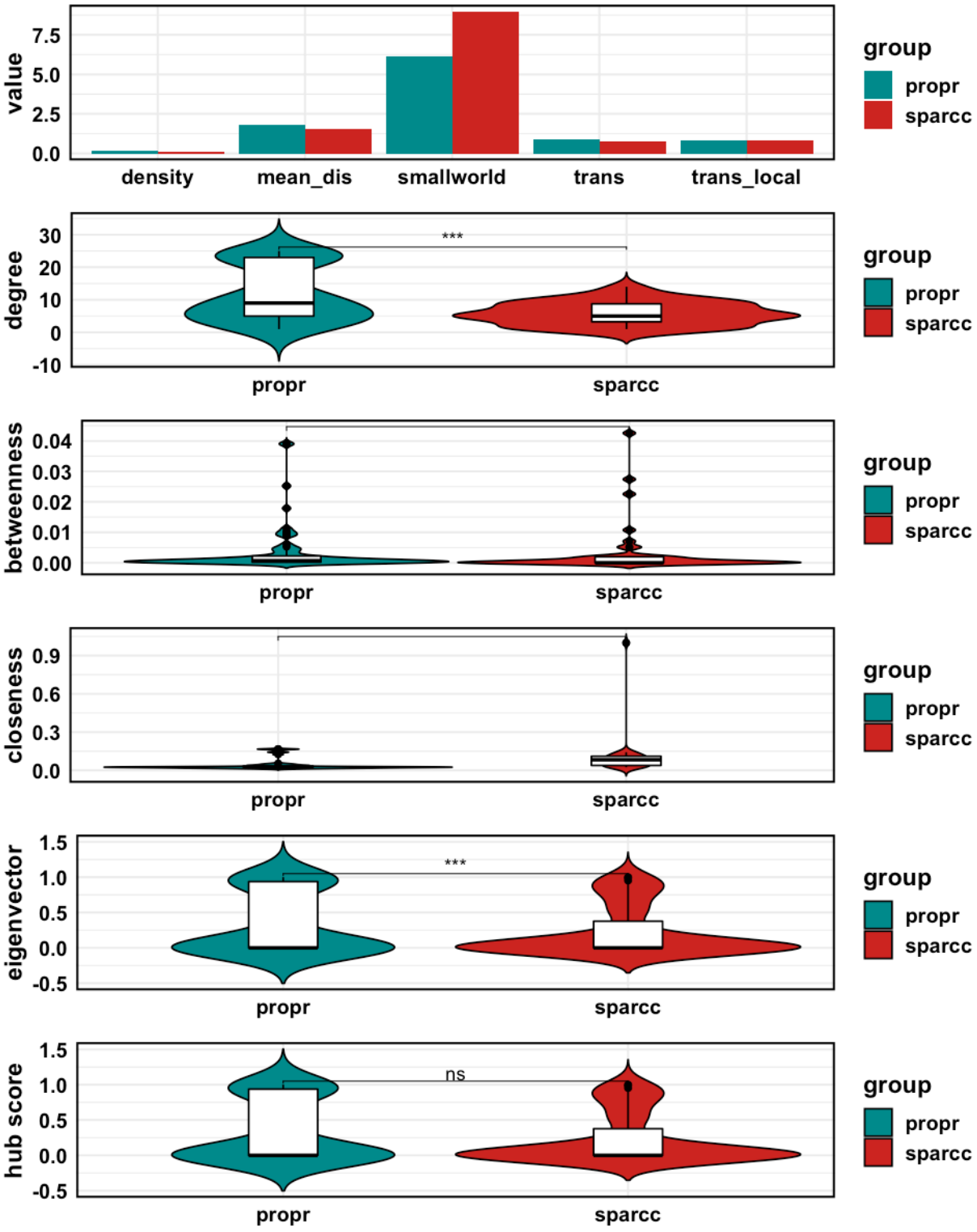


Sparcc



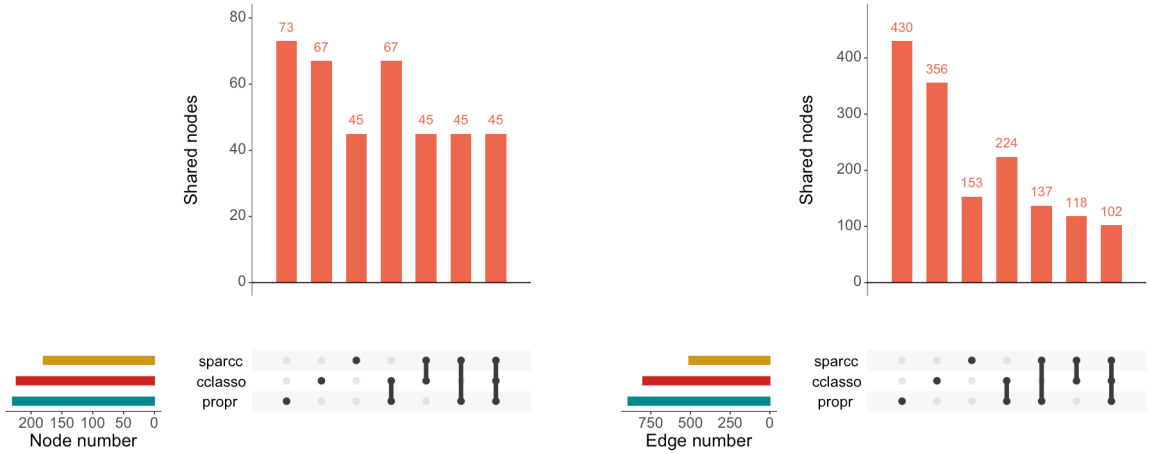
Then, we calculate the node-level and network-level properties of these two networks and perform statistical testing.

```
options(repr.plot.width = 8, repr.plot.height = 10)
networks_properties_plot(otu, tax, sample, pre_threshold = 0.15, fdr_t
hreshold = 0.1, cor_threshold = 0.6, permutation = 100, propr = TRUE,
sparcc = TRUE, cclasso = FALSE, propr_col = 'darkcyan', sparcc_col = '
firebrick3', cclasso_col = 'goldenrod3')
```



Finally, we compare the common/shared nodes and edges between these two networks.

```
options(repr.plot.width = 16, repr.plot.height = 6)
networks_shared_ne_plot(otu, tax, sample, pre_threshold = 0.15, fdr_th
reshold = 0.1, cor_threshold = 0.6, permutation = 100, propr = TRUE, s
parcc = TRUE, cclasso = TRUE, propr_col = 'darkcyan', sparcc_col = 'fi
rebrick3', cclasso_col = 'goldenrod3')
```

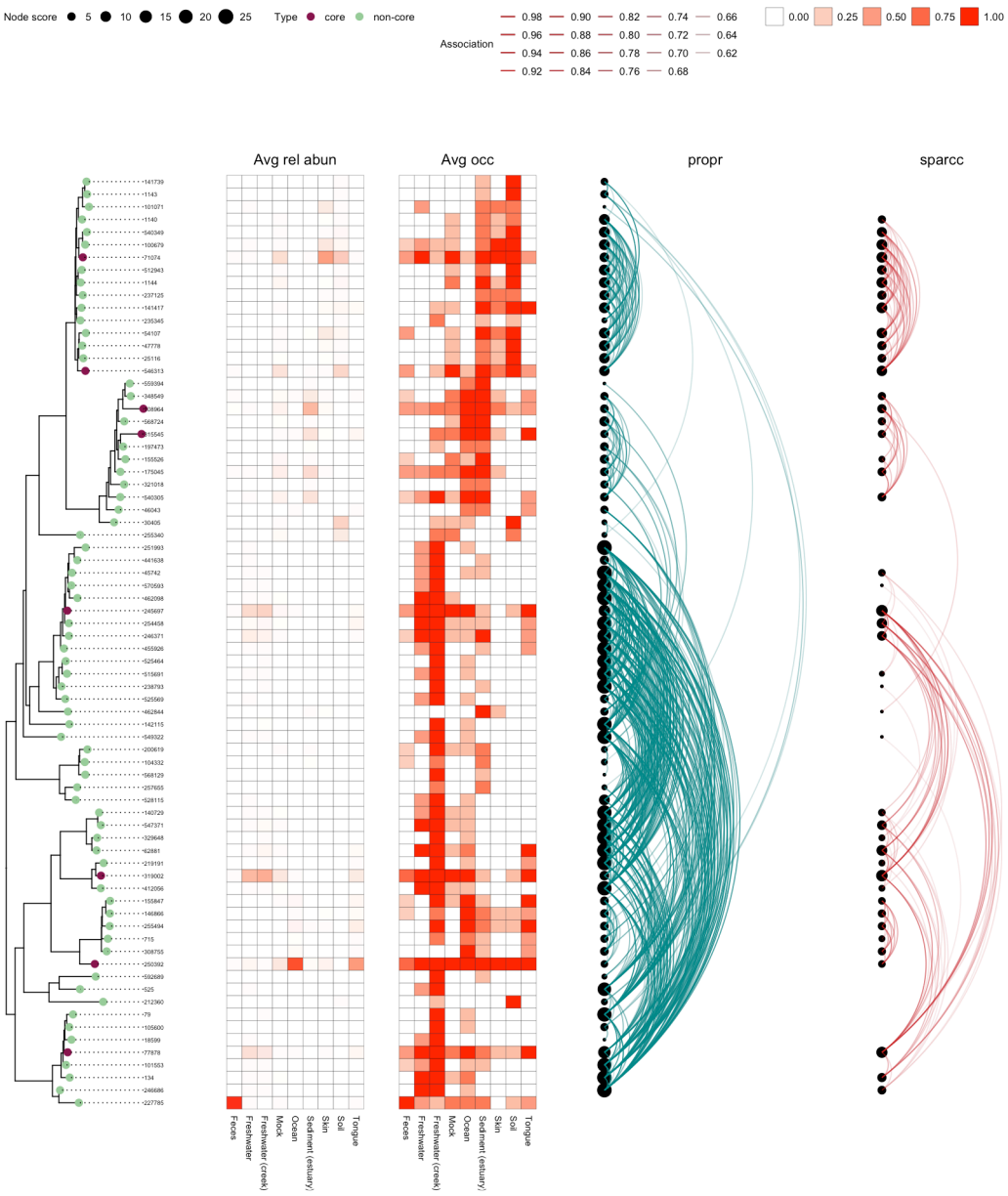


5.4 Combined plot

The function phnetworks implements a user friendly wrapper for visualization in abundance-occupancy analysis and network analysis.

1. Plot phylogenetic tree showing the evolutionary relationship of all taxa.
2. Plot heatmaps showing the average occurrence frequency and average relative abundance of each taxa among all sample groups.
3. Plot the networks constructed by 3 optional compositional association methods: propr, sparcc and cclasso, which shows the co-occurrence role of each taxa.

```
options(repr.plot.width = 16, repr.plot.height = 16)
phnetworks(otu, tax, sample, tree, propr = TRUE, sparcc = TRUE, cclass
o = FALSE, pre_threshold=0.15, fdr_threshold=0.1, cor_threshold=0.6, p
ermutation=10, nscore = 'degree', hcol = 'red', propr_col = 'darkcyan'
, sparcc_col = 'firebrick3', cclasso_col = 'goldenrod3', ttip = 3, tla
b = 2, hlab = 4, alab = 5, llab = 10, offset = 0.3, width = 1, mini_ab
un = 0, threshold = 0.02, sample_name = 'X.SampleID', sample_group = '
SampleType')
```



6. Package versions

```
sessionInfo()
```

```
R version 4.1.1 (2021-08-10)
Platform: x86_64-apple-darwin17.0 (64-bit)
Running under: macOS Big Sur 10.16

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/lib
Rblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/lib
Rlapack.dylib

locale:
```

```
[1] C/UTF-8/C/C/C/C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base


other attached packages:
 [1] propr_4.3.0         UpSetR_1.4.0        reshape2_1.4.4
 [4] compositions_2.0-4  ggraph_2.0.5        qgraph_1.9
 [7] igraph_1.3.0        tidygraph_1.2.0     caret_6.0-89
[10] randomForest_4.6-14 tidyr_1.2.0         dplyr_1.0.8
[13] vegan_2.5-7         lattice_0.20-45     permute_0.9-7
[16] pheatmap_1.0.12     ggpubr_0.4.0        ggtree_3.0.4
[19] gridExtra_2.3       microbiome_1.14.0   ggplot2_3.3.5
[22] phyloseq_1.36.0     coreMicrobiome_1.0  devtools_2.4.3
[25] usethis_2.0.1

loaded via a namespace (and not attached):
  [1] utf8_1.2.2              tidyselect_1.1.2        htmlwidgets_1.5.4
  [4] grid_4.1.1              Rtsne_0.15              pROC_1.18.0
  [7] munsell_0.5.0           codetools_0.2-18        pbdZMQ_0.3-5
 [10] future_1.22.1           withr_2.5.0             colorspace_2.0-3
 [13] Biobase_2.52.0          knitr_1.38              uuid_0.1-4
 [16] rstudioapi_0.13         stats4_4.1.1            robustbase_0.93-9
 [19] bayesm_3.1-4            ggsignif_0.6.3          listenv_0.8.0
 [22] labeling_0.4.2          repr_1.1.3.9000         GenomeInfoDbData_1.2
 [25] mnormt_2.0.2            polyclip_1.10-0         farver_2.1.0
 [28] rhdf5_2.36.0            rprojroot_2.0.3         parallelly_1.28.1
 [31] vctrs_0.4.1             treeio_1.16.2           generics_0.1.2
 [34] ipred_0.9-12            xfun_0.30               R6_2.5.1
 [37] GenomeInfoDb_1.28.4     graphlayouts_0.8.0      VGAM_1.1-6
 [40] bitops_1.0-7            rhdf5filters_1.4.0      cachem_1.0.6
 [43] gridGraphics_0.5-1      assertthat_0.2.1        scales_1.1.1
 [46] nnet_7.3-16             gtable_0.3.0            globals_0.14.0
 [49] processx_3.5.3          timeDate_3043.102       rlang_1.0.2
 [52] splines_4.1.1           rstatix_0.7.0           lazyeval_0.2.2
 [55] ModelMetrics_1.2.2.2    broom_0.7.12            checkmate_2.0.0
 [58] abind_1.4-5             backports_1.4.1         Hmisc_4.6-0
 [61] tensorA_0.36.2          tools_4.1.1             lava_1.6.10
 [64] psych_2.1.9             lavaan_0.6-10           ggplotify_0.1.0
 [67] ellipsis_0.3.2          biomformat_1.20.0       RColorBrewer_1.1-3
 [70] proxy_0.4-26            BiocGenerics_0.38.0     sessioninfo_1.1.1
 [73] Rcpp_1.0.8.3            plyr_1.8.7              base64enc_0.1-3
 [76] zlibbioc_1.38.0         purrr_0.3.4             RCurl_1.98-1.6
 [79] ps_1.6.0                prettyunits_1.1.1       rpart_4.1-15
 [82] pbapply_1.5-0           viridis_0.6.2           cowplot_1.1.1
 [85] S4Vectors_0.30.2        ggrepel_0.9.1           cluster_2.1.2
 [88] fs_1.5.2                magrittr_2.0.3          data.table_1.14.2
```

```
 [91] tmvnsim_1.0-2          pkgload_1.2.4          patchwork_1.1.1
 [94] evaluate_0.15          jpeg_0.1-9             IRanges_2.26.0
 [97] testthat_3.1.3         compiler_4.1.1         tibble_3.1.6
[100] crayon_1.5.1           htmltools_0.5.2        ggfun_0.0.4

[103] mgcv_1.8-36            corpcor_1.6.10         Formula_1.2-4

[106] aplot_0.1.1            lubridate_1.7.10       DBI_1.1.1

[109] tweenr_1.0.2           MASS_7.3-54            boot_1.3-28

[112] Matrix_1.3-4           ade4_1.7-18            car_3.0-12

[115] brio_1.1.3             cli_3.2.0              parallel_4.1.1

[118] gower_0.2.2            pkgconfig_2.0.3        foreign_0.8-81

[121] IRdisplay_1.0.0.9000   recipes_0.1.17         foreach_1.5.2

[124] pbivnorm_0.6.0         multtest_2.48.0        XVector_0.32.0

[127] prodlim_2019.11.13     yulab.utils_0.0.4      stringr_1.4.0

[130] callr_3.7.0            digest_0.6.29          Biostrings_2.60.2

[133] tidytree_0.3.6         htmlTable_2.4.0        curl_4.3.2

[136] gtools_3.9.2           lifecycle_1.0.1        nlme_3.1-153

[139] glasso_1.11            jsonlite_1.8.0         Rhdf5lib_1.14.2

[142] carData_3.0-5          desc_1.4.1             viridisLite_0.4.0

[145] fansi_1.0.3            pillar_1.7.0           DEoptimR_1.0-10

[148] fastmap_1.1.0          pkgbuild_1.2.0         survival_3.2-13

[151] glue_1.6.2             remotes_2.4.2          fdrtool_1.2.17

[154] png_0.1-7              iterators_1.0.14       ggforce_0.3.3

[157] class_7.3-19           stringi_1.7.6          latticeExtra_0.6-2
9
[160] memoise_2.0.0          IRkernel_1.2.0.9000    e1071_1.7-9

[163] future.apply_1.8.1     ape_5.6-2
```

References

McMurdie, P. J., and S. Holmes. 2013. "Phyloseq: An R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data." Journal Article. PLoS One 8 (4): e61217. https://doi.org/10.1371/journal.pone.0061217.

Shade, A., Stopnisek, N. 2019. "Abundance-occupancy distributions to prioritize plant core microbiome membership." Journal Article. Current opinion in microbiology 49: 50-58. https://doi.org/10.1016/j.mib.2019.09.008