

1 ggtreeExtra: A universal package to visualize compact circular layers of phylogenetic tree

Shuangbin Xu, Zehan Dai, Pingfan Guo, Xiaocong Fu, Shanshan Liu, Lang Zhou, Wenli Tang, Tingze Feng, Meijun Chen, Li Zhan and Guangchuang Yu*

*correspondence: guangchuangyu@gmail.com, gcyu1@smu.edu.cn

1.1 Examples of mapping and visualizing associated data on circular layout tree.

It can help to find patterns and generate new hypotheses by integrating and visualizing associated data to the phylogenetic tree. The associated data type of phylogenetic tree can be roughly divided into continuous data and categorical data (discrete data). The continuous data represents measurements, they can be measured but not be counted, such as the height, weight, abundance of species, gene expression and the number of target genes etc. The categorical data represents characteristics, they can not be measured but they can be counted, such as Endemic region information of virus, Taxonomy information of species, type of target gene and sampling location information etc. Certainly, categorical data can also take on numerical values (for example, 1 for target gene A, 2 for target gene B). The associated data are also often multi-dimensional. To display them on the outer of phylogenetic tree more compactly, we developed *ggtreeExtra* to annotate multi-dimensional data to the outer ring of circular phylogenetic tree. It is a universal tool, based on the grammar of graphics(Wilkinson 2012). So user can easily map the variables (abundance of species, length of genome, sampling location) of associated data to aesthetic attributes (size, color, shape) of outer geometric objects (bar, point, boxplot) of circular phylogenetic tree using *ggtreeExtra*. Here, we present several examples to elucidate how to map and display the associated data on the outer rings of circular phylogenetic trees using *ggtreeExtra*. More examples can be found on the chapter10 of online book¹.

1.1.1 Displaying the continuous data to circular phylogenetic tree.

This example reproduce Fig.2 of (Morgan, Segata, and Huttenhower 2013). The data is provided by GraPhlAn (Asnicar et al. 2015), which contained the relative abundance of bacteria (continuous data) at different body sites (categorical data). We use heat map to display the abundance of tip species (the abundance of species was mapped to the transparency of heat map, the different body sites was mapped to the color of heat map) (Fig. S1.C), and the most abundance of species (continuous data) at specific site also was visualized with bar plot (the abundance of species was mapped to the length of bar plot, the different body sites was mapped to the color of bar plot) (Fig. S1.D). And the outer graphic layers can be aligned automatically according to the tip labels of phylogenetic tree. The tree in *geom_fruit* can be fully annotated with multiple layers (high light (Fig. S1.A), clade labels (Fig. S1.B)). The layers can be added step-by-step with +. More step-by-step instructions are also available at the vignette².

```
library(ggtreeExtra)
library(ggtree)
library(treeio)
library(tidytree)
library(ggstar)
library(ggplot2)
library(ggnewscale)
library(patchwork)

tree <- read.tree("../data/HMP_tree/hmptree.nwk")
# the abundance and types of microbes
dat1 <- read.csv("../data/HMP_tree/tippoint_attr.csv")
# the abundance of microbes at different body sites.
dat2 <- read.csv("../data/HMP_tree/ringheatmap_attr.csv")
# the abundance of microbes at the body sites of greatest prevalence.
```

¹<http://yulab-smu.top/treedata-book>

²<http://bioconductor.org/packages/release/bioc/vignettes/ggtreeExtra/inst/doc/ggtreeExtra.html>

```

dat3 <- read.csv("../data/HMP_tree/barplot_attr.csv")

# adjust the order
dat2$Sites <- factor(dat2$Sites, levels=c("Stool (prevalence)", "Cheek (prevalence)",
                                             "Plaque (prevalence)", "Tongue (prevalence)",
                                             "Nose (prevalence)", "Vagina (prevalence)",
                                             "Skin (prevalence)"))
dat3$Sites <- factor(dat3$Sites, levels=c("Stool (prevalence)", "Cheek (prevalence)",
                                             "Plaque (prevalence)", "Tongue (prevalence)",
                                             "Nose (prevalence)", "Vagina (prevalence)",
                                             "Skin (prevalence)"))

# extract the clade label information. Because some nodes of tree are annotated to genera,
# which can be displayed with high light using ggtree.
nodeids <- nodeid(tree, tree$node.label[nchar(tree$node.label)>4])
nodedf <- data.frame(node=nodeids)
nodelab <- gsub("[\\.\.0-9]", "", tree$node.label[nchar(tree$node.label)>4])
# The layers of clade and hightlight
poslist <- c(1.6, 1.4, 1.6, 0.8, 0.1, 0.25, 1.6, 1.6, 1.2, 0.4,
            1.2, 1.8, 0.3, 0.8, 0.4, 0.3, 0.4, 0.4, 0.4, 0.6,
            0.3, 0.4, 0.3)
labdf <- data.frame(node=nodeids, label=nodelab, pos=poslist)
# The circular layout tree.
p <- ggtree(
    tree,
    layout="fan",
    size=0.15,
    open.angle=5
) +
  geom_hilight(
    data=nodedf,
    mapping=aes(node=node),
    extendto=6.8,
    alpha=0.3,
    fill="grey",
    color="grey50",
    size=0.05
) +
  geom_cladelab(
    data=labdf,
    mapping=aes(node=node, label=label, offset.text=pos),
    barsize=NA,
    fontsize=0.7,
    angle="auto",
    hjust=0.5,
    horizontal=FALSE,
    fontface="italic"
) +
  ggtile("A")

p1 <- p %<+% dat1 +
  geom_fruit(
    geom=geom_star,
    mapping=aes(fill=Phylum, starshape=Type, size=Size),
    position="identity",
    starstroke=0.05
) +
  scale_fill_manual(
    values=c("#FFC125", "#87CEFA", "#7B68EE", "#808080", "#800080",

```

```

        "#9ACD32", "#D15FEE", "#FFC0CB", "#EE6A50", "#8DEEEE",
        "#006400", "#800000", "#B0171F", "#191970"),
  guide=guide_legend(keywidth = 0.5, keyheight = 0.5, order=1,
  override.aes=list(starshape=15)),
  na.translate=FALSE
) +
scale_starshape_manual(
  values=c(15, 1),
  guide=guide_legend(keywidth = 0.5, keyheight = 0.5, order=2),
  na.translate=FALSE
) +
scale_size_continuous(
  range = c(0.5, 1.5),
  guide = guide_legend(keywidth = 0.5, keyheight = 0.5, order=3,
  override.aes=list(starshape=15))
) +
new_scale_fill() +
ggtitle("B") +
theme(legend.position="none")

p2 <- p1 +
  geom_fruit(
    data=dat2,
    geom=geom_tile,
    mapping=aes(y=ID, x=Sites, alpha=Abundance, fill=Sites),
    color = "grey50",
    offset = 0.04,
    size = 0.02
) +
scale_alpha_continuous(
  range=c(0, 1),
  guide=guide_legend(keywidth = 0.3, keyheight = 0.3, order=5)
) +
scale_fill_manual(
  values=c("#0000FF", "#FFA500", "#FF0000", "#800000",
          "#006400", "#800080", "#696969"),
  guide=guide_legend(keywidth = 0.3, keyheight = 0.3, order=4)
) +
ggtitle("C") +
theme(legend.position="none")

p3 <- p2 +
  geom_fruit(
    data=dat3,
    geom=geom_bar,
    mapping=aes(y=ID, x=HigherAbundance, fill=Sites),
    pwidth=0.38,
    orientation="y",
    stat="identity"
) +
geom_treescale(fontsize=1.2, linesize=0.3, x=4.9, y=0.1) +
ggtitle("D") +
theme(legend.position="none")

p4 <- (p + p1 + plot_layout(width=c(3.4,4)))/ (p2 + p3 + plot_layout(width=c(3.4,4))) +
  plot_layout(heights=c(3,4))
p4

```

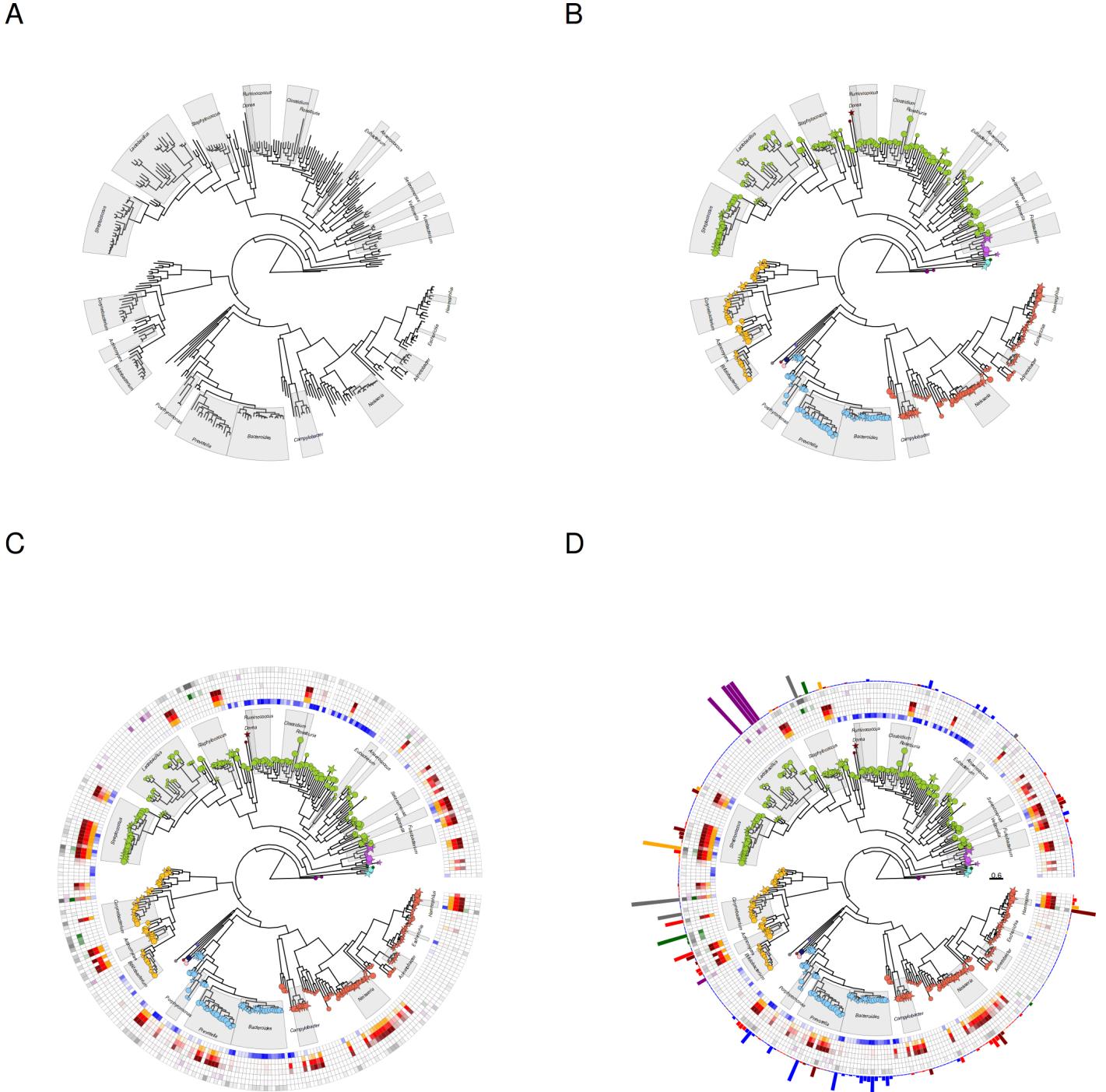


Fig. S1: The abundance of microbes at different sites of human. The shape of tip labels indicated the commensal microbes or potential pathogens. The transparency of heat map indicates the abundance of microbes, and colors of heat map indicate the different sites of human. The bar plot indicates the relative abundance at body site of the most abundance. Fig.S1.A contains the tree layer, high light layer and clade label layer; Fig.S1.B is a tip point layer added on the basis of Fig.S1.A; Fig.S1.C is a heatmap layer added on the basis of Fig.S1.B; Fig.S1.D is a bar chart layer added on the basis of Fig.S1.C. This example shows the abilities of annotate phylogenetic tree through layer overlay using *ggtree* and *ggtreeExtra*.

1.1.2 Visualizing discrete datasets to circular phylogenetic tree.

The categorical data (discrete data) can also be displayed using heat map, scatter plot etc. We reproduce Figure 3.3 of (Smith and Wrighton 2019) to show how to map discrete datasets. When the external datasets have a column of `node` or first column of tip labels (the element of the column must be unique). We can integrated the external data to tree data by `%<+%` operator of *ggtree* (Yu et al. 2018). The `geom_fruit` can extract the tree data integrated automatically,

and the attributes of datasets can be mapped using related attributes of geometry layers. The *y* of *aes* can be ignored, it will be added automatically. In this example, the external ring heat maps represent the different types in corresponding categories (the type of Ecosystem was mapped to the color of innermost ring heat map, the type of sequencing was mapped to the color of middle ring heat map, the type of sample treatment was mapped to the color of outermost ring heat map) (Fig. S2). And the source data is available at this repository³. This example also show the external data can be displayed and aligned to circular phylogenetic tree automatically.

```

library(ggtreeExtra)
library(ggtree)
library(treeio)
library(ggplot2)
library(ggnewscale)
tree <- read.raxml("../data/Methanotroph/Methanotroph_rpS3_Modified_Alignment_RAxML")
# Root the tree to the archaea sequences
tree@phylo <- root(tree@phylo, node=1402, edgelabel=TRUE)
df <- read.csv("../data/Methanotroph/metadata.csv")
# reset the levels of columns to reproduce the order of original figure.
df$Specific.Ecosystem <- factor(df$Specific.Ecosystem,
                                 levels=c("Agriculture", "Alkaline/Hypersaline",
                                         "Contaminated/Wastewater", "Endosymbiont",
                                         "Freshwater", "Forest", "Geothermal",
                                         "Marine", "Natural Seep", "Peat",
                                         "Permafrost", "Wetland", "Unknown"))
df$MetaType <- factor(df$MetaType,
                       levels=c("Metatranscriptome", "Metagenome",
                               "Single-amplified genome", "Fosmid", NA))
df$Treatment <- factor(df$Treatment, levels=c("Native", "Enrichment", "Isolate", "Unknown"))
p <- ggtree(tree, layout="fan", open.angle=30)
p <- rotate_tree(p, 90)
p <- p + geom_treescaling(x=0.2, y=727*6/11, width=1, offset=20) +
  geom_nodepoint(aes(subset=as.numeric(bootstrap)>=70),
                 shape=21, fill="grey", size=1)

# we can use %<+% to integrate the external datasets to tree structure.
# and the y can not be specified.
p <- p %<+% df +
  geom_fruit(
    geom=geom_tile,
    mapping=aes(fill=Specific.Ecosystem),
    offset=0.13,
    width=0.35,
    axis.params=list(axis="x", text="Ecosystem",
                     text.angle=0, hjust=0,
                     text.size=3, family="Times",
                     fontface="bold"))
) +
  scale_fill_manual(
    values=c("green3", "turquoise", "maroon", "orchid",
            "deepskyblue", "forestgreen", "salmon", "cadetblue3",
            "slategray4", "yellowgreen", "gray90", "chocolate2",
            "yellow"),
    guide=guide_legend(title="Ecosystem", keywidth=0.5, keyheight=0.5, order=4),
    na.translate=FALSE
) +
  new_scale_fill() +
  geom_fruit(
    geom=geom_tile,
    mapping=aes(fill=MetaType),

```

³https://github.com/TheWrightonLab/Methanotroph_rpS3Analyses_SmithWrighton2018

```

offset=0.13,
width=0.35,
axis.params=list(
  axis="x",
  text="Sequencing Type",
  text.angle=0,
  hjust=0,
  text.size=3,
  family="Times",
  fontface="bold"
)
) +
scale_fill_manual(
  values=c("red", "black", "dodgerblue", "gray50"),
  guide=guide_legend(title="Sequencing Type",
    keywidth=0.5, keyheight=0.5, order=3),
  na.translate=FALSE
) +
new_scale_fill()+
geom_fruit(
  geom=geom_tile,
  mapping=aes(fill=Treatment),
  offset=0.13,
  width=0.35,
  axis.params=list(
    axis="x",
    text="Sample Treatment",
    text.angle=0,
    hjust=0,
    text.size=3,
    family="Times",
    fontface="bold"
)
) +
scale_fill_manual(
  values=c("red","gray50", "black", "yellow"),
  guide=guide_legend(title="Sample Treatment", keywidth=0.5,
    keyheight=0.5, order=2),
  na.translate=FALSE
) +
theme(legend.background=element_rect(fill=NA), # the background of legend.
  legend.title=element_text(size=9, family="Times", face="bold"),
  legend.text=element_text(size=7, family="Times"), # the text size of legend.
  legend.spacing.y = unit(0.02, "cm"),
  legend.margin=margin(0.1, 0.9, 0.1,-0.9, unit="cm"), # t, r, b, l, cm
  legend.box.margin=margin(0.1, 0.9, 0.1, -0.9, unit="cm"),
  plot.margin = unit(c(-1.2, -1.2, -1.2, 0.1),"cm"))
# To build the external annotation of clades.
p <- p + geom_cladelabel(node=793, angle="auto", label="Gammaproteobacteria", horizontal=FALSE,
  offset=1.4, align=T, fontsize = 3, hjust=0.5, offset.text=0.12, barsize=1,
  family="Times", fontface="bold", color="gray30")+
  geom_cladelabel(node=791, angle="auto", label="Alphaproteobacteria", horizontal=FALSE,
  offset=1.4, align=T, fontsize = 3, hjust=0.5, offset.text=0.12, barsize=1,
  family="Times", fontface="bold", color="black")+
  geom_cladelabel(node=1384, label="Ca.Methylomirabilis", horizontal=TRUE,
  offset=1.4, align=T, fontsize = 3, angle="auto", barsize=1,
  family="Times", fontface="bold", color="gray30")+
  geom_cladelabel(node=1394, label="Methylacidiphilae", angle="auto", horizontal=TRUE,

```

```

offset=1.4, align=T, fontsize = 3, barsize=1,
family="Times", fontface="bold", color="black")+
geom_cladelabel(node=1440, label="ANME-1", angle="auto", horizontal=TRUE,
offset=1.4, align=T, fontsize = 3, barsize=1,
family="Times", fontface="bold", color="gray30")+
geom_cladelabel(node=1405, label="ANME-2", angle="auto", horizontal=TRUE,
offset=1.4, align=T, fontsize = 3, barsize=1,
family="Times", fontface="bold", color="black")

```

p

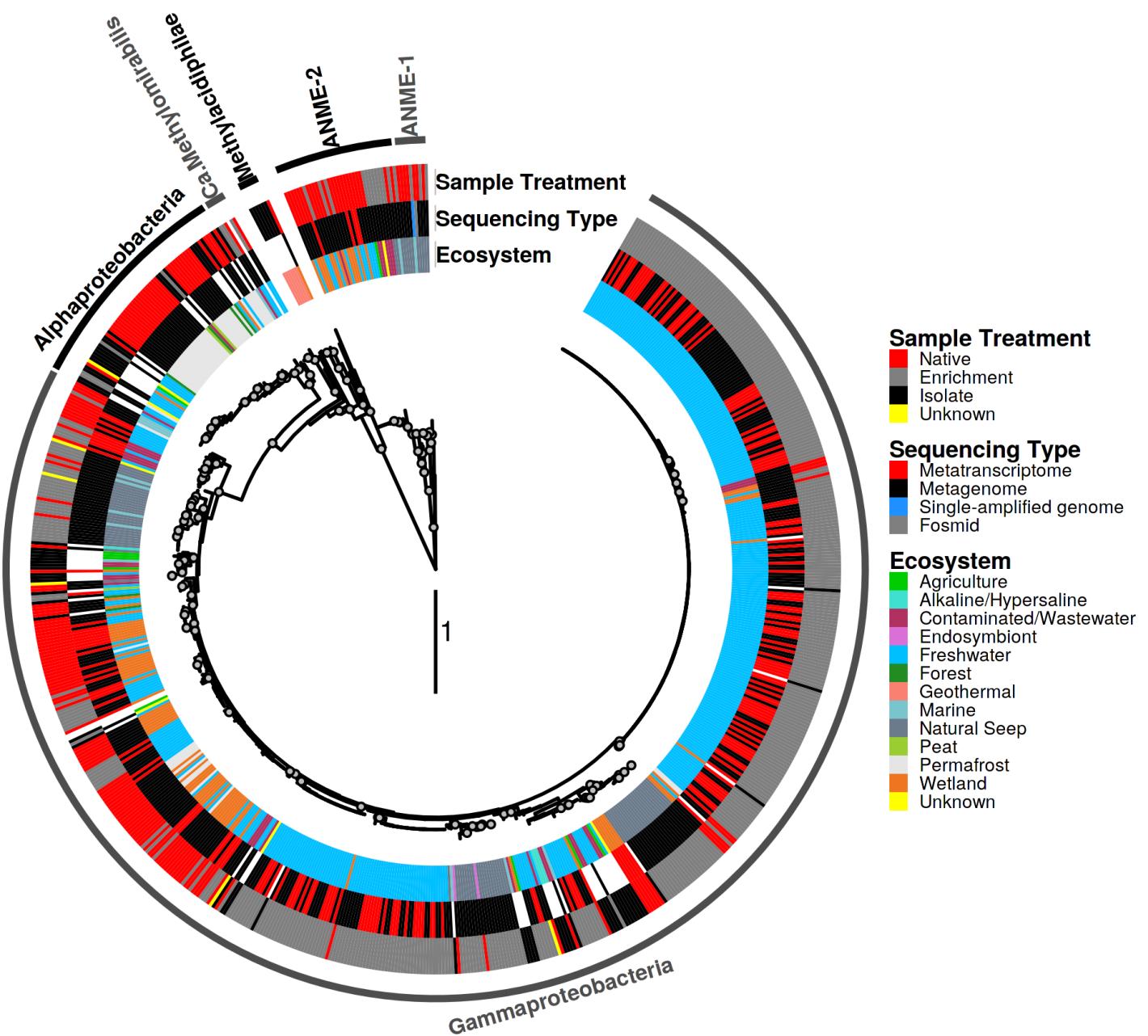


Fig. S2: Phylogeny of methanotroph ribosomal protein S39 (rpS3) genes from Figure 3.3 (Smith and Wrighton 2019). The external ring of circular tree is built with discrete datasets, which were added to tree structure in advance

1.1.3 Annotating multiple associated data to large-scale phylogenetic tree.

The number of tips of phylogenetic tree annotated by *ggtreeExtra* is strictly not limited, the large-scale phylogenetic tree also can be annotated by *ggtreeExtra*, since *ggtreeExtra* inherits the design concept of *ggtree* (Yu et al. 2017) and *ggplot2*

(Wickham 2016). And each outer graphic layer of circular phylogenetic tree is independent, the associated data can be visualized simultaneously at the different outer layer. Then the layers can be aligned and stacked on the outer rings of circular phylogenetic tree according to tree structure. These features allow ggtreeExtra to plot fully annotated large-scale phylogenetic tree with multi-dimensional data.

1.1.3.1 The first example in this section We reproduce the Fig 1 of (Segata et al. 2013), which contains a phylogenetic tree built with 3737 genomes. The discrete data is displayed in the outermost ring using scatter plot, which is provided by *ggstar* (Xu 2020) (The type of genomes was mapped to the color of point), the fraction of 400 proteins is visualized with bar chart (The number of proteins was mapped to the length of bar, the taxonomy of genomes was mapped to the color of bar) (Fig. S3).

```
library(ggtreeExtra)
library(ggtree)
library(treeio)
library(tidytree)
library(ggstar)
library(ggplot2)
library(ggnewscale)

tree <- read.tree("../data/PhyloPhlAn/ppal_tol.nwk")
# The attributes of tip point
da1 <- read.csv("../data/PhyloPhlAn/tipoint_attr.csv")
# The attributes of bar plot
da2 <- read.csv("../data/PhyloPhlAn/barplot_attr.csv")
# The attributes of triangle point
da3 <- read.csv("../data/PhyloPhlAn/ringpoint_attr.csv")

# to reproduce, reorder column Type
da3$Type <- factor(da3$Type, levels=c("Mislabelled", "Insertions",
                                         "Corrections", "Refinements"))

# extract the node label to annotate clade
nodelab <- tree$node.label[nchar(tree$node.label)>0]
nodeids <- nodeid(tree, nodelab)
# the position of high light extends to
extos <- c(rep(5.66, 12), 7.1, rep(5.66, 30))
nodedf <- data.frame(node=nodeids, extos=extos)

# the position of clade label
textex <- c(0.8, 0.8, 1.0, 0.6, 0, 0.8, 0.8, 0.2, -0.05, 0.5,
           0.1, 0.1, -0.1, 0.6, 0.4, 0, 1.2, 0.8, 1.0, 0.6,
           0.6, 1.0, 0.8, 0.2, -0.1, 1.2, 1, 1, 0.6, 0.6,
           0.4, 0.5, 1.2, 0.6, 1.4, 1.3, 0, -0.1, 0.8, 0.6,
           0.8, 1.4, 2.4)

cladelabels <- mapply(function(x, y, z){geom_cladelabel(node=x, label=y, barsize=NA, extend=0,
                                                       offset.text=z, fontsize=1.2, angle="auto",
                                                       hjust=0.5, horizontal=FALSE, fontface="italic")},
                       nodeids, nodelab, textex, SIMPLIFY=FALSE)

# To reproduce the color of original figures, we generated the same colors,
# user can custom their colors.
phylacolor <- c(Proteobacteria="#9ACD32", Firmicutes="#EE6A50",
                Actinobacteria="#87CEFA", Bacteroidetes="#FFC125",
                Euryarchaeota="#D15FEE", Tenericutes="#8DEEEE",
                Cyanobacteria="#800000", Spirochaetes="#006400",
                Chlamydiae="#800080", Crenarchaeota="#808080",
                Fusobacteria="#FFC0CB", Thermi="#B0171F", Other="Black",
                Chloroflexi="#191970", Thermotogae="#7B68EE",
```

```

Aquificae="#00CD00", Synergistetes="#8B4513",
Chlorobi="#BC8F8F", Verrucomicrobia="#303030",
Planctomycetes="#8E8E38", Acidobacteria="#CDCDC1")

tipcolors <- c("#9ACD32", "#EE6A50", "#87CEFA", "#FFC125", "#D15FEE",
               "#8DEEEE", "#800000", "#006400", "#800080", "#808080",
               "#FFC0CB", "#B0171F", "#191970", "#7B68EE", "#00CD00",
               "#8B4513", "#BC8F8F", "#303030", "#8E8E38", "#CDCDC1")
names(tipcolors) <- c("Proteobacteria", "Firmicutes", "Actinobacteria",
                      "Bacteroidetes", "Euryarchaeota", "Tenericutes",
                      "Cyanobacteria", "Spirochaetes", "Chlamydiae",
                      "Crenarchaeota", "Fusobacteria", "Thermi",
                      "Chloroflexi", "Thermotogae", "Aquificae",
                      "Synergistetes", "Chlorobi", "Verrucomicrobia",
                      "Planctomycetes", "Acidobacteria")

# We built a circular layout tree with open a small (6) angle.
p <- ggtree(
  tree,
  layout="fan",
  open.angle=6,
  size=0.1
) +
  geom_hilight(
    data=nodedf,
    mapping=aes(node=node, extendto=extos),
    alpha=0.3,
    fill="grey",
    color="grey50",
    size=0.05)

p <- p %<+% da1 +
  geom_tippoint(
    mapping=aes(fill=Phylum),
    size=1.2,
    shape=21,
    stroke=0.05,
    show.legend=FALSE
) +
  scale_fill_manual(values=tipcolors) +
  new_scale_fill() +
  geom_fruit(
    data=da2,
    geom=geom_bar,
    mapping=aes(x=Abundance, y=ID, fill=Phyla),
    offset=-0.2,
    pwidth=0.1,
    stat='identity',
    orientation="y"
) +
  scale_fill_manual(values=phylacolor,
                    guide=guide_legend(keywidth=0.6,
                                        keyheight=0.6,
                                        order=1, ncol=1)) +
  cladelabels +
  new_scale_fill() +
  geom_fruit(
    data=da3,

```

```

geom=geom_star,
mapping=aes(x=Pos, y=ID, fill=Type, size=Pos),
starshape=26,
alpha=0.8,
starstroke=0,
offset=0.024,
pwidth=0.008
) +
scale_fill_manual(values=c("blue", "black", "green", "red"),
                   guide=guide_legend(keywidth=1, keyheight=1, order=2,
                                       override.aes=list(alpha=1, size=2)),
                   na.translate=FALSE) +
scale_size_continuous(range=c(1.4,2.2), guide="none") +
geom_treescale(fontsize=1.2, linesize=0.3) +
theme(legend.position=c(0.94, 0.5),
      legend.title=element_text(size=7),
      legend.background=element_rect(fill=NA),
      legend.text=element_text(size=6),
      legend.spacing.y = unit(0.02, "cm"))

```

p

1.1.3.2 The second example in this section This example reproduces Fig 2 of (Asnicar et al. 2015). The phylogenetic tree also was built using 3737 microbes (Segata et al. 2013). The associated data contains present or not (discrete data) and type (discrete data) of target gene, the type (discrete data) and capability (continuous data) of fatty acid metabolism, and the length of microbes. The present or not of different target gene was visualized with first inner heat map (the type of target gene was mapped to color of heat map, and the present or not of target gene was mapped to the x value(take on numerical values)). The type and capability of fatty acid metabolism was also visualized with middle heat map (the type was mapped to color of heat map, the capability of fatty acid metabolism was mapped to the transparency of heat map). The length of genome was displayed with bar chart (the length of genome was mapped to length of bar, the phylum information of genome was mapped to the color of bar). (Fig. S4).

```

library(ggtreeExtra)
library(ggtree)
library(treeio)
library(tidytree)
library(ggstar)
library(ggplot2)
library(ggnewscale)

tree <- read.tree("../data/kegg/kegg.nwk")
# The attributes of tip point
dt1 <- read.csv("../data/kegg/tippoint_attr.csv")
# The attributes of first ring
dt2 <- read.csv("../data/kegg/firstring_attr.csv")
# The attributes of second ring
dt3 <- read.csv("../data/kegg/secondring_attr.csv")
# The attrihutes of bar plot
dt4 <- read.csv("../data/kegg/barplot_attr.csv")

#reorder the Phyla column
dt1$Phyla <- factor(dt1$Phyla, levels=c("Actinobacteria", "Aquificae", "Bacteroidetes",
                                             "Chlamydiae", "Chlorobi", "Chloroflexi", "Crenarchaeota",
                                             "Cyanobacteria", "Euryarchaeota", "Firmicutes", "Proteobacteria",
                                             "Spirochaetes", "Tenericutes", "Thermi", "Thermotogae", "Other"))

# reorder the Type2 column
dt3$Type2 <- factor(dt3$Type2, levels=c("FA synth init", "FA synth elong",
                                         "acyl-CoA synth", "beta-Oxidation",
                                         "Glyoxylate cycle", "TCA cycle", "HMP shunt", "Pentose phosphate"))

```

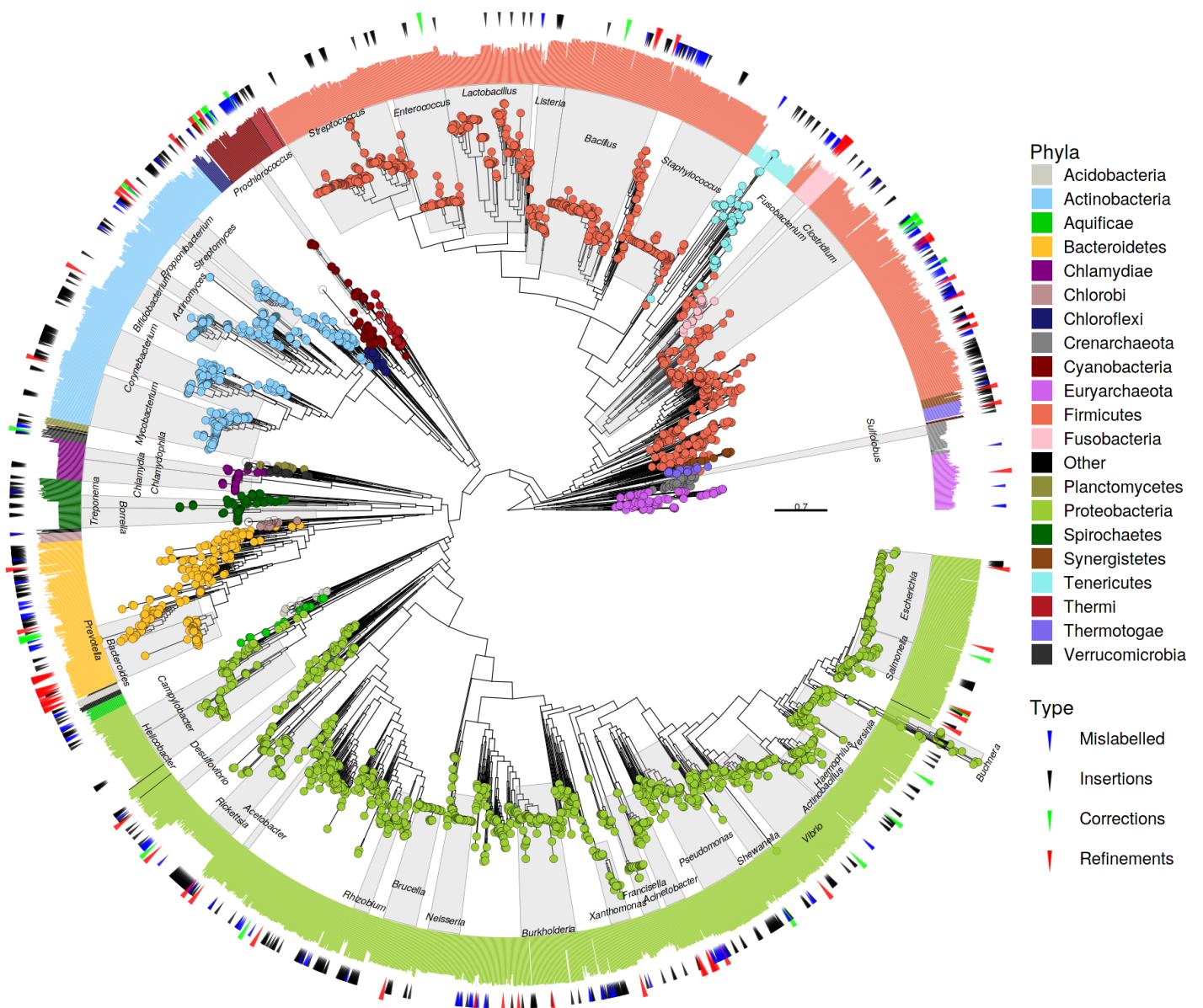


Fig. S3: The bacterial and archaeal (3737 genomes) phylogenetic tree built with 400 conserved proteins. The tip points are built using discrete data, the different colors represent the different phyla. The external bar is created using continuous data, the length represents the fraction of the 400 proteins contained in each genome. The external triangle is also created using discrete data, the different colors represent the different types of genomes (mislabelled and confidently replaced see Fig 1 of (Segata et al. 2013)).

```

    "Ketone biosynth"))

dt4$Phyla <- factor(dt4$Phyla, levels=c("Actinobacteria", "Aquificae", "Bacteroidetes",
                                             "Chlamydiae", "Chlorobi", "Chloroflexi", "Crenarchaeota",
                                             "Cyanobacteria", "Euryarchaeota", "Firmicutes", "Proteobacteria",
                                             "Spirochaetes", "Tenericutes", "Thermi", "Thermotogae", "Other"))

# extract node label for the clade layers
nodelab <- tree$node.label[nchar(tree$node.label)>0]
nodeids <- nodeid(tree, nodelab)

# the position of clade label
textex <- c(1.0, 0.4, 0.2, 1.4, 1.4, 0.4, 1.4, 1.4, 0.4, 0.4,
          0.8, 1, 0.6, 0.6, 0.4, 0.3, 0, 0.4, 0.1, 0.25,

```

```

    0.2, 0.3, 0.8, 0.8, 0.6, 2.4)
# clade layers
cladelabels <- mapply(function(x, y, z){geom_cladelabel(node=x, label=y, barsize=NA, extend=0.3,
                                         offset.text=z, fontsize=1.2, angle="auto",
                                         hjust=0.5, horizontal=FALSE, fontface="italic")},
                           nodeids, nodelab, textex, SIMPLIFY=FALSE)

# high light layers
fills <- c("#808080", "#808080", "#808080", "#808080",
           "#191970", "#87CEFA", "#FFC125", "#B0171F", "#B0171F",
           "#B0171F", "#B0171F", "#B0171F", "#B0171F", "#B0171F",
           "#B0171F", "#B0171F", "#B0171F", "#B0171F", "#B0171F",
           "#B0171F", "#B0171F", "#9ACD32", "#9ACD32", "#9ACD32",
           "#006400", "#800000")
highlights <- mapply(function(x, y){geom_hilight(node=x, extendto=5.8, alpha=0.3,
                                                 fill=y, color=y, size=0.05)},
                      nodeids, fills, SIMPLIFY=FALSE)

# to reproduce the original figures, we use the same colors.
# uses can custom set it.
colors <- c("#9ACD32", "#EE6A50", "#87CEFA", "#FFC125", "#D15FEE", "#8DEEEE", "#800000",
            "#006400", "#800080", "#808080", "#B0171F", "#B0171F", "#191970", "#7B68EE",
            "#00CD00", "Black")

p <- ggtree(tree, layout="circular", size=0.1) + highlights

p <- p +
  geom_fruit(
    data=dt1,
    geom=geom_point,
    mapping=aes(y=ID, fill=Phyla),
    shape=21,
    size=1.2,
    stroke=0.05,
    position="identity",
    show.legend=FALSE
  ) +
  scale_fill_manual(values=colors) +
  cladelabels +
  new_scale_fill() +
  geom_fruit(
    data=dt2,
    geom=geom_tile,
    mapping=aes(y=ID, x=ring, fill=Type1),
    offset=-0.02,
    pwidth=0.14,
    addbrink=TRUE
  ) +
  scale_fill_manual(
    name="ATP synthesis",
    values=c("#339933", "#dfac03"),
    guide=guide_legend(keywidth=0.5, keyheight=0.5, order=1)
  ) +
  new_scale_fill() +
  geom_fruit(
    data=dt3,
    geom=geom_tile,
    mapping=aes(y=ID, alpha=Abundance, x=Type2, fill=Type2),
    show.legend=FALSE
  )

```

```

    offset=0.001,
    pwidth=0.18
) +
scale_fill_manual(
  name="Fatty Acid metabolism",
  values=c("#b22222", "#005500", "#0000be", "#9f1f9f", "#793a07"),
  guide=guide_legend(keywidth=0.5, keyheight=0.5, order=2)
) +
scale_alpha_continuous(range=c(0, 0.4),
                       guide=guide_legend(keywidth=0.5, keyheight=0.5, order=3)) +
new_scale_fill() +
geom_fruit(data=dt4,
            geom=geom_bar,
            mapping=aes(y=ID, x=Length, fill=Phyla),
            stat="identity",
            orientation="y",
            pwidth=0.3,
            position=position_dodge()) +
scale_fill_manual(values=colors,
                  guide=guide_legend(keywidth=0.5, keyheight=0.5, order=4)) +
geom_treescale(fontsize=1.2, linesize=0.3) +
theme(legend.position=c(0.95, 0.5),
      legend.background=element_rect(fill=NA),
      legend.title=element_text(size=7),
      legend.text=element_text(size=6),
      legend.spacing.y = unit(0.02, "cm"))

```

P

1.1.4 Obtain more power with some packages

The *geom_fruit* not only work with some *geom* function defined in *ggplot2* (Wickham 2016), but also can work with *geom* function defined in ggplot2-based packages. This ability will make *ggtreeExtra* more power to present dataset. We use an example from Fig.1 of (Song et al. 2020) to show *geom_fruit* can work with *geom_phylopic* defined in *ggimage* (Yu 2020). The attributes (such as color, size) of subplot also can be mapped to variables of external dataset. This is often not available in the existing tools. The source data of the example is available at this repository⁴.

```

library(ggtree)
library(treeio)
library(ggplot2)
library(ggtreeExtra)
library(tidytree)
library(ggnewscale)
library(ggimage)

tr <- read.tree("../data/VertebrateGutMicrobiomes/annotated_host_tree.tre")
corda <- read.csv("../data/VertebrateGutMicrobiomes/mantel.jaccard.pearson.csv")
corda$r <- abs(corda$r)
barda <- read.csv("../data/VertebrateGutMicrobiomes/data_diet_bar.csv", check.names=F)
barda <- reshape2::melt(barda, id.vars="ID", variable.name="Diet", value.name="mete")
barda$Diet <- factor(barda$Diet, levels=c("Fruit", "Invertebrates",
                                             "Nectar", "Plants", "Scavenging",
                                             "Seeds", "Meat (Ectotherms)",
                                             "Meat (Endotherms)",
                                             "Meat (Fish)", "Meat (Unknown)"))

cladeda <- read.csv("../data/VertebrateGutMicrobiomes/data_clade_class.csv", check.names=F)

```

⁴https://github.com/tanaes/tetrapod_microbiome_analysis

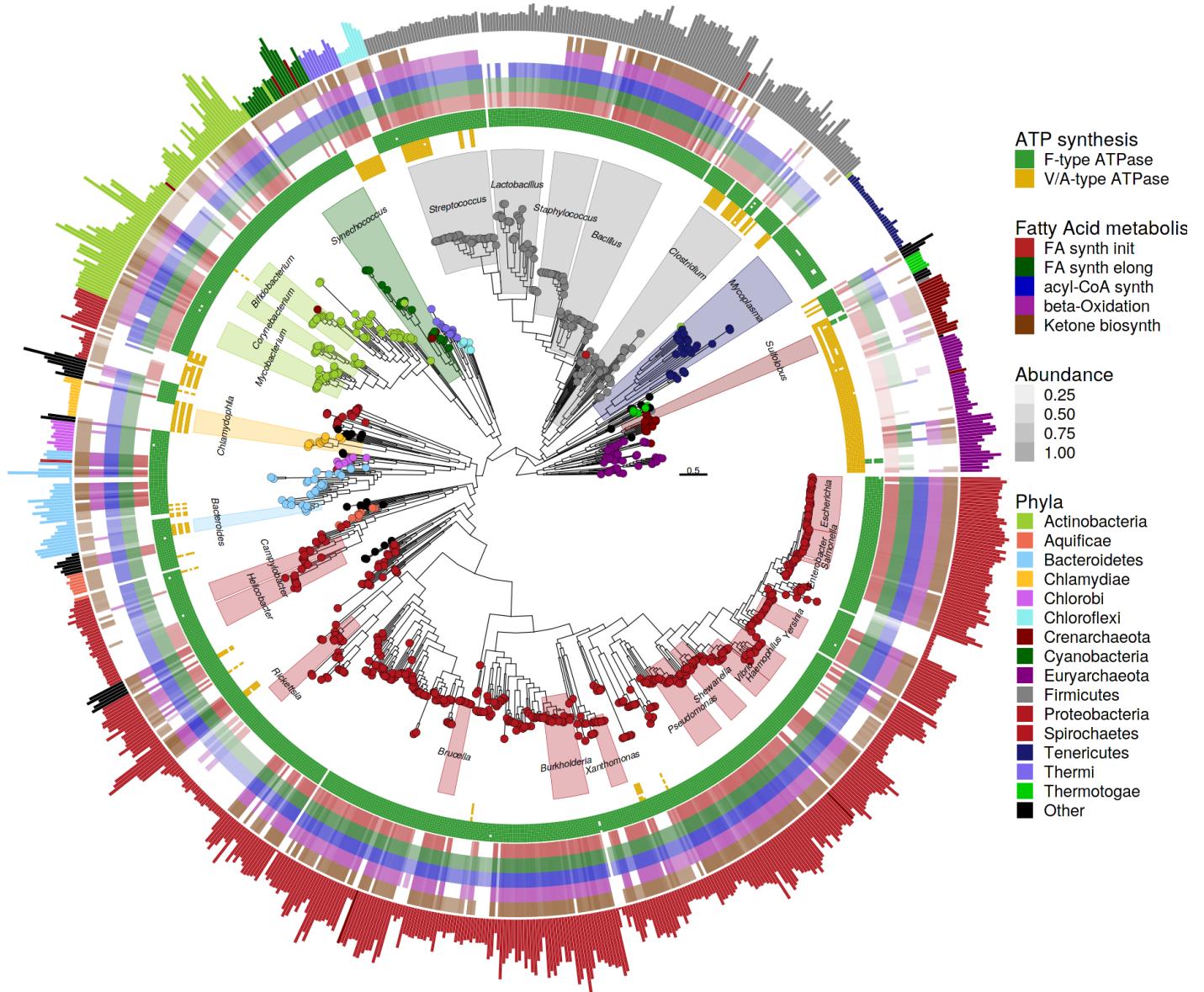


Fig. S4: The phylogenetic tree built using in (Segata et al. 2013) using 3737 microbial genomes. The first and second ring heat map were built with discrete data, it represents the presence or absence of each module, the other heat map rings were created with continuous data, the transparency represents the capability of fatty acid metabolism, the different colors represent the types of fatty acid metabolism. The length of bar represents the genome length of corresponding microbes.

```

cladedda$id <- nodeid(tr, cladedda$id)
cladedda$class <- factor(cladedda$class, levels=c("Amphibia", "Chelonia", "Lepidosauria",
                                                 "Crocodylomorpha", "Aves", "Mammalia"))

flightda <- read.csv("../data/VertebrateGutMicrobiomes/data_flight_bar.csv")

phylopicda <- read.csv("../data/VertebrateGutMicrobiomes/data_phylopic_uid.csv")
phylopicda$class <- factor(phylopicda$class, levels=c("Amphibia", "Chelonia", "Lepidosauria",
                                                       "Aves", "Mammalia"))

p <- ggtree(tr, layout="fan", open.angle=15)

p <- p %<+% corda
p$data$width <- ifelse(is.na(p$data$r), 0.1, 0.6)

```

```

r <- NULL
p <- p +
  aes(color=r, size=I(width)) +
  scale_colour_viridis_c(
    name="Mantel Correlation",
    option="C",
    guide=guide_colorbar(
      barheight=0.6,
      order=4,
      title.position="top",
      label.position="bottom",
      direction="horizontal"
    )
  )

p1 <- p +
  geom_fruit(
    data=barda,
    geom=geom_bar,
    mapping=aes(x=mete, y=ID, fill=Diet),
    orientation="y",
    stat="identity",
    colour=NA,
    pwidth=0.25,
    offset=0.008
  ) +
  scale_fill_manual(
    values=c("#a6cee3", "#cab2d6",
             "#1f78b4", "#33a02c",
             "#6a3d9a", "#b2df8a",
             "#fb9a99", "#e31a1c",
             "#ff7f00", "#fdbf6f"),
    guide=guide_legend(keywidth=0.5, keyheight=0.5, order=1)
  )

p2 <- p1 +
  new_scale_colour() +
  geom_cladelab(
    data=cladeda,
    mapping=aes(node=id, label=class, colour=class),
    textcolour=NA,
    barsize=4,
    extend=0.2,
    offset=100) +
  scale_colour_manual(
    name="Host Class",
    values=c("#b2df8a", "#33a02c", "#fb9a99",
             "#e31a1c", "#EACB47", "#6a3d9a"),
    guide=guide_legend(
      keywidth=0.5,
      keyheight=0.5,
      order=2,
      override.aes=list(size=3, alpha=1)
    )
  )

p3 <- p2 +
  new_scale_fill() +

```

```

geom_fruit(
  data=flightda,
  geom=geom_tile,
  mapping=aes(y=ID, fill=flight),
  size=0,
  width=14,
  offset=0.11,
  pwidth=0.4,
) +
scale_fill_manual(
  name="Flight Status",
  values=c("black", "white"),
  guide=guide_legend(keywidth=0.5, keyheight=0.5, order=3,
                     override.aes=list(color="black", size=0.3)))
p4 <- p3 +
  new_scale_colour() +
  geom_fruit(
    data=phylopicda,
    geom=geom_phylopic,
    mapping=aes(y=taxa, image=uid, color=class),
    size=0.035,
    offset=0.16,
    alpha=0.8,
    position=position_identityx()
) +
  scale_colour_manual(
    values=c("#b2df8a", "#33a02c", "#fb9a99",
             "#EACB47", "#6a3d9a"),
    guide="none"
) +
  theme(
    legend.background=element_rect(fill=NA),
    legend.title=element_text(size=9),
    legend.text=element_text(size=6.6),
    legend.spacing.y = unit(0.02, "cm")
)
p4

```

1.1.5 Obtain new patterns by annotating tree with relationship data

The `geom_fruit` is a general function to link `ggtree` (Yu et al. 2017) and `ggplot2` (Wickham 2016), so it can be applied in many fields. It can give user more convenience and reproducibility to visualize and annotate phylogenetic trees with associated data for better exploring phylogenetic patterns behind multi-dimensional data. For example, Fig.1b and Fig.2 of (Helfrich et al. 2018) show the directional interactions and the biosynthetic potential of isolates from *Arabidopsis* leaf microbiome in phylogenetic tree. However, because GraPhAn (Asnicar et al. 2015) did not display the stack bar plot, some phylogenetic patterns behind multi-dimensional data were not be found easily. Here, we can use `ggtree` and `ggtreeExtra` to show the total information of Fig.1b and Fig.2 of (Helfrich et al. 2018). We found some strains from Firmicutes and from Grammaproteobacteria have more inhibitor interactions. However, many strains from Alphaproteobacteria and Betateobacteria prefer the interaction of sensitivity. In addition, These strains that prefer the interactions of sensitivity might be have more BGCs from ribosomally synthesized and post-translationally modified peptide (*RiPP*).

```

library(ggtree)
library(ggtreeExtra)
library(ggplot2)
library(MicrobiotaProcess)
library(ggstar)
library(ggnewscale)
library(grid)

```

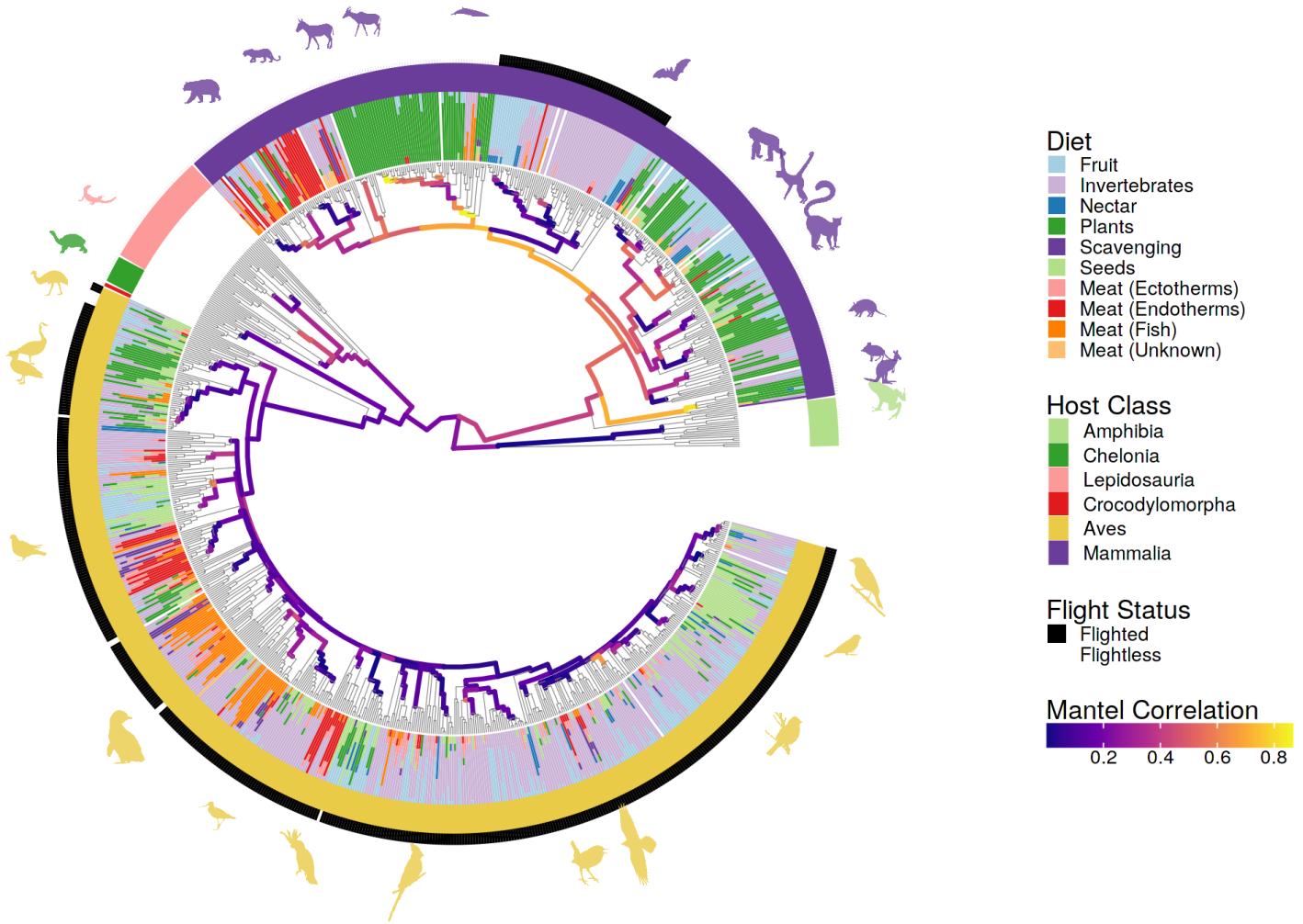


Fig. S5: Host tree was obtained from TimeTree (Kumar et al. 2017). The branch colour represents the Mantel Pearson correlation. The stack bar chart represents the diet composition of host. The inner ring represents the host taxonomic class, and some representative species are also displayed in the outermost circle. The middle ring represents flight status.

```

alltax <- read.csv("../data/Arabidopsis_leaf_microbiome/all_stain_taxonomy.csv")
linktab <- read.csv("../data/Arabidopsis_leaf_microbiome/Interaction_link_tab.csv")
weighttab <- read.csv("../data/Arabidopsis_leaf_microbiome/Interaction_weight.csv")
tippoint <- read.csv("../data/Arabidopsis_leaf_microbiome/stain_tippoint.csv")
BGCsda <- read.csv("../data/Arabidopsis_leaf_microbiome/BGCs_heatmap.csv")

tippoint$Taxa <- factor(tippoint$Taxa,
                         levels=c("Actinobacteria",
                                 "Bacteroidetes",
                                 "Firmicutes",
                                 "Deinococcus-Thermus",
                                 "Alphaproteobacteria",
                                 "Betaproteobacteria",
                                 "Gammaproteobacteria"
                         )
)
tippoint$names <- gsub("s__Leaf","",tippoint$Isolation)

BGCsda$BGCs <- factor(BGCsda$BGCs,
                       levels=c("modular.PKS",
                               "modular.PKS.NRPS.hybrid",

```

```

        "non_modular.PKS", "NRPS",
        "RiPP",
        "Quorum.sensing",
        "terpene",
        "other"
    )
)
BGCsda$Count <- log10(BGCsda$Count+1)
BGCsda$Count <- ifelse(BGCsda$Count==0, NA, BGCsda$Count)

trda <- convert_to_treedata(alltax)
p <- ggtree(
  trda,
  layout="inward_circular",
  size=0.2,
  xlim=c(18,NA)
)
p <- p %<+% tippoint

p1 <- p +
  geom_tippoint(
    mapping=aes(
      color=Taxa,
      shape=Level
    ),
    size=1,
    alpha=0.8
  ) +
  scale_color_manual(values=c("#EF3B2C", "#1D91C0", "#FEB24C", "grey60",
                             "#7FBC41", "#4D9221", "#276419"),
                     guide=guide_legend(
                       keywidth=0.5,
                       keyheight=0.5,
                       order=2,
                       override.aes=list(shape=c("Actinobacteria"=20,
                                                 "Bacteroidetes" =20,
                                                 "Firmicutes"     =20,
                                                 "Deinococcus-Thermus" =20,
                                                 "Alphaproteobacteria" =18,
                                                 "Betaproteobacteria"   =18,
                                                 "Gammaproteobacteria" =18
                                               ),
                       size=2
                     ),
                     na.translate=TRUE
  )
) +
  scale_shape_manual(
    values=c("Phylum"=20, "Class"=18),
    guide="none"
  )

p2 <- p1 +
  new_scale_color() +
  geom_taxalink(
    data=linktab,
    mapping=aes(
      taxa1=Inhibitor,

```

```

            taxa2=Sensitive,
            color=Interaction
        ),
        alpha=0.6,
        offset=0.1,
        size=0.15,
        ncp=10,
        hratio=1,
        arrow=grid::arrow(length = unit(0.005, "npc"))
    ) +
    scale_colour_manual(values=c("chocolate2", "#3690C0", "#009E73"),
        guide=guide_legend(
            keywidth=0.8,
            keyheight=0.5,
            order=1,
            override.aes=list(alpha=1,
                size=0.5)
        )
    )
}

p3 <- p2 +
  geom_fruit(
    data=BGCsda,
    geom=geom_tile,
    mapping=aes(y=Strain, x=BGCs, alpha=Count, fill=BGCs),
    offset=-0.9,
    pwidth=1,
    size=0.02,
    color = "grey50"
  ) +
  scale_alpha_continuous(range=c(0.1, 1),
    name=bquote(paste(Log[10], .(("Count+1")), "))),
    guide=guide_legend(
        keywidth = 0.4,
        keyheight = 0.4,
        order=4
    )
  ) +
  scale_fill_manual(
    values=c("#66C2A5", "#FC8D62", "#8DA0CB", "#E78AC3",
      "#A6D854", "#FFD92F", "#E5C494", "#B3B3B3"),
    guide=guide_legend(keywidth = 0.4, keyheight = 0.4, order=3)
  )
)

p4 <- p3 +
  geom_tiplab(
    mapping=aes(label=names),
    align=TRUE,
    size=1,
    linetype=NA,
    offset=7.8
  )

p5 <- p4 +
  new_scale_fill() +
  geom_fruit(
    data=weighttab,
    geom=geom_bar,

```

```

mapping=aes(
    x=value,
    y=Strain,
    fill=Number
),
stat="identity",
orientation="y",
offset=0.48,
pwidth=2,
axis.params=list(
    axis="x",
    text.angle=-45,
    hjust=0,
    vjust=0.5,
    nbreak=4,
)
) +
scale_fill_manual(
    values=c("#E41A1C", "#377EB8", "#4DAF4A", "#984EA3"),
    guide=guide_legend(keywidth=0.5, keyheight=0.5, order=5)
) +
theme(
    legend.background=element_rect(fill=NA),
    legend.title=element_text(size=6.5),
    legend.text=element_text(size=5),
    legend.spacing.y = unit(0.02, "cm"),
    legend.margin=margin(0.1, 0.9, 0.1, -0.9, unit="cm"),
    legend.box.margin=margin(0.1, 0.9, 0.1, -0.9, unit="cm"),
    plot.margin = unit(c(-1.2, -1.2, -1.2, 0.1), "cm")
)

```

p5

1.2 Geometric layers that supported by *geom_fruit* of *ggtreeExtra*

As mentioned above, *geom_fruit* is an general function for linking multiple geometric graph to tree. It supports multiple layout type tree, such as circular layout, inward circular layout, rectangular layout etc. It can re-order the input data based on tree structure and displayed the data at the external of tree with the geometric function, which was defined in *ggplot2* (Wickham 2016) and other *ggplot2* extension packages. Here is the list of the geometric functions that work seamlessly with *geom_fruit* (Table S1). Since the *ggplot2* (Wickham 2016) community keeps expanding and more *geom* layers will be implemented in either *ggplot2* (Wickham 2016) or other extensions, *ggtreeExtra* also will support more *geom* layers and will gain more power to explore data in future. Several examples of *ggtreeExtra* working with other geometric function has been listed in supplementary file2.

1.3 Adding further annotation to specific layers

When the annotation information should be add to same panel, we developed *geom_fruit_list* to add multiple layers to the same external ring of circular tree (Fig S7).

```

library(ggtreeExtra)
library(ggtree)
library(treeio)
library(ggplot2)
set.seed(1024)
tree <- rtree(100)
df <- data.frame(id=tree$tip.label, value=sample(seq_len(200), 100, replace=TRUE),
                  group=c(rep("A", 50), rep("B", 50)))

```

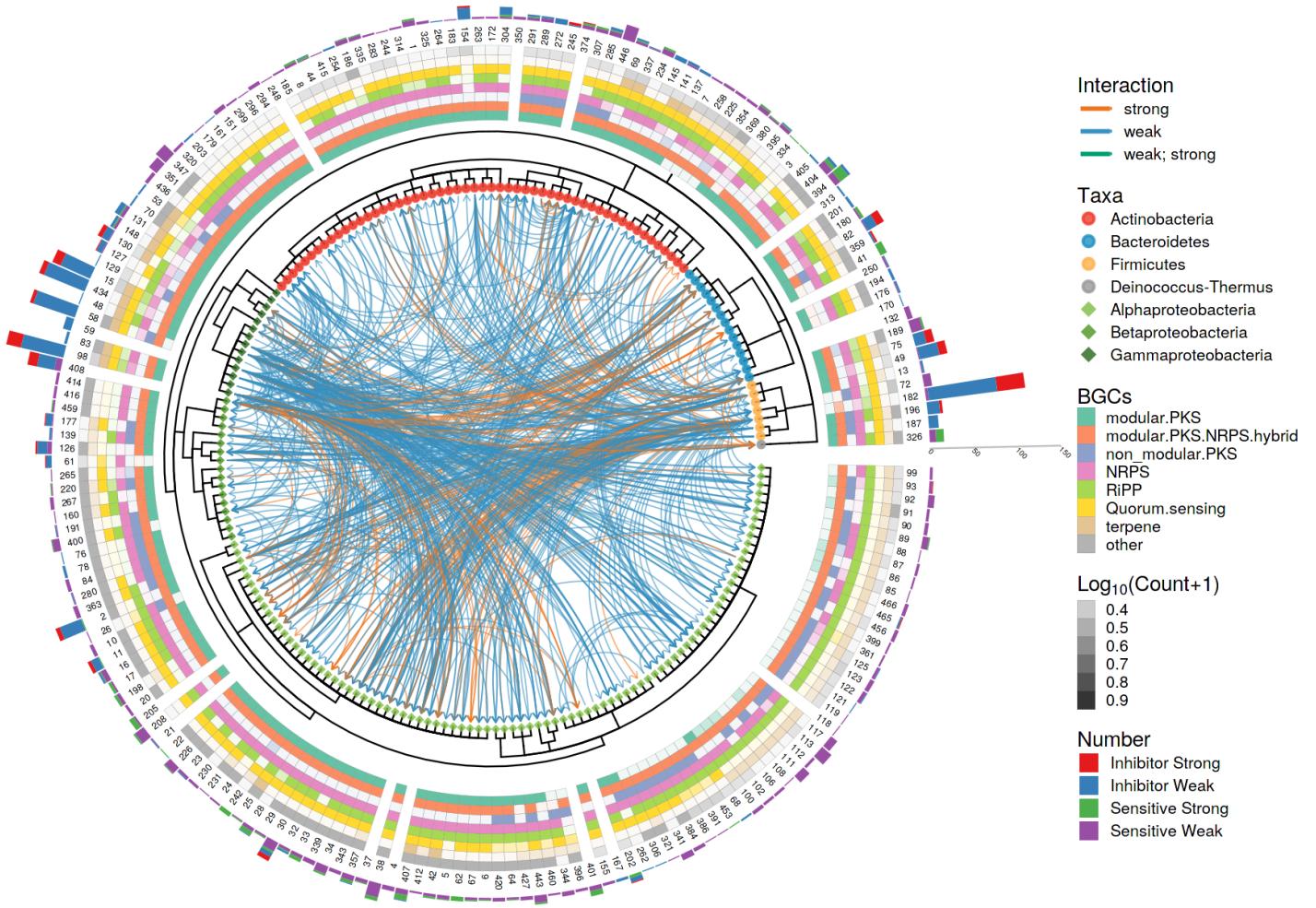


Fig. S6: The Phylogenetic tree of isolates from *Arabidopsis* leaf microbiome (Bai et al. 2015). The lines in inner represent the directional inhibitory interactions, the direction of arrow represents from inhibitor to sensitive, and the color of line represents the weak or strong interactions. The color of tip point represents the taxonomy annotation, circle represents the phylum, square represents the class from Proteobacteria. The heat map of external ring represents the number of detected BGCs (biosynthesis gene clusters). The ring from inner to outside represents the Modular polyketide synthase, Modular PKS non-ribosomal peptide synthetase (NRPS) hybrid, Non modular PKS, NRPS, ribosomally synthesized and post-translationally modified peptide (RiPP), Quorum sensing, Terpene, and Others. The external bar represent the number of interactions of inhibitions or sensitivities per strain.

```

p <- ggtree(tree, layout="fan", open.angle=180) + xlim(-2, NA)

p <- p + geom_fruit_list(data=df,
                           geom=geom_bar,
                           mapping=aes(x=value, y=id, fill=group),
                           orientation="y",
                           stat="identity"
                           ),
      geom_fruit(data=df,
                  geom=geom_text,
                  mapping=aes(x=value, y=id, label=value),
                  size=2,
                  offset=0.055,
                  pwidth=0.26,
                  position=position_identityx(),
                  ) +
      scale_fill_manual(values=c("chocolate2", "#009E73"),

```

Table S1: List of geometric layers supported by 'geom_fruit()'

Package	Geom Layer	Description
ggdist	geom_dotsinterval	creates dots, intervals, and quantile dotplots
	geom_pointinterval	creates point and multiple uncertainty interval
	geom_slab	creates slab geom
	geom_slabinterval	creates slab, point and interval meta-geom
ggimage	geom_image	visualizes image files
	geom_phylopic	queries image files from phylopic database and visualizes them
ggpattern	geom_bar_pattern	draws bar charts with support for pattern fills
	geom_boxplot_pattern	draws box and whiskers plot with support for pattern fills
	geom_col_pattern	draws bar charts using 'stat_identity()' with support for pattern fills
	geom_tile_pattern	draws rectangle by using the center of the tile and its size with support for pattern fills
ggplot2	geom_bar	draws bar charts
	geom_boxplot	draws box and whiskers plot
	geom_col	draws bar charts using 'stat_identity()'
	geom_label	draws a rectangle behind the text
	geom_point	creates scatterplots
	geom_raster	a high performance special case for all the tiles are the same size
	geom_text	adds text to the plot
	geom_tile	draws rectangle by using the center of the tile and its size
ggrepel	geom_text_repel	adds text to the plot. The text labels repel away from each other and away from the data points
	geom_label_repel	draws a rectangle underneath the text. The text labels repel away from each other and away from the data points
ggstance	geom_barh	horizontal version of 'geom_bar()'
	geom_boxploth	horizontal version of 'geom_boxplot()',
	geom_colh	horizontal version of 'geom_col()'
ggstar	geom_star	creates scatterplots
ggsymbol	geom_symbol	creates scatterplots
scatterpie	geom_scatterpie	creates scatter pie plot

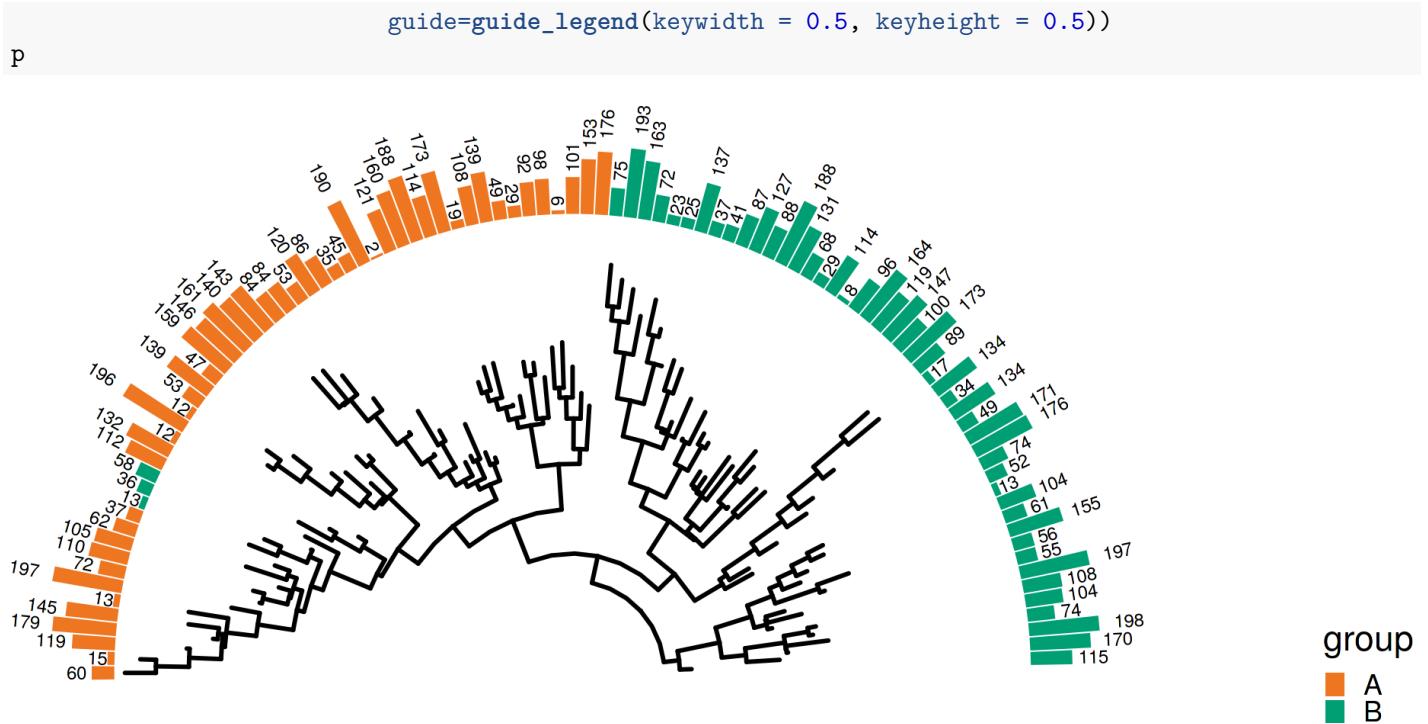


Fig. S7: visualizing multiple layers at them same external ring using ggtreeExtra

1.4 Summary

As far as we known, None of R packages support visualizing external data and align the plot to circular layout trees or other layout trees. There are some tools depending on python or other platform, they provide some features to annotate and visualize circular phylogenetic tree, but they are not general and still have some shortcomings. For example, *GraPhlAn* (Asnicar et al. 2015) and *ETE3* (Huerta-Cepas, Serra, and Bork 2016), both based on python, *GraPhlAn* (Asnicar et al. 2015) provided few graph layers to display associated data, while *ETE3* (Huerta-Cepas, Serra, and Bork 2016) did not support visualizing associated dataset in the outer ring of circular phylogenetic tree. There are some web tools support visualizing phylogenetic tree, such as *iTOL* (Letunic and Bork 2019), *Microreact* (Argimón et al. 2016) and *EvolView* (Subramanian et al. 2019). But the feature to link tree with external datasets is often missing, and they often need user to provide predefined data types, which make it tedious to use and debug. They are also not conducive to integration into the relevant data analysis pipeline.

ggtreeExtra is designed to work with *treeio*, *tidytree* and *gtree*. Since it depends R, an ideal environment for statistical analysis and visualization, the user can use it with other R package, such as *ggstar* (Xu 2020), *ggnewscale* (Campitelli 2020) and *ggplot2* (Wickham 2016). It inherits the merits of *ggplot2* (Wickham 2016), built based on grammar of graphics. So it is easy to use *ggtreeExtra*. In addition, there are no restriction of data types (Fig. S3 and S4). Combining multiple geometric layers to display associated data is supported (Fig. S4 and S1). The internal data of tree data or data added by %<+% operator also can be used in *geom_fruit* (Fig. S2). It is more efficient and convenient to visualize multi-dimensional data using multiple external ring layers (Fig. S4). In addition, the *radial*, *rectangular* and *slanted* layouts also are supported by *ggtreeExtra*. The tree in *geom_fruit* can be fully annotated with multiple layers (high light, clade labels, group clade, etc functions of *gtree*) (Fig. S1, S3 and S4).

Reference

- Argimón, S., K. Abudahab, R. J. E. Goater, A. Fedosejev, J. Bhai, C. Glasner, E. J. Feil, et al. 2016. “Microreact: Visualizing and Sharing Data for Genomic Epidemiology and Phylogeography.” Journal Article. *Microb Genom* 2 (11): e000093. <https://doi.org/10.1099/mgen.0.000093>.
- Asnicar, Francesco, George Weingart, Timothy L Tickle, Curtis Huttenhower, and Nicola Segata. 2015. “Compact Graphical Representation of Phylogenetic Data and Metadata with Graphlan.” *PeerJ* 3: e1029. <https://doi.org/10.717/peerj.1029>.
- Bai, Yang, Daniel B Müller, Girish Srinivas, Ruben Garrido-Oter, Eva Potthoff, Matthias Rott, Nina Dombrowski, et al. 2015. “Functional Overlap of the Arabidopsis Leaf and Root Microbiota.” *Nature* 528 (7582): 364–69. <https://doi.org/10.1038/nature16192>.
- Campitelli, Elio. 2020. *Ggnewscale: Multiple Fill and Colour Scales in 'Ggplot2'*. <https://CRAN.R-project.org/package=ggnewscale>.
- Helfrich, Eric J. N., Christine M. Vogel, Reiko Ueoka, Martin Schäfer, Florian Ryffel, Daniel B. Müller, Silke Probst, Markus Kreuzer, Jörn Piel, and Julia A. Vorholt. 2018. “Bipartite Interactions, Antibiotic Production and Biosynthetic Potential of the Arabidopsis Leaf Microbiome.” Journal Article. *Nature Microbiology* 3 (8): 909–19. <https://doi.org/10.1038/s41564-018-0200-0>.
- Huerta-Cepas, Jaime, François Serra, and Peer Bork. 2016. “ETE 3: Reconstruction, Analysis, and Visualization of Phylogenomic Data.” *Molecular Biology and Evolution* 33 (6): 1635–8. <https://doi.org/10.1093/molbev/msw046>.
- Kumar, Sudhir, Glen Stecher, Michael Suleski, and S. Blair Hedges. 2017. “TimeTree: A Resource for Timelines, Timetrees, and Divergence Times.” *Molecular Biology and Evolution* 34 (7): 1812–9. <https://doi.org/10.1093/molbev/msx116>.
- Letunic, Ivica, and Peer Bork. 2019. “Interactive Tree of Life (iTOL) V4: Recent Updates and New Developments.” *Nucleic Acids Research* 47 (W1): W256–W259. <https://doi.org/10.1093/nar/gkz239>.
- Morgan, Xochitl C., Nicola Segata, and Curtis Huttenhower. 2013. “Biodiversity and Functional Genomics in the Human Microbiome.” *Trends in Genetics* 29 (1): 51–58. <https://doi.org/10.1016/j.tig.2012.09.005>.
- Segata, Nicola, Daniela Börnigen, Xochitl C. Morgan, and Curtis Huttenhower. 2013. “PhyloPhlAn Is a New Method for Improved Phylogenetic and Taxonomic Placement of Microbes.” Journal Article. *Nature Communications* 4 (1): 2304. <https://doi.org/10.1038/ncomms3304>.

- Smith, Garrett J, and Kelly C Wrighton. 2019. “Metagenomic Approaches Unearth Methanotroph Phylogenetic and Metabolic Diversity.” *Curr Issues Mol Biol* 33: 57–84. <https://doi.org/10.21775/9781912530045.03>.
- Song, Se Jin, Jon G. Sanders, Frédéric Delsuc, Jessica Metcalf, Katherine Amato, Michael W. Taylor, Florent Mazel, et al. 2020. “Comparative Analyses of Vertebrate Gut Microbiomes Reveal Convergence Between Birds and Bats.” Edited by Joerg Graf. *mBio* 11 (1). <https://doi.org/10.1128/mBio.02901-19>.
- Subramanian, Balakrishnan, Shenghan Gao, Martin J Lercher, Songnian Hu, and Wei-Hua Chen. 2019. “Evolview V3: A Webserver for Visualization, Annotation, and Management of Phylogenetic Trees.” *Nucleic Acids Research* 47 (W1): W270–W275. <https://doi.org/10.1093/nar/gkz357>.
- Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <https://ggplot2.tidyverse.org>.
- Wilkinson, Leland. 2012. “The Grammar of Graphics.” In *Handbook of Computational Statistics: Concepts and Methods*, edited by James E. Gentle, Wolfgang Karl Härdle, and Yuichi Mori, 375–414. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-21551-3_13.
- Xu, Shuangbin. 2020. *Ggstar: Star Layer for 'Ggplot2'*. <https://CRAN.R-project.org/package=ggstar>.
- Yu, Guangchuang. 2020. *Ggimage: Use Image in 'Ggplot2'*. <https://CRAN.R-project.org/package=ggimage>.
- Yu, Guangchuang, Tommy Tsan-Yuk Lam, Huachen Zhu, and Yi Guan. 2018. “Two Methods for Mapping and Visualizing Associated Data on Phylogeny Using Ggtree.” *Molecular Biology and Evolution* 35 (2): 3041–3. <https://doi.org/10.1093/molbev/msy194>.
- Yu, Guangchuang, David Smith, Huachen Zhu, Yi Guan, and Tommy Tsan-Yuk Lam. 2017. “Ggtree: An R Package for Visualization and Annotation of Phylogenetic Trees with Their Covariates and Other Associated Data.” *Methods in Ecology and Evolution* 8 (1): 28–36. <https://doi.org/10.1111/2041-210X.12628>.