Tree/Heatmap Plots

John Quensen

4/21/2021

This script details how to make a tree plot based on ANI distances among genomes and align it to a heatmap of orthologous genes. This aids in interpretation of the tree structure.

Load packages.

```
suppressPackageStartupMessages(library(ape))
suppressPackageStartupMessages(library(phangorn))
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(hablar))
suppressPackageStartupMessages(library(RDPutils))
suppressPackageStartupMessages(library(ggtree))
suppressPackageStartupMessages(library(aplot))
```

Read in data. Edit the path to the MiGA project in the first line of the next code block. The other paths are those within MiGA's output directory structure.

```
# Set path to the MiGA project.
path_to_miga_project <- "./"

# Read in the ANI tree file.
path_to_ani_tree <- "data/10.clades/02.ani/"
ani.tree <- ape::read.tree(paste0(path_to_miga_project, path_to_ani_tree, "miga-project.ani.nwk"))

# Read in the orthologous genes file.
path_to_ogs <- "data/10.clades/03.ogs/"
ogs <- read.table(paste0(path_to_miga_project, path_to_ogs, "miga-project.ogs"),
    header = TRUE, stringsAsFactors = FALSE, sep = "\t")
```

ogs is a text file giving the genomes in columns and orthologs in rows. A sample is:

```
ogs[1:5, 1:5]
```

```
##   GCF_000011905_1_ASM1190v1_genomic GCF_001610775_1_ASM161077v1_genomic
## 1                            1_1415                1_1310,1_1354,1_1279
## 2                     1_1100,1_1462                         1_259,1_978
## 3                            1_1484                       1_1475,1_1474
## 4                                 -                       1_1258,1_1444
## 5                                 -                1_1311,1_1355,1_1280
##   GCF_002007845_1_ASM200784v1_genomic GCF_000341655_1_ASM34165v1_genomic
## 1                              1_1417                             1_1348
## 2                              1_1106                        1_1016,1_85
## 3                              1_1464                             1_1418
## 4                              1_1396                             1_1295
## 5                              1_1418                             1_1349
##   GCF_000830925_1_ASM83092v1_genomic
## 1                      1_1382,1_1332
## 2                        1_1374,1_93
## 3                             1_1497
## 4                             1_1465
## 5                      1_1383,1_1333
```

The cell entries are the gene IDs as assigned by MiGA.

Convert ogs to a binary (presence/absence) matrix. Assign arbitrary row names to serve as the ortholog names. Shorten the genome names (row names) to the GenBank assembly accession IDs only.

```
ogs.bin <- ogs
ogs.bin[ogs.bin != "-"] <- 1
ogs.bin[ogs.bin == "-"] <- 0
ogs.bin[1:5, 1:5]
```

```
##   GCF_000011905_1_ASM1190v1_genomic GCF_001610775_1_ASM161077v1_genomic
## 1                                 1                                   1
## 2                                 1                                   1
## 3                                 1                                   1
## 4                                 0                                   1
## 5                                 0                                   1
##   GCF_002007845_1_ASM200784v1_genomic GCF_000341655_1_ASM34165v1_genomic
## 1                                   1                                  1
## 2                                   1                                  1
## 3                                   1                                  1
## 4                                   1                                  1
## 5                                   1                                  1
##   GCF_000830925_1_ASM83092v1_genomic
## 1                                  1
## 2                                  1
## 3                                  1
## 4                                  1
## 5                                  1
```

```
rownames(ogs.bin) <- RDPutils::make_otu_names(1:nrow(ogs.bin))
rownames(ogs.bin) <- sub("OTU", "OG", rownames(ogs.bin))
colnames(ogs.bin) <- substring(colnames(ogs.bin), 1, 13)
ogs.bin[1:5, 1:4]
```

```
##         GCF_000011905 GCF_001610775 GCF_002007845 GCF_000341655
## OG_0001             1             1             1             1
## OG_0002             1             1             1             1
## OG_0003             1             1             1             1
## OG_0004             0             1             1             1
## OG_0005             0             1             1             1
```

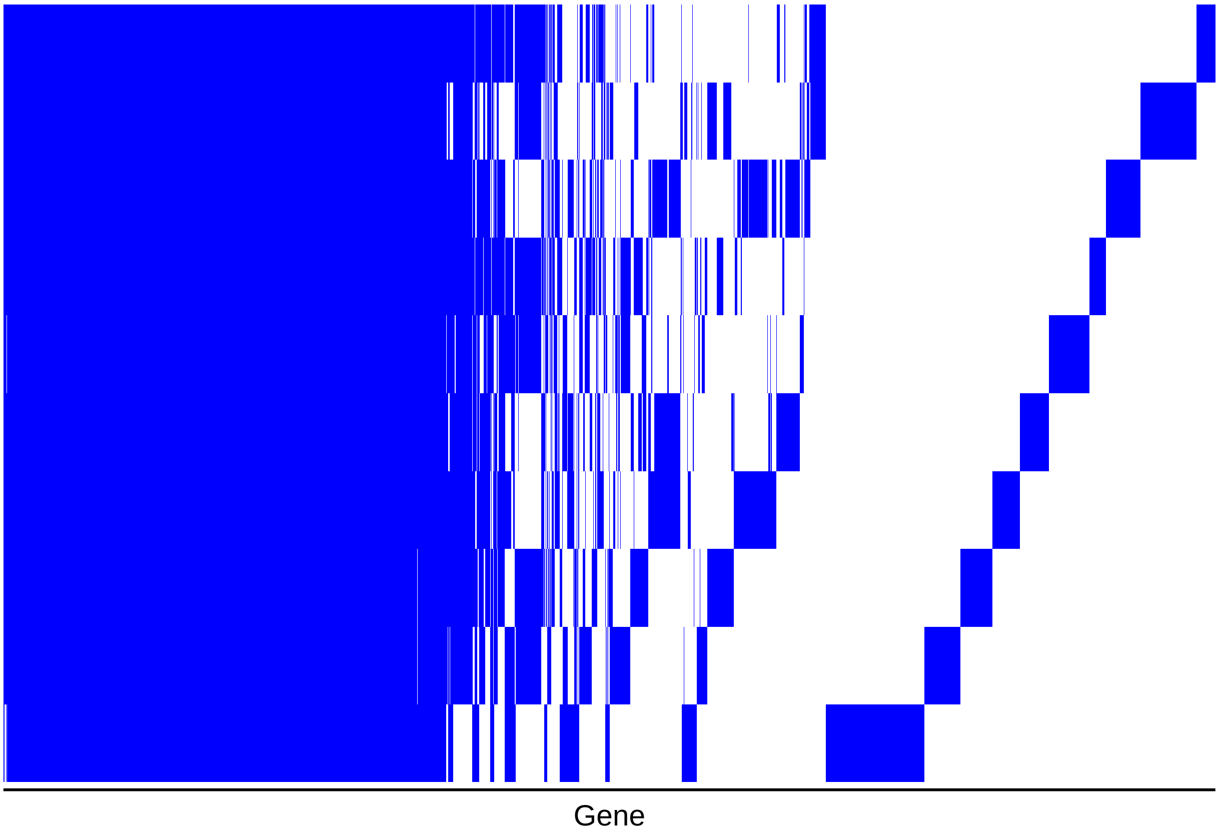
Convert to a tibble for making the heatmap plot.

```
n.cols <- ncol(ogs.bin) + 1
df.ogs <- ogs.bin %>% as.data.frame() %>% tibble::rownames_to_column(var = "Gene") %>%
    hablar::convert(int(colnames(.)[2:n.cols])) %>% hablar::convert(fct(Gene)) %>%
    tidyr::pivot_longer(-Gene, names_to = "Genome") %>% tibble::as_tibble()

df.ogs
```

```
## # A tibble: 29,180 x 3
##    Gene    Genome          value
##    <fct>   <chr>           <int>
##  1 OG_0001 GCF_000011905     1
##  2 OG_0001 GCF_001610775     1
##  3 OG_0001 GCF_002007845     1
##  4 OG_0001 GCF_000341655     1
##  5 OG_0001 GCF_000830925     1
##  6 OG_0001 GCF_001010485     1
##  7 OG_0001 GCF_004684285     1
##  8 OG_0001 GCF_000341695     1
##  9 OG_0001 GCF_001547795     1
## 10 OG_0001 GCF_001889305     1
## # ... with 29,170 more rows
```

Make the heatmap plot.

```
heatmap.plt <- ggplot(df.ogs, aes(x = Gene, y = Genome, fill = as.factor(value))) +
    geom_tile() + scale_fill_manual(values = c("white", "blue")) + theme(axis.text.x = element_blank(),
    axis.ticks.x = element_blank(), axis.line.x.bottom = element_line(size = 0.5),
    axis.text.y = element_blank(), axis.ticks.y = element_blank()) + theme(legend.position = "none") +
    ylab(NULL)

heatmap.plt
```



Trim the tree tip labels to match the genome names in `df.ogs` used to make the heatmap plot and root the tree. Tree tip labels and rows in the heatmap cannot be aligned with `aplot` unless the tree is rooted. A simple solution is to use `phanghorn`'s `midpoint` function.

```
ani.tree$tip.label <- substring(ani.tree$tip.label, 1, 13)
ani.tree <- phangorn::midpoint(ani.tree)
ani.tree
```

```
##
## Phylogenetic tree with 10 tips and 9 internal nodes.
##
## Tip labels:
##   GCF_004684285, GCF_001610775, GCF_002007845, GCF_001889305, GCF_000830925, GCF_001010485, ...
##
## Rooted; includes branch lengths.
```

Make a plot of the tree. Expand the plot so that the tip labels show.

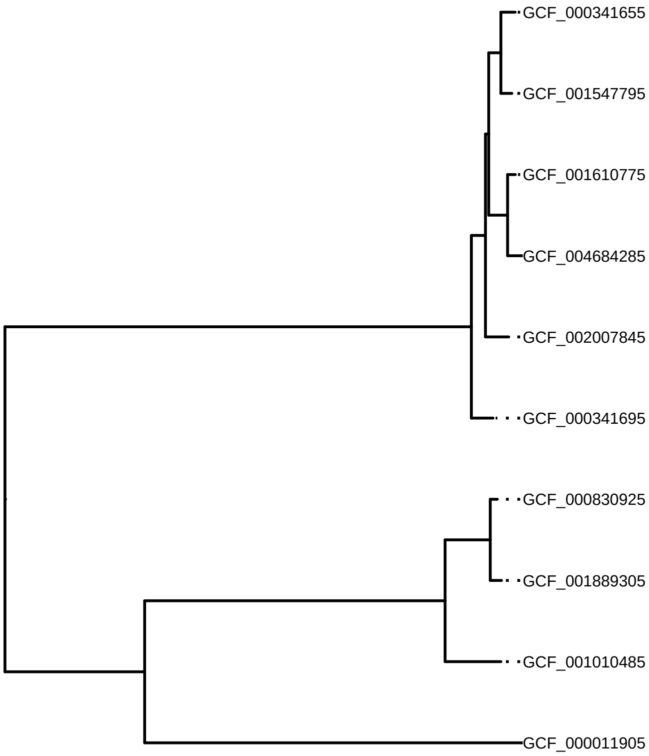
```
gg_tr <- ggtree(ani.tree) + geom_tiplab(align = TRUE, size = 2) + scale_x_continuous(expand = expansion
    0.05))
```

```
## Found more than one class "phylo" in cache; using the first, from namespace 'phyloseq'
```

```
## Also defined by 'tidytree'
```

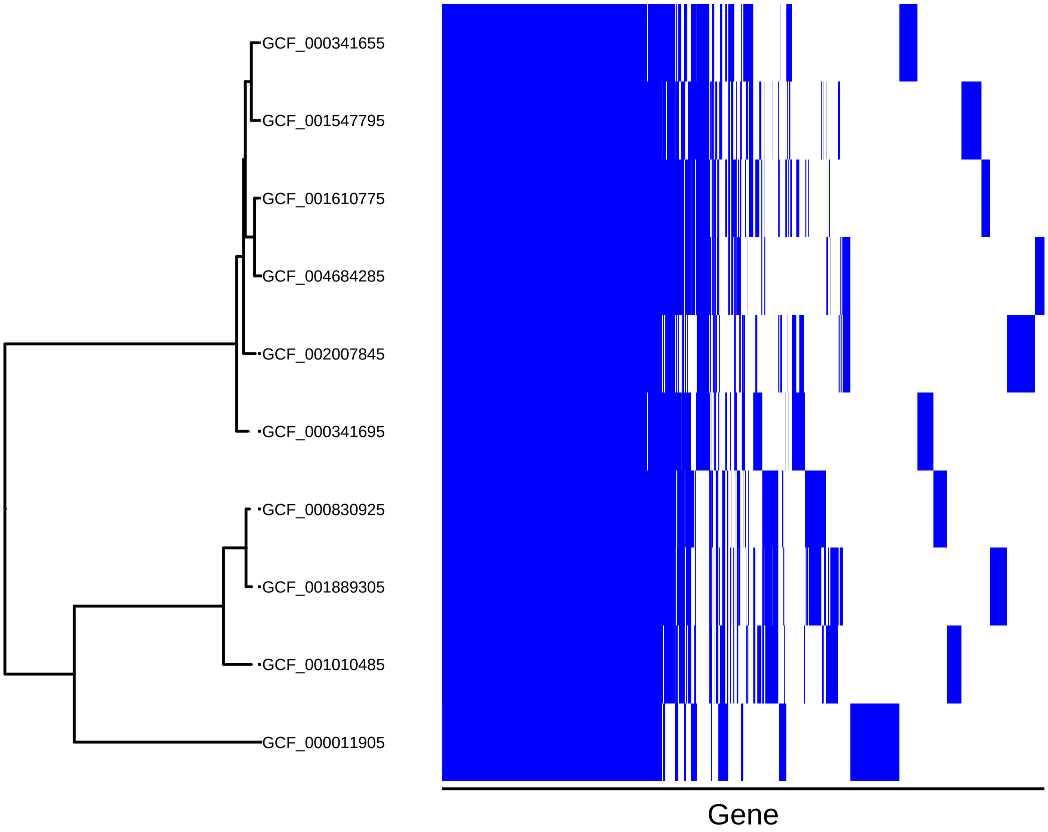
```
## Warning: Duplicated aesthetics after name standardisation: size
```

```
gg_tr
```



Align the plot by the tip labels, side-by-side. The `aplot` package automatically sizes the plots so that the tree tips and heatmap rows align. The plot can be saved with the `ggsave` function.

```
plt <- heatmap.plt %>% insert_left(gg_tr)
plt
```



```
ggsave(plt, file = "tree_heatmap.png", width = 7, height = 4)
```