MEDICAL MICROBIOLOGY AND INFECTIOUS DISEASES CODING WORKSHOP

Presents

Data visualization using ggtree

INSTRUCTED BY

Taylor Davedow (PhD student)

Email: davedowt@myumanitoba.ca

Educational series supported by the Department of MMID to promote knowledge translation and dissemination among students and researchers

INFORMATION FOR PARTICIPANTS

All workshops are being recorded and posted to the [MMID Coding Workshop - YouTube]

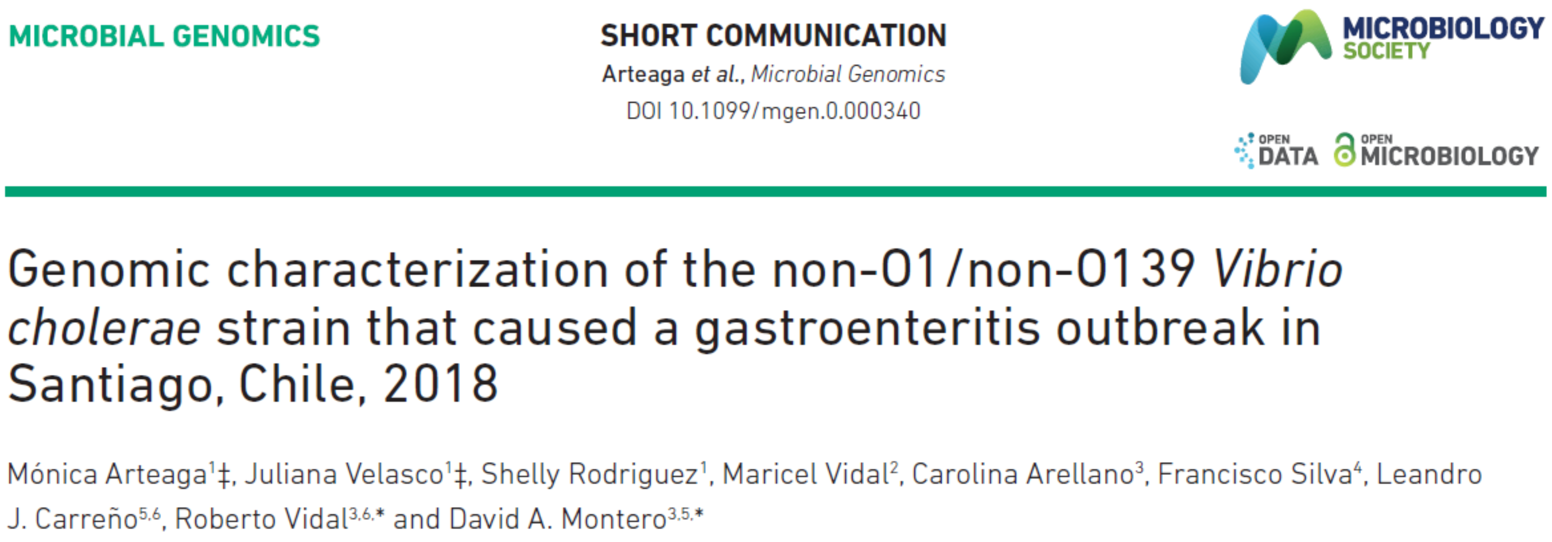
Question and Answer period will not be recorded.

LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Data for workshop

- Data set was **modified** from Arteaga et al. 2020. Microbial Genomics. https://doi.org/10.1099/mgen.0.000340
- Based on 26 samples (reduced from 70)



MICROBIAL GENOMICS

SHORT COMMUNICATION
Arteaga *et al.*, *Microbial Genomics*
DOI 10.1099/mgen.0.000340

MICROBIOLOGY SOCIETY

OPEN DATA OPEN MICROBIOLOGY

Genomic characterization of the non-O1/non-O139 *Vibrio cholerae* strain that caused a gastroenteritis outbreak in Santiago, Chile, 2018

Mónica Arteaga[1]‡, Juliana Velasco[1]‡, Shelly Rodriguez[1], Maricel Vidal[2], Carolina Arellano[3], Francisco Silva[4], Leandro J. Carreño[5,6], Roberto Vidal[3,6,]* and David A. Montero[3,5,]*

Data for workshop

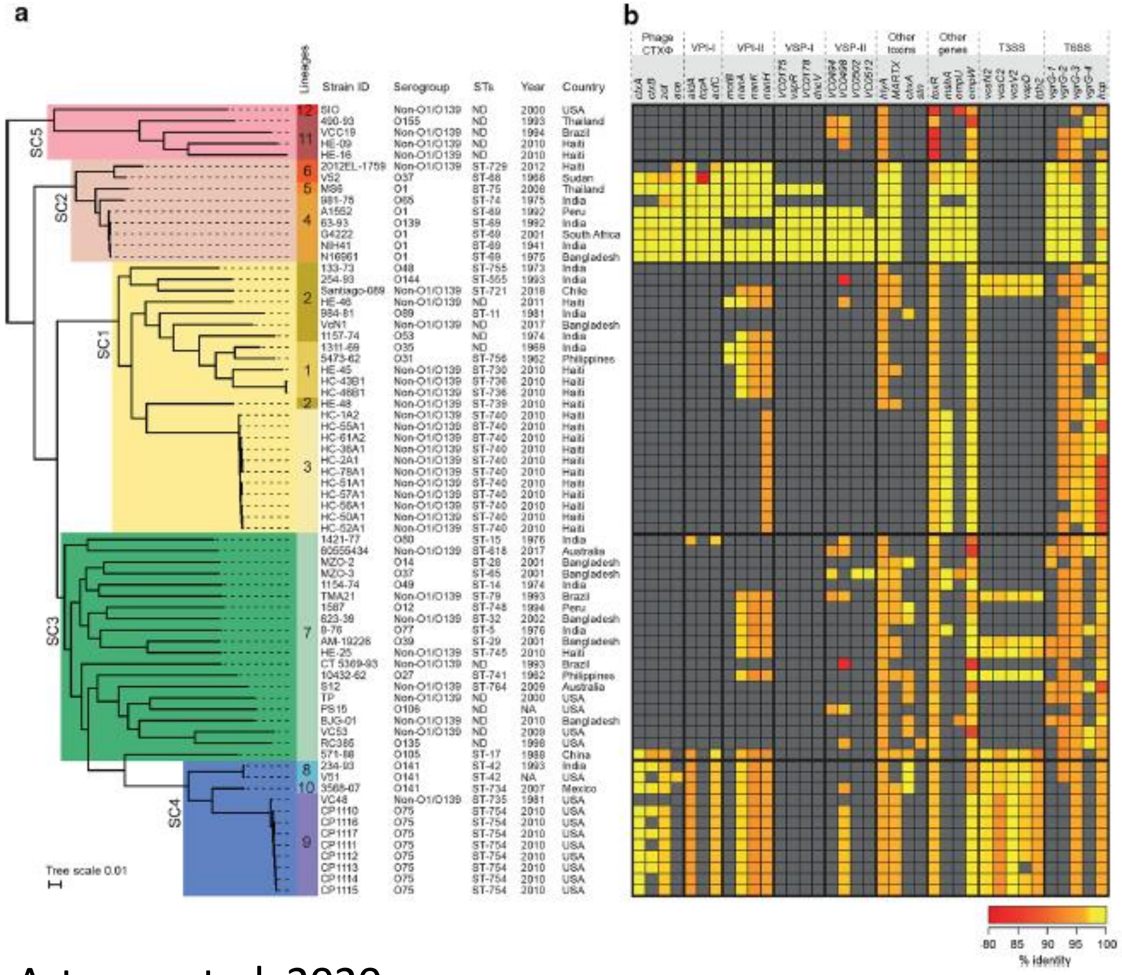
Data files we'll be using for today's workshop:

1. Tree file (newick)
   - Created by downloading genome accessions listed in manuscript and running SNVPhyl pipeline in Galaxy
2. Metadata (xlsx)
   - Information compiled from manuscript (strain ID, serogroup, year etc.)
3. Simulated BlastN (xlsx) 7 genes (reduced from 37)
   - Percent identity was produced using random number generator so will not reflect the results in the paper

Please **download** and **save** materials in your working directory
https://github.com/MMID-coding-workshop

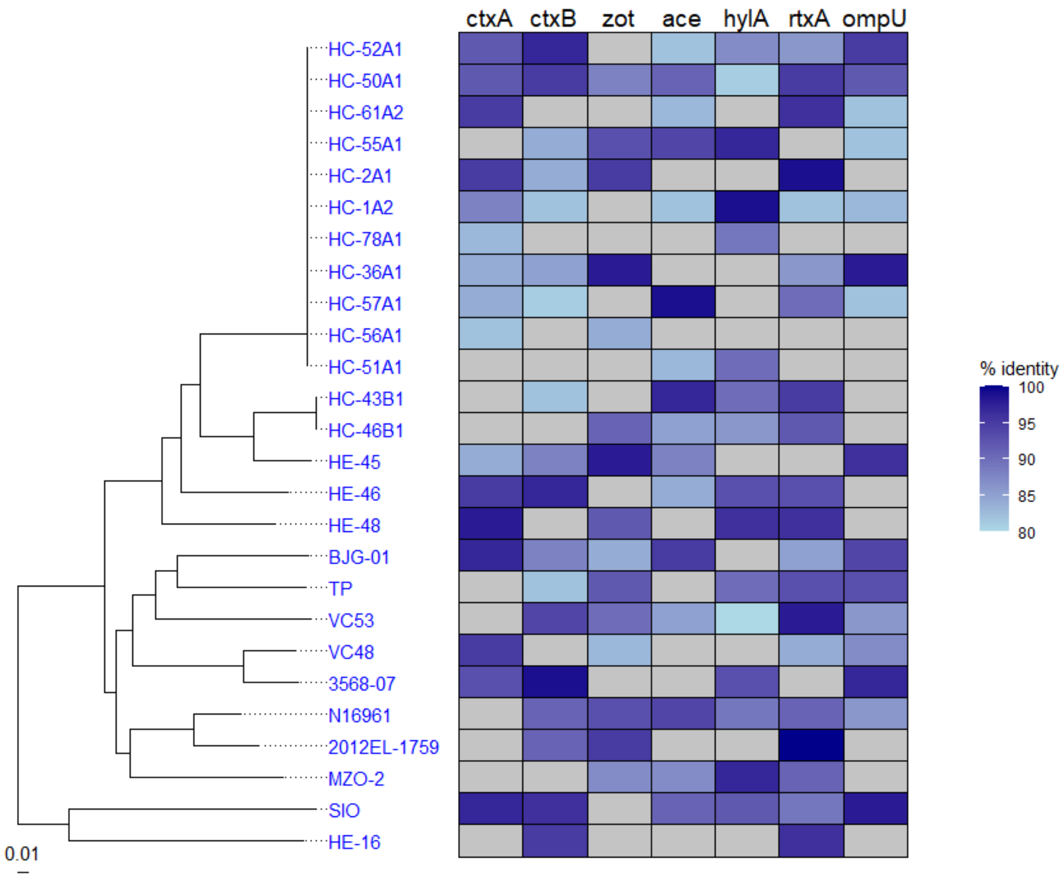
Preview

Original Figure



Arteaga et al. 2020

What we'll be creating!



ggtree

- Package for R programming language
- Under Bioconductor project
- Guangchuang Yu: https://yulab-smu.top/treedata-book/
- Data integration, manipulation and visualization of phylogenetic trees
- Customized annotation of tree

LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Packages for today's workshop

```
Readxl        # for reading xlsx files
BiocManager   # for installing ggtree
Treeio        # for read.newick function
Phytools      # for midpoint.root (also has read.newick)
Tidyverse     # tidying data
ggtree        # tree visualization & annotation
ggplot2       # for additional plotting support
```

Install packages

Install the following packages using install.packages function:

readxl, BiocManager, treeio, and tidyverse

```
> install.packages(”package name”)
```

Then install ggtree using BiocManager:

```
> BiocManager::install(”ggtree”)
```

Load packages

Load the packages using:

```
> library(”package name”)
```

Note: package installation only has to be done once, but we must load our libraries each time we want to use them

LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. ***Load a newick tree and compliment files***
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Load in files

```
# tree file
tree <- read.newick(”data/sample_tree.newick”)

# metadata file
metadata <- read_xlsx(here(”data”,”metadata.xlsx”))

# blast results file
blast_raw <- read_xlsx(here(”data”,”blast_results.xlsx”))
```

LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. ***Learn how to create a basic tree***
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Creating a basic tree

To create a tree, use the ggtree() function:

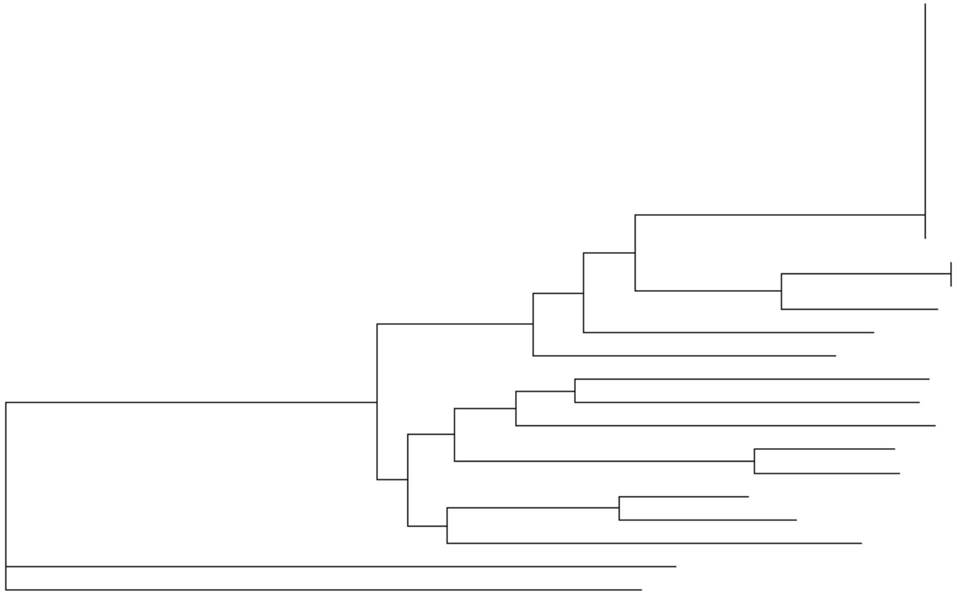
```
> ggtree(tr)
# or
> tr %>% ggtree()
```

- tr is the phylo object, so for us this would be:

```
> ggtree(tree)
```

Note: for a list of other arguments check out the ggtree help page

```
> ?ggtree
```



LEARNING OBJECTIVES

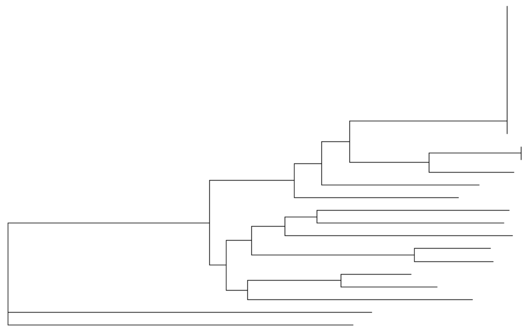
1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. ***Changing tree layout***
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Changing tree layout

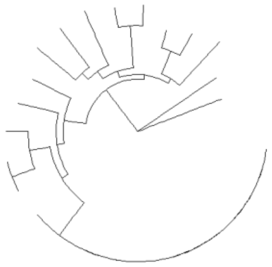
To change the layout in ggtree use the **layout** = "shape" argument:

```
> ggtree(tree, layout = ”rectangular”)
```

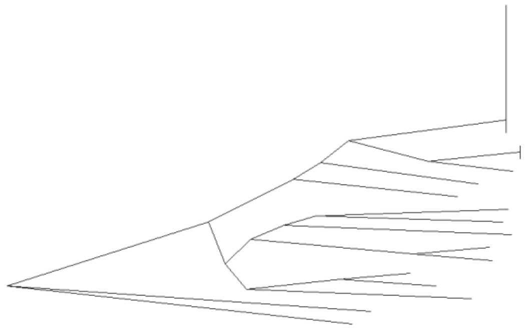
rectangular



circular



slanted



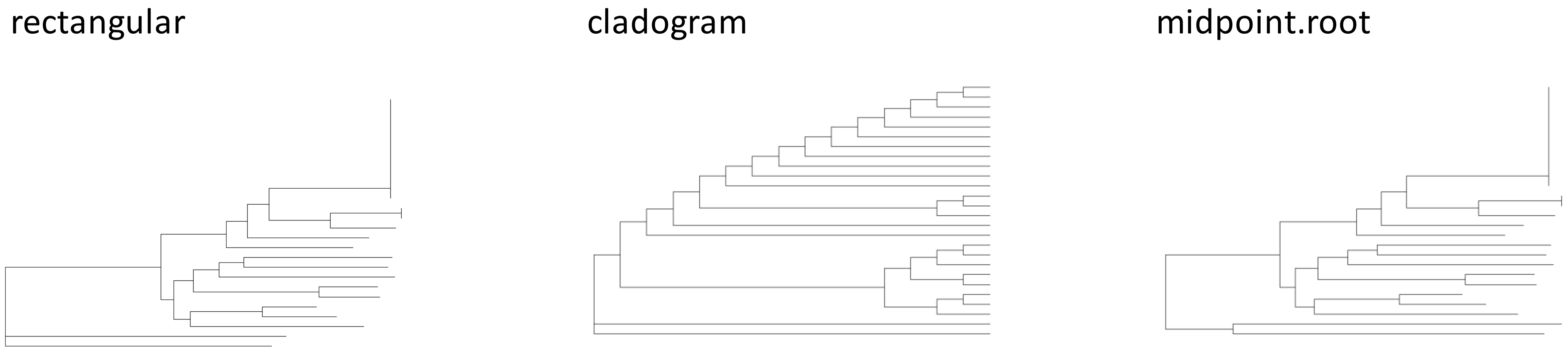
Changing tree layout

A **cladogram** will show topology without branch length information

```
> ggtree(tree, branch.length = ”none”)
```

Midpoint root: roots the tree at the midpoint of the longest point between two tips.

```
> ggtree(midpoint.root(tree))
```



rectangular

cladogram

midpoint.root

Try changing position of the root node using **root.position** argument

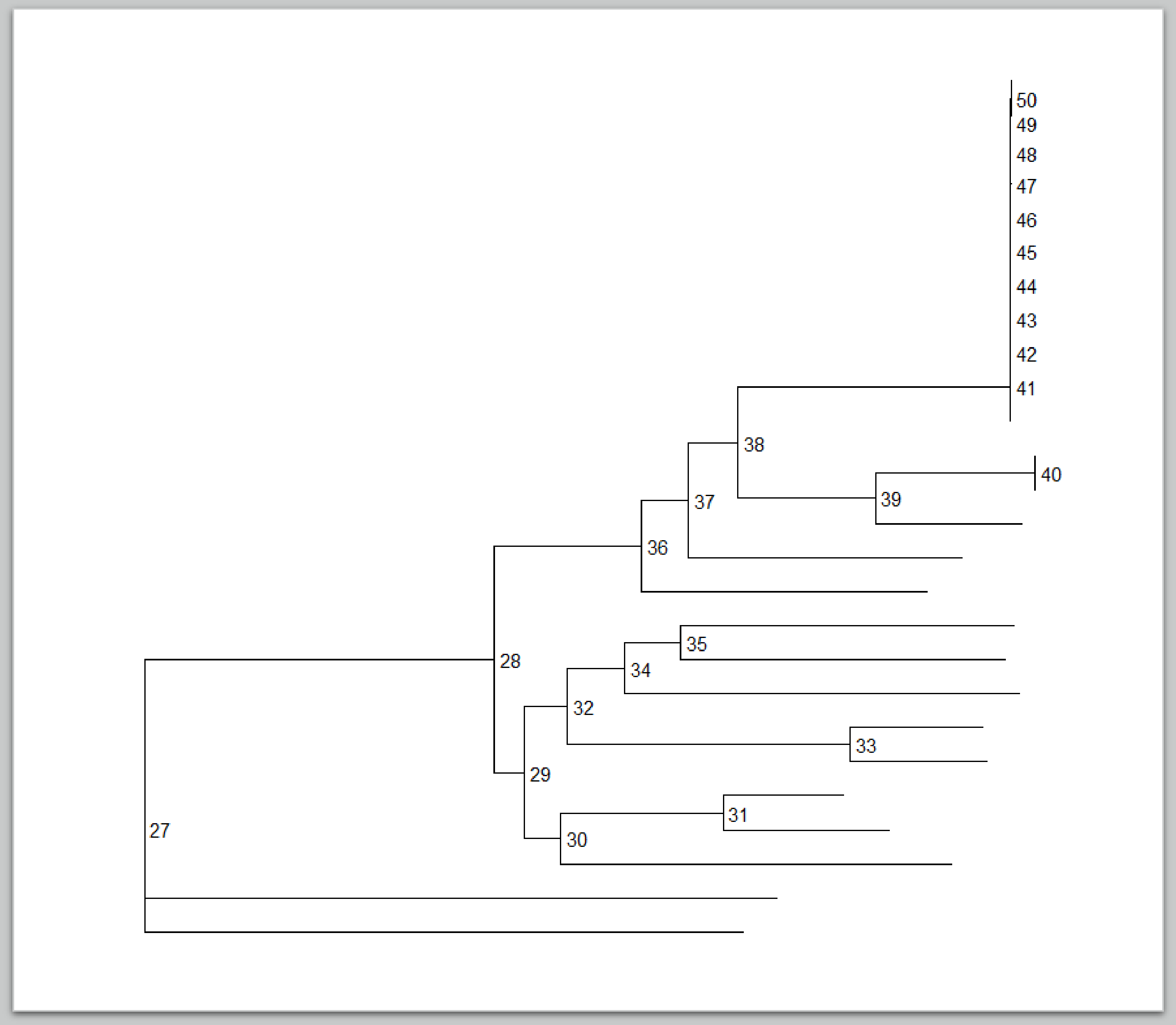
**Caution: rooting *can* drastically change tree topology and you must use an appropriate method

Identify nodes

First, we need to identify the nodes

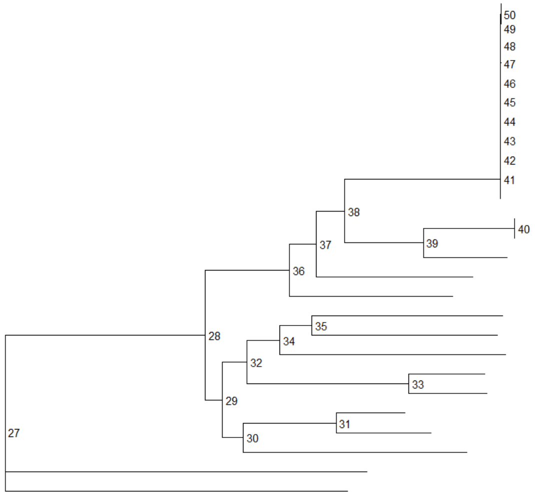
We will do this using geom_text2

```
> ggtree(tree) +
  geom_text2(aes(subset=!isTip,
                 label=node),
             hjust = -.3)
```

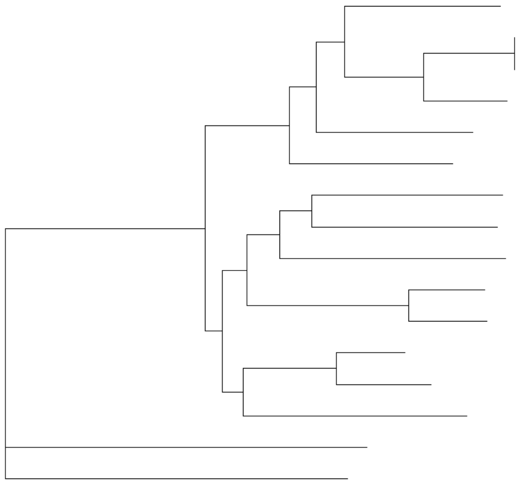


Tree manipulation

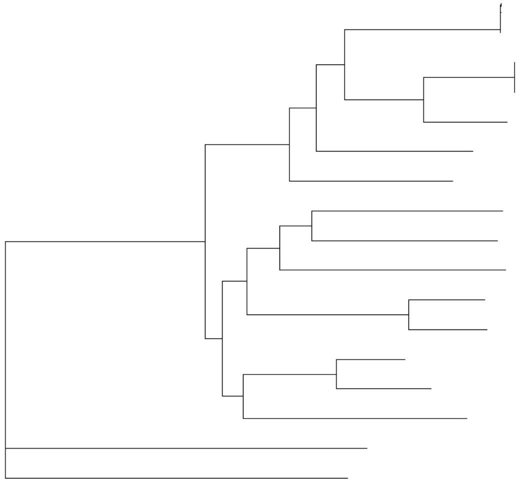
Original tree with node labels



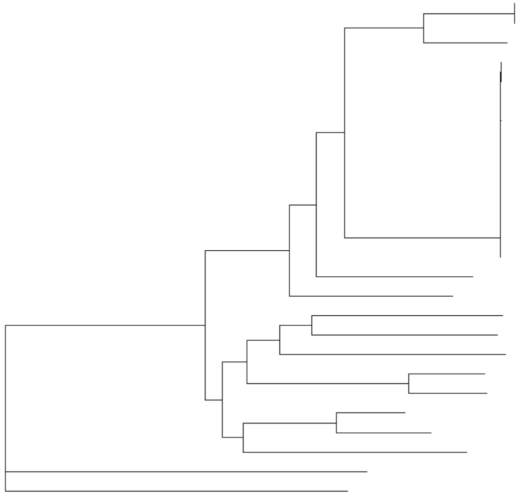
```
> ggtree(tree) %>%
  collapse(node = 41)
```



```
> ggtree(tree) %>%
  scaleClade(node = 41,
             scale = 0.1)
```

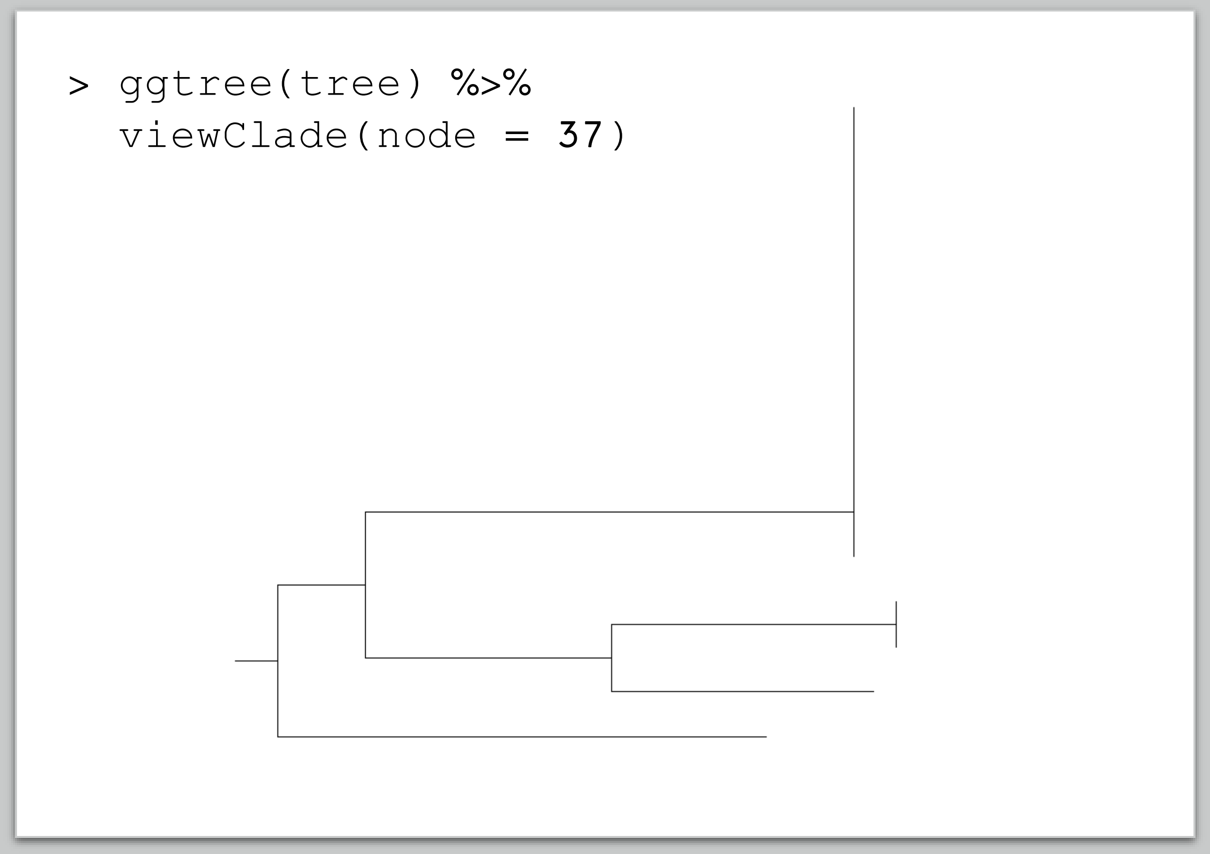


```
> ggtree(tree) %>%
  flip(39,41)
```

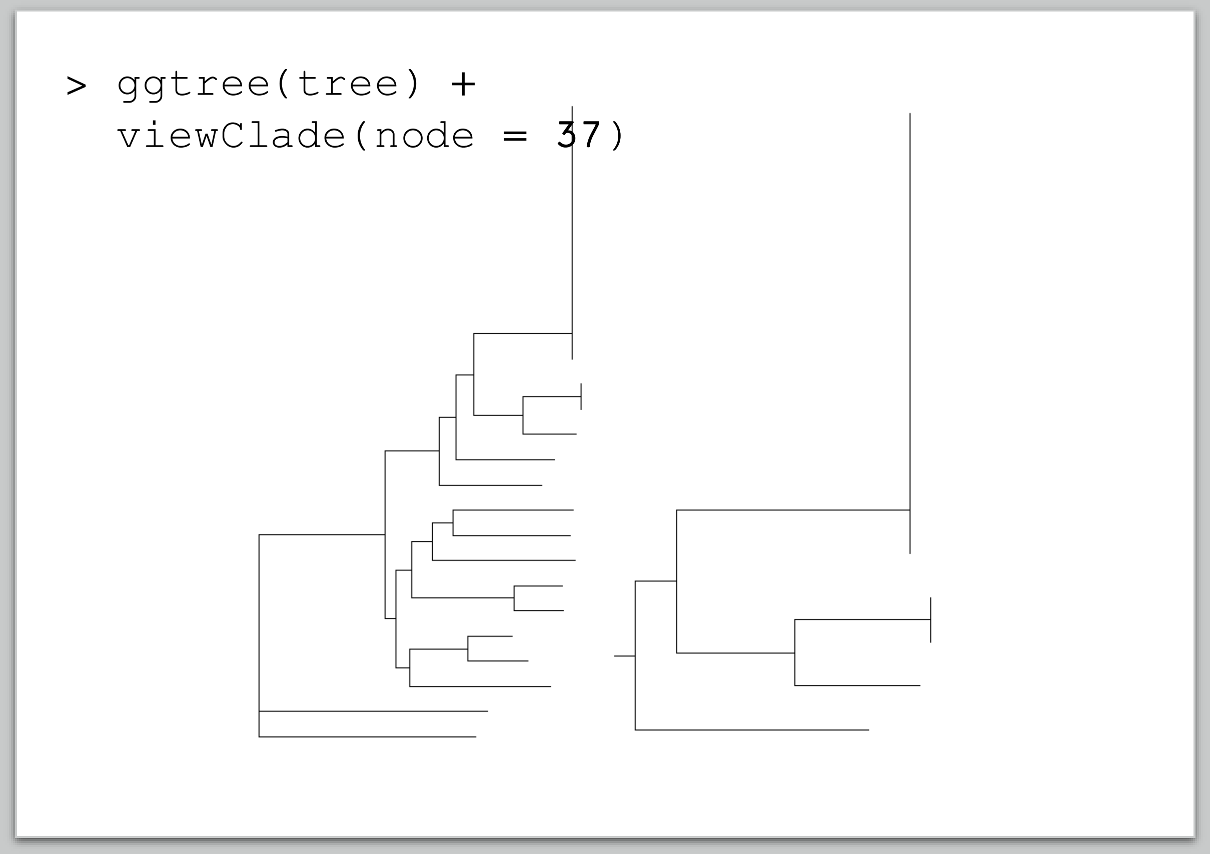


Tree manipulation

- To view a particular clade, we can use viewClade()
- Notice the difference between operators
  - +		Plus
  - %>%	Pipe



```
> ggtree(tree) %>%
  viewClade(node = 37)
```



```
> ggtree(tree) +
  viewClade(node = 37)
```

LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. ***Adding and customizing tip labels***
6. *How to merge your tree with a heatmap*
7. *How to export the final tree*

Adding tip labels

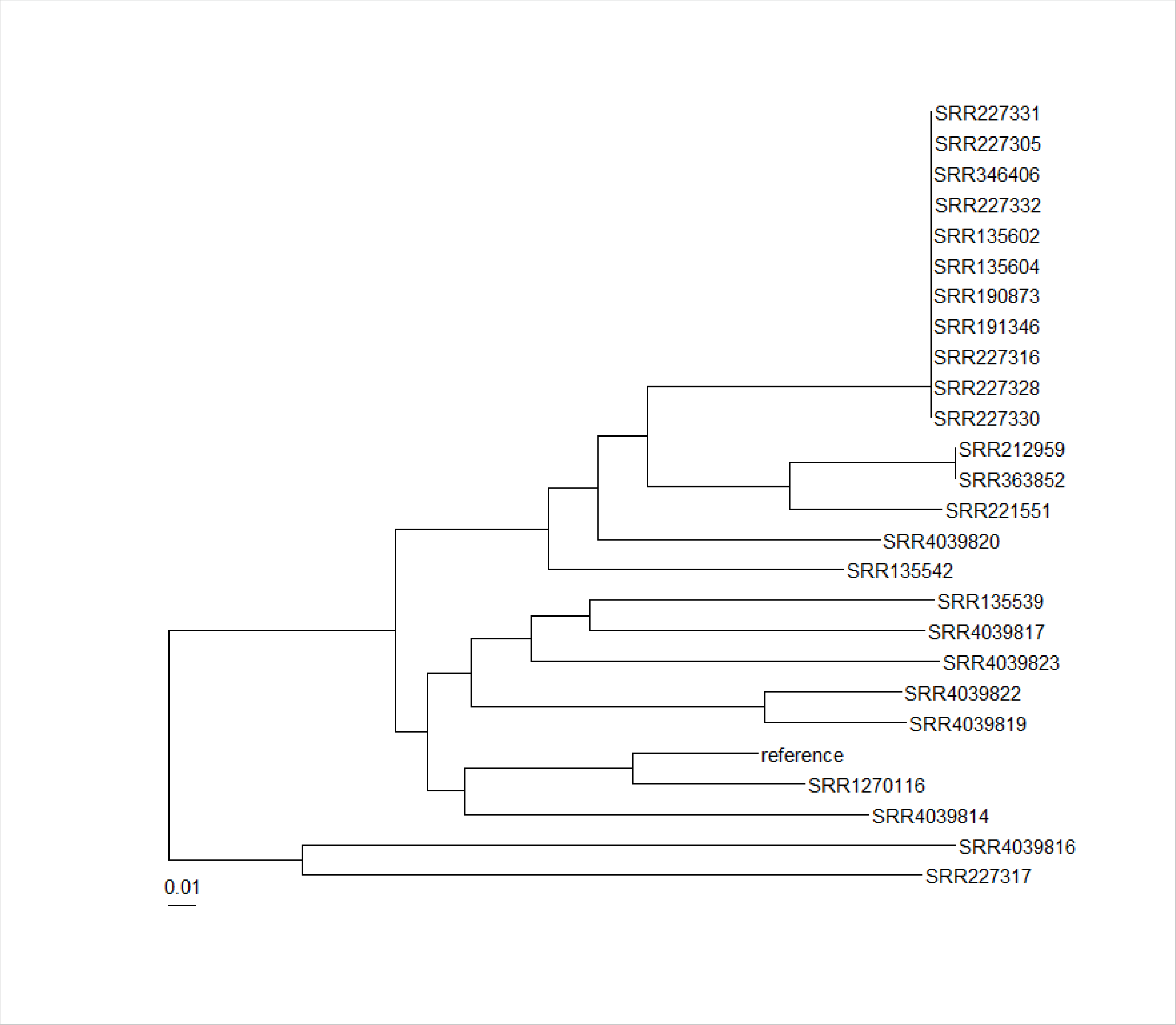
Check tip labels:

```
> head(tree$tip.label)
[1] "SRR227317"  "SRR4039816" "SRR4039814" "SRR1270116" "reference"  "SRR4039819"
```

```
> ggtree(midpoint.root(tree)) +
  geom_treescale(x = 0, y = 0, # x and y position of the tree scale
                 width = 0.01) + # width of scale
  geom_tiplab(size = 4) + # displaying tip labels (size four)
  coord_cartesian(clip = ’off’)+ # turning off the plot limits
  theme(plot.margin = margin(1,2,1,1, ”cm”)) # add margin
```

Adding tip labels

```
> ggtree(midpoint.root(tree)) +
  geom_treescale(x = 0, y = 0,
                       width = 0.01) +
   geom_tiplab(size = 4) +
  coord_cartesian(clip = ’off’)+
  theme(plot.margin =
  margin(1,2,1,1, ”cm”))
```

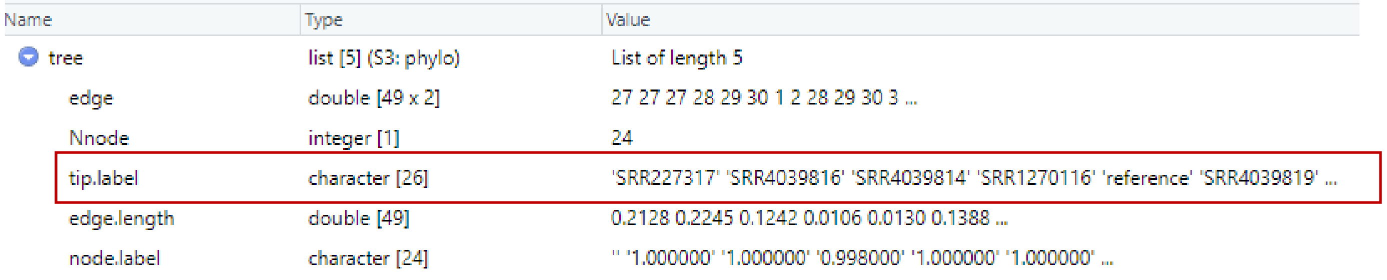


Customizing tip labels

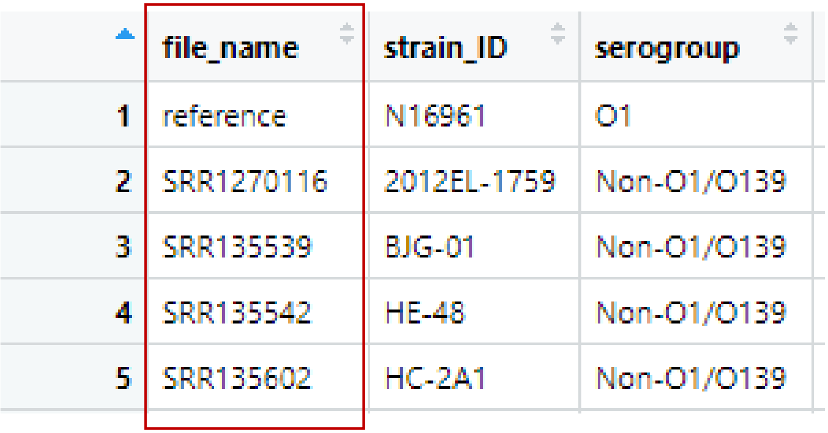
Since we will be linking the metadata file to the tree, we can use a vector to check if there are any file_name observations that are not in the tree

```
> metadata$file_name[!tree$tip.label %in% metadata$file_name]
[1]  character(0) # all 26 observations match b/w tree and metadata file
```

We can also view the data files from our environment:



Name	Type	Value
tree	list [5] (S3: phylo)	List of length 5
edge	double [49 x 2]	27 27 27 28 29 30 1 2 28 29 30 3 ...
Nnode	integer [1]	24
tip.label	character [26]	'SRR227317' 'SRR4039816' 'SRR4039814' 'SRR1270116' 'reference' 'SRR4039819' ...
edge.length	double [49]	0.2128 0.2245 0.1242 0.0106 0.0130 0.1388 ...
node.label	character [24]	'' '1.000000' '1.000000' '0.998000' '1.000000' '1.000000' ...



	file_name	strain_ID	serogroup
1	reference	N16961	O1
2	SRR1270116	2012EL-1759	Non-O1/O139
3	SRR135539	BJG-01	Non-O1/O139
4	SRR135542	HE-48	Non-O1/O139
5	SRR135602	HC-2A1	Non-O1/O139

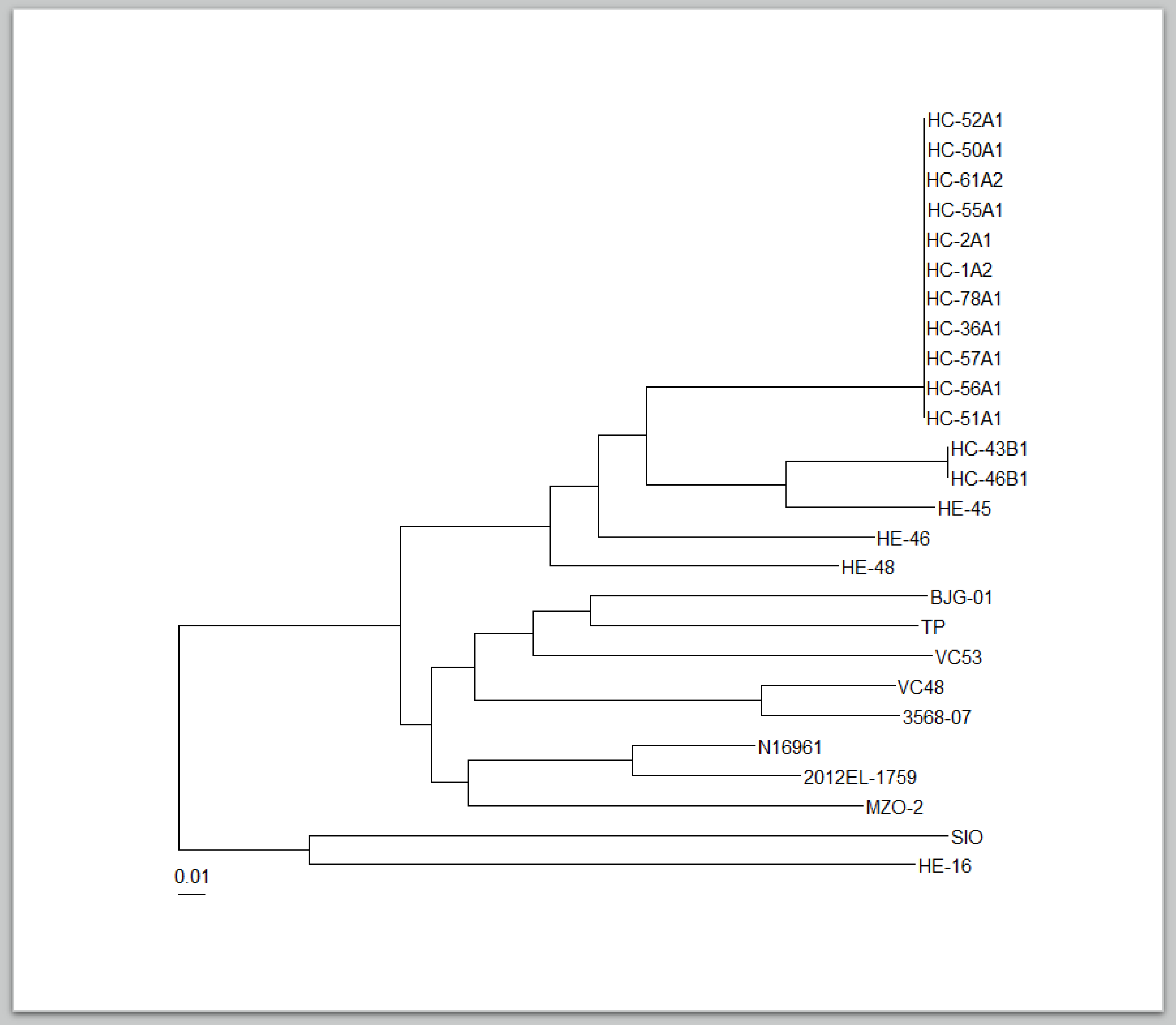
Any data we want linked to the tree must have a column that **EXACTLY matches** the tip.label

Customizing tip labels

```
> ggtree(midpoint.root(tree)) %<+% # operator to attach annotation data
  to tree
  metadata + # link our metadata file here
  geom_treescale(x = 0, y = 0,
                 width = 0.01)+
  coord_cartesian(clip = ’off’)+
  theme(plot.margin = margin(1,2,1,1, ”cm”)) +
  geom_tiplab(aes(label = strain_ID)) # change the tip label to strain_ID
```

Customizing tip labels

```
> ggtree(midpoint.root(tree)) %<+%
  metadata +
  geom_treescale(x = 0, y = 0,
                     width = 0.01)+
  coord_cartesian(clip = ’off’)+
  theme(plot.margin =
  margin(1,2,1,1, ”cm”)) +
  geom_tiplab(aes(label
  = strain_ID))
```

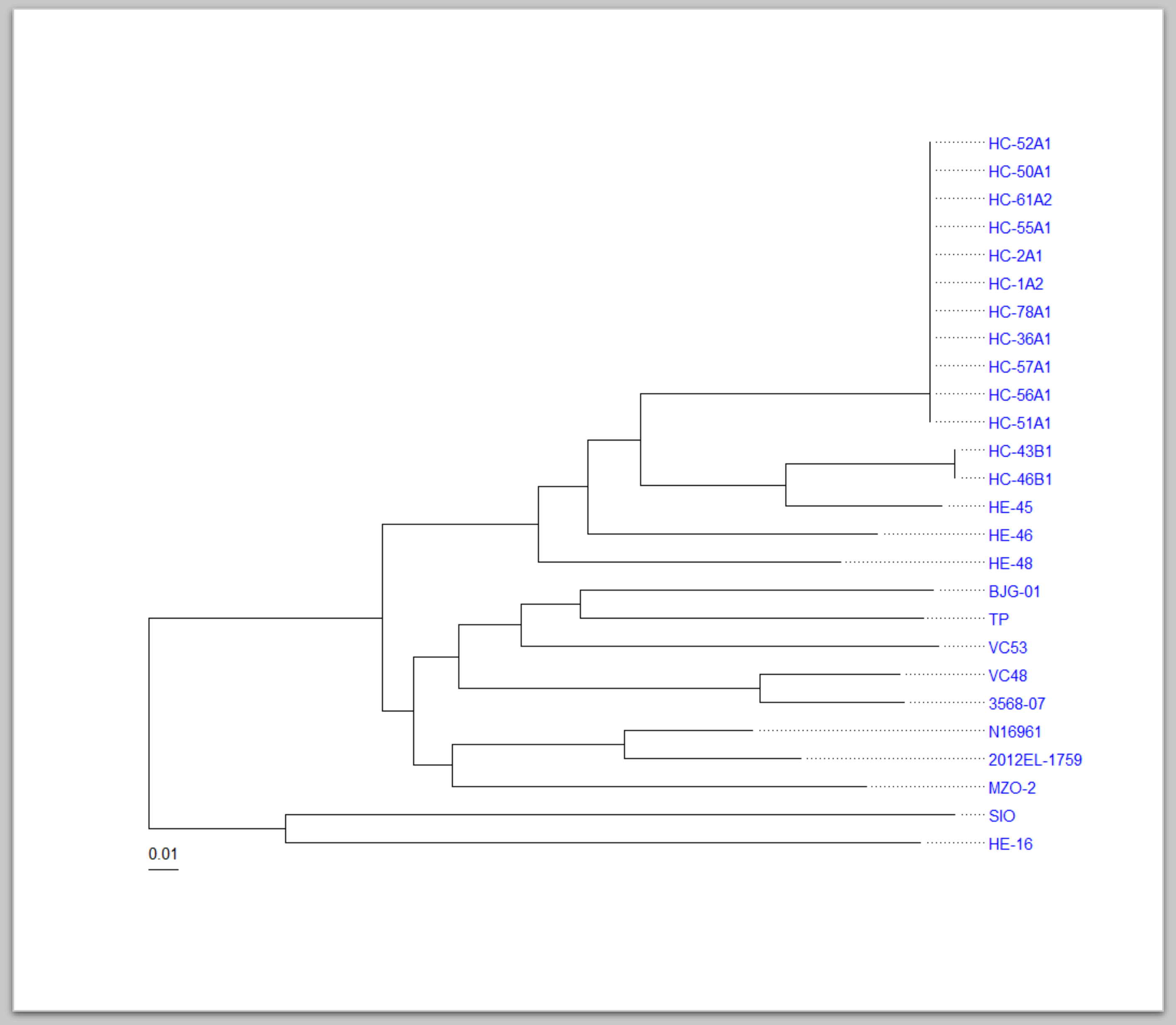


Customizing tip labels

```
> ggtree(midpoint.root(tree)) %<+%
  metadata +
  geom_treescale(x = 0, y = 0,
                 width = 0.01)+
  coord_cartesian(clip = ’off’)+
  theme(plot.margin = margin(1,2,1,1, ”cm”)) +
  geom_tiplab(aes(label = strain_ID),
            color = ”blue”, # changing font color
            size = 4,  # changing font size
            offset = 0.01, # horizontal adjustment of tip labels
            align = TRUE) # this creates a dotted leader line
```

Customizing tip labels

```
> gg_simple <-
   ggtree(midpoint.root(tree)) %<+%

  metadata +

  geom_treescale(x = 0, y = 0,

                 width = 0.01)+

  coord_cartesian(clip = 'off')+

  theme(plot.margin = margin(1,2,1,1,
   "cm")) +

  geom_tiplab(aes(label = strain_ID),

             color = "blue",

             size = 4,

             offset = 0.01,

             align = TRUE)
```



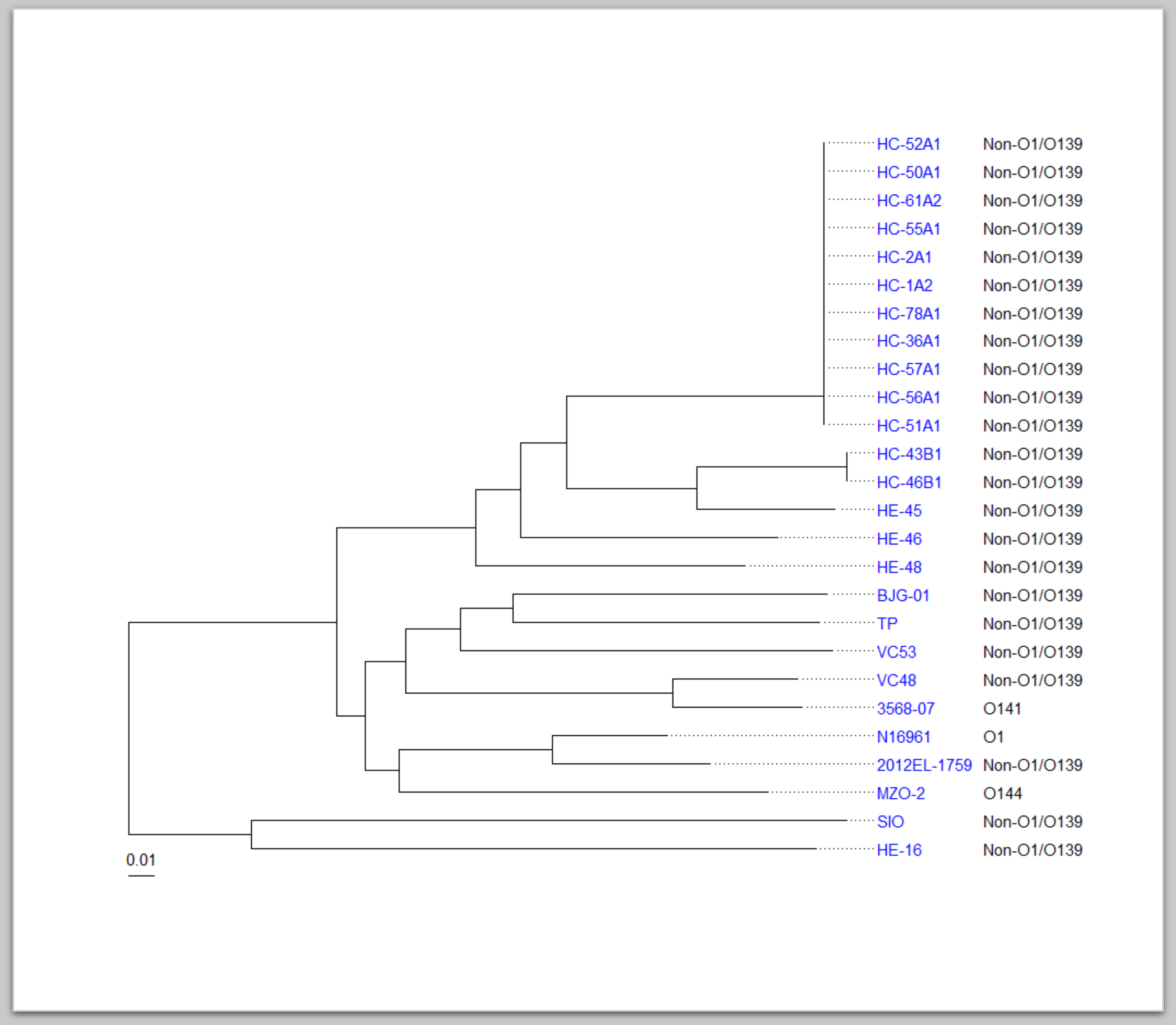
Adding another layer of tip labels

```
> gg_simple +
  geom_tiplab(aes(label = serogroup), # add in serogroup information
            color = ”black”,
            offset = 0.05, # horizontal adjustment so the tiplabs don’t overlap
            size = 4,
            align = TRUE, # align tiplab
            linetype = NA) # remove the dotted line b/w strain_ID and serogroup
```

You can keep adding layer by layer by adding a new geom_tiplab, but will need to keep adjusting the offset

Adding another layer of tip labels

```
> gg_simple +
  geom_tiplab(aes(label =
   serogroup), # add in serogroup
   information
     color = ”black”,
     offset =.05,
     size = 4,
     align = TRUE,
     linetype = NA)
```



LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. ***How to merge your tree with a heatmap***
7. *How to export the final tree*

Merging tree with a heatmap

gheatmap(): a function in ggTree that joins a heatmap matrix and phylogenetic tree

```
> gheatmap(p,       # tree view
           data)    # matrix or data.frame
```

Merging tree with heatmap

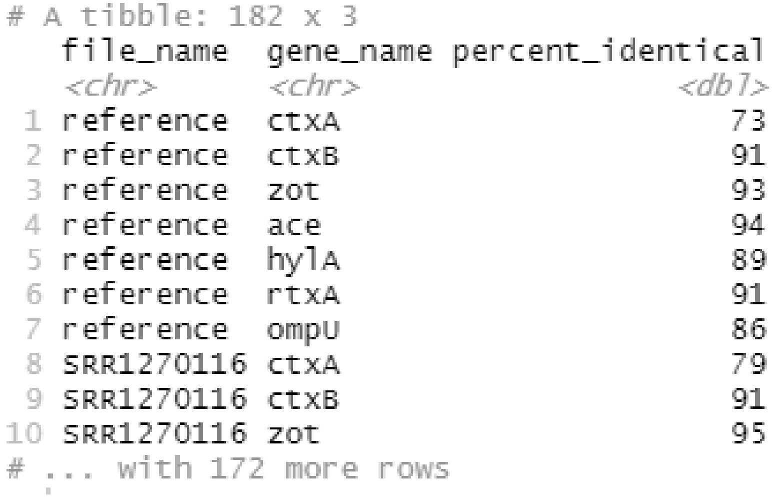
We need to prepare the blast file so it will align with tree

1. Tidy the blast file
2. Set row names to file_name
3. Select columns of interest

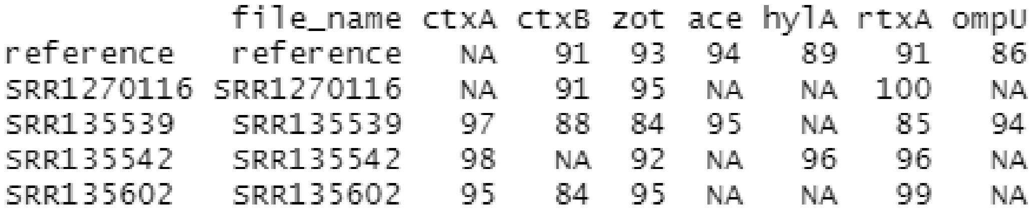
Merging tree with heatmap

1. Tidy blast file

```
blast_df <- blast_raw %>% filter(percent_identical >= 80) %>%
  pivot_wider(names_from = gene_name,
              values_from = percent_identical)%>%
  relocate(ctxA, .before = ctxB) %>% as.data.frame()
```



```
# A tibble: 182 x 3
   file_name  gene_name percent_identical
   <chr>      <chr>                 <dbl>
 1 reference  ctxA                     73
 2 reference  ctxB                     91
 3 reference  zot                      93
 4 reference  ace                      94
 5 reference  hylA                     89
 6 reference  rtxA                     91
 7 reference  ompU                     86
 8 SRR1270116 ctxA                     79
 9 SRR1270116 ctxB                     91
10 SRR1270116 zot                      95
# ... with 172 more rows
```



```
             file_name ctxA ctxB zot ace hylA rtxA ompU
reference    reference   NA   91  93  94   89   91   86
SRR1270116 SRR1270116   NA   91  95  NA   NA  100   NA
SRR135539    SRR135539   97   88  84  95   NA   85   94
SRR135542    SRR135542   98   NA  92  NA   96   96   NA
SRR135602    SRR135602   95   84  95  NA   NA   99   NA
```

Merging tree with heatmap

2. Set row names to file_name

```
# row names before
> rownames(blast_df)
[1] ”1”  ”2”  ”3”  ”4”  ”5” …

# set row names
> rownames(blast_df) <- blast_df$file_name

# row names after
> rownames(blast_df)
[1] ”reference”  ”SRR1270116” ”SRR135539” …
```

Merging tree with heatmap

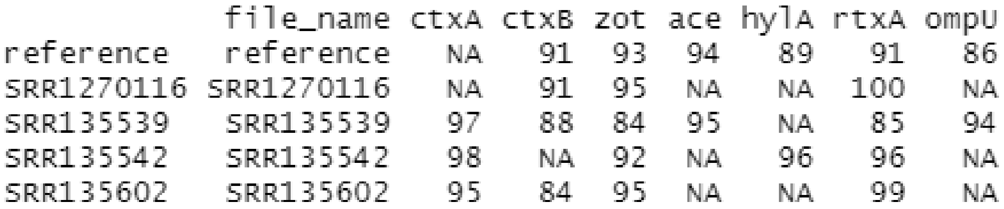
3. Select columns of interest

We can now get rid of the redundant file_name column

We are only interested in columns 2:8 with the genes

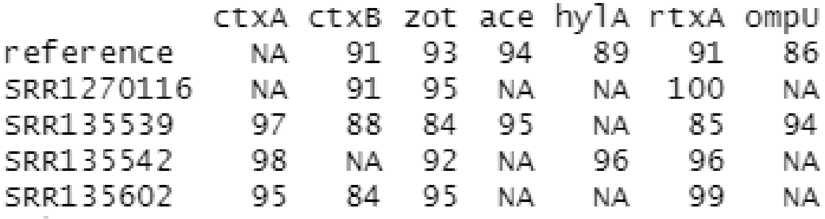
Double check before removing, using view() or:

```
> blast_df %>% head(5)
```



```
            file_name ctxA ctxB zot ace hylA rtxA ompU
reference   reference   NA   91  93  94   89   91   86
SRR1270116 SRR1270116   NA   91  95  NA   NA  100   NA
SRR135539   SRR135539   97   88  84  95   NA   85   94
SRR135542   SRR135542   98   NA  92  NA   96   96   NA
SRR135602   SRR135602   95   84  95  NA   NA   99   NA
```

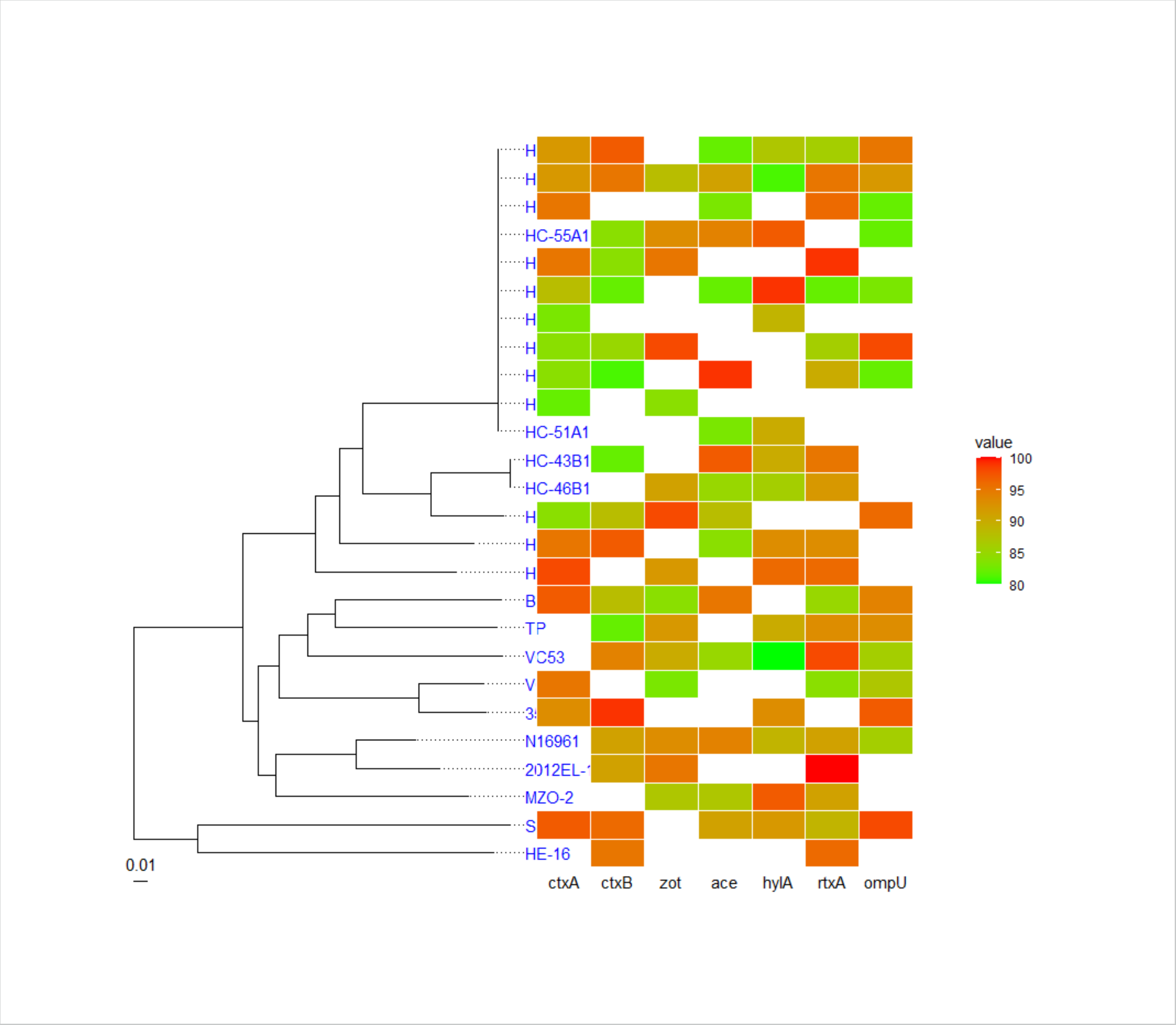
```
Select columns of interest (! means not)

> blast_df <- select(blast_df, !file_name)

> blast_df %>% head(5)
```



```
           ctxA ctxB zot ace hylA rtxA ompU
reference    NA   91  93  94   89   91   86
SRR1270116   NA   91  95  NA   NA  100   NA
SRR135539    97   88  84  95   NA   85   94
SRR135542    98   NA  92  NA   96   96   NA
SRR135602    95   84  95  NA   NA   99   NA
```

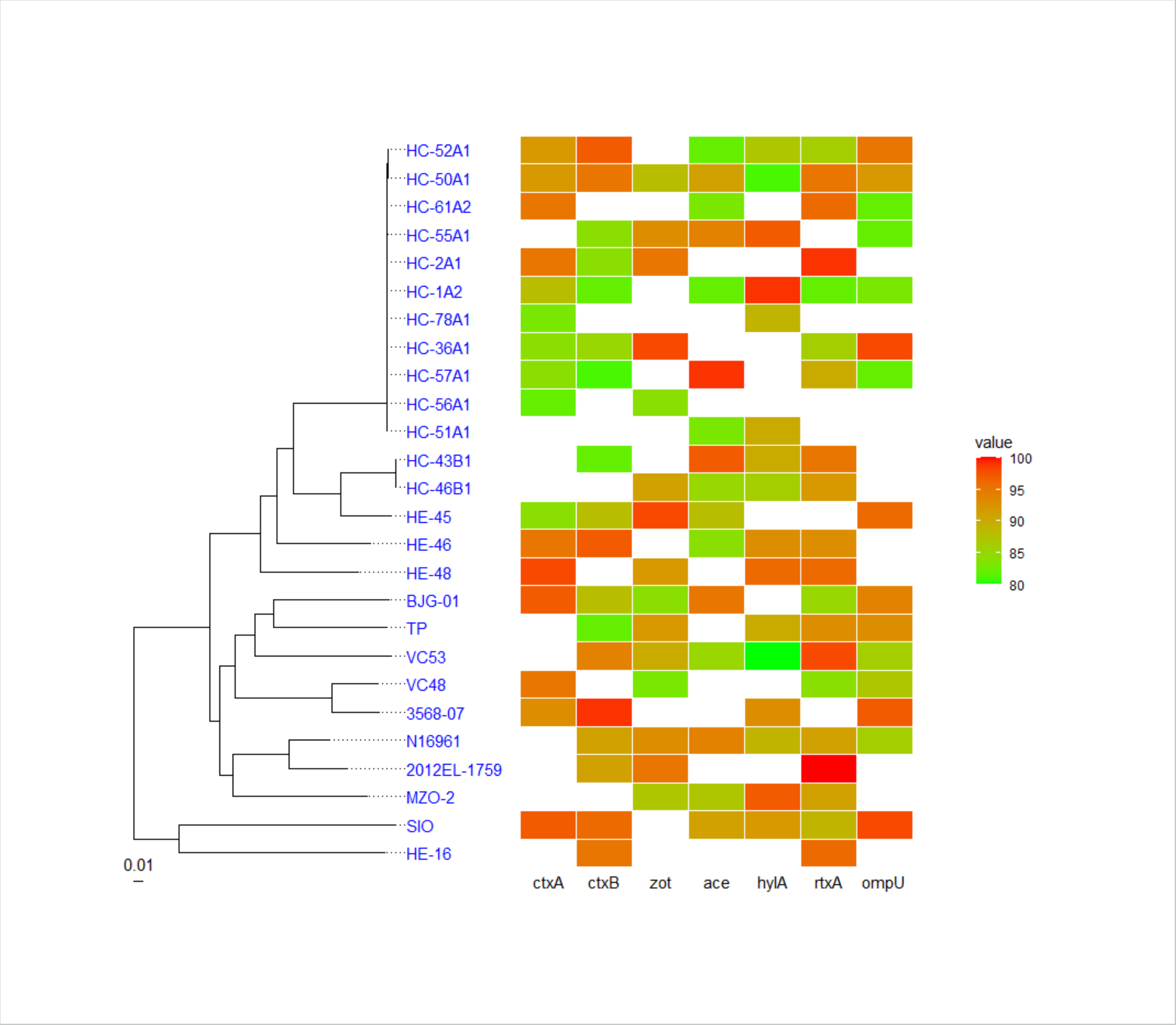
Merging tree with heatmap

```
> gheatmap(gg_simple, # tree
           blast_df)   # heatmap
```



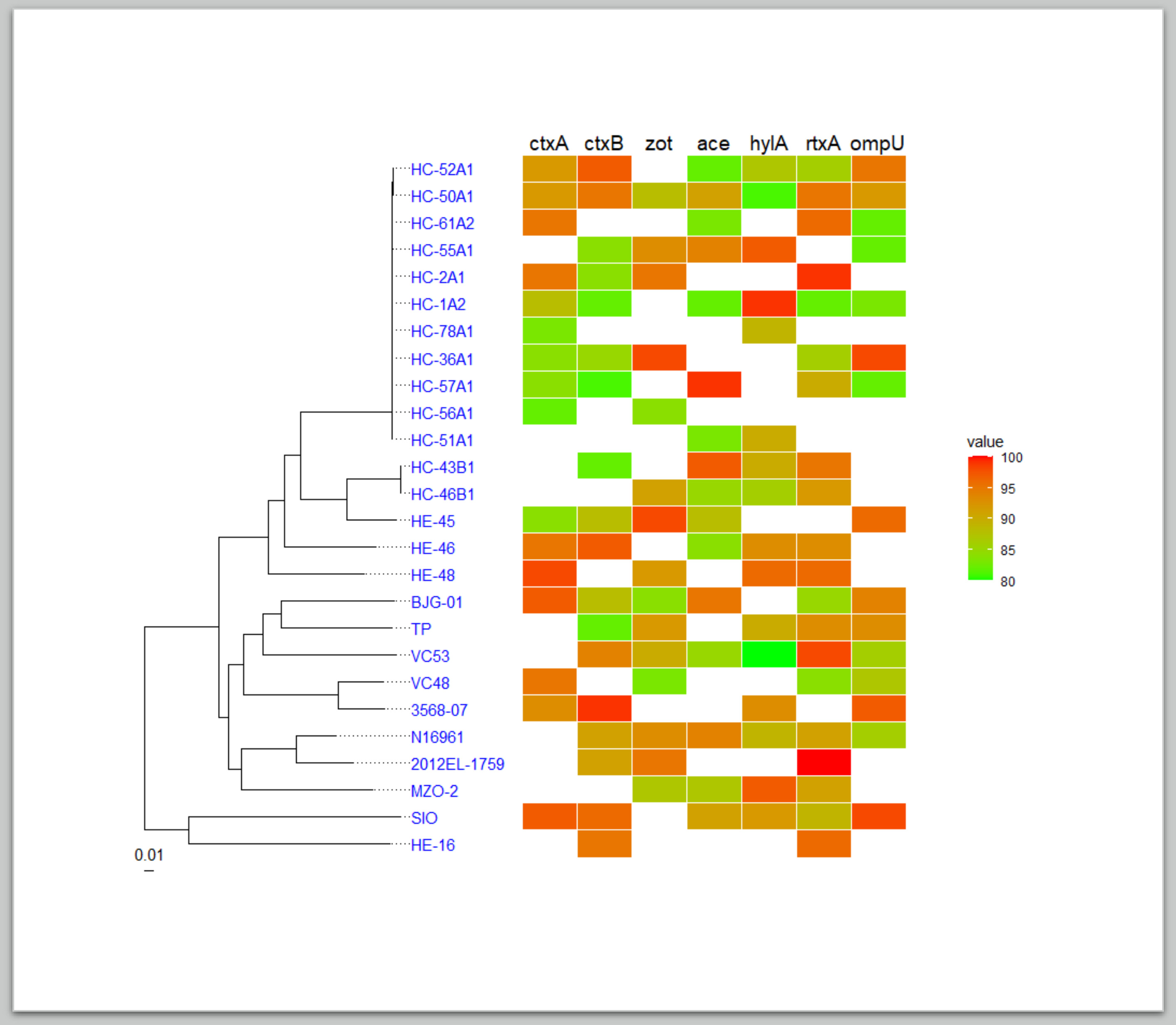
Merging tree with heatmap

```
> gheatmap(gg_simple,
      blast_df,
      offset = 0.1, # offset distance
      width = 1.5) # width of heatmap
```



Merging tree with heatmap

```
> gheatmap(gg_simple,
    blast_df,
    offset = 0.1,
    width = 1.5,
    font.size = 5,
    colnames_position = ”top”)
```



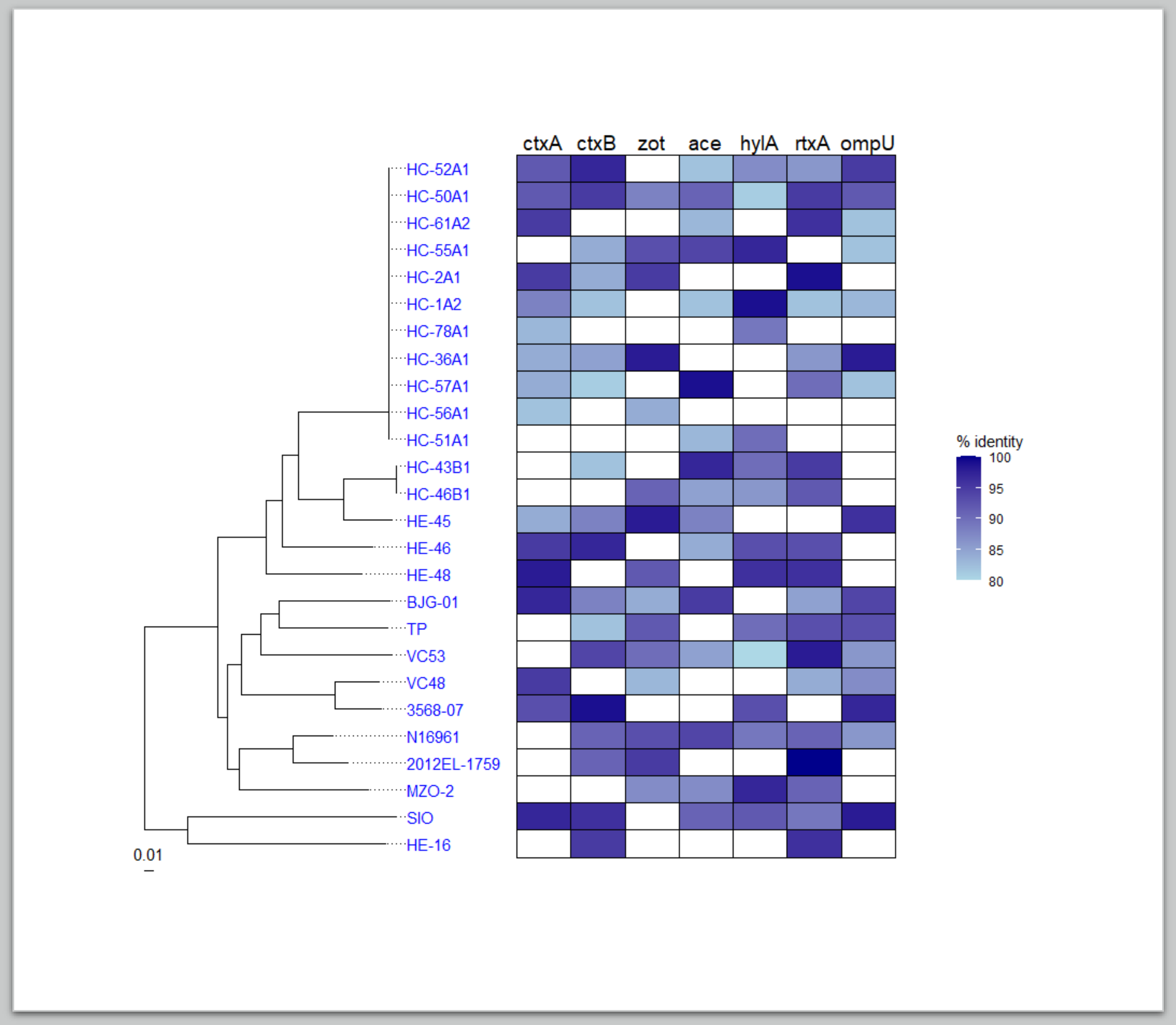
Changing heatmap colors: Option 1

You can change the heatmap colors directly in **gheatmap** using arguments color, low and high

```
> gheatmap(gg_simple,
           blast_df,
           offset = 0.1,
           width = 1.5,
           font.size = 5,
           colnames_position = ”top”,
           color = ”black”, # color of cell border
           legend_title = ”% identity”, # title of legend
           low = ”lightblue”, # color of lowest value
           high = ”darkblue”) # color of highest value
The NA values will remain white
```

Option 1

```
> gheatmap(gg_simple,
   blast_df,
   offset = 0.1,
   width = 1.5,
   font.size = 5,
   colnames_position = ”top”,
   color = ”black”,
   legend_title = ”% identity”,
   low = ”lightblue”,
   high = ”darkblue”)
```



Changing heatmap colors: Option 2

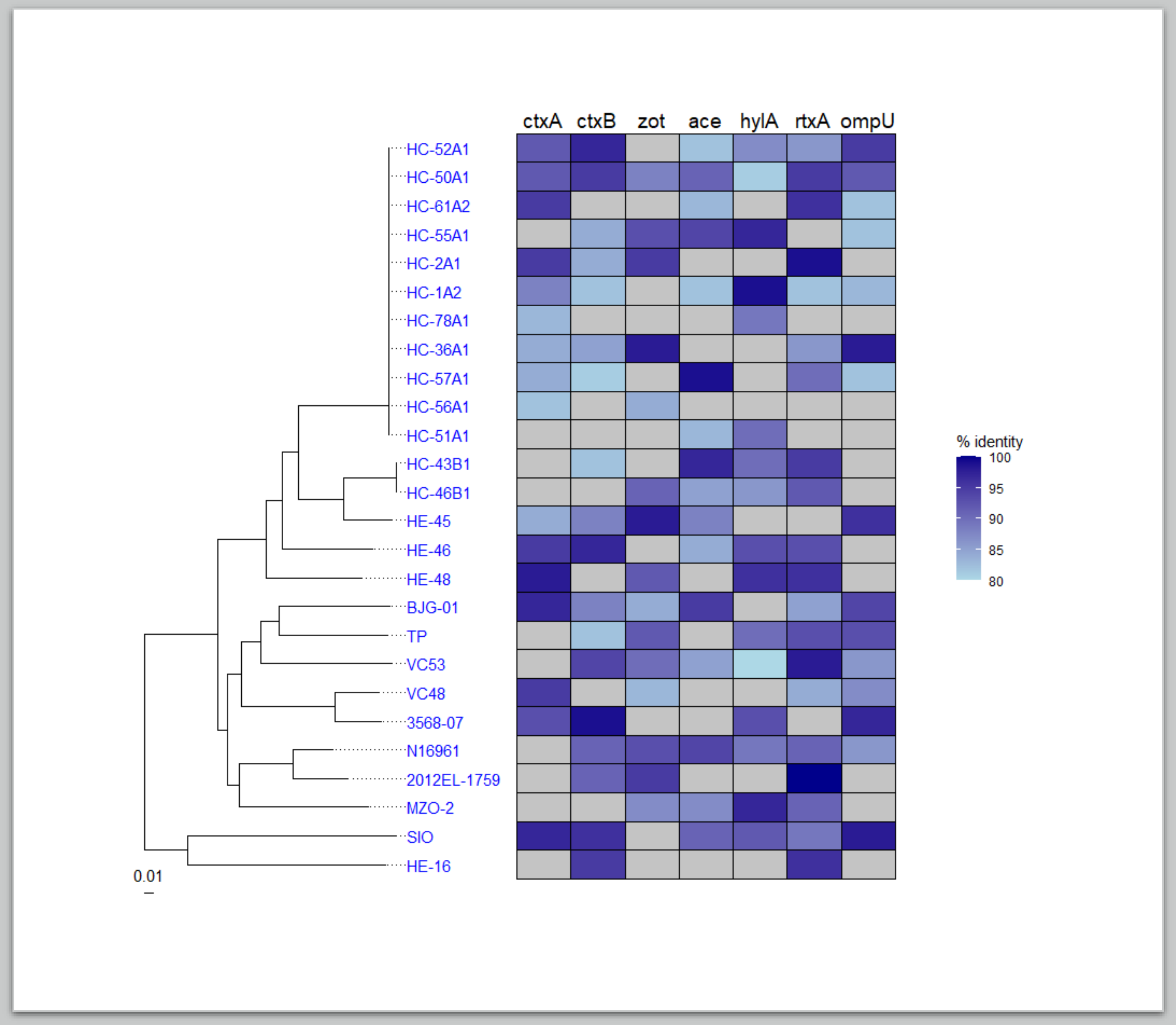
You can change the heatmap colors using **ggplot2**:

```
> gheatmap(gg_simple,
         blast_df,
         offset = 0.1,
         width = 1.5,
         font.size = 5,
         colnames_position = ”top”,
         color = ”black”) +
  scale_fill_gradient(name = ”% identity”, # title of legend
                      low = ”lightblue”, high = ”darkblue”,
                      na.value = ”grey77”)
```

A warning message will appear because we are overriding the previous color scale from gheatmap with the new scale from ggplot2

Option 2

```
> gheatmap(gg_simple,
   blast_df,
   offset = 0.1,
   width = 1.5,
   font.size = 5,
   colnames_position = ”top”,
   color = ”black”) +
scale_fill_gradient(name = ”%
  identity”,
   low = ”lightblue”,
   high = ”darkblue”,
   na.value = ”grey77”)
```



LEARNING OBJECTIVES

1. *Install and load packages into RStudio*
2. *Load a newick tree and compliment files*
3. *Learn how to create a basic tree*
4. *Changing tree layout*
5. *Adding and customizing tip labels*
6. *How to merge your tree with a heatmap*
7. ***How to export the final tree***

Exporting the final tree

```
> tree_heat <- gheatmap(gg_simple,
           blast_df,
           offset = 0.1,
           width = 1.5,
           font.size = 5,
           colnames_position = ”top”,
           color = ”black”) +
      scale_fill_gradient(name = ”% identity”, # title of legend
                      low = ”lightblue”, high = ”darkblue”,
                      na.value = ”grey77”)

ggsave(”output/tree_heat.jpeg”, tree_heat, dpi = 300)
```

HELPFUL RESOURCES

1. Data integration, manipulation and visualization of phylogenetic trees: https://yulab-smu.top/treedata-book/index.html
2. ggtree github: https://github.com/YuLab-SMU/ggtree
3. Enhanced annotation practice: http://www.randigriffin.com/2017/05/11/primate-phylogeny-ggtree.html
4. Colors and scales: https://ggplot2-book.org/scale-colour.html

THANK YOU FOR ATTENDING!

The Q&A Session will now begin.

Please make sure to fill out the Exit Survey

We value your feedback!

More questions? Please email us at
mmid.coding.workshop@gmail.com or post them to the workshop slack channel