Code

Assignment 6

- Nadejda Boev (20056079)

- Due Date - 2022/03/02
- Github user - 16nbb1
- Github link - https://github.com/16nbb1/Biol432_A6 (https://github.com/16nbb1/Biol432_A6)

Assignment Brief

After graduating from Queen's, you land a job as a research scientist at the Canadian Public Health Agency in Canada when a clinician sends you a sample of blood from a patient with life-threatening illness.

You use nanopore sequencing of the patient's biofluids and a custom bioinformatics pipeline that filters out human DNA. Of the remaining (non-human) DNA, you find one sequence that seems odd.

Use the knowledge you have gained to generate an alignment and build a phylogeny in R to analyze the sequence. Determine if it is human or another organism. Write a report in R Markdown explaining to the clinician whether this is something to be concerned about, using graphics with text to explain your analysis. Remember to pay attention to formatting to make the report look professional.

Accessing libraries we'll need, most are dependent on BiocManager

Hide

```
library(BiocManager)
library(genbankr)
library(Biostrings)
library(ggtree)
library(annotate)
library(muscle)
library(reshape2)
library(rentrez)
library(ape)
library(dplyr)
library(ggplot2)
library(cowplot)
```

Sequence provided

Hide

```
seq = 'ATGTCTGATAATGGACCCCAAAATCAGCGAAATGCACCCCGCATTACGTTTGGTGGACCCTCAGATTCAACTGGCAGTAACCAGAATGGAGAACGCAGTGGGGCGC
GATCAAAACAACGTCGGCCCCAAGGTTTACCCAATAATACTGCGTCTTGGTTCACCGCTCTCACTCAACATGGCAAGGAAGACCTTAAATTCCCTCGAGGACAAGGCGTTCCA
ATTAACACCAATAGCAGTCCAGATGACCAAATTGGCTACTACCGAAGAGCTACCAGACGAATTCGTGGTGGTGACGGTAAAATGAAAGATCTCAGTCCAAGATGGTATTTCTA
CTACCTAGGAACTGGGCCAGAAGCTGGACTTCCCTATGGTGCTAACAAAGACGGCATCATATGGGTTGCAACTGAGGGAGCCTTGAATACACCAAAAGATCACATTGGCACCC
GCAATCCTGCTAACAATGCTGCAATCGTGCTACAACTTCCTCAAGGAACAACATTGCCAAAAGGCTTCTACGCAGAAGGGAGCAGAGGCGGCAGTCAAGCCTCTTCTCGTTCC
TCATCACGTAGTCGCAACAGTTCAAGAAATTCAACTCCAGGCAGCAGTAGGGGAACTTCTCCTGCTAGAATGGCTGGCAATGGCGGTGATGCTGCTCTTGCTTTGCTGCTGCT
TGACAGATTGAACCAGCTTGAGAGCAAAATGTCTGGTAAAGGCCAACAACAACAAGGCCAAACTGTCACTAAGAAATCTGCTGCTGAGGCTTCTAAGAAGCCTCGGCAAAAAC
GTACTGCCACTAAAGCATACAATGTAACACAAGCTTTCGGCAGACGTGGTCCAGAACAAACCCAAGGAAATTTTGGGGACCAGGAACTAATCAGACAAGGAACTGATTACAAA
CATTGGCCGCAAATTGCACAATTTGCCCCCAGCGCTTCAGCGTTCTTCGGAATGTCGCGCATTGGCATGGAAGTCACACCTTCGGGAACGTGGTTGACCTACACAGGTGCCAT
CAAATTGGATGACAAAGATCCAAATTTCAAAGATCAAGTCATTTTGCTGAATAAGCATATTGACGCATACAAAACATTCCCACCAACAGAGCCTAAAAAGGACAAAAAGAAGA
AGGCTGATGAAACTCAAGCCTTACCGCAGAGACAGAAGAAACAGCAAACTGTGACTCTTCTTCCTGCTGCAGATTTGGATGATTTCTCCAAACAATTGCAACAATCCATGAGC
AGTGCTGACTCAACTCAGGCCTAA'
```

Searching for sequence using NCBI's blast via blastSequences function

We will output the top 10 hits (hitListSize) as a dataframe

Hide

```
myst_blast = blastSequences(seq, as='data.frame', hitListSize = 10, timeout = 600)
```

```
estimated response time 31 seconds
elapsed time 31 seconds
elapsed time 42 seconds
```

Investigating the Accession numbers from our Blast search

Hide

```
(myst_blast$Hit_accession)
```

```
 [1] "OM836578" "OM833768" "OM831491" "OM831484" "OM831469" "OM831457" "OM831448" "OM831445" "OM831427" "OM83142
3"
```

Making a MSA using Muscle

We need to feed in the sequences from the table, collapse, unlist them and convert them into DNAStringSet sequences muscle can use.We then run the MSA using default parameters

Hide

```
myst_blastDNAstring =
  myst_blast$Hsp_hseq %>%
  as.character %>%
  lapply(.,paste0,collapse="") %>%
  unlist %>%
  DNAStringSet

# Allows for unique IDs among sequences that may be similar
names(myst_blastDNAstring)<-paste(1:nrow(myst_blast),myst_blast$Hit_accession, sep='_')

# Making an alignment with Muscle
BbAlign<-muscle::muscle(myst_blastDNAstring, quiet=T)
```

Inspecting/ Preprocessing

Since we are using sequencing, specifically nanopore sequencing, we should confirm if there is variation in the length of sequences for our 10 hits

Hide

```
# We're going to pull the length of the sequence from the myst_blastDNAstring object
SeqLen<-as.numeric(lapply(myst_blastDNAstring,length))

# Plotting length
plot1 = qplot(SeqLen)+
  theme_bw() +
  xlab ('Sequence length')+
  ylab('Number of sequences')
```

Distance matrix

We are interested in pairwise comparisons across for each sequence - Note: all the values are 0.

Hide

```
# Storing the aligned sequence in "internal" format
BBSubAlignBin <- as.DNAbin(BbAlign)

# We picked the K80 model
(BbDM<-dist.dna(BBSubAlignBin, model="K80"))
```

```
             1_OM836578 2_OM833768 3_OM831491 4_OM831484 5_OM831469 6_OM831457 7_OM831448 8_OM831445 9_OM831427
2_OM833768            0
3_OM831491            0          0
4_OM831484            0          0          0
5_OM831469            0          0          0          0
6_OM831457            0          0          0          0          0
7_OM831448            0          0          0          0          0          0
8_OM831445            0          0          0          0          0          0          0
9_OM831427            0          0          0          0          0          0          0          0
10_OM831423           0          0          0          0          0          0          0          0          0
```

Hide

```
# class(BbDM)
# length(BbDM)
```

We need to make a linear matrix we can plot as a heatmap then “melt” the matrix so we have a dataframe we can search rows/columns from

Hide

```
# Makes 1 long dataframe instead of matrix
(BbDMmat<-as.matrix(BbDM))
```

```
             1_OM836578 2_OM833768 3_OM831491 4_OM831484 5_OM831469 6_OM831457 7_OM831448 8_OM831445 9_OM831427 10
_OM831423
1_OM836578           0          0          0          0          0          0          0          0          0
0
2_OM833768           0          0          0          0          0          0          0          0          0
0
3_OM831491           0          0          0          0          0          0          0          0          0
0
4_OM831484           0          0          0          0          0          0          0          0          0
0
5_OM831469           0          0          0          0          0          0          0          0          0
0
6_OM831457           0          0          0          0          0          0          0          0          0
0
7_OM831448           0          0          0          0          0          0          0          0          0
0
8_OM831445           0          0          0          0          0          0          0          0          0
0
9_OM831427           0          0          0          0          0          0          0          0          0
0
10_OM831423          0          0          0          0          0          0          0          0          0
0
```

Hide

```
(PDat<-melt(BbDMmat))
```

Var1 <fctr>	**Var2** <fctr>	**value** <dbl>
1_OM836578	1_OM836578	0
2_OM833768	1_OM836578	0
3_OM831491	1_OM836578	0
4_OM831484	1_OM836578	0
5_OM831469	1_OM836578	0
6_OM831457	1_OM836578	0
7_OM831448	1_OM836578	0
8_OM831445	1_OM836578	0
9_OM831427	1_OM836578	0
10_OM831423	1_OM836578	0

1-10 of 100 rows Previous **1** 2 3 4 5 6 … 10 Next

Hide

```
#dim(PDat)
```

Now we can plot the heatmap - Colors with a value of 0, are NOT different

Hide

```
plot2 = ggplot(data = PDat, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile()+
  theme_bw() +
  xlab ('Sequence length')+
  ylab('Number of sequences')+
  theme(axis.text.x = element_text(angle = 20, vjust = 0.5, hjust=1))
```

Building a tree

We’ll use an NJ approach when building tree - Note there are 10 tips as we were expecting

Hide

```
(BbTree<-nj(BbDM))
```

```
Phylogenetic tree with 10 tips and 8 internal nodes.

Tip labels:
  1_OM836578, 2_OM833768, 3_OM831491, 4_OM831484, 5_OM831469, 6_OM831457, ...

Unrooted; includes branch lengths.
```

Hide

```
#str(BbTree)
# class(BbTree), "phylo"
```

Visualizing the tree

Since we aren't expecting major differences based on Figure 2, we won't prune this tree

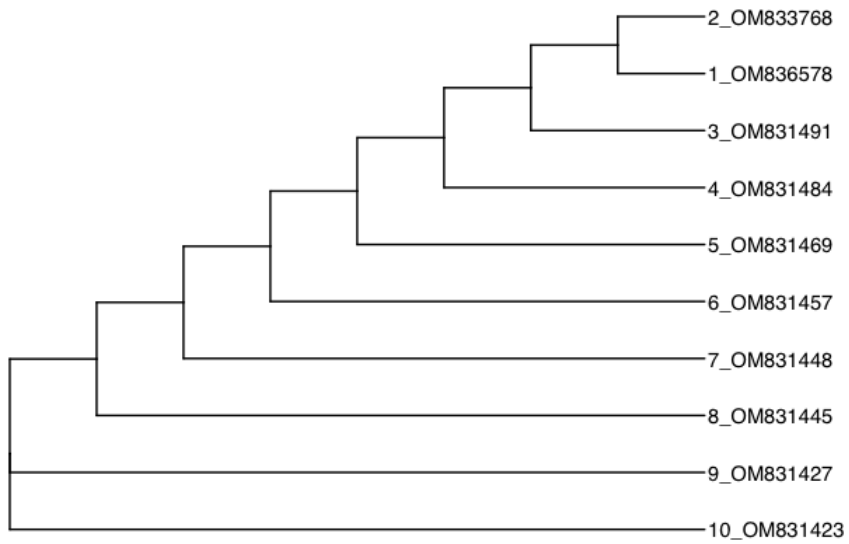
Hide

```
# Plotting the phylogenetic tree
plot3 = ggtree(BbTree) +
  geom_tiplab(hjust=.00003)+
  xlim(0, 10)
```

Visualizing relationships instead

Hide

```
ggtree(BbTree,branch.length='none')+
  geom_tiplab(hjust=.0003)+
  xlim(0, 10)
```



Conclusions for Clinician

To begin, with preprocessing, we found all the sequences are the same length (Figure 1) This means our alignment will have no gapping

Hide

```
ggdraw(add_sub(plot1, size =10,
               'Figure 1. Histogram of length of sequences (n=10) from Top 10 NCBI hits which matched to
               Human isolate, unknown sequence.'))
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

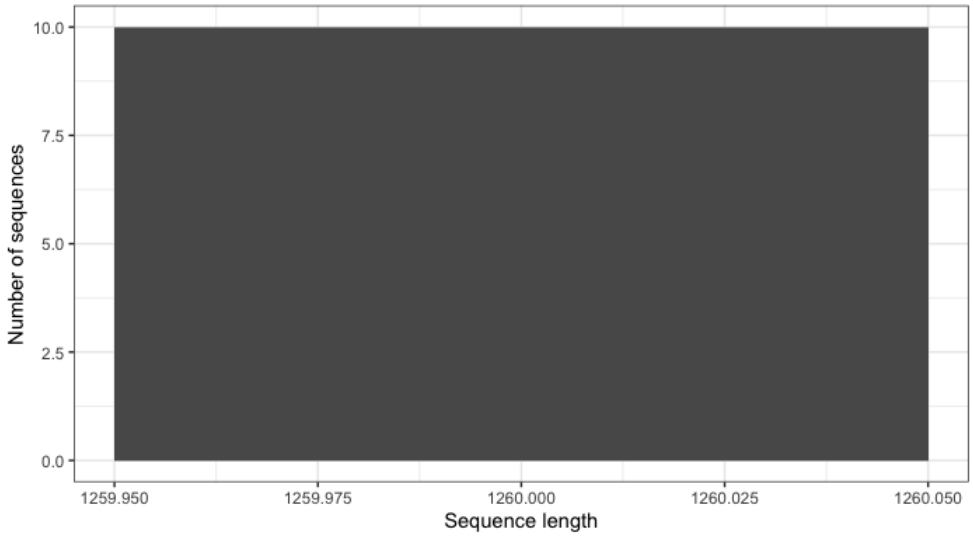


Figure 1. Histogram of length of sequences (n=10) from Top 10 NCBI hits which matched to Human isolate, unknown sequence.

We found the distances matrices showed pairwise distances were all 0. Further confirmed with a pairwise heatmap (Figure 2). This would be consistent with all 10 sequences having the same sequence.

Hide

```
ggdraw(add_sub(plot2, size =10,
               'Figure 2. Heatmap of pairwise distances among  Top 10 NCBI hits which matched to
               Human isolate, unknown sequence.'))
```

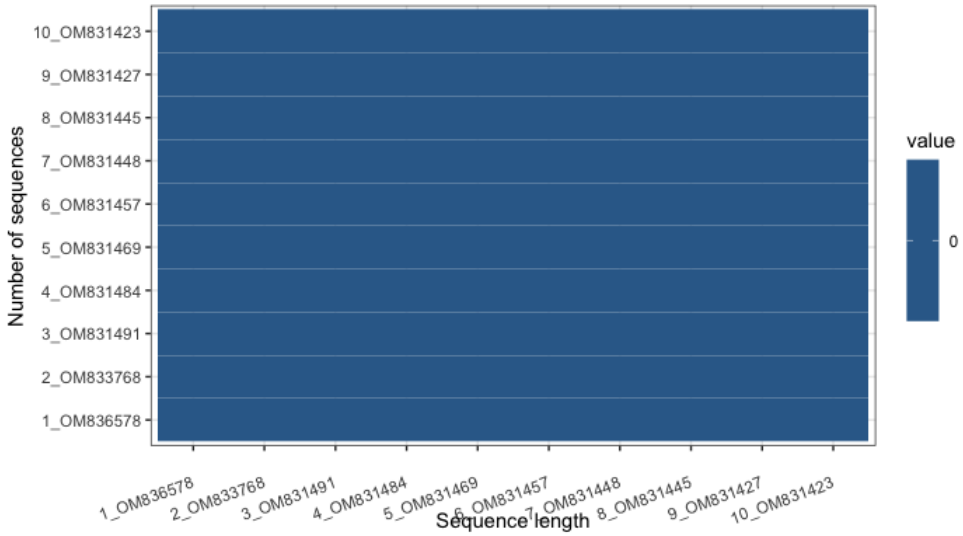


Figure 2. Heatmap of pairwise distances among Top 10 NCBI hits which matched to Human isolate, unknown sequence.

We found there is not variation in the sequences, confirmed with no branches in the phylogenetic tree (Figure 3)

Hide

```
ggdraw(add_sub(plot3, size =10,
               'Figure 3. Phylogenetic tree of Top 10 NCBI hits which matched to
               Human isolate, unknown sequence once aligned with Muscle'))
```

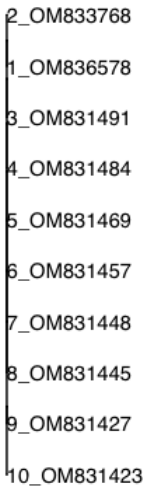


Figure 3. Phylogenetic tree of Top 10 NCBI hits which matched to Human isolate, unknown sequence once aligned with Muscle

Hide

```
myst_blast$Hit_def
```

```
 [1] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/974471/2020, complete genome"
 [2] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/CA-CDC-ASC210555624/2022, comp
lete genome"
 [3] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U122MD00289/2020, complete g
enome"
 [4] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U172MD00253/2020, complete g
enome"
 [5] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U120MD00104/2020, complete g
enome"
 [6] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U172MD00254/2020, complete g
enome"
 [7] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U117MD00099/2020, complete g
enome"
 [8] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U183MD00611/2020, complete g
enome"
 [9] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U189MD00114/2020, complete g
enome"
[10] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/20U119MD00334/2020, complete g
enome"
```

Upon inspection, these samples map to the SARS-CoV-2 sequence, therefore we may generally argue, given the transmissibility this is a concerning finding. If the analysis' purpose is to surveil for new strains or concerning mutations, we can confirm there is no variation in these sequences (Figure 2,3)