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 as a dataframe

Hide

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

```
estimated response time 4 seconds
elapsed time 4 seconds
elapsed time 15 seconds
```

Investigating the Accession numbers from our Blast search

Hide

```
(myst_blast$Hit_accession)
```

```
 [1] "OM797449" "OM793753" "OM779898" "OM766143" "OM766139" "OM766138" "OM765571" "OM765560" "OV888164" "OV88626
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='_')

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

```
SeqLen<-as.numeric(lapply(myst_blastDNAstring,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

```
BBSubAlignBin <- as.DNAbin(BbAlign)

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

```
            1_OM797449 2_OM793753 3_OM779898 4_OM766143 5_OM766139 6_OM766138 7_OM765571 8_OM765560 9_OV888164
2_OM793753           0
3_OM779898           0          0
4_OM766143           0          0          0
5_OM766139           0          0          0          0
6_OM766138           0          0          0          0          0
7_OM765571           0          0          0          0          0          0
8_OM765560           0          0          0          0          0          0          0
9_OV888164           0          0          0          0          0          0          0          0
10_OV886263          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

```
(BbDMmat<-as.matrix(BbDM))
```

```
            1_OM797449 2_OM793753 3_OM779898 4_OM766143 5_OM766139 6_OM766138 7_OM765571 8_OM765560 9_OV888164 10
_OV886263
1_OM797449          0          0          0          0          0          0          0          0          0
0
2_OM793753          0          0          0          0          0          0          0          0          0
0
3_OM779898          0          0          0          0          0          0          0          0          0
0
4_OM766143          0          0          0          0          0          0          0          0          0
0
5_OM766139          0          0          0          0          0          0          0          0          0
0
6_OM766138          0          0          0          0          0          0          0          0          0
0
7_OM765571          0          0          0          0          0          0          0          0          0
0
8_OM765560          0          0          0          0          0          0          0          0          0
0
9_OV888164          0          0          0          0          0          0          0          0          0
0
10_OV886263         0          0          0          0          0          0          0          0          0
0
```

Hide

```
#dim(BbDMmat)
```

Hide

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

Var1 <fctr>	**Var2** <fctr>	**value** <dbl>
1_OM797449	1_OM797449	0
2_OM793753	1_OM797449	0
3_OM779898	1_OM797449	0
4_OM766143	1_OM797449	0
5_OM766139	1_OM797449	0
6_OM766138	1_OM797449	0
7_OM765571	1_OM797449	0
8_OM765560	1_OM797449	0
9_OV888164	1_OM797449	0
10_OV886263	1_OM797449	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_OM797449, 2_OM793753, 3_OM779898, 4_OM766143, 5_OM766139, 6_OM766138, ...

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

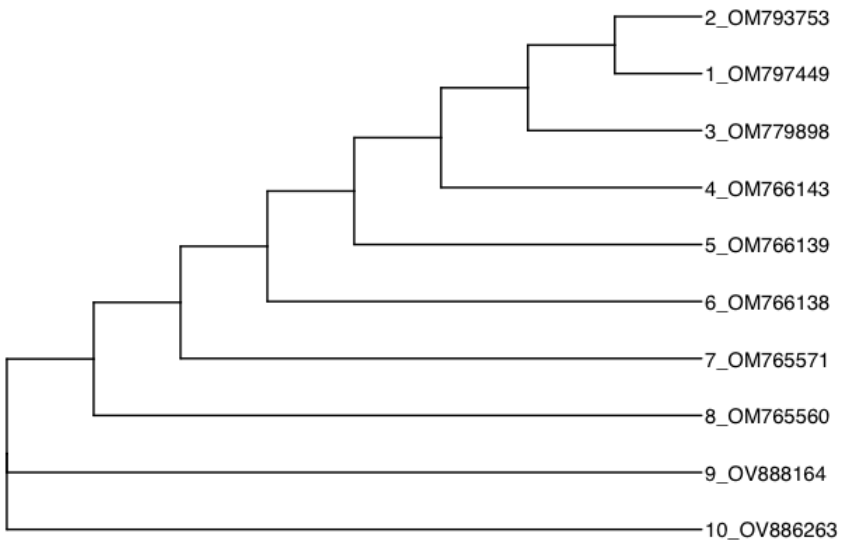
Hide

```
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`.
```

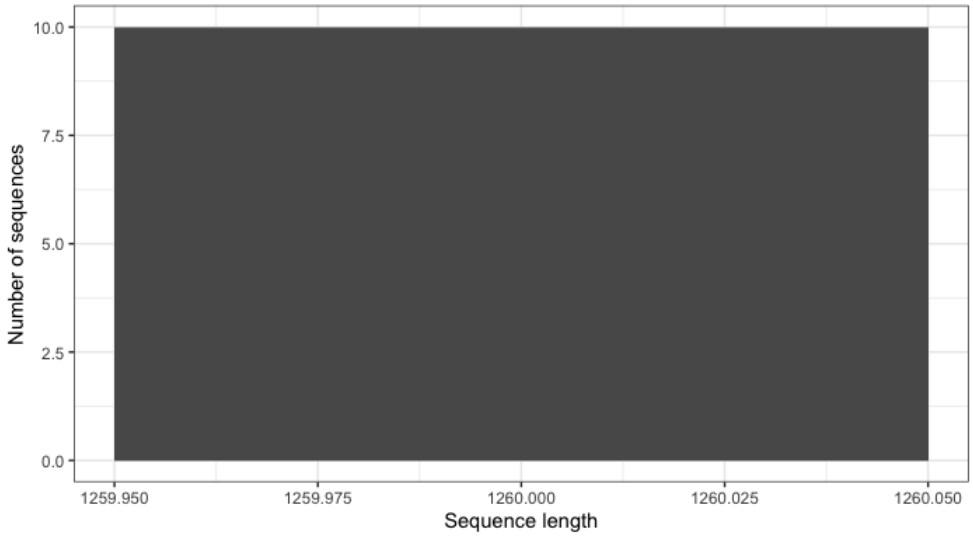


Figure 1. Histogram of length of sequences (n=10) from Top 10 NCBI hits which matched to Human isolate, unknown sequence.

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.'))
```

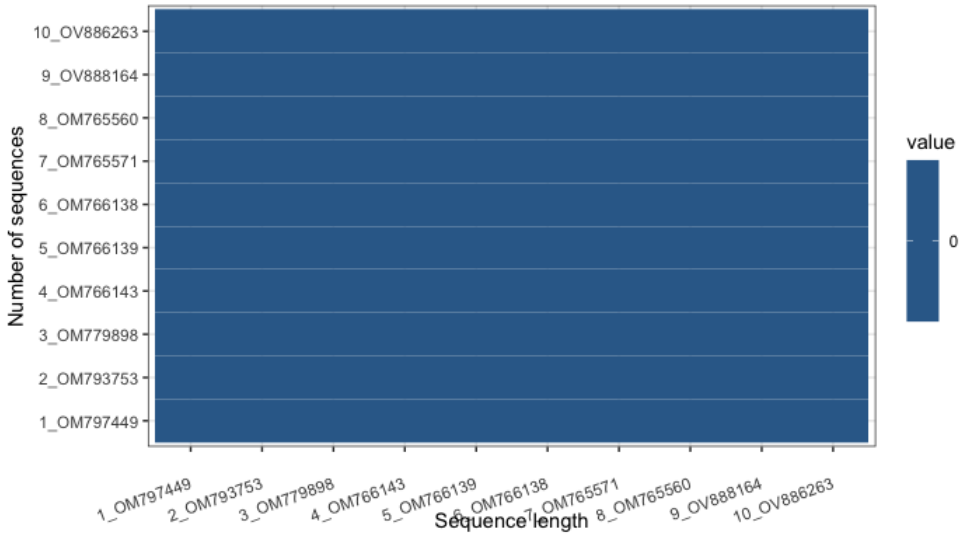


Figure 2. Heatmap of pairwise distances among Top 10 NCBI hits which matched to Human isolate, unknown sequence.

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'))
```

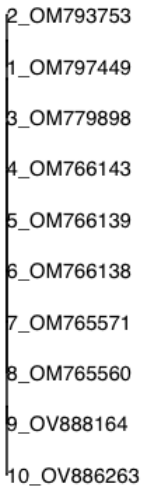


Figure 3. Phylogenetic tree of Top 10 NCBI hits which matched to Human isolate, unknown sequence once aligned with Muscle

Hide

```
myst_blast$Hit_def
```

```
 [1] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/CO-CDPHE-2100025094/2020, comp
lete genome"
 [2] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/UT-UPHL-220213027980/2022, com
plete genome"
 [3] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/USA/V01337/2021 ORF1ab polyprotein
(ORF1ab) gene, complete cds; ORF1a polyprotein (ORF1ab) gene, partial cds; and surface glycoprotein (S), ORF3a pr
otein (ORF3a), envelope protein (E), membrane glycoprotein (M), ORF6 protein (ORF6), ORF7a protein (ORF7a), ORF7b
(ORF7b), ORF8 protein (ORF8), nucleocapsid phosphoprotein (N), and ORF10 protein (ORF10) genes, complete cds"
 [4] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/SouthAfrica/NHLS-UCT-LA-9012/2020,
complete genome"
 [5] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/SouthAfrica/NHLS-UCT-LA-9008/2020
ORF1ab polyprotein (ORF1ab), ORF1a polyprotein (ORF1ab), surface glycoprotein (S), ORF3a protein (ORF3a), envelop
e protein (E), membrane glycoprotein (M), ORF6 protein (ORF6), ORF7a protein (ORF7a), ORF7b (ORF7b), ORF8 protein
(ORF8), nucleocapsid phosphoprotein (N), and ORF10 protein (ORF10) genes, complete cds"
 [6] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/SouthAfrica/NHLS-UCT-LA-9007/2020
ORF1ab polyprotein (ORF1ab), ORF1a polyprotein (ORF1ab), surface glycoprotein (S), ORF3a protein (ORF3a), envelop
e protein (E), membrane glycoprotein (M), ORF6 protein (ORF6), ORF7a protein (ORF7a), ORF7b (ORF7b), ORF8 protein
(ORF8), nucleocapsid phosphoprotein (N), and ORF10 protein (ORF10) genes, complete cds"
 [7] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/SouthAfrica/NHLS-UCT-GS-C167/2021,
complete genome"
 [8] "Severe acute respiratory syndrome coronavirus 2 isolate SARS-CoV-2/human/SouthAfrica/NHLS-UCT-GS-C154/2021
ORF1ab polyprotein (ORF1ab), ORF1a polyprotein (ORF1ab), surface glycoprotein (S), ORF3a protein (ORF3a), envelop
e protein (E), membrane glycoprotein (M), ORF6 protein (ORF6), ORF7a protein (ORF7a), ORF7b (ORF7b), ORF8 protein
(ORF8), nucleocapsid phosphoprotein (N), and ORF10 protein (ORF10) genes, complete cds"
 [9] "Severe acute respiratory syndrome coronavirus 2 genome assembly, chromosome: 1"
[10] "Severe acute respiratory syndrome coronavirus 2 genome assembly, chromosome: 1"
```

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)

It is however interestingly, some sequences map both to sequences found in both the USA and South Africa, potentially indicating a sick individual traveled and carried this version of the virus and infected others.