Individual Exercise in Molecular Phylogenetics

Cennen Del Rosario

Introduction

SARS-CoV-2 is the new coronavirus that causes coronavirus disease 2019 or COVID-19. The virus is believed to have originated in Wuhan, China in late 2019. Currently it has spread worldwide affecting 8.68 million people and bringing more than 400,000 deaths as of this writing [1]. Collecting samples of the virus and determining how they are related in terms of phylogeny is one of the first tasks in understanding the nature of this virus.

In this exercise, various DNA sequences of SARS-CoV-2 have been analyzed and put into hierarchical clusters. Different methods of substitution have been explored using a very popular tool among data sciencntists. Thus, it does not only serve as an exercise of bioinformatics tools but this merges a few data science techniques to existing bioinformatics practice.

The same pipeline is used: sequence alignment, distance matrix generation, and plotting into Phylogenetic trees. The author hopes that this may serve as a basis for future consideration in integrating a wide number of fragmented tools in bioinformatics.

Exploratory Data Analysis

Through out this project, Google Colab has been used to conduct phylogenetic analysis. Together with existing tools in bioinformatics, nine DNA sequence samples of SARS-CoV-2 were analyzed. **Table 1** summarizes the details about these samples. Samples from Wuhan, Hong Kong, Hangzhou, Yunnan, Singapore, Valencia, Netherlands, and US were collected. A reference DNA sample from a bat will be used as outgroup. All of them except the sample from the Netherlands clearly indicate complete genome sequence. As for the length, their nucleotide base-pairs stretch up to around 30 kbp. Many of them, except the reference sample, have a GC content of around 38%.

`Table 1. Summary of DNA samples of SARS-CoV-2`

```
In [570]: pd_eda_data
```

Out[570]:

	Sample ID	Description	Kind	Length (bp)	GC Content (%)	Origin
0	NC045512-Wuhan	SARS Cov 2 isolate Wuhan-Hu-1	complete genome	29903	37.97	Wuhan
1	GQ153539.1-Bat-SARS-CoV	Bat SARS coronavirus HKU3-4	complete genome	29704	41.17	Bat
2	MN938384-HKU-SZ	SARS Cov 2 isolate 2019-nCoV/HKU-SZ-002a/2020	complete genome	29838	38.02	Hong Kong
3	MN997409-US-AZ1	SARS Cov 2 isolate 2019-nCoV/USA-AZ1/2020	complete genome	29882	37.99	US
4	MT253708-Hangzhou	SARS Cov 2 isolate SARS-CoV-2/HZ-79/human/2020...	complete genome	29781	38.02	Hangzhou
5	MT066175-NTU	SARS Cov 2 isolate SARS-CoV-2/NTU01/2020/TWN	complete genome	29870	38.01	Singapore
6	MT198652-Valencia	SARS Cov 2 isolate SARS-CoV-2/Valencia003/huma...	complete genome	29781	38.01	Valencia
7	MT049951-Yunnan	SARS Cov 2 isolate SARS-CoV-2/Yunnan-01/human/...	complete genome	29903	37.97	Yunnan
8	EPI_ISL_413572	hCoV-19/Netherlands/Haarlem_1363688/2020		29786	38.01	Netherlands

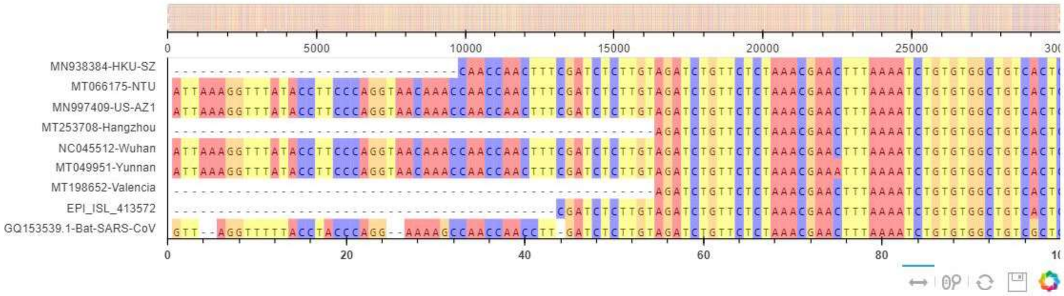
Phylogenetic Analysis

1. Multiple Sequence Alignment

Using the wrapper function of Biopython, MUSCLE alignment was made. **Figure 1** shows the multiple sequence alignment of the nine DNA samples. The image was rendered using the Python package *Bokeh*. Note that in the actual notebook, the image is draggable.

`Figure 1. Multiple sequence alignment of SAR-CoV-2 DNA sequences.`

```
In [548]: show(align_plot)
```



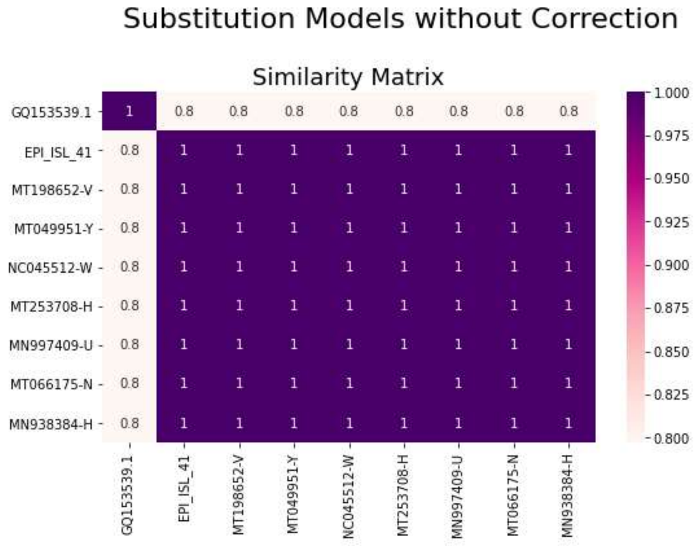
2. Distance Matrices

In order to view the relative distances and similary of the aligned samples, *fdnadist* function of EMBOSS was run through another Biopython wrapper function. The resulting distance matrix shown in **Figure 2** uses no correction. Specifically, the substitution method was *p-distance*. Values closer to 0 indicate closeness, whereas values near 1 tend to be more distant. Although it is not ideally used in Phylogenetic analysis, it verifies that all SARS-CoV-2 samples except the one extracted from a bat are very near to each other.

```
Figure 2. Similarity matrix indicates the relative closeness of the DNA sequences.
          No correction was made in the matrix shown.
```

```
In [551]: # Similarity matrix
          fig, axn = plt.subplots(figsize=(9,6))
          fig.suptitle("Substitution Models without Correction", fontsize=22)
          fig.tight_layout(pad = 7.0)

          hm = display_dist_matrix(p_distance_matrix, colors='RdPu',
                                   title = 'Similarity Matrix', ax=axn)
          ax.set_ylabel('')
          ax.set_xlabel('')
          ax.tick_params(axis='both', which='both', length=0)
```



On the other hand, 4 substitution models with corrections have been considered to analyze the distantness of the samples. They were Jukes-Cantor, Kimura 2-parameters, F84, and LogDet. All four are available in the same Biopython wrapper function for EMBOSS. The distance matrices are shown in **Figure 3**.

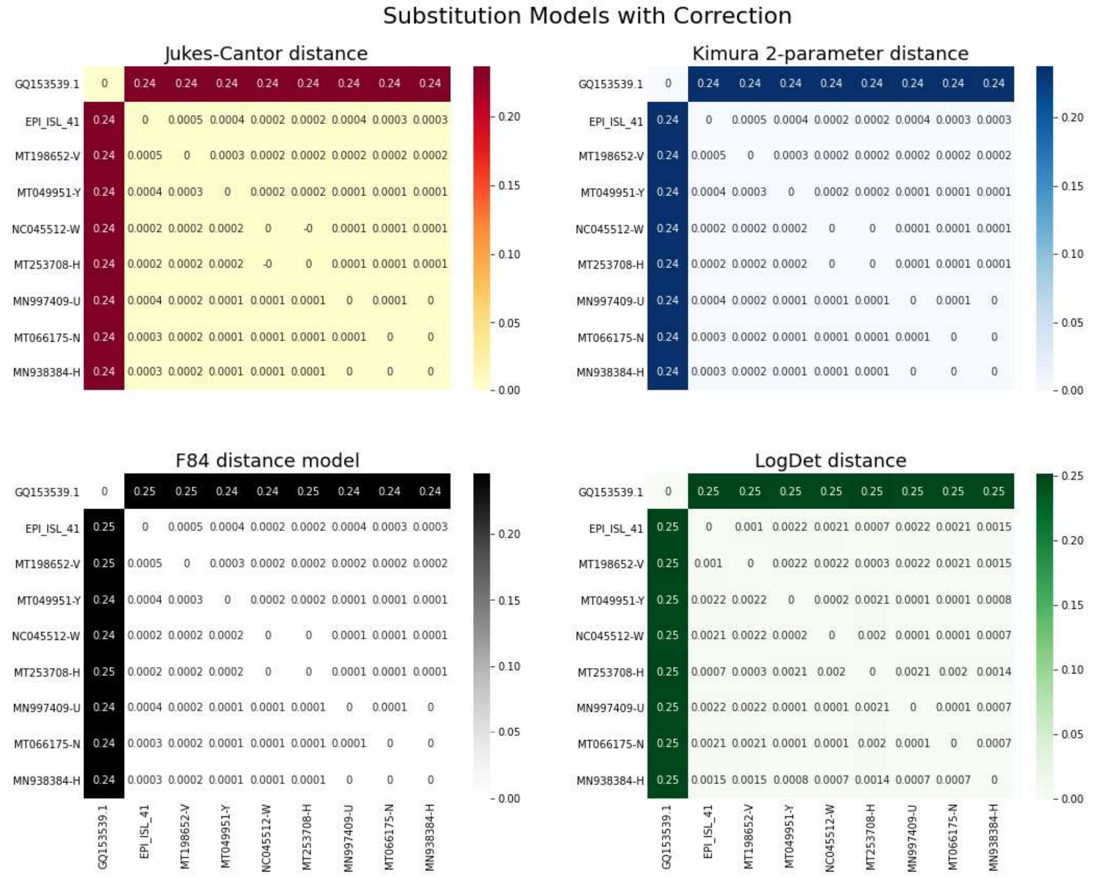
The very powerful tool *Seaborn* provides many colorful and adjustable plots in Python. It is suited for analysis of pairwise variations. In many instances the heatmap plot similarly shown here are used extensively to present confusion matrices. Here, it is best suited for distance matrices.

```
Figure 3. Corrected distance matrices. Four substititin methods were used:
          Jukes-Cantor, Kimura 2-parameter, F84 and LogDet.
```

In [553]:

```python
# Corrected raw distance matrices
fig, axn = plt.subplots(2, 2, sharex=True, figsize=(16,12))
fig.suptitle("Substitution Models with Correction", fontsize=22)
fig.tight_layout(pad = 7.0)

for i, ax in enumerate(axn.flat):
    hm = display_dist_matrix(distance_matrices[i], colors=heatmap_colors[i],
                        title = matrix_names[i], ax=ax, round_digit = 4)
    ax.set_ylabel('')
    ax.set_xlabel('')
    ax.tick_params(axis='both', which='both', length=0)
```



Substitution Models with Correction

Jukes-Cantor distance

Kimura 2-parameter distance

F84 distance model

LogDet distance

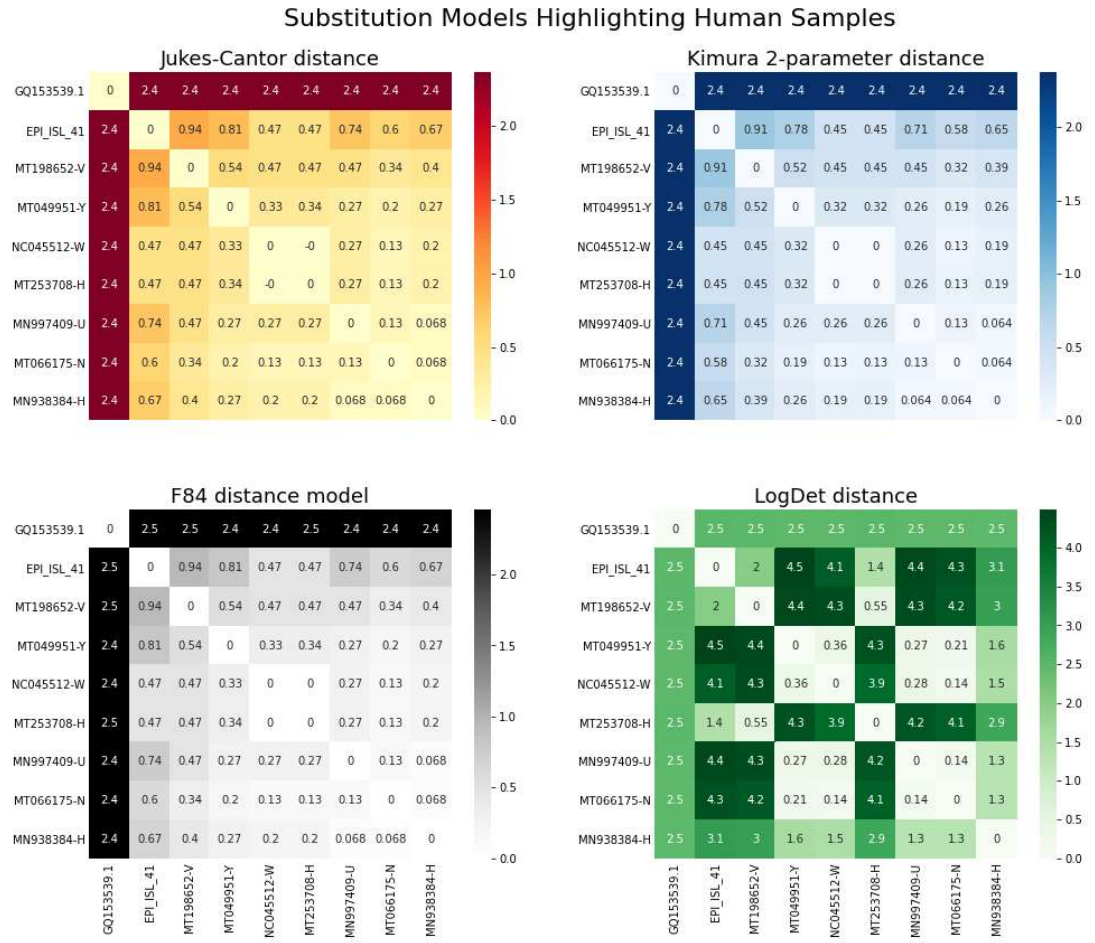
With the corrected substitution models, the Bat sample shows comparatively large distance from other samples. SARS-CoV-2 genome samples taken from various places appear closer with each other. Up to 2 decimal places their distance measures are zero. This does not show their relative distance though.

To highlight how they (the non-Bat samples) differ from each other, further adjustment was made. Weights have been applied into the distance values of the Bat and eight other samples. The result is shown in **Figure 4**. Three substitution models (Jukes-Cantor, Kimura and F84) showed that SARS-C0V-2 sample from the Netherland is relatively distant to samples from Valencia, Yunnan, Singapore and Hong Kong. Meanwhile, samples from Wuhan and Hangzhou exhibit nearly perfect closeness with each other.

Figure 4. Adjusted distance matrices highlighting relative distances among non-Bat samples. It can be seen that EPI_ISL_413572 (Netherland) is relatively distant to MT198652 (Valencia), MT049951 (Yunnan), MT066175 (Singapore) and MN938384 (Hong Kong). The rest have common and strong closeness with each other. LogDet model shows distance values belonging to this group.

```
In [554]: # Corrected distance matrices with additional weight adjustment
          fig, axn = plt.subplots(2, 2, sharex=True, figsize=(15,12))
          fig.suptitle("Substitution Models Highlighting Human Samples", fontsize=22)
          fig.tight_layout(pad = 7.0)

          for i, ax in enumerate(axn.flat):
              hm = display_dist_matrix(display_matrices[i], colors=heatmap_colors[i],
                                  title = matrix_names[i], ax=ax, round_digit = 4)
              ax.set_ylabel('')
              ax.set_xlabel('')
              ax.tick_params(axis='both', which='both', length=0)
```



Jukes-Cantor distance

	GQ153539.1	EPI_ISL_41	MT198652-V	MT049951-Y	NC045512-W	MT253708-H	MN997409-U	MT066175-N	MN938384-H
GQ153539.1	0	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4
EPI_ISL_41	2.4	0	0.94	0.81	0.47	0.47	0.74	0.6	0.67
MT198652-V	2.4	0.94	0	0.54	0.47	0.47	0.47	0.34	0.4
MT049951-Y	2.4	0.81	0.54	0	0.33	0.34	0.27	0.2	0.27
NC045512-W	2.4	0.47	0.47	0.33	0	-0	0.27	0.13	0.2
MT253708-H	2.4	0.47	0.47	0.34	-0	0	0.27	0.13	0.2
MN997409-U	2.4	0.74	0.47	0.27	0.27	0.27	0	0.13	0.068
MT066175-N	2.4	0.6	0.34	0.2	0.13	0.13	0.13	0	0.068
MN938384-H	2.4	0.67	0.4	0.27	0.2	0.2	0.068	0.068	0

Kimura 2-parameter distance

	GQ153539.1	EPI_ISL_41	MT198652-V	MT049951-Y	NC045512-W	MT253708-H	MN997409-U	MT066175-N	MN938384-H
GQ153539.1	0	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4
EPI_ISL_41	2.4	0	0.91	0.78	0.45	0.45	0.71	0.58	0.65
MT198652-V	2.4	0.91	0	0.52	0.45	0.45	0.45	0.32	0.39
MT049951-Y	2.4	0.78	0.52	0	0.32	0.32	0.26	0.19	0.26
NC045512-W	2.4	0.45	0.45	0.32	0	0	0.26	0.13	0.19
MT253708-H	2.4	0.45	0.45	0.32	0	0	0.26	0.13	0.19
MN997409-U	2.4	0.71	0.45	0.26	0.26	0.26	0	0.13	0.064
MT066175-N	2.4	0.58	0.32	0.19	0.13	0.13	0.13	0	0.064
MN938384-H	2.4	0.65	0.39	0.26	0.19	0.19	0.064	0.064	0

F84 distance model

	GQ153539.1	EPI_ISL_41	MT198652-V	MT049951-Y	NC045512-W	MT253708-H	MN997409-U	MT066175-N	MN938384-H
GQ153539.1	0	2.5	2.5	2.4	2.4	2.5	2.4	2.4	2.4
EPI_ISL_41	2.5	0	0.94	0.81	0.47	0.47	0.74	0.6	0.67
MT198652-V	2.5	0.94	0	0.54	0.47	0.47	0.47	0.34	0.4
MT049951-Y	2.4	0.81	0.54	0	0.33	0.34	0.27	0.2	0.27
NC045512-W	2.4	0.47	0.47	0.33	0	0	0.27	0.13	0.2
MT253708-H	2.5	0.47	0.47	0.34	0	0	0.27	0.13	0.2
MN997409-U	2.4	0.74	0.47	0.27	0.27	0.27	0	0.13	0.068
MT066175-N	2.4	0.6	0.34	0.2	0.13	0.13	0.13	0	0.068
MN938384-H	2.4	0.67	0.4	0.27	0.2	0.2	0.068	0.068	0

LogDet distance

	GQ153539.1	EPI_ISL_41	MT198652-V	MT049951-Y	NC045512-W	MT253708-H	MN997409-U	MT066175-N	MN938384-H
GQ153539.1	0	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
EPI_ISL_41	2.5	0	2	4.5	4.1	1.4	4.4	4.3	3.1
MT198652-V	2.5	2	0	4.4	4.3	0.55	4.3	4.2	3
MT049951-Y	2.5	4.5	4.4	0	0.36	4.3	0.27	0.21	1.6
NC045512-W	2.5	4.1	4.3	0.36	0	3.9	0.28	0.14	1.5
MT253708-H	2.5	1.4	0.55	4.3	3.9	0	4.2	4.1	2.9
MN997409-U	2.5	4.4	4.3	0.27	0.28	4.2	0	0.14	1.3
MT066175-N	2.5	4.3	4.2	0.21	0.14	4.1	0.14	0	1.3
MN938384-H	2.5	3.1	3	1.6	1.5	2.9	1.3	1.3	0

Phylogenetic Inference

With the distance matrices previously shown, phylogenetic trees were generated. Two distance-based approaches were used to construct the trees for each of the 4 substitution models: UPGMA and Neighbor-Joining. Biopython's *Phylo* package provides the function to build and import these trees into Newick format.

For the presentation, a Python package called *ToyTree* is used. Although it can render trees from Newick files, there are many other tree-plotting libraries in Python (those used in hierarchichal clustering). But *ToyTree* is very simple and easy to use because it has many clear examples in its documentation.

Figures 5a-5d list down all the output trees images. In the figures, tree branches are not scaled. Weight adjustment added in the computation of distance matrices above was used to specify the visible length of each and every branch. To avoid confusion, proper branch length has been placed beside each node.

Furthermore, due to time-constraints, bootstraping was not implemented in this exercise. Though *Phylo* has available functions to do that, reading its documentation for proper code implementation may take too much time, especially for beginners.

```
Figure 5-a. Phylogenetic trees of Jukes-Cantor model. (left) UPGMA method and (right)  Neighbor-Joining method.
```

```
In [572]: display_upgma_nj_trees(distance_matrices, i=0)
```

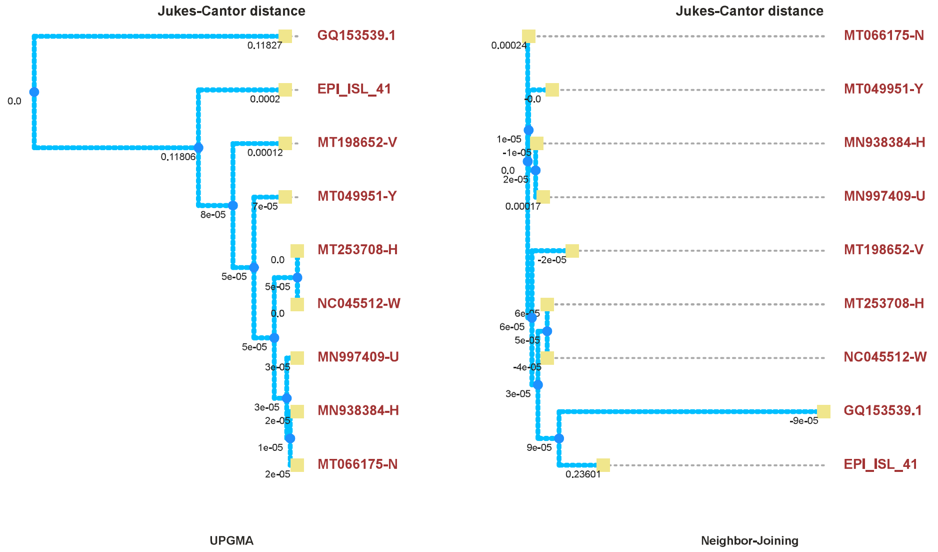


Figure 5-b. Phylogenetic trees of Kimura 2-parameter model. (left) UPGMA method and (right) Neighbor-Joining method.

```
In [573]: display_upgma_nj_trees(distance_matrices, i=1)
```

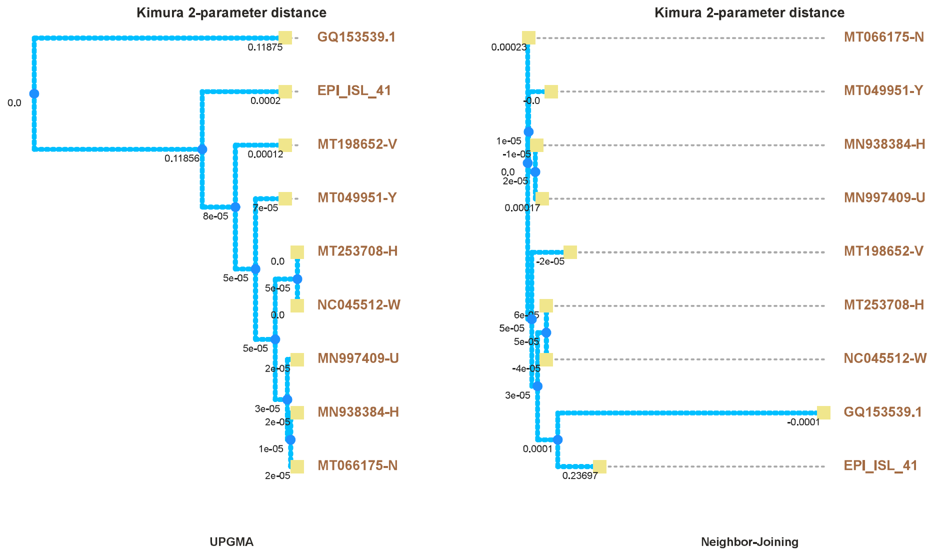


Figure 5-c. Phylogenetic trees of F84 model. (left) UPGMA method and (right) Neighbor-Joining method.

```
In [574]: display_upgma_nj_trees(distance_matrices, i=2)
```

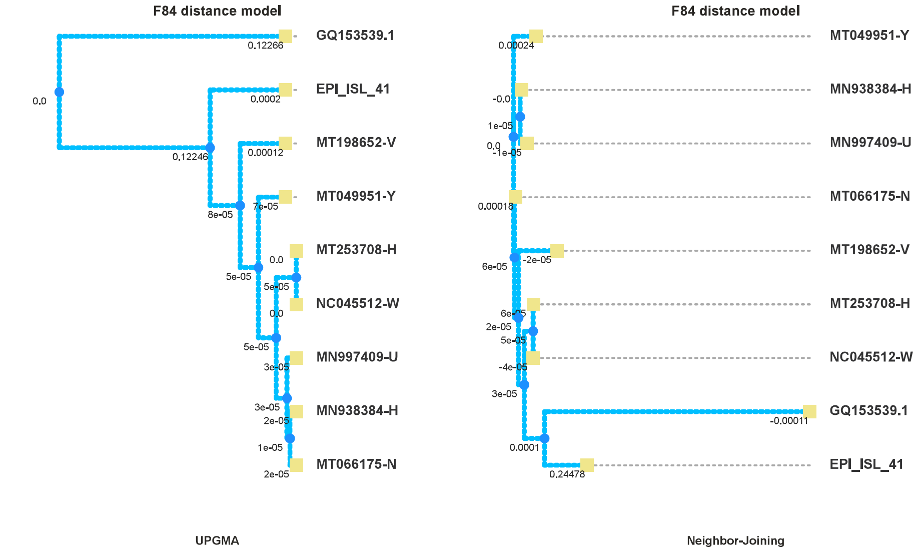
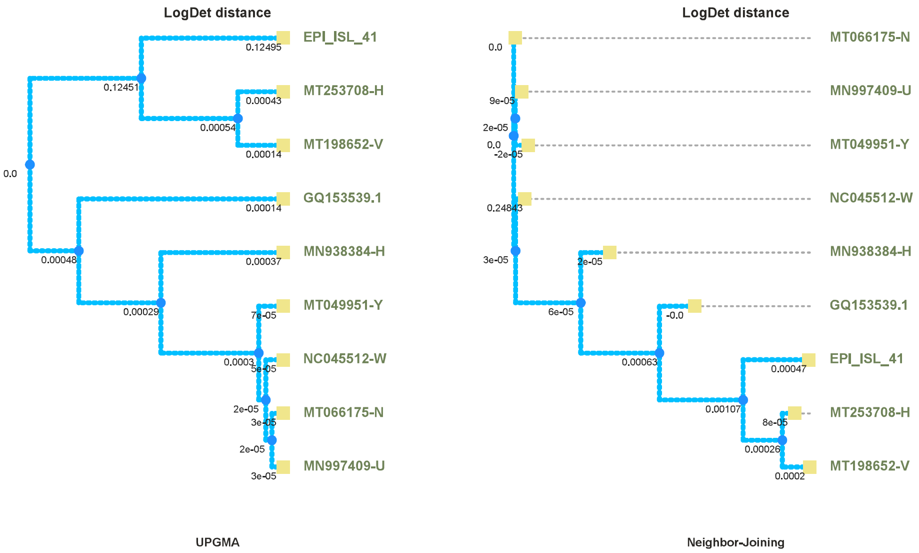


Figure 5-d. Phylogenetic trees of Log-Determinant model. (left) UPGMA method and (right) Neighbor-Joining method.

```
In [575]: display_upgma_nj_trees(distance_matrices, i=3)
```



Tree clustering shows that for the UPGMA method, there are two immediate groups among the SARS-CoV-2 samples. One group consists of MT066175 (Singapore), MN938384 (Hong Kong) and MN997409 (US). Another group is comprised of MT253708 (Hangzhou) and NC045512 (Wuhan). The rest, except the outgroup Bat, is relatively close with each other. Meanwhile, LogDet model shows a different set of groupings - with the sample from Valencia being homologous with the one from Hanghzhou.

On the other side of the graphs, with Neighbor-Joining, still MT253708 (Hangzhou) and NC045512 (Wuhan) are very close with each other. But the method takes away MT066175 (Singapore) from the gang of MN938384 (Hong Kong) and MN997409 (US). Similarly, LogDet shows no good result.

Lastly, intermediate nodes vary for UPGMA. Whereas, it is indeterminate for NJ method because the variations in distances are clearly negligible.

Discussion, Conclusions and Recommendations

The Phylogenetic analysis showed that there are two clusterings among the nine samples provided: 1) Wuhan-Hangzhou, and 2) US-Hong Kong. Analysis from the distance matrix also showed that the samples from Valencia and Netherlands are relatively distant from rest.

For the techniques used, the use of heatmaps for presentation speeds up the identification of clusterings at the distance matrix stage of pipeline. Moreover, adjusting the weights of the outgroup and ingroup samples helps in the identification of subtle differences in their distance values, especially that they are highly imbalanced. It is also worth noting that LogDet methods produced indeterminate results. In either methods, bootstrapping may provide more accurate findings.

Lastly, the use of modern integrated tools like Jupyter notebook or google Colab for this task (pipeline) could take sometime for beginners because of hard details in the code implementation (reading the documentation and installation), and the availability of already established click-and-drag tools. However, as a trade off, they could provide anyone with greater flexibility in control and visualization.

References

> 1. https://www.worldometers.info/coronavirus/ (https://www.worldometers.info/coronavirus/)
> 2. Pevsner, J. Bioinformatics and Functional Genomics, 3rd Edition. 2015. Wiley Blackwell. pp 168-206.

.