NS112 - Tree Thinking

January 17, 2022

1   Question 1

I've marked the most recent ancestor with snakes in green and the most recent ancestor with birds in blue. We can see that, according to the morphological and microRNA evidence, turtles are most closely related to snakes. However, according to the mtDNA and nucDNA evidence, they are more closely related to birds. Here, "more closely related" means their lines diverged more recently.
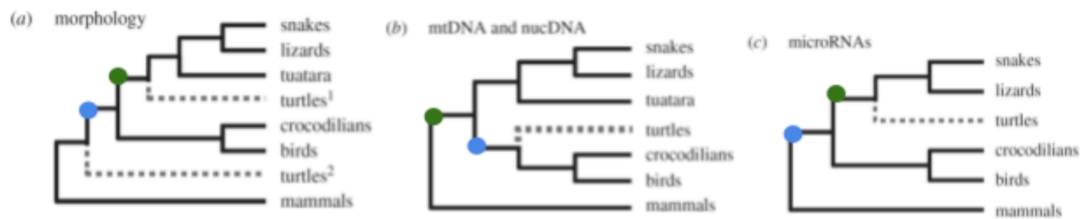

Figure 1 - Hypothesized phylogenetic placements of turtles using different data types.

2   Question 2

Humans were an appropriate choice of outgroup for an analysis of birds and turtles because the most recent common ancestor of humans and any bird or reptile is older than the most recent common ancestor between any two organisms from that group. Any mammal would work here, for example, dogs. Crocodiles would be an inappropriate choice because, as shown in the trees above, crocodiles diverged more recently from birds than they did from reptiles.

3   Question 3

My guess is that the root was either green or red. There was an initial split where the other color occurred, and that is the earliest split in to evolutionary tree. Then, on the left branch, red and black both emerged independently a few times (an example of homoplasy). On the right side, blue and green both emerged from the red as number of times as well.

Since most of the history of the right branch is unknown, other theories are possible. It would be that the right branch was blue or green and red emerged twice independently on the right branch. If the right branch was initially blue, that would explain why it can be found in two of the subbranches of the right branch: the subbranches containing SUNT and CREN. It seems unlikely that the right branch was black, because this color never appears among lines with known character there.

More broadly, this visualization seems to support the idea that phylogeny and chemical composition are related. However, it seems relatively easy for lines to change chemical composition to fit the needs of the environment.

4 Question 4

For this question, we need to find branches that start in India, then move to Sri Lanka, then come back to India. The only phylogeny with well-documented evidence for a single migration of this kind is the crab in figure F. The lines leading from the root to the clade with the families *spiralothelphusa wuellerstorfi*, *spiralothelphusa sp 1*, *oziotelphusa sp 7*, *oziotelphusa sp 6*, *oziotelphusa sp 5* is first brown, then green, then brown. The phylogenic process here is extremely confident; bootstrapping assigns probabilities of 94%+ to every clade containing this one. So, as long as we are confident about the geographic information assigned to ancestor species, then we should be confident that a migration occured.

The fish (figure d) also show evidence of migration, but it is clear that there must have been at least two separate migrations.

5 Question 5

```
[4]:  bugs = {
          "Escherichia coli": "AACGTTCTAGGCCCATACGG",
          "Bacillus anthracis": "AACGTTCTAGGGCCATACGG",
          "Synechococcus": "AACGTCGTAGGACCATCCGG",
          "Chlorobium": "AACGTCATAGGACCATGCGG",
          "Methanococcus": "ATCGTATAACGTCGATTCGG",
      }

      def dist(a, b):
          assert len(a) == len(b)
          return sum(x == y for x, y in zip(a, b))/len(a)

      print("Distance Matrix\n" + ("-" * 59))
      for i, bug in enumerate(bugs):
          distances = [f"{bug:16}"] + ["----" for _ in range(i + 1)]
          for other_bug in list(bugs)[i + 1:]:
              distances.append(dist(bugs[bug], bugs[other_bug]))
          print("\t".join(str(s) for s in distances))
```

```
Distance Matrix
-----------------------------------------------------------
Escherichia coli        ----    0.95    0.8     0.8     0.6
Bacillus anthracis      ----    ----    0.8     0.8     0.6
Synechococcus           ----    ----    ----    0.9     0.6
Chlorobium              ----    ----    ----    ----    0.6
Methanococcus           ----    ----    ----    ----    ----
```

```
[60]: import networkx as nx
      from itertools import combinations, count
      import matplotlib.pyplot as plt

      distance_graph = nx.Graph()
      for a, b in combinations(bugs, 2):
          distance_graph.add_edge(a, b, dist=dist(bugs[a], bugs[b]))

      plt.figure(figsize=(6, 6))
      layout = nx.spring_layout(distance_graph)
      nx.draw(distance_graph, layout, with_labels=True, node_size=500)
      nx.draw_networkx_edge_labels(distance_graph, layout, edge_labels=nx.
       →get_edge_attributes(distance_graph, "dist"))
      plt.title("Genetic Distances between the Organisms")
      plt.show()
```

Genetic Distances between the Organisms
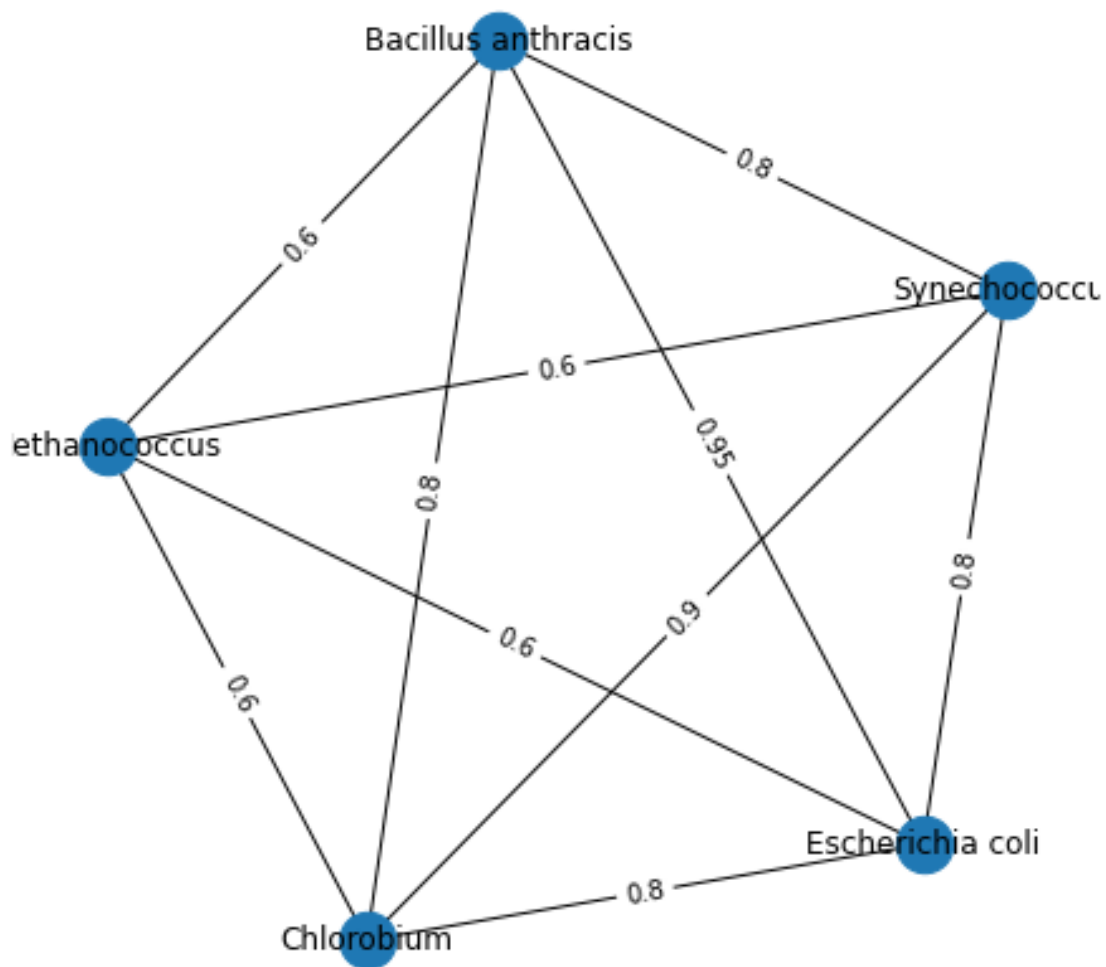

```
[56]: def draw_tree(tree, root, w=1, h=-1, kill_nums=True):
          stack = [new]
          layout = {new: (w, h)}
          while stack:
              node = stack.pop()
              x, y = layout[node]
              for i, child in enumerate(tree.successors(node)):
                  stack.append(child)
                  layout[child] = (x + w, y + (i - 0.5)*h)

          if kill_nums:
```

```python
        labels_to_use = {}
        for node in tree:
            try:
                float(node)
            except ValueError:
                labels_to_use[node] = node

    plt.figure(figsize=(8, 6))
    nx.draw(tree, layout, node_size=0)
    nx.draw_networkx_labels(tree, layout, labels=labels_to_use)
    nx.draw_networkx_edge_labels(tree, layout, edge_labels=nx.
→get_edge_attributes(tree, 'dist'))
```

```python
[63]: distances = distance_graph.copy()
      tree = nx.DiGraph()
      i = count()

      while len(distances.nodes) > 1:
          a, b, d_ab = min(distances.edges.data('dist'), key=lambda x: x[2])
          new = next(i)
          tree.add_node(new)
          tree.add_edge(new, a, dist=d_ab/2)
          tree.add_edge(new, b, dist=d_ab/2)

          for node in distances.nodes - {a, b}:
              avg_dist = (distances[a][node]['dist'] + distances[b][node]['dist'])/2
              distances.add_edge(new, node, dist=avg_dist)
          distances.remove_nodes_from({a, b})

      draw_tree(tree, new)
      plt.title("Inferred Phylogenic Tree")
      plt.show()
```
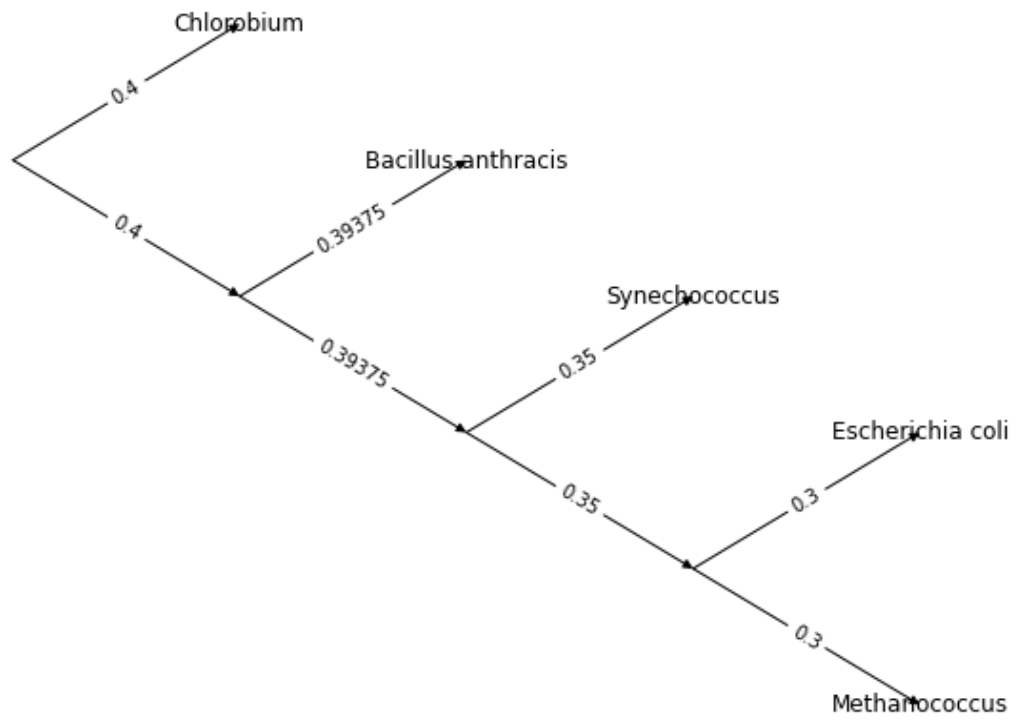
Inferred Phylogenic Tree

Chlorobium

0.4

Bacillus anthracis

0.4

0.39375

Synechococcus

0.39375

0.35

Escherichia coli

0.35

0.3

0.3

Methanococcus

[ ]: