**GS 373 Homework 3**

Due April 26th before 1:30 PM on Canvas

* (100 points): 4 bioinformatics questions (80 points), 1 programming assignment (20 points).
* Submit answers to the bioinformatics questions in a Microsoft Word document or PDF via Canvas. Your answers do not need to contain the text of the questions, but they need to be clearly labeled (e.g., 1a., 3b., etc.)
* Tree topologies can be depicted by inserting shapes and text boxes into a Word Document, by inserting a photo of a hand-drawn tree, or by any method you prefer.
* Submit the programming assignment as a separate .py file onto Canvas. The script should be able to be directly run by Python.

**Bioinformatics Questions (80 points)**

1. (20 points, 3 parts) Tree topologies
   1. (6 points) **Draw and label three distinct *unrooted* topologies for trees with 4 leaves** labeled A, B, C, and D. **Circle the tree that can be rooted such that A and D share a more recent common ancestor** than either of them share with B or C.
   2. (6 points) Choose one of the trees you drew and **draw three topologically distinct rooted variations of that tree**, with all branches parallel to each other.
   3. (8 points) **How many distinct *rooted* tree topologies with 5 species exist?**
2. (25 points, 2 parts) Distance-based tree methods
   1. (5 points) **What is the upper bound for the maximum raw sequence divergence (fraction of unmatched bases) between two sequences, assuming equal nucleotide frequencies?** Explain why. Suppose the assumption of equal frequencies does not hold and you are comparing two sequences that both have very high G-C content – **how would this upper bound change?**
   2. (20 points) **Use the UPGMA algorithm to construct a tree based on the following distance matrix**. Label the length of each branch on your final tree.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | A | B | C | D |
| A | 0 | 7 | 10 | 5 |
| B |  | 0 | 8 | 6 |
| C |  |  | 0 | 9 |
| D |  |  |  | 0 |

1. (20 points, 2 parts) Molecular Evolution
   1. (15 points) For situation below, indicate whether it would be **more appropriate to use a dN/dS method to detect positive selection or a population method and why**.
      1. You have sequencing data from many individuals from two recently diverged populations.
      2. You are interested in a promoter sequence of a human protein.
      3. You are interested in identifying sites on a viral protein which repeatedly changed identity due to selection from the immune system.
   2. (5 points) Give one reason why a deleterious allele may persist in a population.
2. (15 points) Take a look at the **2016 paper by Hug *et al.***describing a new reconstruction of the evolutionary tree of life. **Focus on Figures 1 and 2 and the first three paragraphs of the Methods**.
   1. (8 points) What **method** did the authors use to construct this tree? For the trees in Figures 1 and 2, what **type of sequences** were used from each organism?
   2. (7 points) Describe **one novel finding or observation based on this phylogenetic analysis.**

**Programming problem (20 points)**

**Write a program that calculates the Jukes-Cantor distance between any two sequences of equal length.**

Your output should be formatted as follows:

python calculateJukesCantor.py AGCCCT ATCGCC

**Sequence 1: AGCCCT**

**Sequence 2: ATCGCC**

**Number of nucleotide differences: 3**

**Raw sequence distance: 0.5**

**Jukes-Cantor distance: 0.83**

Notes + Hints

* Your program should take as input (using sys.argv) two DNA sequences.
* It should first check that the two sequences are the same length, and print “Error!” if they are not.
* If they are the same length, it should print each sequence and then calculate
  + 1) the number of nucleotide sites that differ between the two sequences,
  + 2) the raw fractional nucleotide divergence (Draw), and
  + 3) the Jukes-Cantor divergence.
* The Jukes-Cantor divergence is defined as Note that the Jukes-Cantor divergence is only mathematically defined if the raw distance is less than 0.75.
* To calculate a logarithm, you need to import the “math” module. For example:

>>> import math

>>> print math.log(2.7183) #Default is natural log (base e)

**1.00**