GS 373 Homework 5

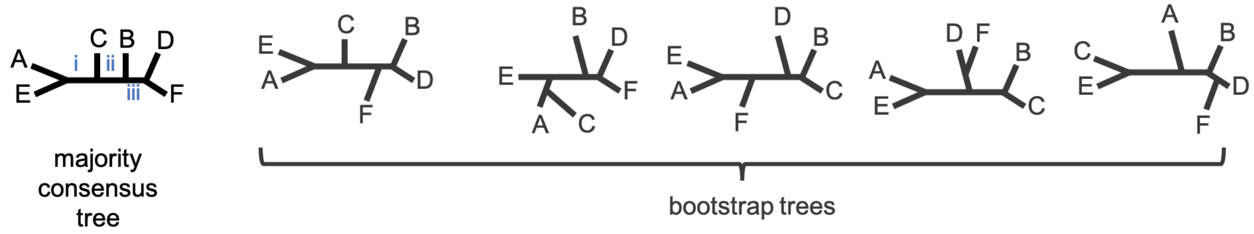
Due May 10th before 1:30 PM on Canvas

- (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.)
- 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. (25 points) Primates and rodents both have two copies of gene A, called A1 and A2. You have sampled sequences of A1 and A2 from humans (H), chimps (C), rats(R), and mice(M).
   a. (11 points) Draw the rooted phylogenetic tree you would expect if A1 and A2 arose from an ancient gene duplication in the ancestors of all mammals and have evolved independently since. Your tree should have eight tips (H-A1, H-A2, C-A1, C-A2, R-A1, R-A2, M-A1, and M-A2). You don't have to worry about branch lengths.
   b. (11 points) Draw the rooted phylogenetic tree you would expect if A1 and A2 arose in the ancestor of all mammals but are next to each other on the chromosome and experience strong concerted evolution.
   c. (3 points) Explain why the trees in parts a and b are the same or different.

2. (25 points)
   Consider the following majority consensus tree. Please give the bootstrap values as proportions for the subtrees marked by i, ii, and iii.



Hint:
Try constructing a table like this

subtree split name	subtree split description	number of trees
i	AE I BCDF	

3. (20 points) Read Kellis *et al*. (2004). Specifically, the introduction up until but not including "Genome sequencing and alignment" and "Evolutionary Analysis" sections "Pattern of gene loss", "Accelerated protein divergence", and "Ancestral and derived functions."

a. (5 points) Name one concept or algorithm we discussed in class which was crucial for this paper.

b. (15 points) What are the competing hypotheses for post-duplication divergence of gene pairs? Which hypothesis does this paper support and what is their evidence?

4. (10 points)

    a. (5 points) Explain why you want to build a species tree with orthologues rather than paralogues. Make sure you define each term in your response.

    b. (5 points) Explain the relationship between gene duplication and gene family.

Programming question (20 points)

This week, **you will implement a simple hill climbing algorithm using a while loop**. The job of your function will be to **find the optimal value for y (height) given a single parameter x**. This is a one-dimensional landscape example – imagine that x is the only parameter that determines some aspect of a tree and y is how well the tree performs on some scoring metric.

I will provide a scoring function `scoring_func` which will provide y for some value of x you give it. You will write a function that finds the highest scoring value of x by taking small steps to the top of the hill. (Yes, the function I provided can be solved analytically, but pretend it cannot).

Your function called `hill_climb` will take one argument called start. Start will be the starting x value of your hill climb. Your step size should be 0.1. In addition to returning the best x value, print the best x and the maximum y as well as the number of iterations it took to get there.

Some hints:

- Start by initializing your current x and y values using `scoring_func`
- For each iteration of the while loop, decide which direction you should go by going **one step (0.1) forward and one step backwards from your current location and comparing the y values.** Then update both your current x and your current y based on the results
- Make sure you have a way to stop the loop when a step in any direction gives you a lower value.

```
# Starter code

def scoring_func(x):
    return(-2*x**2 + 4*x + 5)

def hill_climb(start):
    # Initialize values

    while(): # insert test
        # your code here
    print("Best x: {0}\nMaximum y: {1}\nIterations:
    {2}\n".format(current_x, current_y, iters))
```

Some example output

```
>>> hill_climb(10)
Best x: 1.0
Maximum y: 7.0
Iterations: 91

>>> hill_climb(-2)
Best x: 1.0
Maximum y: 7.0
Iterations: 31

>>> hill_climb(1)
Best x: 1.0
Maximum y: 7.0
Iterations: 1.0
```