Orthlogs and Paralogs

author: Julin Maloof
date: April 11, 2019

Teams

A few pointers

incremental: true

- You must look at the output and files that your commands create.
  - IT is necessary to make sure that the command is doing what you think it should.
  - example…
- Do not just copy and paste the commands from the lab manual.
  - You need to learn these commands and how they are working
  - Cut and paste without thinking may save you time now, but will hurt you in the long run.

Alignment methods

incremental: true

Why are sequence alignment methods important?

- genome annotation
- trying to determine gene function
- evolutionary history of the gene
- using sequences to determine taxonomic relationships
- and more…

GOALS

incremental: true

Goals:

- Familiarity with BLAST at the command line.
- Automating tasks with "for" loops
- Orthology and Paralogy

What does BLAST stand for?

Basic Local Alignment Search Tool

We already learned about `water` why do we also need to know about `BLAST`?

BLAST and Smith-Waterman compared

incremental: true

- Both are both local alignment algorithms
  - What does "local" mean?
  - When might a local alignment be preferred over global?
- Smith-Waterman is guaranteed to provide the best possible local alignments, but this is a time intensive process
- BLAST may not provide the best possible alignment but is faster.
- When would you chose one or the other?

Orthologs and Paralogs

incremental: true

- Two genes are homologs if they have descended from a common ancestor.

- We can distinguish different kinds of homologs:

  - Orthologs are homologs that arose by speciation.

  - Paralogs are homologs that arose by gene duplication in a species.

(Illustration on board)

For more info see paper by Fitch

For loops

incremental:true

Often it is necessary to repeat a computational task many times with subtle variation or to perform the same task on a large number of objects.

For example, you might want to:

- Run `water` many times but use different scoring matrices or penalties or queries
- In today's lab run `BLAST` many times with different word sizes
- You can also imagine wanting to `BLAST` a gene against many different genomes, each separately.
- There are many other examples that you will encounter…

For loop example

incremental: true

Imagine that we want to use a computer and robot to automatically measure the weight of every student in the room.

First lets describe in detail what the steps would be for a single student. We use "pseudo-code" to describe what we want our program to do for a single student.

```
pick up the student
bring them to the scale
record their weight
return them to their seat
```

For loop example: pseudo-code

How would we describe the process of measuring all the students?

```
for each student in the classroom:

    pick up the student
    bring them to the scale
    record their weight
    return them to their seat
```

Note the use of the word for. This is natural English in this context, but it is also why these are called `for loops` in computer languages.

In this example student is a variable that takes on a different value each time we go through the loop.

For loop example: real code

Lets try it! Unfortunately I don't have a robot or scale, so we will just have the computer pretend that it is doing the task and tell us what it is doing.

Type the commands below into the Linux shell to see what happens

First we create a list of all the students, contained in the variable `classroom_students`

```
set classroom_students Chris Carine Cole Surbhi Sankalp
echo $classroom_students
```

We use the `echo` command to confirm that the classroom_students list has been created successfully

For loop example: real-code continued

Now we run the loop. The the following into your terminal.

```
for student in $classroom_students
    echo "picking up $student"
    echo "bringing $student to the scale"
    echo "recording $student's weight"
    echo "returning $student to their seat"
    echo \n
  end
```

Note the use of `end` telling the computer the end of the loop.

Discussion Questions

incremental:true

- Like last week, part of tomorrow's discussion will be a discussion in which both I and your peers answer questions about this week's material
- Today go to canvas and complete the assignment…list at least one question. This is due by 6pm and will be part of the 5% "participation points".
    - Don't be shy about asking questions that you think might be "too simple". I won't judge and I won't include your name if I use your question.
- Tomorrow We will break up into groups to answer the most common questions. Attendance and participation will be part of the 5% "participation points"