Lecture 20 - Introduction to Bioinformatics

What does bioinformatics mean to you?

At a practical level, what is bioinformatics?

At a practical level, what is bioinformatics?

- installing tools
- figuring out how to use tools
- reformating files
- searching for patterns

Compiled vs. interpreted languages

- **Compiled** - source code is translated to computer-understandable instructions once prior to its use. The compiling process is specific to a given computer based on its hardware, operating system, etc. Examples include C, C++, and sometimes Java.

- **Interpreted** - human-readable code is read and evaluated by an interpreter each time it is run. However, the interpreter itself is a compiled program. Examples include Python, R, and Perl.

Source code vs. binaries

- **Source code** - this is often written in a compiled language and must be *compiled* on your machine. Many times these tools will have *dependencies* that may or may not be included with the source code. A common tool for distributing and compiling biocomputational tools is called `make`, usually three steps are required - `./configure`, `make`, `make install`.

- **Binaries** - these are previously compiled versions of the tool. These are operating system specific and sometimes may not work for your system because they were compiled on a different system. However, when these are available and work for you, it allows you to avoid the sometimes difficult compiling process.

Installing tools

This week one tool we will use is an aligner called `Muscle`. This tool is available for free on the creator's website:

`https://www.drive5.com/muscle/downloads.htm`

We'll also be using a pattern matching tool called `hmmer`. This tool is also available for free on the web:

`http://hmmer.org/download.html`

Installing tools - example

Often the tools come in a compressed format to make it easier to download. In Unix this compressed format is called a "tarball" because the function used to generate the compressed file is `tar`. Thes "tarballs" are often also "zipped" to further compress them. As a result many tools you download will have the file names like *toolname.tar.gz*.

We have to "unpack" these compressed files to use the tools. This is accomplished using two tools (`gzip` and `tar`).

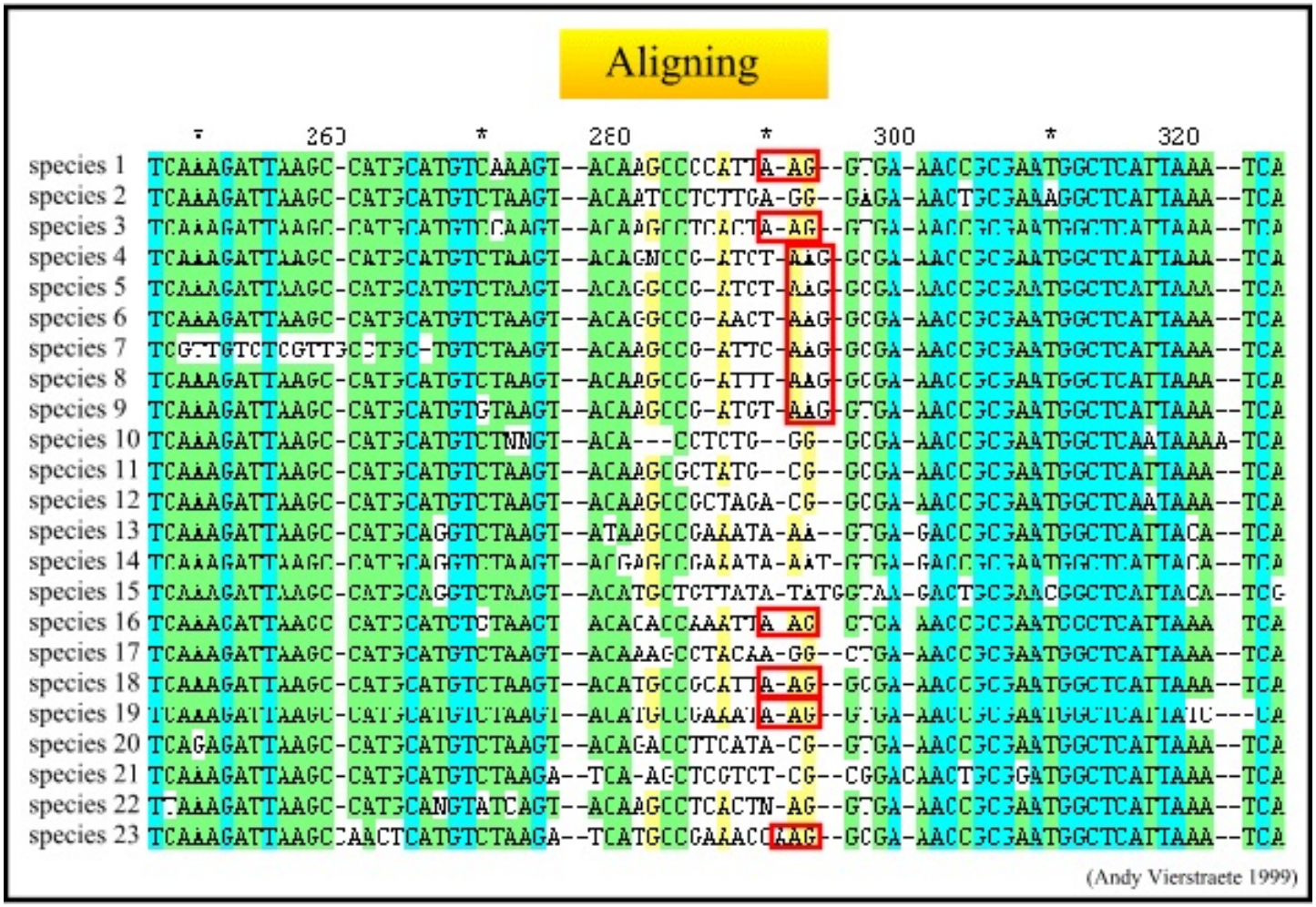
`gzip -d toolname.tar.gz`, which will "unzip" our file leaving us with a new file `toolname.tar`

`tar -xf toolname.tar`, which will "untar" our file leaving us with our tool or a directory containing components of our tool

How do we figure out how to use tools?

Sequence alignment

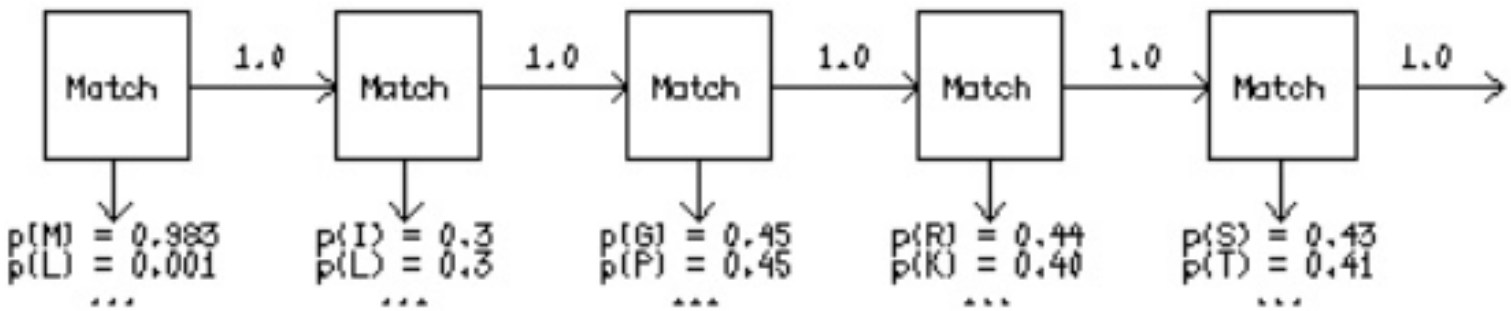
Sequence alignment attempts to evaluate the level of sequence conservation or *homology* amongst a set of sequences. Once aligned the sequences can be used to evaluate genetic change and to infer evolutionary history.



Profile Hidden Markov Models (HMM)

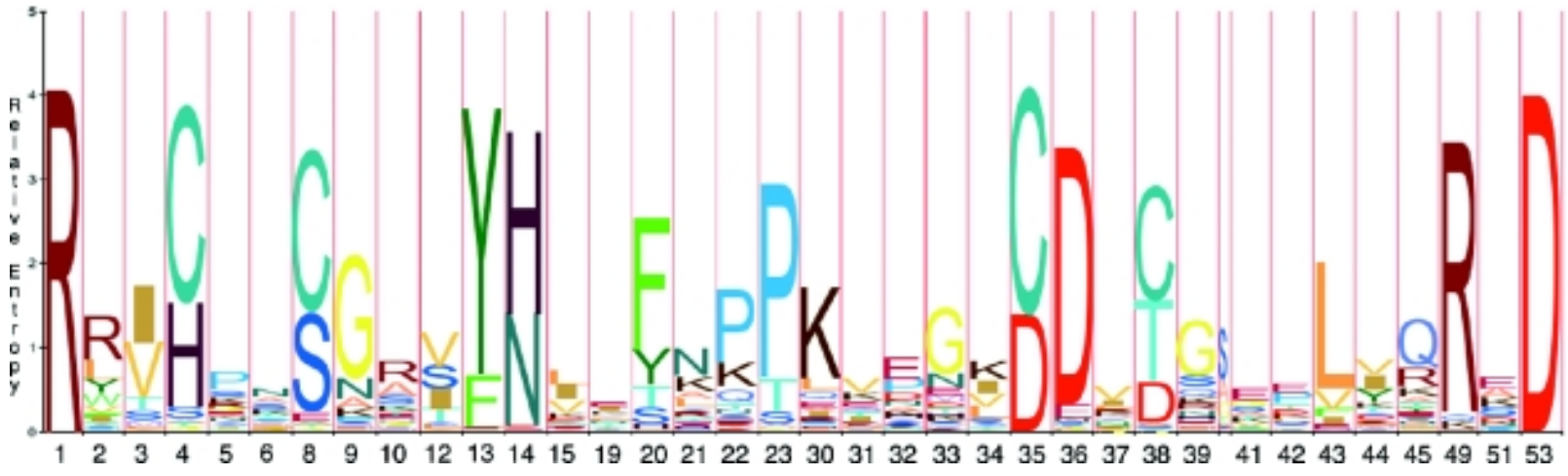
HMMs have been used to conduct flexible and fast sequence comparisons in a probabalistic framework.

For a simple model that does not allow for insertions or deletions, the probability of a given amino acid or nucleic acid is given.



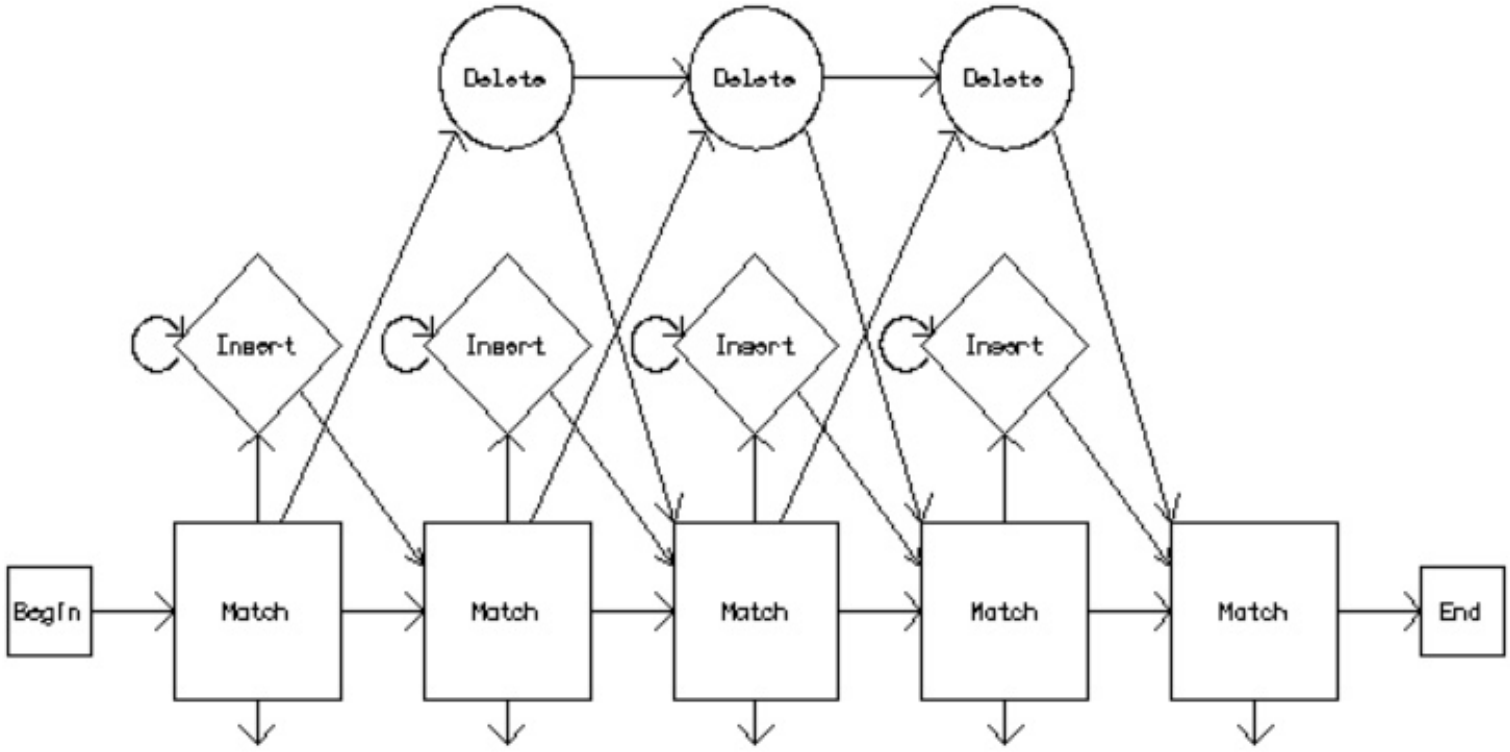
Profile Hidden Markov Models (HMM)

This simple profile HMM is also sometimes depicted as a sequence with the size of the residue proportional to its probability.



Profile Hidden Markov Models (HMM)

More complex profile HMMs allow for insertions or deletions of amino acids or nucleic acids.



Reformating files and searching for patterns

You've done some of this already!

What were some tools we could use to do this?

Wednesday

practice the use of these new tools, as well as using Unix and Python/R functionality to complete some bioinformatics

Download and "install" Muscle and hmmer3 for Wednesday

- download correct version for your OS; Muscle for Cygwin - use the Windows binaries
- run within Unix/Cygwin
- put them somewhere easy to find in Unix/Cygwin
- don't forget you may need a `./` to execute the command
- you know they work if you can run the version (-v) or help (-h) option