

**CMPSC 300
Bioinformatics
Spring 2021**

**Activity 09:
Sequence assembly and Investigation
Submit deliverables through the Google Form.**

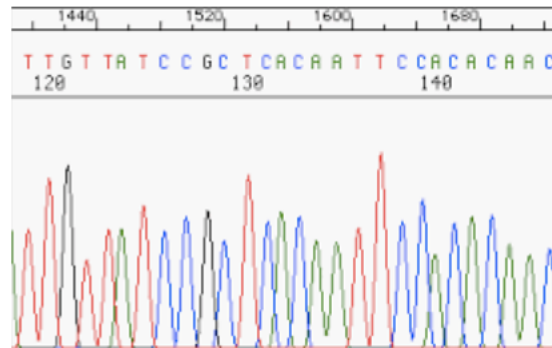


Figure 1: A DNA Sequence Chromatogram: This manual procedure to create strings of DNA text took a long time to complete. In the early days of Bioinformatics, a person's job in the lab was to use a magnifying glass to inspect the colored plots and to write down sequences as they were spotted. Now, this system has been automated and machines perform this inspection which means that our ability to sequence DNA takes less time per sequence sample.

Objectives

To learn how to use online tools in conjunction with Blast to investigate the origins of an unknown sequence of DNA. In particular, we will be studying how to use automated gene assembly techniques to reassemble an unknown sequence that will then be identified using BLAST.

Reading Assignment

Chapter 8 in the *Exploring Bioinformatics* textbook.

Introduction

As you know, the genome contains the complete instruction code to build all the proteins in the life of an organism from which it has been obtained. The genome is also a very specific biological signature which may be used to identify an organism. In addition, while this genome may be similar between related organisms, an individual's genome is very likely to contain mutations that may be used to describe how one particular organism is different from another. Using tools and methods from Bioinformatics, you can conveniently employ techniques of comparative genome analysis to determine genetic trends of the individual's family, as well as, to uncover patterns

of the individual's (unique) genetic properties. The study of these trends and patterns could be exploited for the purpose of classifying and identifying the organism and its family, in relation to other individuals and families.

Much of the DNA data that exists for research in Bioinformatics was generated using dye terminator sequencing technologies. Previously, it was a time-consuming task to obtain samples of DNA as text to study from unknown organisms. The work to isolate the molecular DNA and then determine its base-composition using machines was still highly manual task and may have taken months to years to create a reliable sequence of DNA for research. More importantly, even once the “final” genome sequence had been sequenced (i.e., a textual record was made of the order of its nucleotides), errors were likely to have been found during quality control which may have required a fresh re-sequencing task.

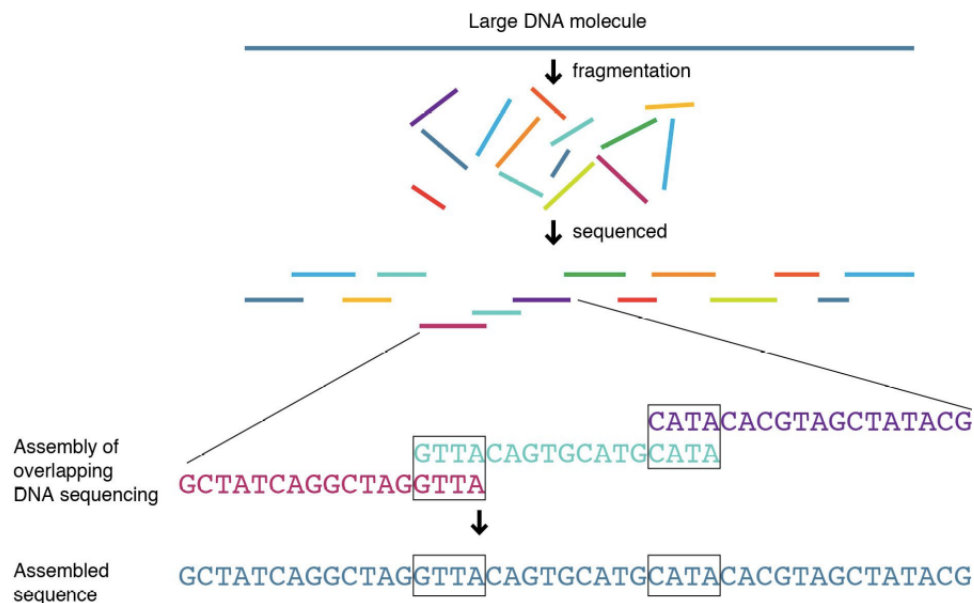


Figure 2: **Genome Sequencing and Assembly:** The process starts with the *reads* (strings of text) that are obtained from DNA samples (molecular material), are combined to make longer strings of DNA material called *contigs*. These larger fragments of text are further combined into a single DNA string similar to completing a jigsaw puzzle. Overlapping pieces of the fragments are very important and are used to determine the order that they are placed to make the DNA genome.

Advances in Sequencing Technologies

More recently, advances in DNA automated sequencing technologies have been made that significantly improved our ability to quickly and efficiently collect DNA sequence data. Some of the highly automated technologies are collectively referred to as *Next Generation* sequencing and can offer completed sequencing tasks in seemingly hours or minutes. However, one disadvantage is that

the length of a given sequence generated using *Next Generation* sequencing technologies is that the reads (i.e., the fragments of DNA which serve as a starting point to a gene assembly project) are typically smaller than the average lengths of read sequences generated using former technologies such as the dye terminator sequencing. This means that while *Next Generation* sequencing technologies can offer faster results, researchers are faced with many smaller reads which must be combined to re-create the original sequence of DNA.

The amount of information contained in DNA does matter in terms of sequencing. For instance, the smallest known genome belongs to the *Porcine circovirus* organism and is just 1759 bp (base pairs) in length. This contrasts with the largest known genome that belongs to the *marbled lungfish* and is 130,000,000,000 bp. Even the smallest known genome (and clearly the largest known genome) would contain too much information to process for a single DNA sequencing read operation.

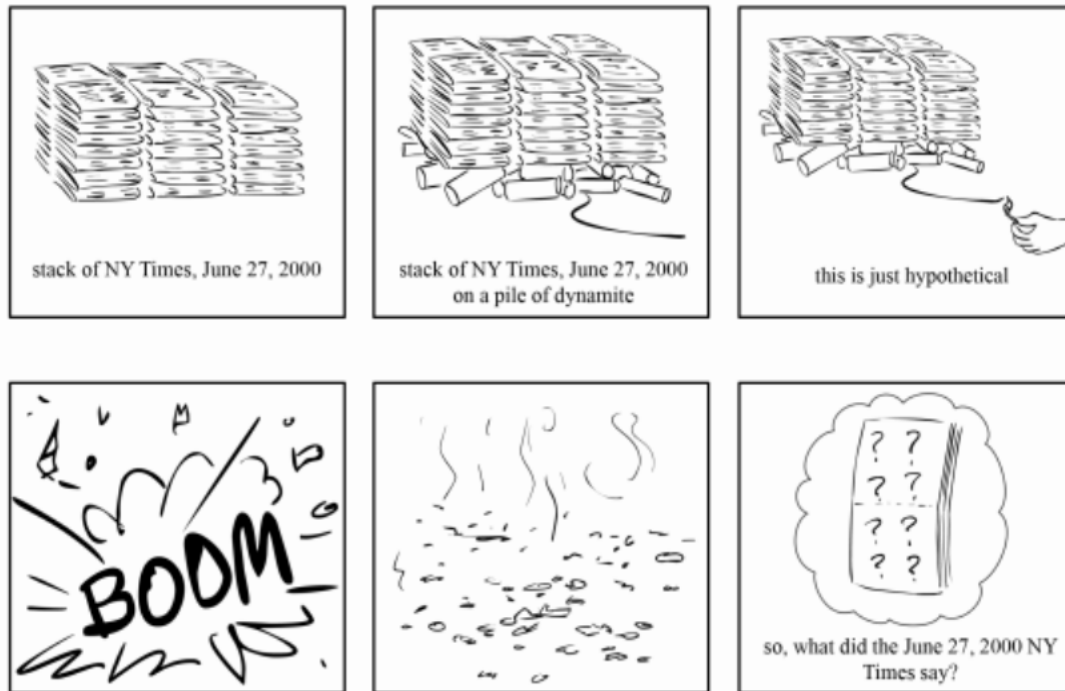
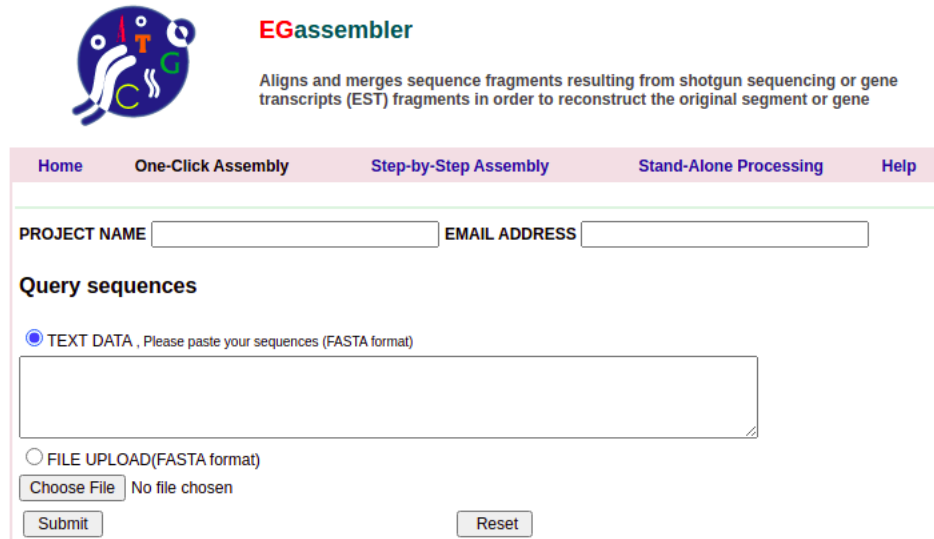


Figure 3: Exploding newspaper analogy

It's Like Working with Jigsaw Puzzles

The solution to this challenge is shown in Figure 2. The first step of a genome sequencing project is to isolate many copies of the target genome and then break the large genome into smaller pieces, randomly. These smaller pieces are then sequenced and the genome may be reassembled based on the overlapping nucleotides present in each sequence that are used to establish an order. This process has been compared to putting together a jigsaw puzzle or trying to read the New York Times after it has been blown apart into millions of pieces as noted in Figure 3.

What To Do



The screenshot shows the EGAssembler web interface. At the top left is a logo with a stylized 'G' and 'E' and a DNA helix. To its right is the text 'EGAssembler' in red and green. Below this is a description: 'Aligns and merges sequence fragments resulting from shotgun sequencing or gene transcripts (EST) fragments in order to reconstruct the original segment or gene'. The main interface has a pink header with five tabs: 'Home', 'One-Click Assembly', 'Step-by-Step Assembly', 'Stand-Alone Processing', and 'Help'. Below the tabs are two input fields: 'PROJECT NAME' and 'EMAIL ADDRESS'. Underneath is a section titled 'Query sequences' with two radio buttons. The first is selected and labeled 'TEXT DATA , Please paste your sequences (FASTA format)', with a large text area below it. The second is labeled 'FILE UPLOAD(FASTA format)' and has a 'Choose File' button next to it, with the text 'No file chosen' below the button. At the bottom are two buttons: 'Submit' and 'Reset'.

Figure 4: **The main screen of the the EGAssembler online web tool.** If you are having trouble with your analysis, please first visit the help tab which may offer you quick solution.

For today's class activity, you will use **EGAssembler** from link: <https://www.genome.jp/tools/egassembler/> to assemble the genome sequence of an unknown organism, shown in Figure 4. The raw data you will assemble is contained in a file called, `data/reads-fasta.txt` (note, this file is found in the `data/` directory of today's lesson directory in `classDocs/`). This file contains about 2,500 sequences in the FASTA format representing the fragmented genome of the unknown organism. These sequences range in length from 100 to 500 bases and contain between 1 and 10 random substitutions or single-nucleotide deletions each, representing the errors inherent in sequencing data. Your task is to follow the below steps to uncover what this unknown virus is and to learn more about its relatedness to other viruses.

Survey Questions

You will be responding to question-in-blue in the below steps. Please place your answers in the Google Doc which can be found at the link:

<https://forms.gle/4Fzs4coKTTuayEKF9>

Steps

1. Download the `data/reads-fasta.txt` file in today's lesson directory of the `classDocs/` repository.

2. Navigate to the *EGassembler* website and either upload or copy and paste the sequences from the reads text file into the input field. You could equally use the form's upload feature to upload the file. *EGassembler* includes software to scan for low-quality sequence (i.e., sequences containing many *N*'s, indicating unreadable nucleotides in the sequence) and remove sequences matching databases of other DNA sources (i.e., organelles such as mitochondrial DNA in a human nuclear genome project) as well as highly repetitive sequences.
3. For our purposes, turn off the options other than sequencing cleaning and the assembly step itself by unchecking the box next to the word "enable" for repeat masking, vector masking, and organelle masking. Run the program. You should immediately see the results of sequencing cleaning; You can view a `.cln` file to identify reads that were discarded and then examine these reads in the original sequence file.

In a few minutes, the results should become functional (i.e., the links become active). From the results page, you can view;

- The contig or contigs that resulted from the assembly of your sequence reads;
- Any "singletons", which are reads that could not be assembled into the contigs or that were not used in creating the contig; and
- An alignment of the individual sequence reads showing how they led to the generation of the consensus contig sequence.

4. Process 1 describes the *Sequence Cleaning*.

Looking at the text on the results page, how many sequence reads were rejected in the sequence cleaning process? Why do you think they were rejected?

5. Download the `.contigs` file and view the contents in an editor. Use BLAST (*Nucleotide*) to compare your contig sequence(s) with known sequences in Genbank (note: you can copy and paste sequences into BLAST directly). The assembled sequence should match a known sequence with a high degree of similarity.

According to the results from BLAST, what genome have you just assembled?

6. Because next-generation sequencing produces random short reads, there is no guarantee that even 2,500 reads would be sufficient to completely sequence a particular genome.

Did you find that the sequence reads that were assembled cover the entire genome, or did gaps remain? Please explain. *Hint – You could click on the genome accession number to determine the length of the previously published genome. Equally, you could check the Graphic Summary on BLAST's results page and look for breaks in red lines to indicate mis-alignments between sequences.*

7. You used the default parameters in your *EGassembler* analysis. In a real sequencing project, however, you might want to change variables in the Options such as the *Overlap percent identity cut-off*, which is the minimum percentage of nucleotides that must be identical in the overlapping region of two fragments. By default, the assembler is quite tolerant of sequencing errors, and also automatically compensates for some of the common problems of high-throughput sequencing such as as low-quality sequence at the beginnings and ends of fragments.

Due: 19th April, by 10:50am

6

To see how these parameters affect the assembly, in the *Enable Sequence Assembly Process Options*, try setting the overlap percent identity cut-off to 100%. What change was there to your contig results? Did the quality of your alignment change?

Required Deliverables

- Complete Google Doc survey containing the questions-in-blue above. See above link.

Grading

The grade that you receive for this lab assignment will be based on the following:

- This activity is a check-mark grade.

Please see the Technical Leaders or the instructor if you have questions about the assignment submission.