COMP BIO MINI-PROJECT
15pts
Due via Sakai by 10:00PM March 4, 2021

Description: The mini-project will focus on integrating some of the software tools discussed in class. Students will work *independently* to develop a Python wrapper to automate the execution of the software tools. You will create a GitHub repo and post your code there (Introduction to GitHub presented on 2/18 in class). You should include straightforward documentation and sample data in your repository. The code should have adequate documentation in the repository's README.md file such that anyone could download the code, install it or get it to run, and run through the test data without encountering any problems; this should be thought of as a User's Manual. If we cannot run the code with the test data, we cannot grade the functionality of the code. This means that you cannot hardcode paths in your code!

The code will be graded as follows:
• **3 pts.** Code comments
• **3 pts.** Documentation -- including what tools need to be installed and how to use the code
• **1 pts.** Test data (include a ***small subset of input reads*** so we can test your code quickly (total runtime < 5 min), but still include enough reads so the entire pipeline runs. Note, individual files in your GitHub repo must be less than 50MB.)
• **8 pts.** Functionality (include ***miniProject.log*** in your GitHub repo with the requested output from running your pipeline with **all input reads**, so we can check your answers. The full run will take a few hours, so don't wait until the last minute.)

Submit the URL to your GitHub repo to Sakai.

Human herpesvirus 5 is also known as Human cytomegalovirus and is typically abbreviated as HCMV.
From Wikipedia: *Although they may be found throughout the body, HCMV infections are frequently associated with the salivary glands. HCMV infection is typically unnoticed in healthy people, but can be life-threatening for the immunocompromised, such as HIV-infected persons, organ transplant recipients, or newborn infants. Congenital cytomegalovirus infection can lead to significant morbidity and even death. After infection, HCMV remains latent within the body throughout life and can be reactivated at any time. Eventually, it may cause mucoepidermoid carcinoma and possibly other malignancies such as prostate cancer.*

Cheng et al. 2017 (https://www.ncbi.nlm.nih.gov/pubmed/29158406) produced the transcriptomes of HCMV post infection. Write a Python script to automate the following and produce the output file requested named "miniProject.log" and other output files in a folder named "miniProject_FirstName_LastName" [where you've indicated your first and last name]. ALL results generated by you or programs called should be written to this folder. The easiest way to guarantee this is to create the folder using an os.system() call and then move into it via an os.chdir() call.

1. We would like to compare HCMV transcriptomes 2- and 6-days post-infection (dpi). First, retrieve the following transcriptomes from two patient donors from SRA and convert to paired-end fastq files. You can use wget (by constructing the path based on the SRR numbers for each of these samples).
Donor 1 (2dpi): https://www.ncbi.nlm.nih.gov/sra/SRX2896360
Donor 1 (6dpi): https://www.ncbi.nlm.nih.gov/sra/SRX2896363
Donor 3 (2dpi): https://www.ncbi.nlm.nih.gov/sra/SRX2896374
Donor 3 (6dpi): https://www.ncbi.nlm.nih.gov/sra/SRX2896375

2. We will quantify TPM in each sample using kallisto, but first we need to build a transcriptome index for HCMV (NCBI accession EF999921). Use Biopython to retrieve and generate the appropriate input and then build the index with kallisto. You will need to extract the CDS features from the GenBank format. Write the following to your log file (replace **#** with the number of coding sequences in the HCMV genome):

The HCMV genome (EF999921) has # CDS.

3. Quantify the TPM of each CDS in each transcriptome using kallisto and use these results as input to find differentially expressed genes between the two timepoints (2pi and 6dpi) using the R package sleuth. Write the following details for each significant transcript (FDR < 0.05) to your log file, include a header row, and tab-delimit each item:

target_id test_stat pval qval

4. It has been proposed that HCMV disease and pathogenesis may be related to the genetic diversity of the virus (Renzette *et al*. https://www.ncbi.nlm.nih.gov/pubmed/25154343/). Which publicly available strains are most similar to these patient samples? To compare to other strains, we will assemble these transcriptome reads. We don't expect assembly to produce the entire genome, but enough to be useful in BLAST. Virus sequencing experiments often include host DNAs. It is difficult to isolate the RNA of just the virus (as it only transcribes during infection of the host cell). Before assembly, let's make sure our reads map to HCMV. Using Bowtie2, create an index for HCMV (NCBI accession EF999921). Next, save the reads that map to the HCMV index for use in assembly. Write to your log file the number of reads in each transcriptome before and after the Bowtie2 mapping. For instance, if I was looking at the Donor 1 (2dpi) sample, I would write to the log (numbers here are arbitrary):

Donor 1 (2dpi) had 230000 read pairs before Bowtie2 filtering and 100000 read pairs after.

5. Using the Bowtie2 output reads, assemble all four transcriptomes together to produce 1 assembly via SPAdes. Write the SPAdes command used to the log file.

6. Write Python code to calculate the number of contigs with a length > 1000 and write the # to the log file as follows (replace # with the correct integer):

There are # contigs > 1000 bp in the assembly.

7. Write Python code to calculate the length of the assembly (the total number of bp in all of the contigs > 1000 bp in length) and write this # to the log file as follows (replace # with the correct integer):

There are # bp in the assembly.

8. Write Python code to retrieve the longest contig from your SPAdes assembly. Use the longest contig as blast+ input to query the nr nucleotide database limited to members of the *Betaherpesvirinae* subfamily. You will need to make a local database of just sequences from the *Betaherpesvirinae* subfamily. Identify the top 10 hits. For the top 10 hits, write the following to your log file: Subject accession, Percent identity, Alignment length, Start of alignment in query, End of alignment in query, Start of alignment in subject, End of alignment in subject, Bit score, E-value, and Subject Title.

Include the following header row in the log file, followed by the top 10 hits, and tab-delimit each item:

sacc pident length qstart qend sstart send bitscore evalue stitle