BPSM Assignment 1

Due 14:00 Mon 14 October 2019

Background to the technology

Next Generation Sequencing (NGS) is the broad term used to describe the process whereby millions of sequence reads (or read-pairs) are generated from samples at a comparatively low cost. If you'd like to find out more about the technology and its uses, https://www.illumina.com/ is a good place to start!

RNA-Seq (RNA sequencing), as a generic technique, enables us to assess the levels of gene transcripts (usually mRNA molecules) in a group of samples, perhaps to see how gene expression levels change over time, or, perhaps to answer a question like: "what happens when drug X is administered to cultured cells: how do the cells respond at the gene level, and what role do those genes play?".

As you can probably guess, there are actually many different applications for the technology, some of which are described here:
https://en.wikipedia.org/wiki/RNA-Seq.

Background to the biology

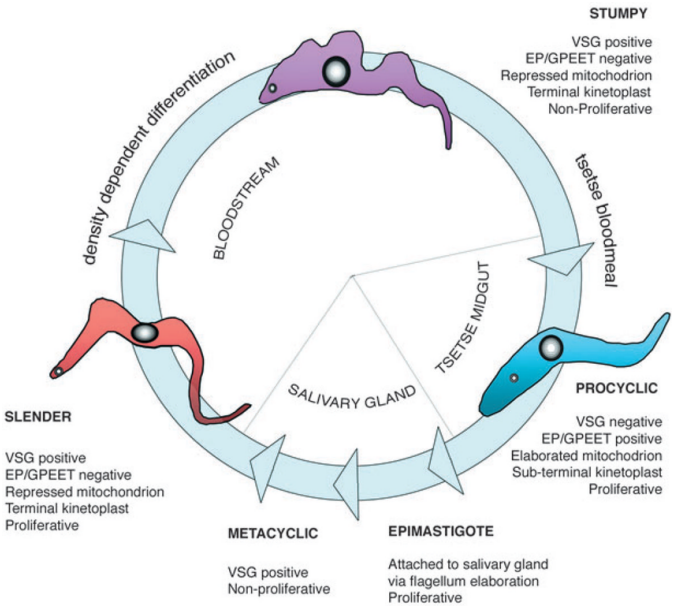
Trypanosoma brucei spp (comprising *Trypanosoma brucei brucei* and the human infective forms *T. b. rhodesiense* and *T. b. gambiense*) are eukaryotic protozoan parasites responsible for African sleeping sickness in 36 countries of sub-Saharan Africa, which are the poorest developing countries worldwide.

Sleeping sickness has a devastating impact on human health and prosperity: >0.5 million cases and 70,000 deaths per year are a result of the parasite and the disease can be fatal unless treated. The parasite also infects cattle and game animals (where it causes the disease "nagana").

The trypanosome is transmitted between mammalian hosts by the tsetse fly, *Glossina* spp, in which it initially establishes in the midgut after a bloodmeal but then migrates to the salivary glands in preparation for transmission to a new mammalian host (see image below). In mammals, the parasite survives free in the bloodstream, being able to evade antibody responses through antigenic variation.

Trypanosomes proliferate in the mammalian bloodstream as morphologically "slender" forms, these being replaced by non-proliferative "stumpy" forms as parasite numbers increase. The accumulation of division-arrested forms limits the increase in parasite numbers and thereby prolongs host survival (and hence the probability of

disease transmission back into the insect host for the cycle to continue).



https://www.researchgate.net/publication/8077434_The_developmental_cell_biology_of_Trypanosoma_bruce

Your task

You have been asked to have a look at some RNA-Seq data that have been generated from *Trypanosoma brucei brucei*. The research lab is particularly interested in whether there are any differences in gene expression levels in two different life cycle stages: "slender" and "stumpy". There are currently three biological replicates for each of the two conditions, and the technology used was paired-end sequencing.

There might be more biological replicate samples coming, so the analysis might need to be done all over again with the bigger dataset, but we don't know for sure yet. If there are more data, they will get put in the same directory.

There are online tools that could be used (at least in part) to process and analyse this kind of data (e.g. https://usegalaxy.org/), but in this instance, we are going to do all the processing from scratch using the command-line interface (i.e. in a terminal) on our MSc course server: the goal for this assignment is to write a pipeline programme that, when executed in a Unix terminal on our MSc server, will process the DNA sequence data.

During the development of your pipeline code, you should use git for version control as detailed in the git/GitHub lecture. The git repository is a major part of this (and the next) Assignment, so do not neglect this component...!

The paired-end sequence data are provided in the directory
`/localdisk/data/BPSM/Assignment1/fastq`

The sample details are also there, in a file called `fqfiles`.

The pipeline should have the following minimum overall design components/modules:

1. perform a quality check on the paired-end raw sequence data (which are in gzip compressed fastq format; https://support.illumina.com/bulletins/2016/04/fastq-files-explained.html) using the installed programme `fastqc`

2. assess the numbers and quality of the raw sequence data based on the output of `fastqc`

3. align the read pairs to the *Trypanosoma brucei brucei* genome using installed programme `bowtie2`, converting the output to indexed "bam" format with `samtools`; the *Trypanosoma brucei brucei* genome sequence is provided in fasta format in the directory `/localdisk/data/BPSM/Assignment1/Tbb_genome/`

4. generate counts data: the number of reads that align to the regions of the genome that code for genes; this is to be done using the installed programme `bedtools` and the `Tbbgenes.bed` "bedfile" that contains the information about the gene locations in the genome. The gene names are in the 4th column of the "bedfile". The bedfile is provided in the directory `/localdisk/data/BPSM/Assignment1/`; for the purposes of this analysis, you should assume, **incorrectly**, that all genes have no introns.

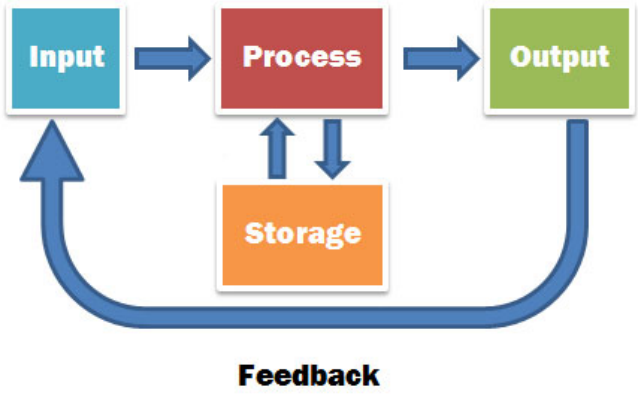
5. generate a single plain text tab-delimited output file that gives the statistical mean (average) of the counts per gene for each group, e.g. gene name, mean for slender samples, mean for stumpy samples.

Considerations

1. don't panic: it isn't as bad as it first sounds!

2. **DO NOT** immediately try to start coding! This is a pipeline made up of various component "bits": try drawing it out as a flowchart first on a piece of paper, so that you can clearly understand what the inputs and outputs might be for each "bit" that gets "stuff" done to it. This is time very well spent...

My favourite picture:



3. most programmes have many parameters/flags that can be set by the user: the defaults are **not** always the best, so see what is available: it might make it run faster/better

4. most programmes have a "-h" or "--help" for help

5. try to make it so the user doesn't actually have to do much, other than execute the programme

6. the "genome sequence" actually comprises many sequences, and will need to be made into a "bowtie2" database before it can be used

7. normally, when you get raw sequence data, you'd get lots more that this, and each sample wouldn't have the same number of read-pairs...

8. Google is your friend

9. the demonstrators and I, we are your friends too, but we really can't tell you the answers, no matter how politely you ask, sorry...!

10. doing this Assignment should use some/most/all/more of the things we cover in the Unix/awk/scripting/git parts of the course, so that includes variables, loops, and so on. The pipeline programme/code doesn't need to be run on eddie3, so just concentrate on getting it to run on the bioinformatics server!

What should be submitted for this assignment

There are two things that need to be completed for this Assignment.

1. a GitHub repository containing a pass-worded zip file called Bxxxxxx-2019.Assignment1.zip, as detailed in the git/GitHub lecture. The zip file should contain the pipeline programme/code for me to run on our server, as well as the full git log of your Assignment1 coding project (the `all_the_things_I_did` file). The pipeline programme/code itself should contain lots of "comments" so than anyone looking at the code knows what each bit is doing

2. a PDF file called Bxxxxxx-2019.Assignment1.pdf, submitted via Learn, that:
    - gives the link for your GitHub Assignment1 repository (e.g. https://github.com/Bxxxxxx-2019/Bxxxxxx-2019.Assignment1)
    - gives the password to open the Bxxxxxx-2019.Assignment1.zip file ...!
    - has an overview/flowchart that describes how the pipeline processes the data
    - briefly indicates why you have chosen the parameters/flags that you have for each step
    - indicates any things the user will have to do to make things work
    - highlights any difficulties, if any, that you have come across
    - indicates any additional/alternative features that you think might be beneficial to include

What I'll be looking for (in no particular order or weighting)

1. successful use of git and GitHub

2. a pipeline programme/code that works and produces the requested output in the correct format: note that it must be your own work (but by all means discuss things with your colleagues)

3. comments in the code so that I can see what it does

4. any additional "flexibility" or features that have been added/suggested that indicate to me that you understand what is being done/not done AND why your pipeline helps biology (this is **BIO**informatics after all!)

What I would rather not see...

I will have over 50 assignments to mark/run programmes for, so....

1. (please) keep the PDF content "focussed"! It is quality, not quantity, that is more important ...

2. the pipeline itself can actually be written using quite a small number of (carefully constructed) commands, so if your programme is hundreds and hundreds of lines, you should probably check it is doing the "right" thing!

3. GitHub has a size limit of ~100Mb, so don't even think about including the raw data in your repository! We know where the raw data are/might be, accessible using the full path to the files!