Bi/BE/CS183 2020
Profs. Matt Thomson and Lior Pachter
Problem Set 4
Due on Moodle by Thursday February 13th at 12:00pm

Homework policy: You may collaborate with anyone on the homework and consult/use any resources. Please write down on the homework who you worked with and explain what, if any, help you received. Also list all resources that were used for the homework and explain how they helped.

Problem 1 (20 points)

This question explores how Next Generation Sequencing works (a.k.a. NGS, Illumina Sequencing), which currently is by far the most widely used sequencing technology today. It is important to understand how the technology works and to be aware of its capabilities and limitations. This is helpful not only when analyzing data, but especially when designing experiments.

a) Explain how NGS Sequencing by Synthesis works.

b) Discuss the MiSeq, HiSeq and NovaSeq Illumina sequencing platforms. Take into account how many reads per run are possible, the read length, and instrument cost.

c) Estimate the sequencing costs per run (cost per million bases and total run cost) on each one. Find out how much it would cost if you had to run them for an experiment (e.g. buying the instrument and running it yourself, using a core facility, or using a sequencing service). The simplest way to do this is to consider the cost of a run is the price of the reagent kit sold by Illumina (see one of the links below for NovaSeq kit prices). You'll need to sign up in order to view retail prices. Keep in mind there are different prices for different read lengths (number of cycles)

Some resources you may find helpful:
https://www.illumina.com/systems/sequencing-platforms.html
Illumina Sequencing by Synthesis on YouTube: https://www.youtube.com/watch?v=fCd6B5HRaZ8
Illumina whitepaper introducing their sequencing technology: https://www.illumina.com/content/dam/illumina-marketing/documents/products/illumina_sequencing_introduction.pdf

Illumina kit prices for NovaSeq:
https://www.illumina.com/products/by-type/sequencing-kits/cluster-gen-sequencing-reagents/novaseq-reagent-kits.html

Problem 2 (30 points) Single-cell RNA sequencing (often abbreviated scRNA-seq) is enabled by several technological advances. The underlying idea is the same: to barcode transcripts according to which cell they came from, but there are many ways to achieve this. We have uploaded on moodle the 2018 review article *Exponential scaling of single-cell RNA-seq in the past decade*, mentioned in class. Read it and answer the following questions:

a) Describe (in just a few sentences each!) the different principles used for performing single-cell RNA sequencing, as categorized by the review in figure 1a. These are:
i) Manual
ii) Multiplexing
iii) Integrated fluidic circuits
iv) Liquid-handling robotics
v) Nanodroplets
vi) *In situ* barcoding

b) Describe the steps for performing Drop-seq, one of the droplet based methods. Be sure to explain how the following things happen in the protocol: i) Second strand generation ii) Library construction iii) Barcoding

To help you with this question, we have uploaded to moodle the 2018 Annual Review *From Tissues to Cell Types and Back: Single-Cell Gene Expression Analysis of Tissue Architecture*. The information boxes on p. 33 as well as Figure 2 (it's actually a table) should help you answer these questions. This review is very comprehensive so don't be dismayed if you don't understand everything! Bring any questions you have to class, office hours or moodle. You may also want to consult the original 2015 Drop-seq paper, *Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets*, DOI:10.1016/j.cell.2015.05.00

If you'd like to do a deep dive and see how the nucleotide sequences are changing at each step of the library preparation, look up your the Drop-seq technology here: `https://github.com/Teichlab/scg_lib_structs`

c) Think back at bulk RNA-seq methods (remember that last homework you read the RNA-seqlopedia `https://rnaseq.uoregon.edu/`). Compare what experimental aspects are similar and different between bulk and single cell RNA sequencing. Keep the discussion short (one or two paragraphs).

Problem 3 (10 points)

Explain the Simpson's paradox. How can it arise when analyzing single cell data, and what should the analyst be mindful of to avoid it?

Note: You may want to read Lior's blog post discussing it.
`https://liorpachter.wordpress.com/2015/07/13/how-to-average-genome-wide-data/`

Problem 4 (20 points)

In this problem you will do something amazing, which usually requires many hours or even days on a computing cluster. You will start from the *raw single-cell data* just as the it comes out of the Illumina sequencer and process it. *Raw, fresh, juicy raw data*, as described in https://youtu.be/KrZ17PE7SfQ?t=99.

But you process it *on your computer* and obtain the cell barcode by gene count matrix. Processing the raw Illumina reads usually takes several hours or even days because the data is usually aligned using an algorithm such as Needleman-Wunsch. What enables you to process the raw data on your computer instead of a cluster is an order of magnitude speedup by using the concept of *pseudoalignment*: Instead of performing base pair level alignment, only exact matches of a certain length (the **k-mer** size) are considered.

For more information of pseudoalignment, see the kallisto original 2016 paper: *Near-optimal probabilistic RNA-seq quantification*, doi:10.1038/nbt.3519. Similar approaches have since been implemented in other tools, such as Salmon (https://combine-lab.github.io/salmon/), discussed informally in this blog post: http://robpatro.com/blog/?p=248

To do the processing we'll use **kallisto bus**, with usage documented at https://pachterlab.github.io/kallisto/manual. The **bus** option was added in **kallisto v0.45** and lets you process single cell data generated with various technologies, here we'll process 10x Chromium v2 data. Unfortunately there's a bug in **kallisto v0.45** that displays the version as v0.44 instead of v0.45, so to be sure that you got the right version installed run `kallisto bus --list` and you should see the following output:

```
List of supported single cell technologies

short name       description
----------       -----------
10Xv1            10X chemistry version 1
10Xv2            10X chemistry verison 2
DropSeq          DropSeq
inDrop           inDrop
CELSeq           CEL-Seq
CELSeq2          CEL-Seq version 2
SCRBSeq          SCRB-Seq
```

In addition to **kallisto v0.45** you'll also need another program called **bustools**, available for mac and linux at https://github.com/BUStools/bustools/releases. This program is necessary because **kallisto** will output the processed data as a `.bus` file which is in binary format and needs a specialized parser. The bus format is described in the 2018 preprint *The Barcode, UMI, Set format and BUStools*: https://www.biorxiv.org/content/10.1101/472571v2.

Usage of **bustools** is documented at https://github.com/BUStools/bustools, but hopefully you'll only need two commands: `bustools sort`, which sorts the binary `.bus` file, and `bustools text`, which converts a binary `.bus` file to a `.txt` plaintext file. For example, usage and output in the notebooks provided is as below:

```
> bustools sort -o ./out_hgmm_100/output_sorted.bus ./out_hgmm_100/output.bus
Read in 5933252 number of busrecords
All sorted
> bustools text -o ./out_hgmm_100/output_sorted.txt ./out_hgmm_100/output_sorted.bus
Read in 4847729 number of busrecords
```

Finally, because this is a species mixing experiment, with human and mouse cells, in order to align the reads with kallisto you need to build the index using both the human and mouse transcriptomes. This can be done by simply concatenating the GTF files for two transcriptomes and then building the indices (done in the Python notebook provided). However, because the concatenated transcriptomes result building an index about twice as big as usual, most desktop computers do not have enough RAM to build the human-mouse index quickly. To save you the hassle, we're providing the human-mouse index here (3.3GB download): https://gitlab.com/munfred/human-mouse-transcriptome-index/raw/master/human_mouse_transcriptome_index.idx.gz

You'll process the dataset **100 1:1 Mixture of Fresh Frozen Human (HEK293T) and Mouse (NIH3T3) Cells**, available at https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/hgmm_100.

The link for direct FTP download (we're not dealing with GEO and SRA *this time*) of the paired end reads (about 760MB) is http://cf.10xgenomics.com/samples/cell-exp/2.1.0/hgmm_100/hgmm_100_fastqs.tar

a) Process this data with kallisto bus by running (and adapting as you see fit) one of the notebooks below. Note that the notebooks are slightly different so it may be informative to read both and understand what they do.

Python notebook: https://github.com/BUStools/BUS_notebooks_python/blob/master/dataset-notebooks/10x_hgmm_100_python/10x_hgmm_100.ipynb
R notebook: https://bustools.github.io/BUS_notebooks_R/10xv2.html

b) Explain the process of mapping equivalence classes (ECs) to genes. The meaning of equivalence class in this context is discussed in this blog post from Lior:
https://liorpachter.wordpress.com/tag/transcript-compatibility-counts/

c) Do you think any of the steps in these notebooks could be improved? (not a trick question - we wrote them and want to make them better and clearer)

Problem 5 (20 points) Now that you have processed the raw sequencing data and have the barcode gene matrix at hand, we can do some exploratory data analysis. However, processing the same dataset with just 100 cells would be no fun, so you'll adapt the notebook from problem 4 to process another 10x v2 chemistry dataset of species mixing, but with about 1000 cells. This dataset is called **1k 1:1 Mixture of Fresh Frozen Human (HEK293T) and Mouse (NIH3T3) Cells** and is available at https://support.10xgenomics.com/single-cell-gene-expression/datasets/2.1.0/hgmm_1k.

The link for direct FTP download of the paired end reads (about 6GB) is http://cf.10xgenomics.com/samples/cell-exp/2.1.0/hgmm_1k/hgmm_1k_fastqs.tar

a) Visualize the number of counts per cell. Choose a cuttoff (for what you think are real cells) that you deem suitable and explain why you chose it.

b) Plot all cells above your chosen cutoff. On the x axis show the number of mouse gene counts and on the y axis the number of human gene counts. Do you think there are human mouse doublets in this data? Explain.

c) Plot the number of total counts per cell on the x axis and fraction of human/mouse transcript counts on the y axis. Is there a clean cutoff separating them? How can you explain this plot? (e.g. by considering the difference in size between the human and mouse cells used - you should look them up to understand what might be relevant for this question)

d) Perform PCA on this data and make plots of $PC1 \times PC2$, $PC2 \times PC3$, $PC1 \times PC3$. Discuss what you see. Why is it important to plot other principal components than just, e.g., $PC1 \times PC2$?