RNA-SEQ COMPUTATIONAL WORKFLOW:
GENERATING COUNT MATRICES FOR DIFFERENTIAL GENE EXPRESSION ANALYSIS

VERSION 2, 5/21/18

i Useful terms

- **RNA-Seq**. A Next-Generation Sequencing application to study and quantify transcriptomes. This can be performed for single-cells or whole tissues, including primary cells and cell cultures. There are many variation on the technique, but the biggest distinction is whether the sample contains protein coding transcripts only (**mRNA-Seq**) or the whole transcriptome (i.e. all RNAs, including rRNAs, tRNAs, siRNAs, lncRNAs). The Stevens lab is mostly interested in mRNA-Seq, since this kind of data could be directly correlated to proteomic data (in theory). Being a Next-Generation Sequencing technology application, RNA-Seq data is in the form of short reads (ranging from 50 to 300 bp).
- **GEO – Gene Expression Omnibus**. The data depository where almost all gene expression data can be found.
- **HPC cluster– High Powered Computing cluster.**

ii Important considerations before you begin

- It is vital to know what your data looks like. Hence, using either a) the published paper or b) the GEO database, you must identify whether your reads are single-end or paired-end. You must choose the correct program that takes the proper parameters into consideration.
- Text highlighted in `GREY` indicates that you should type said text onto the command line and hit 'Enter'
- Text highlighted in `BLACK` indicates files and text/code as it appears on your screen when using the terminal.
- Make sure the following programs are downloaded and installed, and set in your path:
    - Sratoolkit
        - Used to convert .sra to .fastq with the 'fastq-dump' command.
        - Find more information at https://github.com/ncbi/sra-tools
    - FastQC
        - Used to do quality control analysis of the raw and trimmed fastq data.
        - Find more information at https://www.bioinformatics.babraham.ac.uk/projects/download.html
    - TrimGalore!
        - Used to quality trim and remove adapters from raw fastq data.
        - Find more information at https://github.com/FelixKrueger/TrimGalore
    - Miniconda
        - A dependency of TrimGalore!
        - Find more information at https://conda.io/miniconda.html
    - Cutadapt
        - A depdency of TrimGalore!
        - See TrimGalore! Github page for details
    - STAR

        - Used to map reads from fastq data to a reference genome
        - Find more information at https://github.com/alexdobin/STAR
    - samtools
        - Used to convert binary read mapping files to gene count matrices
        - Find more information at https://github.com/samtools/samtools
    - DESeq2
        - A Bioconductor statistical R package used to perform differential gene expression analysis (and other analyses) using gene count matrices
        - Find more information at https://bioconductor.org/packages/release/bioc/html/DESeq2.html

Logging onto the HPC nodes

There are two login nodes which you will use. The first, 'hpc-cmb.usc.edu' is where you will submit all your jobs to the cluster. The second, 'hpc-transfer.usc.edu' is where you will download online data from. Note, do NOT download or transfer data on 'hpc-cmb.usc.edu.' The data will not download and you will be using too much RAM on a node that's shared by all of USC's Computational Biology department. For this exercise, you will be using my HPC cluster account.

1) `ssh hpc-cmb.usc.edu`
2) Enter the password.
3) Voila! For hpc-transfer.usc.edu, simply type

    `ssh hpc-transfer.usc.edu`

    and enter the password.

NOTE: MobaXTerm remembers previous sessions and their login credentials, so just click on the hpc-cmb.usc.edu or hpc-transfer.usc.edu session if you are using the virtual terminal.

Downloading RNA-Seq pipeline from GitHub

1) While logged onto hpc-transfer.usc.edu, navigate to your staging directory with the `cd` command.
2) Download the RNA-Seq_pipeline repository from David Manahan's Github account:

    ```
    git clone https://github.com/dmanahan/RNA-Seq_pipeline.git
    ```

3) Navigate to the new RNA-Seq_pipeline directory with the `cd` command and then make all shell scripts executable:

    ```
    chmod 775 *sh
    ```

Downloading and building genome references

1) While logged on to hpc-transfer.usc.edu, navigate to the RNA-Seq_pipeline directory using the `cd` command and execute the downloads_references.sh script:

```
../download_references.sh
```

2) Log out of hpc-transfer.usc.edu and then log on to hpg-cmb.usc.edu and navigate back to the RNA-Seq_pipeline directory using the `cd` command.
3) Execute the build_references.sh script in the 'RNA-Seq_pipeline' directory, which uses the downloaded genome data to build annotated references to map reads to during RNA-Seq analysis.

```
./build_STAR_indices.sh
```

Downloading RNA-Seq datasets

START HERE IF THIS IS NOT YOUR FIRST TIME USING THIS PIPELINE

Create your accession list.

1.) Identify the accession number assigned to the RNA-Seq dataset deposited in GEO. The number is usually found at the end of the paper, and generally starts with the letters 'GSE.' Ctrl + F function for 'deposit' should help your search.
2.) Go to GEO's website, https://www.ncbi.nlm.nih.gov/geo/ (Googling GEO NCBI works too). Enter the full-length GEO accession number into the search bar. A results page should appear corresponding with your entered dataset.
3.) Scroll to the bottom of the webpage until you reach the 'Relations' section. Open the 'SRA' link (usually the link starts with the letters 'SRP').
4.) On this page, find the 'Send to:' drop down menu in the upper-right hand coner. Under 'Choose Destination' select 'File.' A new drop down menu will appear asking you to select a format. Choose 'Accession list' and then 'Create File.' This should download the list as 'SraAccList.txt' and open the text file.
5.) Open the 'hpc-transfer.usc.edu' node and navigate onto the 'staging' directory. The home directory has very little room and likely will not be able to store all your files produced in a single analysis.
6.) While in the RNA-Seq_pipeline directory (just cloned from Github), create a new subdirectory where you'll store all the output files produced by this analysis pipeline. In this example, we'll call the directory 'sample_data'

```
mkdir sample_data
```

7.) Change your working directory to sample_data.

```
cd sample_data
```

8.) Create a file using a text editor. In this file, you will list all the names of the sequencing data files found in your chosen dataset on GEO.

```
nano accession_list
```

9.) This will open a blank page. Copy the contents of 'SraAccList.txt' into accession_list. Save the changes with 'ctrl + o' + 'Enter' and then exit the text editor with 'ctrl + x'

10.) Execute the download_SRAs.sh script to download your desired GEO .sra data into your working directory.

```
../download_sra.sh
```

This will download all .sra files from your study of interest in the sample_data directory. Proceed to the next step once the script finishes operating.

11.) Exit out of hpc-transfer.usc.edu

Set-up the RNA-Seq analysis pipeline

1.) While logged into hpc-cmb.usc.edu, open the RNA-Seq_analysis.sh script (located in staging/RNA-Seq_pipeline) in a text editor:

```
nano RNA-Seq_analysis.sh
```

2.) The first part of the script asks the user to define a few variables:

- Whether the library is composed of single-end (SE) or paired-end (PE) reads
- From what organism the RNA-Seq data was collected from (i.e. human, rat, or mouse)
- Whether the raw data is in the .sra or .fastq formats. Online data from public databases tend to be in .sra format, while data sequenced at the UPC Genomic Core will be already be in .fastq format.

3.) Save your changes with 'ctrl + o' + 'Enter' and then exit the text editor with 'ctrl + x'

4.) Navigate to the sample_data directory with the `cd` command and execute the RNA-Seq analysis script.

```
../RNA-Seq_analysis.sh
```

It should take about an hour. When the script finishes, you should have a directory with the following subdirectories:

FASTQ-DUMP
- Contains the .fastq files converted from their .sra format.

FASTQC
- Contains the FastQC output for quality control analysis.

TRIMGALORE
- Contains the adapter and quality trimming results as well as the adapter and quality trimmed .fastq files.

STAR
- Contains .bam output file from read mapping alignment.

HTSEQ
- Contains raw gene count matrices, converted from .bam format.

Protein-coding_matrices
- Contains raw gene count matrices that had non-protein-coding information removed. Raw gene count matrices and TPM-normalized gene count matrices are included. These data files are prefixed with "curated" to indicate that they only contain protein-coding information.

Glossary of commands, etc. for bash

*	Indicates a "regular expression." Used to define a search pattern (e.g. *sh means 'all instances ending in sh' and sh* means "all strings starting with sh')
`.`	Tells the terminal "look in the present working directory." Commonly used in paths, but also is used to inform the terminal that the script you want to execute is in your present working directory.
`..`	Tells the terminal "look in the parent directory." Commonly used in paths, but also is used to inform the terminal that the script you want to execute is in the parent directory of your present working directory.
`cat`	"Computer-aided translation." This commands prints the contents of a file onto your screen. Input is a filepath.
`cd`	"change directory." The input for this command is a path.
`chmod`	"change mode." This changes permissions for files and directores. Inputs include numerical permissions followed by a filename/directory. The most common use for chmod is to free permissions using the command `chmod 775 <filepath>`
`git clone`	Clones a repository from Github onto your present working directory. Input is the link to a repository.
`less`	This commands prints "less" than the cat command does, i.e. it only prints one page at a time. Use the less command instead of the cat command when you are dealing with large files. Input is a filepath.
`ls`	"list." List all the files and subdirectories in the present working directory (default). You can input a path to list the files in that directory instead.
`mkdir`	"make directory." Create a new subdirectory in your present working directory. Input is the desired name of your new subdirectory.
`nano`	This command opens a file using Nano, a text editor. If the filename does not exist, it will create a new and blank file. Input is a filepath.
`path`	A string of directory names that tells the terminal where in the file tree your directory or file is located. Paths can be relative (to your present working directory) or absolute (i.e. starts at the root of the file tree). To specify absolute paths, start with / .
`pwd`	"present working directory." This tells you where your current directory is in the file tree. No input required.
`ssh`	"Secure shell." A shell is a user interface for access to an operating's system. Secure shell is used to operate said system securely over an unsecured network. The command is used to specify what system you want to connect to.