Lecture 0 🡪 Overview - notes:

* Discord = communication software (similar to slack)
  + Instructions on how to download under ‘community’ tab
* all of the homework is available through DataCamp.com

Lecture 1 🡪 Introduction to RNA-seq technology and data – notes:

* the different illumina sequencers (MiSeq vs. NovaSeq) differ based on data output
  + data output = number of reads produced and time to complete sequencing
    - E.g., NovaSeq = 0.6-10 billion reads that takes 5 days to complete vs. MiSeq = 1-25 million reads that takes 1 day to complete
  + Same read length produced per machine more or less
* My samples were run on a NextSeq2000
  + 0.4-1.1 billion reads, 150bp paired end, 24 hour run timeA diagram of a diagram

    Description automatically generated
* Flow cell:
  + glass slide with nucleotide oligos that capture other oligos that you load onto flow cell
  + introduced oligos have bar codes at ends that attach to capture oligos
  + the capture oligos are extended (no fluorescent nucleotides yet) so that complementary to introduced oligos
  + introduced oligo discarded, then the extended capture oligo can bend over and attach at other end to complementary capture oligo on glass slide = bridge synthesis (can be controlled to generate specific sized clusters)
    - Forward and reverse strands
  + this leads to a colony of DNA = PCR colony or polony
    - So, basically, you introduce barcoded library, the capture oligos on the flow cell capture them, and PCR amplify the sequences
    - each colony has a cluster of identical DNA sequences
      * Get rid of the reverse sequences = colony of forward sequences
        + Prevent unwanted priming of the capture oligos from which the reverse strands are cleaved by blocking
      * If doing paired-end sequencing, will then have reverse strands copied, forward strands cleaved off and oligos blocked
    - each colony = ~5000 DNA copies in a 2um spot
  + can control density of the clusters based on how densely the flow cell is seeded with capture oligos
* What is a read?
  + From one PCR colony of forward strands, perform sequencing by synthesis to incorporate complimentary fluorescent nucleotides
    - These nucleotides block the addition of more nucleotides so that the camera can read the fluorescent, but this is reversible
    - The sequencer repeats this process, continuing to add more nucleotides to determine the sequence of the forward strand
    - The number of cycles determines the length of the read
  + *So, the read is the fluorescent nucleotide sequence read off of the forward (or reverse) strands adhered to the flow cell*
    - *Number of reads would then depend mainly on cluster density and number of clusters*
  + There are hundreds of millions of clusters per flow cell
* Each .fastq file is simply a text file
  + First line = sequence identifier
  + the sequence in the second line = nucleotide sequence of a single read
    - Is the sequence the nucleotide sequence of a read? i.e., each sequence is a read? YES
  + Third line = +
  + Fourth line = quality score of each assigned base
* Generally, sequencing more replicates is as good or better compared to increasing sequencing depth (i.e., generating more reads from your sequencing run)
  + This is especially true for highly expressed genes
  + For lowly expressed genes, increasing sequencing depth does help, but so does increasing number of replicates
* How many replicates should I include per experiment if looking at differential gene expression?
  + A screenshot of a computer

    Description automatically generated
    - Look at FC-threshold and green and red columns
      * If you include greater than 12 samples, you can use any tool to measure expression of any gene at any FC level
      * If you have 3 samples and your gene is expressed at a FC of 2 (WT to condition), then you can use DEseq, but if it is below 2, then no you can’t, but can if you include more than 3 samples
* How long should reads be if I only care about differential gene expression?
  + At least 50, but no real differences between 50 vs. 75 vs. 100
  + If you care about more than differential gene expression (e.g., alternative splicing), then longer reads are better

Lecture 2 🡪 Setting up software environment:

* What happens when you can’t use personal laptop for bioinformatic analysis?

A diagram of options for a computer

Description automatically generated

* + Option 1 = buy a better computer
  + Option 2 = use commercially available ‘cloud’ (e.g., Amazon)
  + Option 3 = universities or larger institutions will have their own computing clusters to which you can send jobs
    - steep learning curve = need to have additional command line knowledge to know how to send jobs to the cluster
    - SLURM = scheduler – organizes the jobs to be done on the cluster
* In class, we will be using R studio and R/Bioconductor for our analysis and a little bit of command line
* CRAN = comprehensive R archive network – where you go to download all of the packages associated with R
  + Mirrors = downloadable locations – in different areas of the world, there are other websites that can be used to download R-related packages more effectively than the ‘cloud’ which is location-less
* IDE = integrated development environment
* Packages in R fall into 3 main categories: modeling, statistics, graphing
  + Also Bioconductor as 4th category
  + modular = packages can work together
  + R packages = equivalent to scientific papers
* To install R/Bioconductor, used:
* if (!requireNamespace("BiocManager", quietly = TRUE))

install.packages("BiocManager")

BiocManager::install(version = "3.18")

* successfully installed all of the packages!

Lecture 3 🡪 Ultra-fast read mapping with Kallisto:

* if your computer has multiple cores, can spread a job out to different cores by specifying in command = MULTITHREADING
  + e.g., fastqc \*.gz -t 8
    - if you have 4 cores, each core has 1 virtual thread and one solid core = X2 🡪 so specifying to use all 8 cores for fastqc for all files ending in \*.gz
    - For my computer: fastqc \*.gz -t 8 (because have 4 cores so 8 threads)
* For fastqc, if doing RNAseq, sequence duplication isn’t necessarily bad as you expect some genes in the transcriptome to be expressed at higher levels and therefore for there to be some duplication
* *I did the fastqc for one sample – Daniel from the Beiting lab already did this for me – since I know how to do, stopped after one sample and moved on*
* .fasta file of reference genome is not an ideal way to align sample files to it, so need to make a reference *index*
  + making index is a computational laborious process = makes large file size
  + code to make index file:
  + *kallisto index -i inputFastaName.index inputFastaName.fa*
* mapping single end reads command:
* *A diagram of a single-end data

  Description automatically generated*
* summarize outputs of fastqc, kallisto, and many others with multiqc
  + over 128 software supported by multiqc
* For my data:
  + kallisto quant \
    - -i Mus\_musculus.GRCm39.cdna.all.index \
    - -o 941\_0 \
    - -t 2 \
    - --single -l 250 -s 30 \
    - 941-0\_S1\_R1\_001.fastq.gz \
    - >& 941\_0.log
* To automate shell script, create a document that has all of the code you want to run, save as .sh (not .txt), and run that
  + Dan has readMapping.sh as a shell script that performs fastqc and alignment to indexed referenced file as we did through hard coding in Parts 1 & 2 of module 2
  + Instructions on how to run shell script are included in the file itself
* Burrows-Wheeler Transform (BWT) in genomics – what does it do? it is a way to structure text/text files to make searching for patterns within them more efficient
* Seed and extend:
  + Most aligners will search for a seed instead of individual nucleotides
    - E.g., if have sequence CATTGA, instead of looking everywhere for ‘C’ and then extending, look for seed ‘CATT’ and extend from there
* What is pseudoalignment? narrows the space of possible reference sequences from which a possible query read may have originated
  + fast and memory efficient
  + It is a set of target sequences with which the read is compatible
    - Vs. alignment 🡪 how do the nucleotides in the read match with the nucleotides in the target sequence
  + Kallisto = not a base-wise aligner – aligns based on kmers of specific length (31 nucleotides), but does not do so based on base identity
    - Pros = faster
    - Cons = if interested in snps or nucleotides of interests, not an appropriate aligner
* A diagram of a map

  Description automatically generated
  + Can see that with Kallisto, you are not aligning read based on the nucleotide sequence specifically, but with which of these reference transcripts does the read match based on the kmers (i.e., not the sequences within the kmers even)

Lecture 4 🡪 Understanding RNAseq count data:

* You can create pseudo .bam files from kallisto and visualize in IGV
* can drag and drop .json files into sublime (text editor)
* Can also drag and drop .tsv into sublime to view:
  + 5 columns:
    - target\_id = ensemble id 🡪 can google to figure out actual gene symbol
    - length
    - eff\_length = effective length, always shorter than length
    - est\_counts
    - tpm = transcripts per million – if you were to sample 1 million reads from file, that is the number of transcripts you would expect to find
* what is the effective length of a transcript?
* A screenshot of a computer

  Description automatically generated
  + When talking about gene expression and length of transcript, really mean effective length
  + Why is this important?
    - Two sequences of same length, but depending on sequence composition, likelihood of aligning to one sequence vs. other could be different
      * Sequence with greater likelihood of being sampled = greater effective length
        + These are depend on factors that are sequence specific and others such as GC content
* RNAseq = **relative** measure of gene expression
  + normalization = analogous to compensation in flow cytometry
  + **highly expressed genes tend to have higher standard deviation 🡪 can globally alter data to accommodate this**
    - **e.g., log2 transform the count data**
  + large datasets 🡪 when fix one problem, another arises
    - you choose which problem you want to deal with when analyzing data
  + Normalize between samples and within samples 🡪 reads per kilobase per million reads or RPKM
    - per kilobase = per length of gene (gene A twice as long as gene B so has more reads mapping to it)
    - per million = per sequencing depth (gene A from sample 1 twice as much as gene A from sample 2 because 2X as many reads produced from sample 1 vs. sample 2
    - fragments per kilobase per million or FPKM is used for paired end data
      * So, if single end data, RPKM = FPKM
    - Generally will get small numbers, so arbitrarily multiply RPKM by 10^6, to get fragments per kilobase per million reads
  + Why is RPKM no longer used as normalization method?
    - **if you align 2 samples to same reference genome, the total number of reads or the ‘bucket’ for the 2 samples should be similar when normalized = *same average relative molar concentration***
      * **When using RPKM, this does not occur**
    - So, fix by using transcripts per million, or TPM
    - TPM:
      * read count/gene length = reads per kilobase or RPK
      * Add up RPK for all genes sequenced = total RPK
      * TPM = RPK/total RPK
      * fix = total TPM always adds up to 1
      * also will multiply TPM by 10^6 arbitrarily to get transcripts per million reads
  + For both TPM and RPKM, still cannot compare between samples
    - If have a gene that is wildly differentially expressed between 2 samples and a fixed amount of reads that can be sequenced per sample, the wildly diff. expressed genes will take up larger proportions of the total ‘real estate’ that can introduce further artifacts going from read counts to normalized read counts
      * i.e., gene that looks upregulated by read counts can then be downregulated by normalized read counts
    - To account for this, normalization methods essentially ignore genes that are highly variable between 2 samples 🡪 most normalization methods will use sets of genes with minimal variation/assume few differentially expressed genes to avoid this artifact
* R project =/= R script
* Using first script:
  + R project is more like a working environment that can/will include multiple R scripts
  + drag kallisto\_output folder into R studio
    - didn’t work for whatever reason – had to setwd() to the kallisto\_output folder, then create project
    - seems to have solved the problem
  + save R project in that same working directory
  + open up first script 🡪 Step1\_TxImport.R
  + In R studio, open, then go to Edit > Folding > Collapse all
    - all of Dan’s script written in collapsible chunks of code 🡪 analogous to book chapters
    - what makes things foldable? adding more than 3 –
      * e.g., #code chunk----
        + It will fold every line under the line with ---- until the next ---- is encountered
  + To install packages, can use regular command or ‘install’ tab in ‘packages’ window (i.e., where plots show up, except click ‘packages’)
    - Widen repositories to not just CRAN but also bioconduction
      * R> setRepositories(), then all of them separated by space, i.e., R> setRepositories() #then 1 2 3 4 5 6 7 8
      * Then go to ‘install’ button and type the packages Dan’s script calls for
  + He has the human indexed referenced genome, since the class files are human files – specifies in the notes that you can change this to your organism of interest
    - Website for this: <https://bioconductor.org/packages/3.18/data/annotation/>

Lecture 5 🡪 Starting your R workflow

* the conflict that appears when loading tidyverse:
  + There is another R package called ‘stats’ which has functions identical to dplyr within tidyverse
    - It is saying to resolve this conflict, it will only use the function from dplyr and mask the function from stats
  + another way to resolve is to specify from which R package you would like to use the redundant function
    - e.g., dplyr::filter() vs. stats::filter()
* explore datapasta package
  + Can paste in columns from a text or excel file into R studio as a vector, data.frame, etc.
    - Under ‘addins’ on R studio
* If you are trying to annotate transcript IDs in an organism not commonly studied (e.g., dog, Colorado spotted newt), can use biomart package 🡪 Dan demonstrates how to use this
* data frame = can have a combination of numeric and non-numeric data
* matrix = body of the table only have numeric data

*Generated the TPM table:*

* *checked expression of Igha – saw that KO still have expression, some as high or higher than HET*
* *Is there a way to visualize read abundance against genome to see where the transcripts are aligning to the gene?*
  + *Need IGV and .bam files?*
* *http://labshare.cshl.edu/shares/gingeraslab/www-data/dobin/STAR/STARgenomes/*

Lecture 6 🡪 Wrangling gene expression data:

* What are the different data types in R?
* A diagram of different types of data

  Description automatically generated
* When creating TPM table, you specified that you wanted gene\_id as an output
  + Can also specify transcript\_ID
  + because transcripts per million, when adding up TPM for collapsed gene\_id, they don’t quite add up to 1 million
    - if add up TPM for transcript table, it does add up to 1 million
* TMM = Trimmed mean of M-values 🡪 there is a paper that Dan provides that I can read if I wish to learn more about this
  + is the average read counts after removing the upper and lower x% of the data (assuming most genes are not differentially expressed)

Lecture 7 🡪 Data exploration:

* Want to do this step to look at influence of batch effects, data outliers, etc. on your data before performing differential gene expression analysis
  + Otherwise will not know what is driving differential gene expression results
* tidyverse only works on data frames
* tibble = data frame, but nicer display and subsetting
* Many sources of confounding variables
  + One specific to my data set is all the male mice that are IgAKO come from one cage and the WT come from another = cage effects
* Dimensional reduction analysis: many options, but some are more associated with certain types of data than others
  + microbiome data = PCoA
  + single cell seq = t-SNE
  + two classes of data (WT vs. KO) = linear discriminant analysis or LDA (supervised)
    - Maybe I’ll have to look into for my data?
* PCA:
  + linear combination of variables that captures the greatest variance among data points
  + A diagram of weight and weight

    Description automatically generated
    - Linear = the line
    - combination of variable = height and weight (variables being measured)
    - greatest variance = greatest spread among the data points
* If you can understand the variance of your data, relate the principle components of variance to something real in your data, you can understand what are the primary drivers distinguishing your samples from one another
* Watch stat quest video to better understand some of the terms associated with analysis included in this video (e.g., rotation, center, scale)
* unsupervised = knows nothing about group assignment
* A close-up of a box

  Description automatically generated
* Data wrangling summary:
  + Step 1: make DGE list of count data (DGE = digital gene expression)
  + Step 2: convert DGE count data 🡪 counts per million (cpm)
  + Step 3: log2 transform cpm data
  + Step 4: filter out genes/transcripts with cpm values <=1 in >= 4 samples (4 because IgA-HET group has 4 samples – if all samples in one group have values <=1, probably not expressed)
  + Step 5: trimmed mean of M-values (TMM) normalize your data

Lecture 8 🡪 Accessing public data:

*Will watch after lecture 9 so I have the figures to show Mike*

* *This will be useful though – I can look online to see if anyone has created an RNA-seq data set using the same IgA-KO mouse line that I have 🡪 do they also see expression of Igha?*

Lecture 9 🡪 Differential gene expression:

* Genes can be more accurately measured than transcripts
  + Differential transcript usage (DTU) – from the total output of a gene, what proportion of the gene is derived from isoform A vs. isoform B, etc.
* design matrix =/= study design
  + *can do 2 analyses, one based on genotype and another based on sex*