HIGH PERFORMANCE RESEARCH COMPUTING

ACES: RNA-seq and Differential Expression

on the FASTER cluster

Presented by Wes Brashear

28 March, 2023

High Performance
Research Computing
DIVISION OF RESEARCH

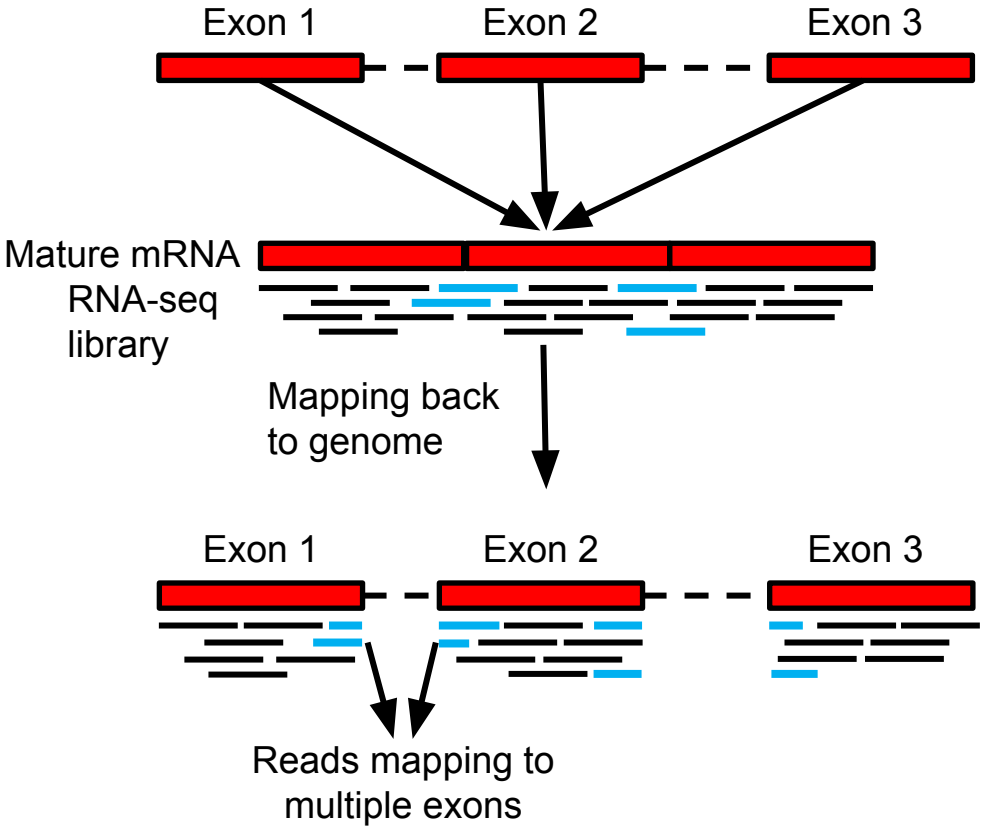
What does RNA-seq data provide?

- Annotate genomes or assemble transcriptomes
- Discover nucleotide variants
- Scaffold genome assemblies
- Measure gene expression and detect differences between groups

RNA-seq Applications

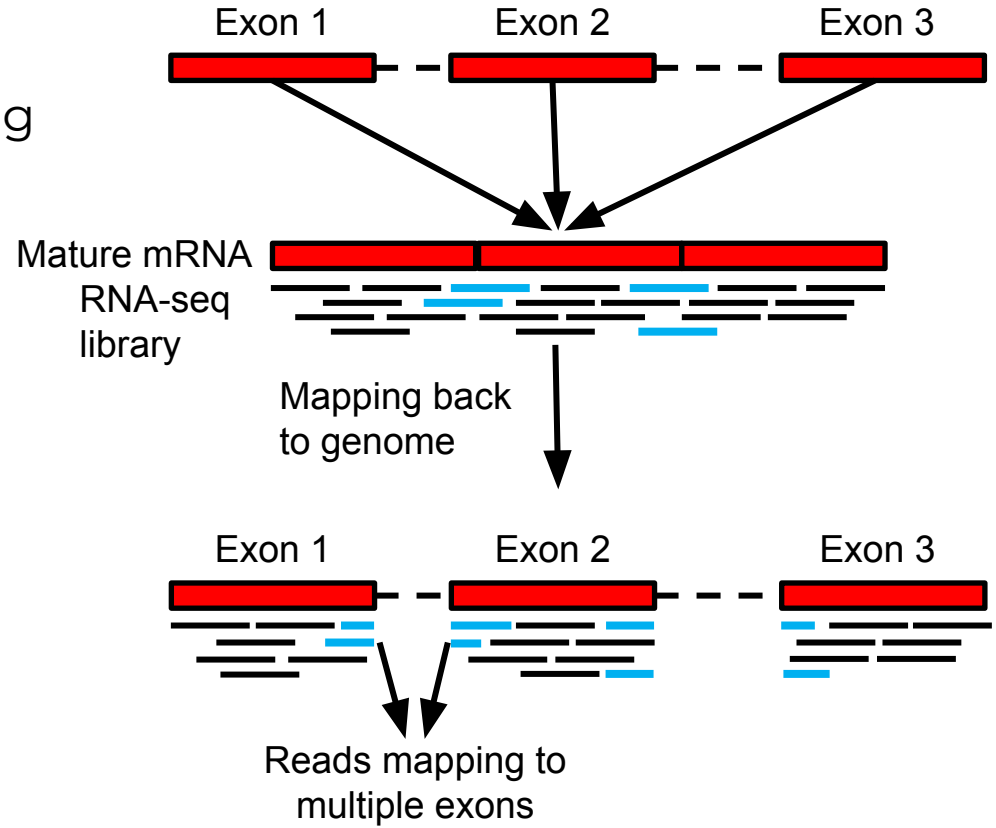
- Transcriptome Assembly
  - ***de novo***: Trinity, Oases, SOAPdenovo-Trans
  - **Reference-based**: Trinity, StringTie, Cufflinks

- Splice-aware alignment
  - HISAT2
  - STAR
  - Clara Parabricks (GPU-accelerated STAR)
  - Tophat



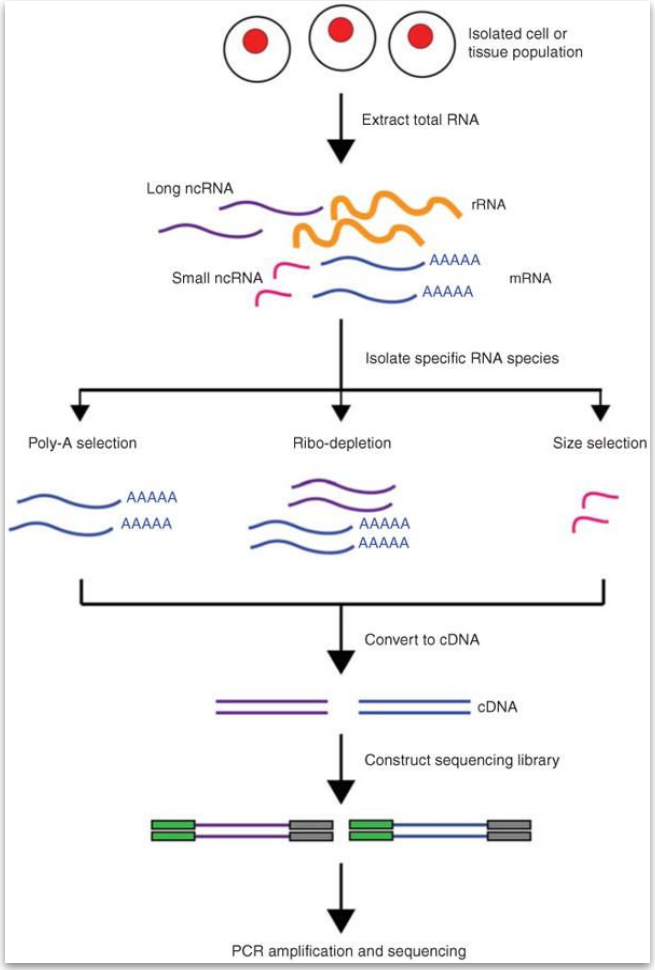
RNA-seq Applications

- File conversion and formatting
  - SAMtools
  - Picard tools
- Variant Calling
  - GATK (Haplotype caller in RNA-seq mode)
- Scaffolding Assemblies
  - L_RNA_scaffolder
  - Rascaf



Sequencing RNA

- Poly-A selection
  - Enriches for mRNA
- Ribosomal depletion
  - Removes rRNA
  - Leaves mRNA, lncRNA, and pre-RNA
- Size selection
  - Used for smRNA (e.g. miRNA)



Kukurba and Montgomery, 2016

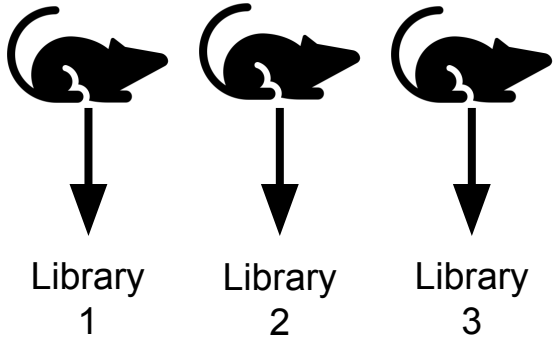
Experimental Design (for Differential Expression)

- Sequencing Depth
  - Minimum 30 million aligned reads per replicate (ENCODE)
  - 30-60 million reads per replicate (Illumina)

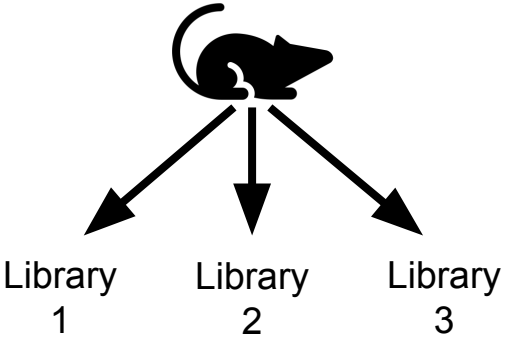
- Replicate Number
  - 3 replicates per condition minimum (will likely recover 20-40% of true DEGs)
  - Schurch et al. (2016) suggest 6 replicates per condition minimum, 12 replicates per condition optimal

Experimental Design (for Differential Expression)

- **Biological Replicates**
  - Independent samples from different populations or individuals



- **Technical Replicates**
  - Multiple libraries from the same individual



Experimental Design (for Differential Expression)

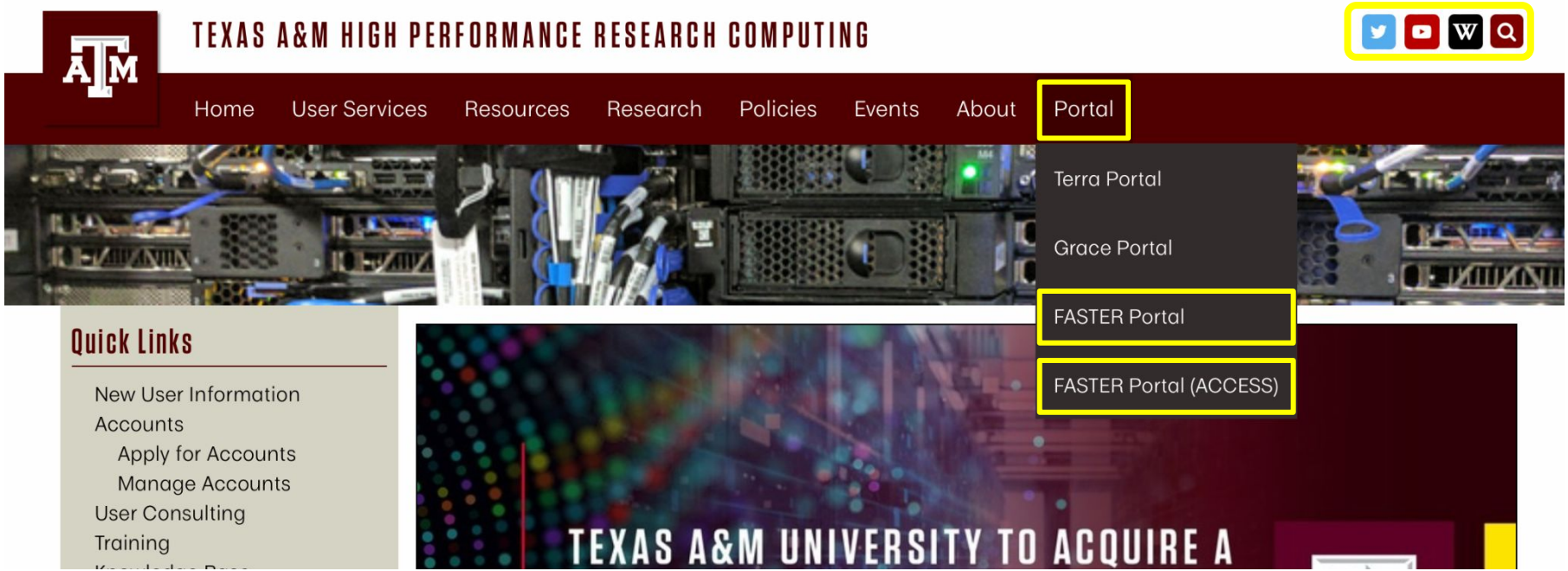
Replicates - Which to use?

- Biological replicates generally increase statistical power more than technical replicates
- Biological variability > Technical Variability
- Biological replicates contain both biological and technical variability

Logging into FASTER via the HPRC Portal

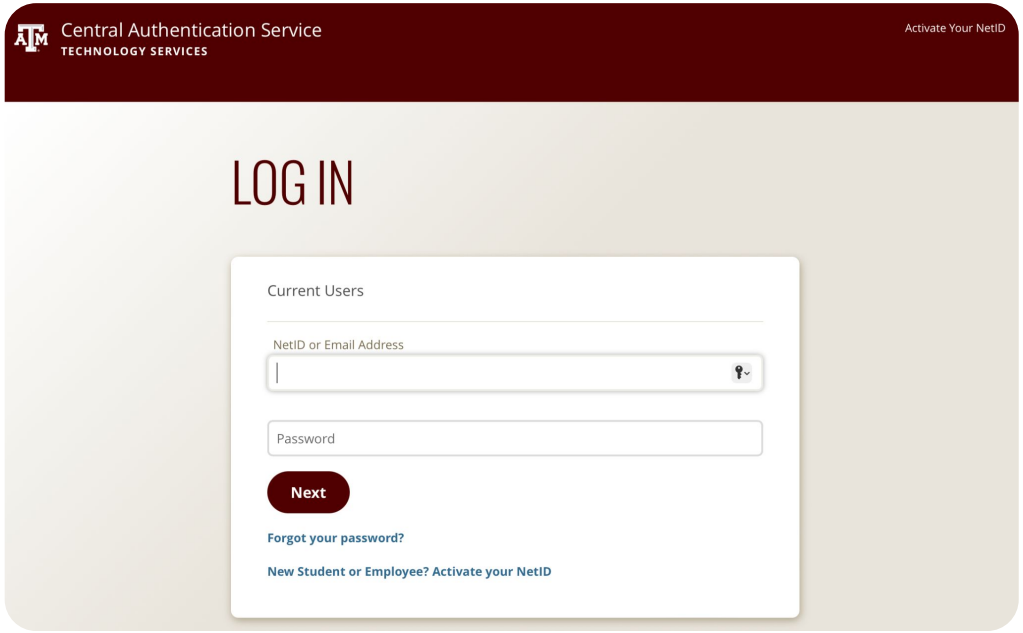
Accessing the HPRC Portal

- HPRC webpage: hprc.tamu.edu Portal dropdown menu



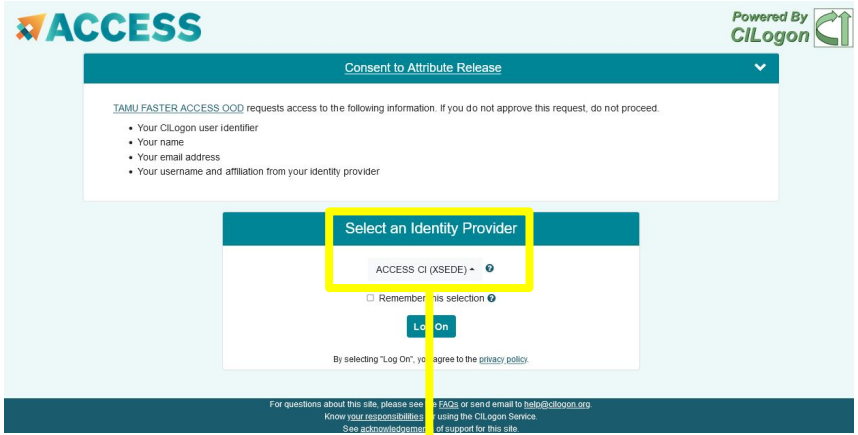
Accessing FASTER via the HPRC Portal (TAMU)

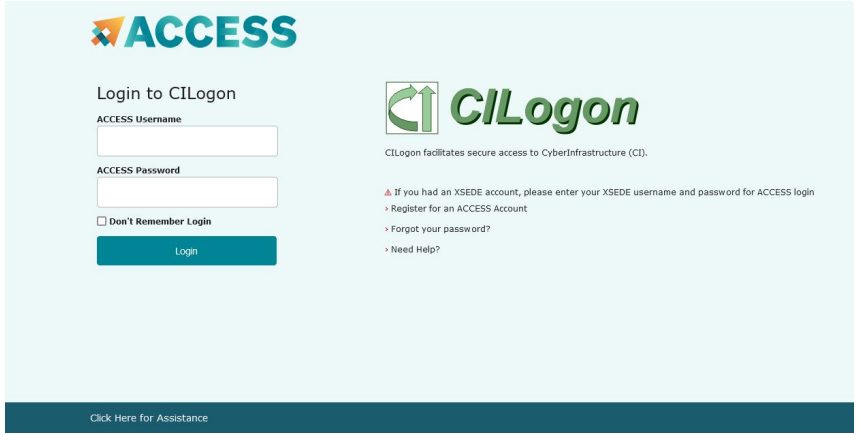
Log-in using your TAMU NetID credentials.

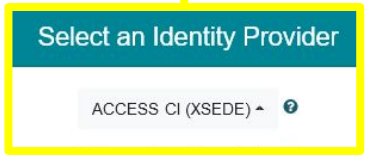


Accessing FASTER via the HPRC Portal (ACCESS)

Log-in using your ACCESS credentials.





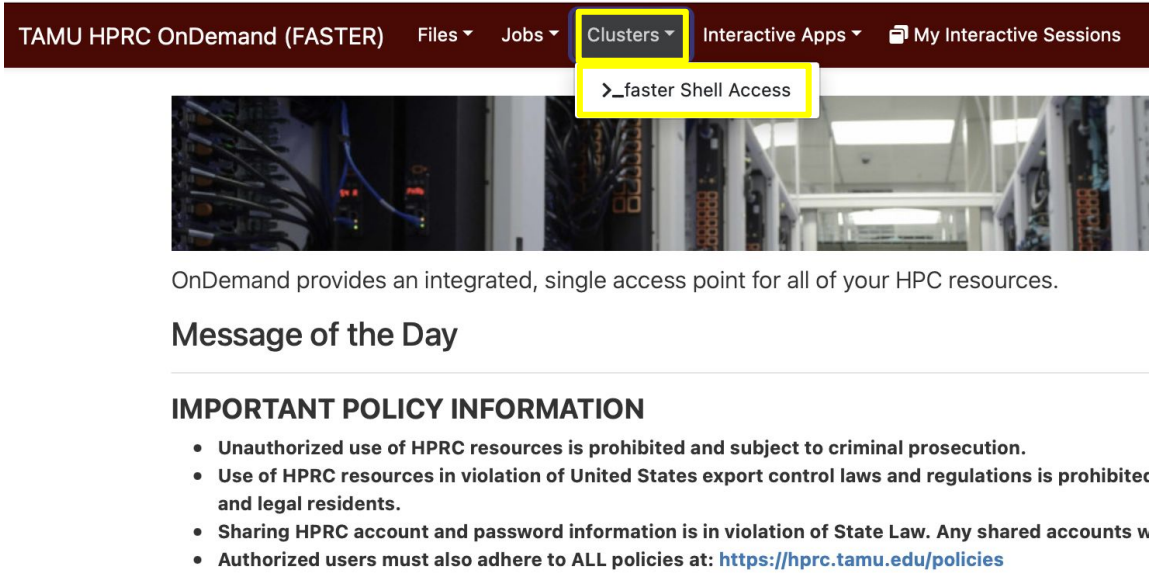


Select the Identity Provider appropriate for your account.

Shell access via the HPRC Portal

Access through (most) web browsers

–Top Banner Menu “Clusters” -> “Shell Access”



Example Data

- Create a new directory in your scratch space

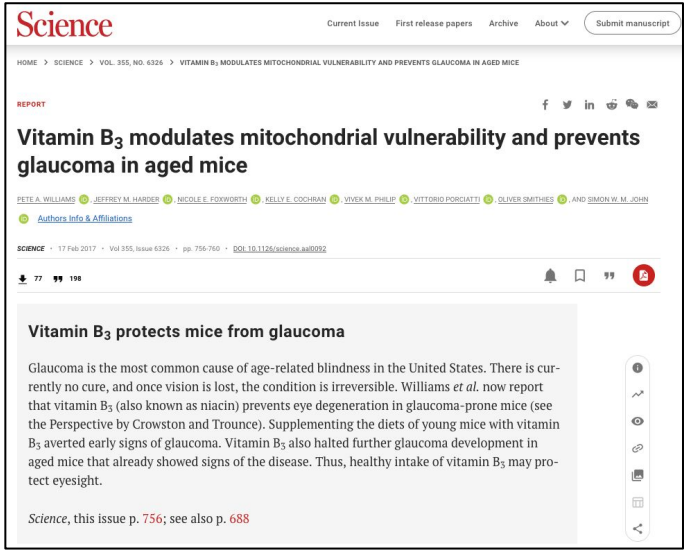
```
$ mkdir $SCRATCH/RNA_class
```

- Change your working directory to the one you just created

```
$ cd $SCRATCH/RNA_class
```

- Copy the example data to your directory

```
$ cp -r /scratch/training/bio/rna-seq/* .
```



Science

Current Issue First release papers Archive About Submit manuscript

HOME > SCIENCE > VOL. 355, NO. 6326 > VITAMIN B_3 MODULATES MITOCHONDRIAL VULNERABILITY AND PREVENTS GLAUCOMA IN AGED MICE

REPORT

Vitamin B_3 modulates mitochondrial vulnerability and prevents glaucoma in aged mice

PETE A. WILLIAMS, JEFFREY M. HARDER, NICOLE E. FOXWORTH, KELLY E. COCHRAN, VIVEK M. PHILIP, VITTORIO PORCIATTI, OLIVER SMITHIES, AND SIMON W. M. JOHN

Authors Info & Affiliations

SCIENCE • 17 Feb 2017 • Vol 355, Issue 6326 • pp. 756-760 • DOI: 10.1126/science.aal0092

77 198

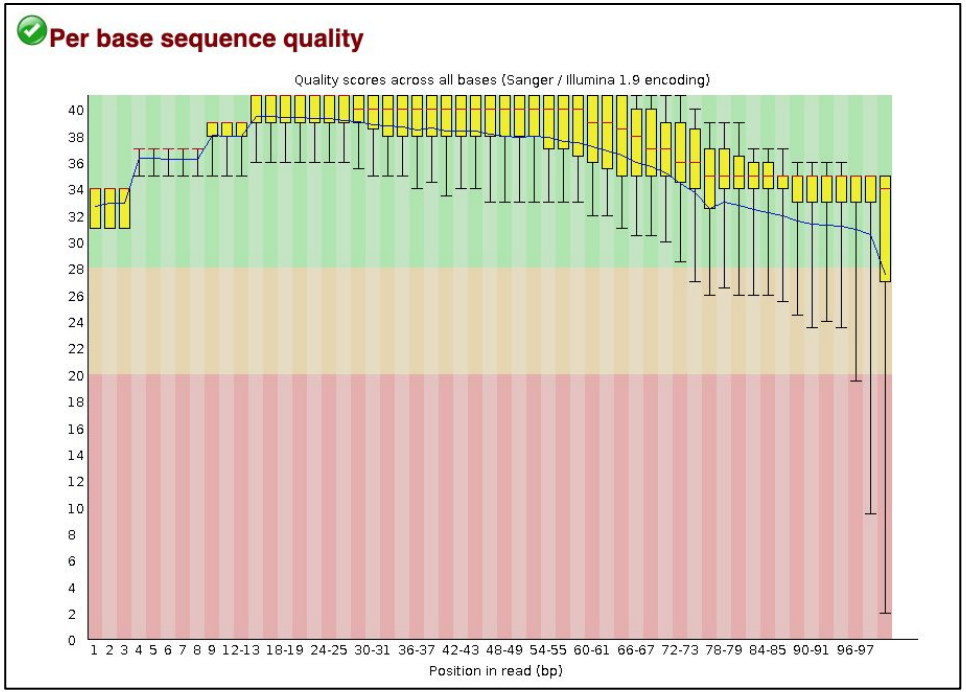
Vitamin B_3 protects mice from glaucoma

Glaucoma is the most common cause of age-related blindness in the United States. There is currently no cure, and once vision is lost, the condition is irreversible. Williams *et al.* now report that vitamin B_3 (also known as niacin) prevents eye degeneration in glaucoma-prone mice (see the Perspective by Crowston and Trounce). Supplementing the diets of young mice with vitamin B_3 averted early signs of glaucoma. Vitamin B_3 also halted further glaucoma development in aged mice that already showed signs of the disease. Thus, healthy intake of vitamin B_3 may protect eyesight.

Science, this issue p. 756; see also p. 688

Quality Control

- NGS libraries should be assessed for adapter content and low-quality reads before downstream analysis
- Low-quality bases and adapters can introduce errors and reduce map rates
- Avoid overly aggressive trimming practices



Quality Control

- Will use FastQC to examine the quality of our example data
- Look for the appropriate module on FASTER:

```
$ module spider fastqc
```

- Clear any previously loaded modules and load FastQC:

```
$ module purge
```

```
$ module load FastQC/0.11.9-Java-11
```

Running jobs on FASTER

- Small jobs can be run on the login nodes (< 60 minutes, up to 8 cores)
- Larger jobs should be submitted to the compute nodes:
  - Slurm job scheduler
  - Can specify computing requirements:
    - Amount of memory required
    - Number of cores
    - Which modules to load
- Template job scripts are available:
  - https://hprc.tamu.edu/wiki/SW:GCATemplates

Quality Control

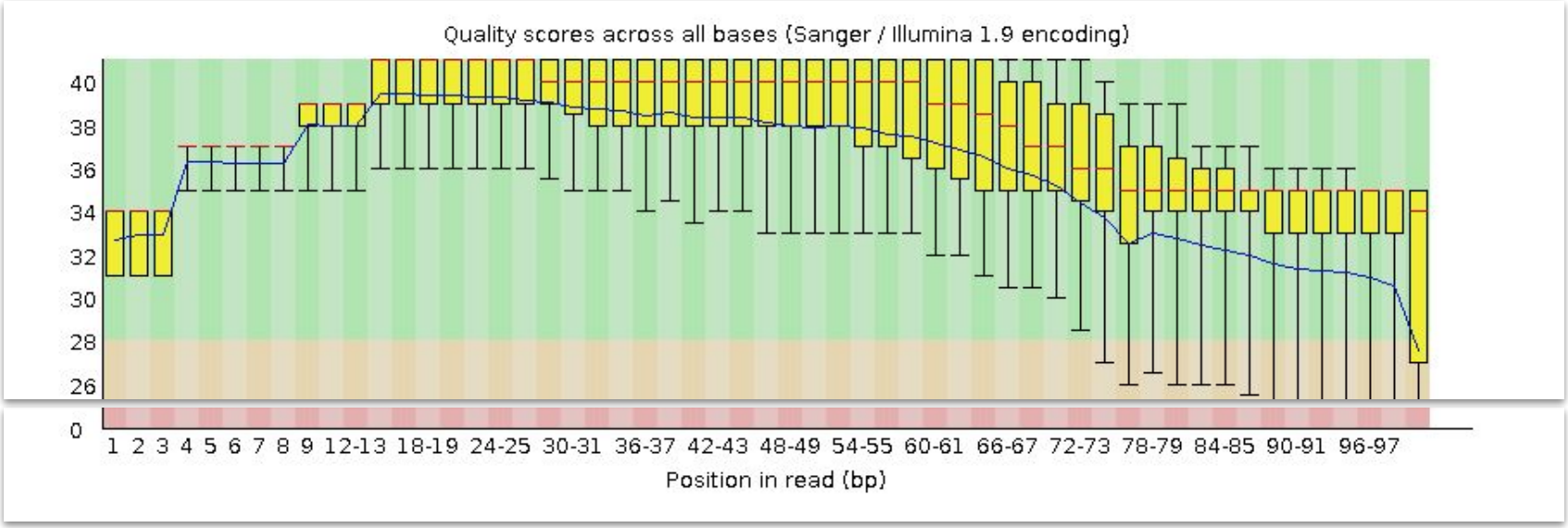
- Run FastQC on our example fastqs:

```
$ fastqc -t 2 -o . Control1_R1.fastq.gz Control1_R2.fastq.gz
```

- Go to “Files” tab in FASTER portal and navigate to the RNA_class directory
- FastQC results saved as html files

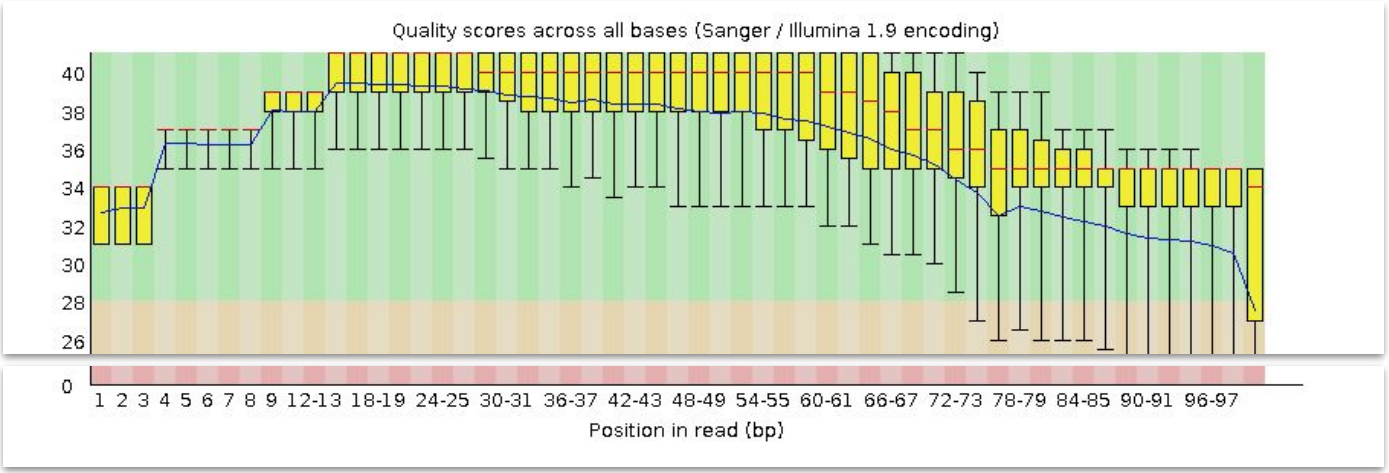
FASTQ Format

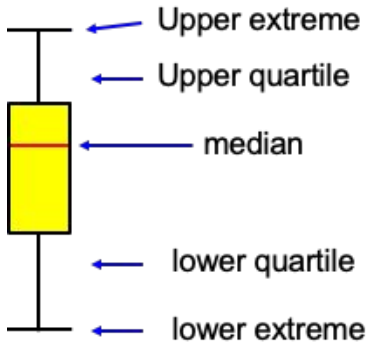
```
@ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
GATCGGAAGAGCACACGTCTGAACTCCAGTCACGATCAGATCTCGTATGCCGTCTTCTGCTTGAAAAAAAAAAAAAAAAACAAAAAACATAATGCCGTAAA
+ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
=:=4AD=B8A:+<A::1<:AE<C3*?F<B???<?:8:6?B*9BD;/638.=-'-.@7=).=A:6?DDDCBBBB9555&&)+((+2&&+(((((((&&&+
@ERR504787.3.1 M00368:15:000000000-A0HKH:1:3:12724:25677-1 length=100
GATGTTTTGTTACTGATTGGAACCATGATTGGTGCTTTACTTGGTTTCTTCCTATTTAACCACAAGCCTGCCAAAGTATTTATGGGAGATGTAGGTAGTT
+ERR504787.3.1 M00368:15:000000000-A0HKH:1:3:12724:25677-1 length=100
BCCFDEFFHHHHHJJJJIJJJJJJJJJIJJJFHIJJJIJJJJJIJJJJJJJIJJJIJIIJJJJJFHJJJJIJJHH=CHHFFFFFFEDDEDEEEDCCDCE
```



FASTQ Format

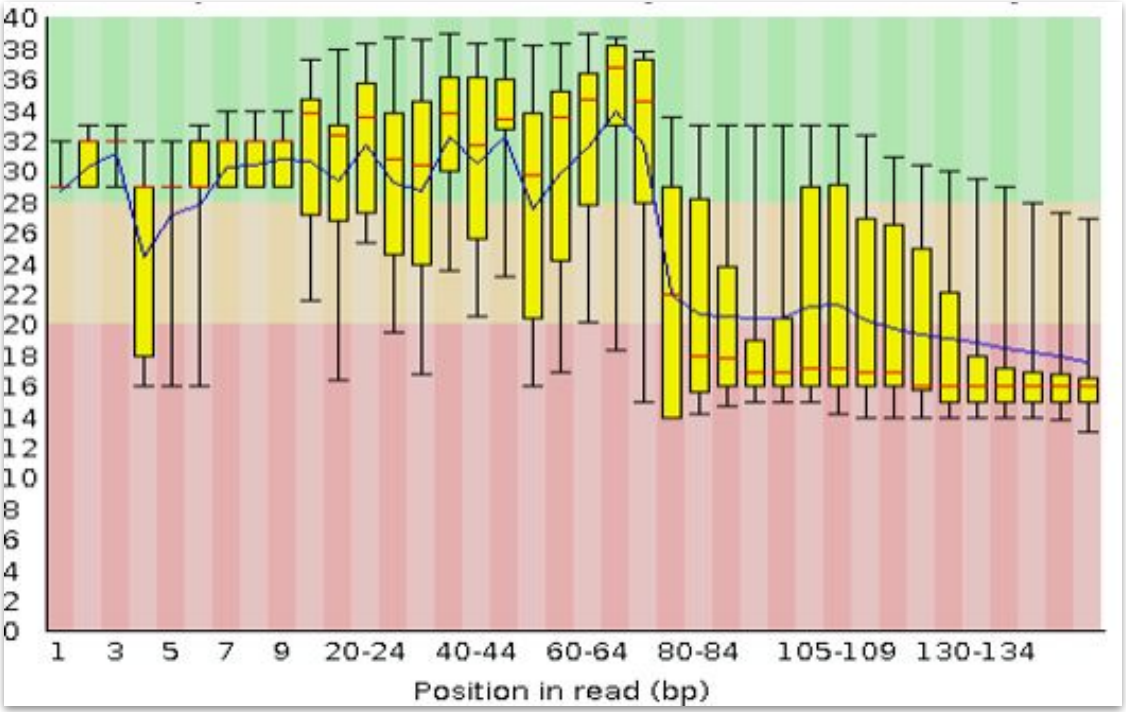
```
@ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
GATCGGAAGAGCACACGTCTGAACTCCAGTCACGATCAGATCTCGTATGCCGTCTTCTGCTTGAAAAAAAAAAAAAAAAACAAAAAACATAATGCCGTAAA
+ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
=:=4AD=B8A:+<A::1<:AE<C3*?F<B???<?:8:6?B*9BD;/638.=-'-.@7=).=A:6?DDDCBBBB9555&&)+((+2&&+((((((()&&&+
@ERR504787.3.1 M00368:15:000000000-A0HKH:1:3:12724:25677-1 length=100
GATGTTTTGTTACTGATTGGAACCATGATTGGTGCTTTACTTGGTTTCTTCCTATTTAACCACAAGCCTGCCAAAGTATTTATGGGAGATGTAGGTAGTT
+ERR504787.3.1 M00368:15:000000000-A0HKH:1:3:12724:25677-1 length=100
BCCFDEFFHHHHHJJJJIJJJJJJJJJIJJJFHIJJJIJJJJJIJJJJJJJIJJJIJIIJJJJJFHJJJJIJJHH=CHHFFFFFFEDDEDEEEDCCDCE
```





Failed QC Examples

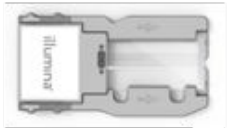
Example 1. Failed per base sequence quality - expired MiSeq kit

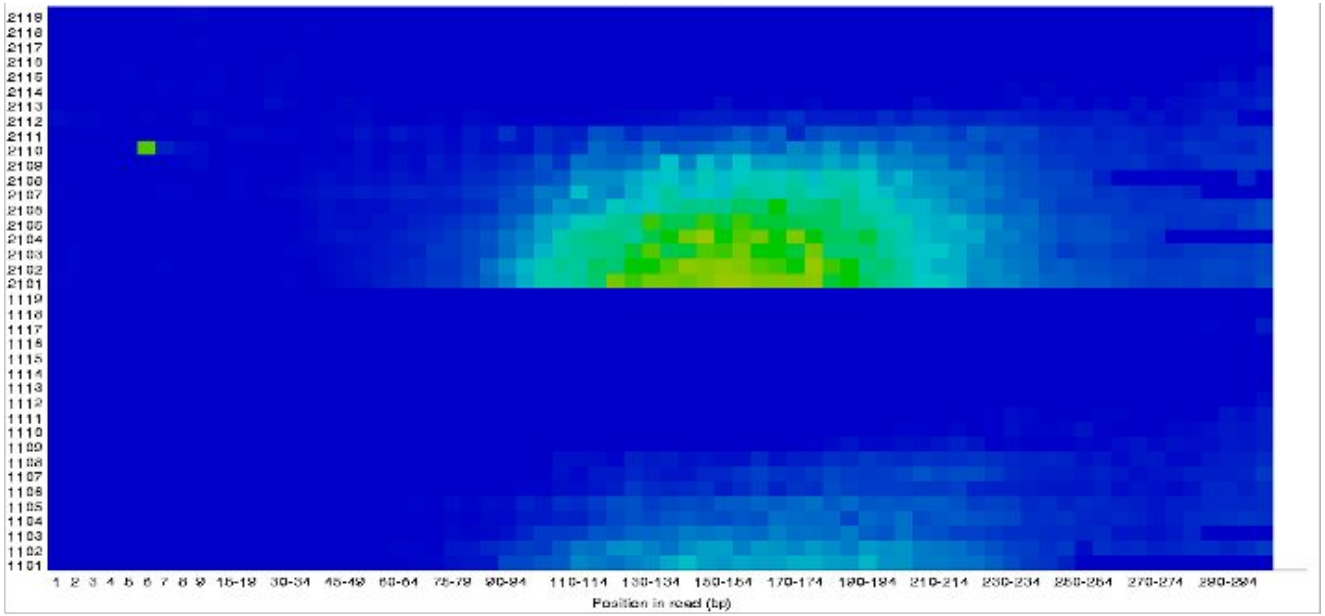


Failed QC Examples

Example 2. Faulty flowcell

MiSeq flowcell

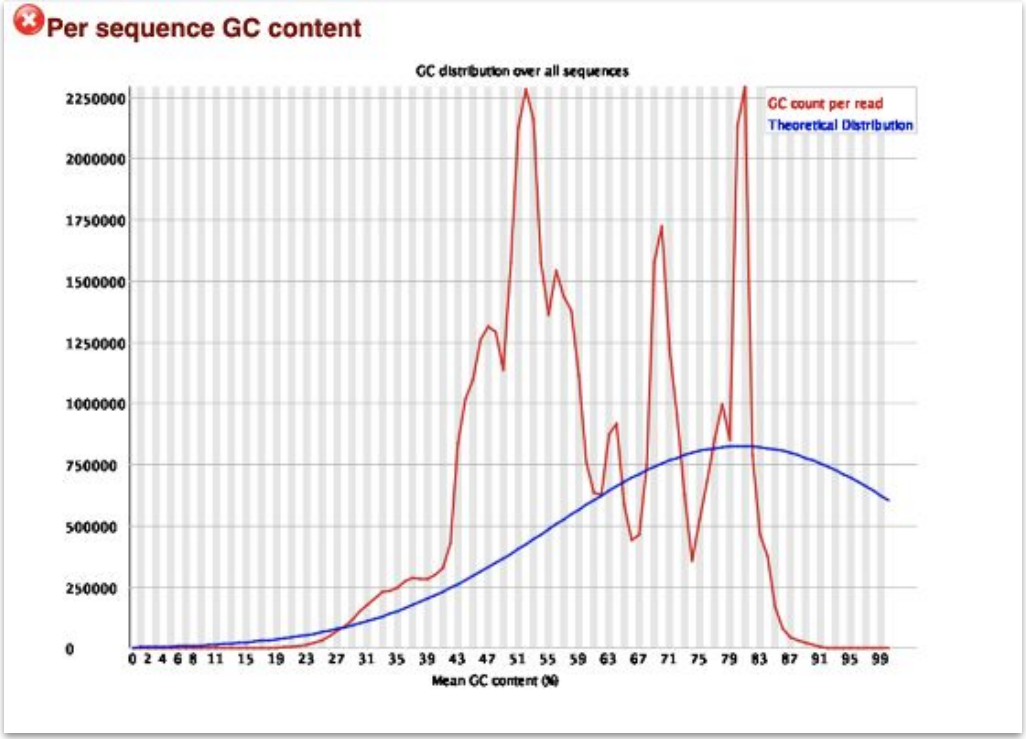




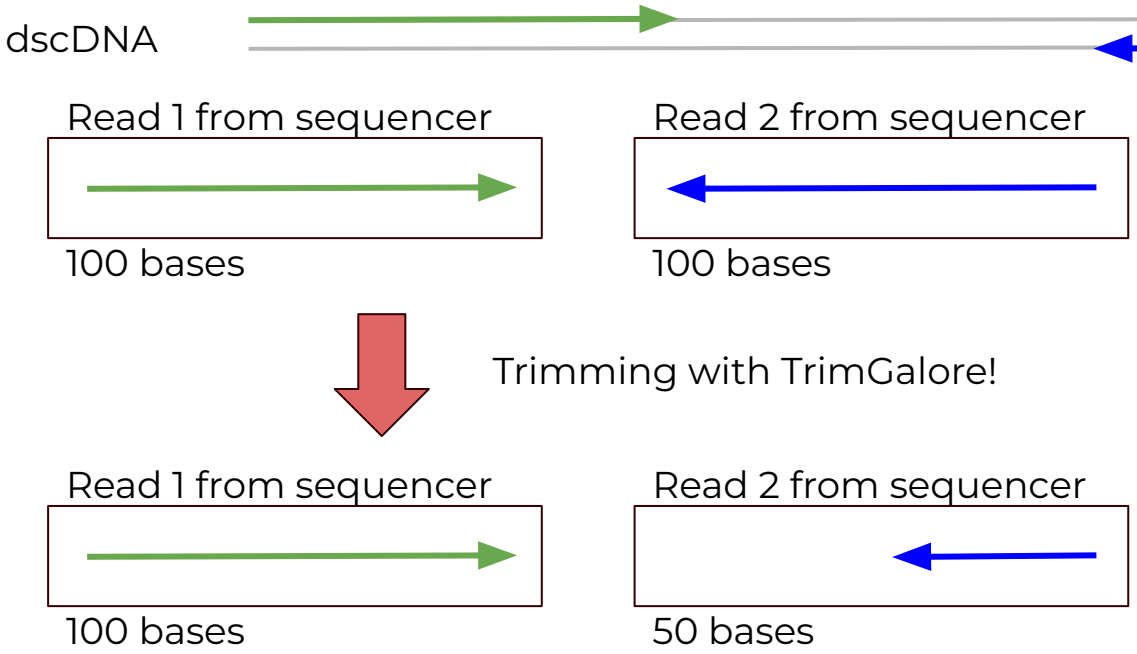
good quality — poor quality

Failed QC Examples

Example 3. Contamination



Library Trimming



- Specify minimum read length (default = 20)
- Return only paired or retain unpaired

Library Trimming

- Remove loaded modules:

```
$ module purge
```

- Find and load the appropriate modules:

```
$ module spider trim_galore
```

```
$ module load GCCcore/11.2.0 Trim_Galore/0.6.10
```

- Run Trim_Galore!

```
$ trim_galore --paired Control1_R1.fastq.gz Control1_R2.fastq.gz
```

Library Trimming

- Let's run FastQC again on our trimmed data to make sure it worked:

```
$ module purge
```

```
$ module load FastQC/0.11.9-Java-11
```

```
$ fastqc Control1_R1_val_1.fq.gz
```

```
$ unzip Control1_R1_val_1_fastqc.zip
```

Aligning Reads to a Reference Genome

- Popular splice-aware aligners
  - STAR (now available for GPUs!)
  - HISAT2
- Alignment software needs to have and indexed genome (software specific)
  - Only needs to be done once
  - Many genomes are already indexed on FASTER:

```
/scratch/data/bio/genome_indexes/
```

  - Email help@hprc.tamu.edu if you would like us to add another indexed genome

Aligning Reads to a Reference Genome

- Clear any previously loaded modules:

```
$ module purge
```

- Search for and load the appropriate modules:

```
$ module spider hisat
```

```
$ module load GCC/9.3.0 OpenMPI/4.0.3 HISAT2/2.2.1
```

- Get information on how to run the program:

```
$ hisat2 -h
```

Aligning Reads to a Reference Genome

- Align our trimmed reads to the mouse genome:
  - Path to previously indexed genome:

```
/scratch/data/bio/genome_indexes/ncbi/mm39/hisat2/GCF_000001635.27_GRCm39_genomic
```

  - Set the path to the indexed genome as a new variable:

```
$ idx_genome=/path/to/genome
```

  - Run the HISAT2 command

```
$ hisat2 -x $idx_genome \
    -1 Control1_R1_val_1.fq.gz \
    -2 Control1_R2_val_2.fq.gz \
    -S Control1.sam
```

Processing Alignment Files

- Alignment files may need to be modified and/or converted before any downstream analyses:
  - Sorting (name or pos/coord)
  - Adding read groups
  - Converting to binary format
- We will use SAMtools to process our alignment file:

```
$ module purge
```

```
$ module spider SAMtools
```

```
$ module spider SAMtools/1.16.1
```

```
$ module load GCC/11.3.0 SAMtools/1.16.1
```

Processing Alignment Files

- Run SAMtools sort to convert and sort the alignment file in one step:

```
$ samtools sort --threads 8 \
    -o Control1_sorted.bam Control1.sam
```

- Index the new bam file:

```
$ samtools index Control1_sorted.bam
```

Generating Count Files

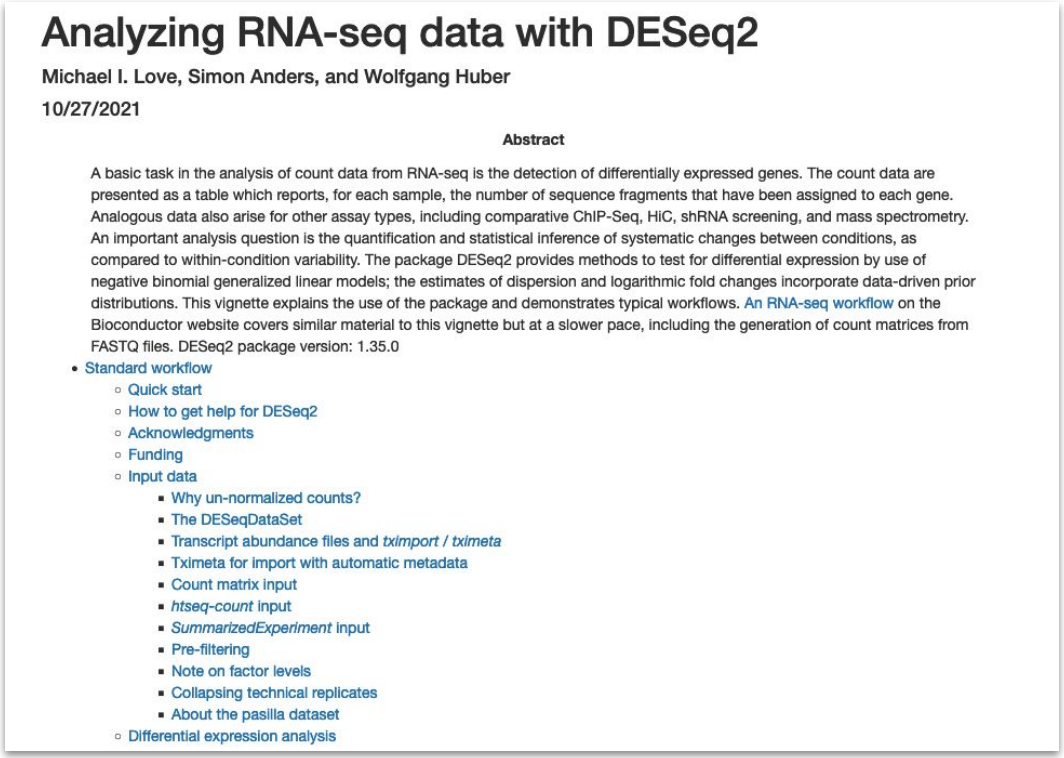
- There are many packages available to generate read counts:
  - featureCounts
  - GenomicRanges (R package)
  - HTSeq
- Load the required modules and produce the count table:

```
$ module purge
```

```
$ module load GCC/11.3.0 OpenMPI/4.1.4 HTSeq/2.0.2
```

```
$ htseq-count -r pos -i gene Control1_sorted.bam \
    GCF_000001635.27_GRCm39_genomic.gff > Control1_counts.txt
```

Differential Expression Analysis with DESeq2



Analyzing RNA-seq data with DESeq2

Michael I. Love, Simon Anders, and Wolfgang Huber

10/27/2021

Abstract

A basic task in the analysis of count data from RNA-seq is the detection of differentially expressed genes. The count data are presented as a table which reports, for each sample, the number of sequence fragments that have been assigned to each gene. Analogous data also arise for other assay types, including comparative ChIP-Seq, HiC, shRNA screening, and mass spectrometry. An important analysis question is the quantification and statistical inference of systematic changes between conditions, as compared to within-condition variability. The package DESeq2 provides methods to test for differential expression by use of negative binomial generalized linear models; the estimates of dispersion and logarithmic fold changes incorporate data-driven prior distributions. This vignette explains the use of the package and demonstrates typical workflows. An RNA-seq workflow on the Bioconductor website covers similar material to this vignette but at a slower pace, including the generation of count matrices from FASTQ files. DESeq2 package version: 1.35.0

http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html

Working with R in FASTER

- Find which versions of R are available and what other modules are needed

```
$ module purge
```

```
$ module spider R_tamu
```

```
$ module spider R_tamu/4.1.2-foss-2021b
```

- Load R and its dependencies:

```
$ module load GCC/11.2.0 OpenMPI/4.1.1 R_tamu/4.1.2-foss-2021b
```

Working with R in FASTER

- Ensure that you are in the correct working directory:

```
$ cd $SCRATCH/RNA_class/counts
```

- Initialize R:

```
$ R
```

```
R version 4.2.1 (2022-06-23) -- "Funny-Looking Kid"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

Working with R in FASTER

- Ensure that you are in the correct working directory:

```
> getwd()
```

```
> list.files()
```

- Let's look at the contents of the sample table:

```
> system("cat sampleTable.csv")
```

Working with R in FASTER

- Load the all of the required packages:

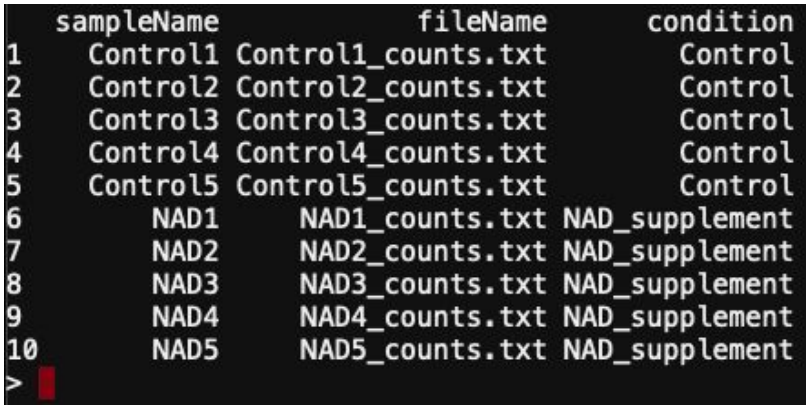
```
> library(ggplot2)
> library(pheatmap)
> library(DESeq2)
> library(EnhancedVolcano)
```

Working with R in FASTER

- Read in the sample table and reformat it:

```
> sampleTable <- read.csv("sampleTable.csv", header=TRUE)
> sampleTable <- as.data.frame(sampleTable)
> sampleTable$condition <- factor(sampleTable$condition)
> sampleTable
```

- Output:



	sampleName	fileName	condition
1	Control1	Control1_counts.txt	Control
2	Control2	Control2_counts.txt	Control
3	Control3	Control3_counts.txt	Control
4	Control4	Control4_counts.txt	Control
5	Control5	Control5_counts.txt	Control
6	NAD1	NAD1_counts.txt	NAD_supplement
7	NAD2	NAD2_counts.txt	NAD_supplement
8	NAD3	NAD3_counts.txt	NAD_supplement
9	NAD4	NAD4_counts.txt	NAD_supplement
10	NAD5	NAD5_counts.txt	NAD_supplement

>

Working with R in FASTER

- Read in the sample table and reformat it:

```
> dds <- DESeqDataSetFromHTSeqCount(sampleTable = sampleTable,
                                    directory = ".",
                                    design = ~ condition)
> dds
```

- Output:

```
> dds
class: DESeqDataSet
dim: 46316 10
metadata(1): version
assays(1): counts
rownames(46316): 0610005C13Rik 0610006L08Rik ... n-TYgta9 n-Tcgca44
rowData names(0):
colnames(10): Control1 Control2 ... NAD4 NAD5
colData names(1): condition
>
```

Working with R in FASTER

- Filter out genes with low read counts:

```
> keep <- rowSums(counts(dds)) >= 10
> dds <- dds[keep,]
```

- Run the differential expression analysis:

```
> dds <- DESeq(dds)
> res <- results(dds)
> res
```

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 23595 rows and 6 columns
                baseMean log2FoldChange     lfcSE       stat    pvalue      padj
               <numeric>      <numeric> <numeric>  <numeric> <numeric> <numeric>
0610005C13Rik   5.99463       0.847191  1.066907   0.794063 0.4271590 0.6270460
0610009B22Rik 229.57285      -0.460666  0.276228  -1.667702 0.0953749 0.2297077
0610009E02Rik  52.71480      -1.185370  0.492924  -2.404771 0.0161826 0.0626895
0610009L18Rik   5.27097       0.500679  1.014609   0.493470 0.6216804 0.7782450
0610010F05Rik 217.32132       0.454101  0.177215   2.562429 0.0103943 0.0444339
...                 ...            ...       ...        ...       ...       ...
n-TRtct2        2.10872      1.6381445  1.355088  1.2088845 0.2267072  0.416331
n-TScga3        5.50077     -0.4540586  0.737908 -0.6153322 0.5383353  0.717329
n-TTagt5        1.22482     -3.3625847  1.792925 -1.8754738 0.0607276        NA
n-TWcca1        3.96509     -0.0540574  1.117400 -0.0483778 0.9614151  0.981356
n-TYgta1        1.05416      1.7174587  1.757956  0.9769635 0.3285872        NA
>
```

Working with R in FASTER

- Filter out genes with low read counts:

```
> keep <- rowSums(counts(dds)) >= 10
> dds <- dds[keep,]
```

- Run the differential expression analysis:

```
> dds <- DESeq(dds)
> res <- results(dds)
> res
```

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526   1.0536757176372  0.803957398183298   0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267   -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158  -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153   1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                  baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                 <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526   1.0536757176372  0.803957398183298   0.4214215791485  0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                 NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267   -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158   -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153   1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

Mean of normalized counts for all samples

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik 5.99463012842517 0.847110388480526 1.0536757176372 0.803957398183298  0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421 -1.33338402962542 2.80181545117752 -0.475900020133387 0.634145607845708                NA
0610009B22Rik  229.572854136365 -0.46059738889209 0.272726760296267  -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124 -1.18516447577791 0.483501805720158 -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362 0.500548878654153 1.0060551707554 0.497536211933899 0.618810973397835 0.779869206330055
```

Log2 fold change: NAD supplement vs Control

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526  1.0536757176372  0.803957398183298   0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267   -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158  -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153  1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

Log fold change standard error

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526   1.0536757176372  0.803957398183298   0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267   -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158   -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153   1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

Wald statistic: NAD supplement vs Control

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue              padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>         <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526   1.0536757176372  0.803957398183298   0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267   -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158  -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153   1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

Wald test p value
(unadjusted)

Differential Expression Analysis with DESeq2

DESeq Results Explained:

```
> res
log2 fold change (MLE): condition NAD supplement vs Control
Wald test p-value: condition NAD supplement vs Control
DataFrame with 46316 rows and 6 columns
                     baseMean     log2FoldChange             lfcSE                stat              pvalue               padj
                    <numeric>          <numeric>         <numeric>           <numeric>           <numeric>          <numeric>
0610005C13Rik  5.99463012842517  0.847110388480526   1.0536757176372  0.803957398183298   0.4214215791485 0.626767086797856
0610006L08Rik 0.595406936513421  -1.33338402962542  2.80181545117752 -0.475900020133387  0.634145607845708                NA
0610009B22Rik  229.572854136365  -0.46059738889209 0.272726760296267    -1.688860265827 0.0912462113650002 0.227131423458176
0610009E02Rik  52.7148015454124  -1.18516447577791 0.483501805720158   -2.45121003015211 0.0142376849790884 0.058533268492583
0610009L18Rik  5.27096640148362  0.500548878654153   1.0060551707554  0.497536211933899  0.618810973397835 0.779869206330055
```

BH corrected p values (corrected for multiple testing)

Working with R in FASTER

- How many genes are differentially expressed?

```
> sum(res$padj <= 0.05, na.rm = TRUE)
```

- Collect all the DEGs and write them to file:

```
> sigGenes <- res[ which(res$padj < 0.05), ]
> sigGenes
> write.csv(sigGenes,
            "Differentrially_Expressed.csv",
            row.names = TRUE)
```

PCA plot

- Log transform the results and calculate the row variance

```
> logTran <- rlog(dds)
> rv <- rowVars(assay(logTran))
```

- Create a list of genes with the greatest variance:

```
> select <- order(rv, decreasing = TRUE)[seq_len(min(100, length(rv)))]
```

PCA plot

- Run the principal component analysis (PCA)

```
> PCA <- prcomp(t(assay(logTran)[select, ]), scale = FALSE)
> summary(PCA)
```

- Output:

```
> summary(PCA)
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7
Standard deviation     13.1129 2.50384 1.94479 1.45805 1.42247 1.24092 1.08253
Proportion of Variance  0.9084 0.03312 0.01998 0.01123 0.01069 0.00814 0.00619
Cumulative Proportion   0.9084 0.94156 0.96154 0.97278 0.98347 0.99160 0.99779
                          PC8     PC9      PC10
Standard deviation     0.52065 0.38289 3.066e-15
Proportion of Variance 0.00143 0.00077 0.000e+00
Cumulative Proportion  0.99923 1.00000 1.000e+00
>
```

PCA plot

- Set up the PCA for ggplot2

```
> percentVar <- round(100*PCA$sdev^2/sum(PCA$sdev^2),1)
> ggPCA_out <- as.data.frame(PCA$x)
> ggPCA_out <- cbind(ggPCA_out, sampleTable)
> head(ggPCA_out)
```

- Output:

```
> head(ggPCA_out)
                PC1         PC2        PC3        PC4        PC5         PC6
Control1 -12.576882  0.679757091  1.4677571  1.4408177 -0.9772907 -2.58170153
Control2 -11.362119 -0.789437801 -4.1149258  0.5846590 -1.6247299  0.15350772
Control3 -12.038043  0.002152241  0.5811305 -3.0992919  1.4657618 -0.97863917
Control4 -13.139919  0.487982477  0.6723550  2.2531953  2.6066210  1.29314839
Control5 -12.530993  0.265744874  1.7077315 -1.1646465 -1.8470308  2.10751112
NAD1       8.795471  0.517771986 -2.7475597 -0.6142266  1.1887028 -0.04330035
```

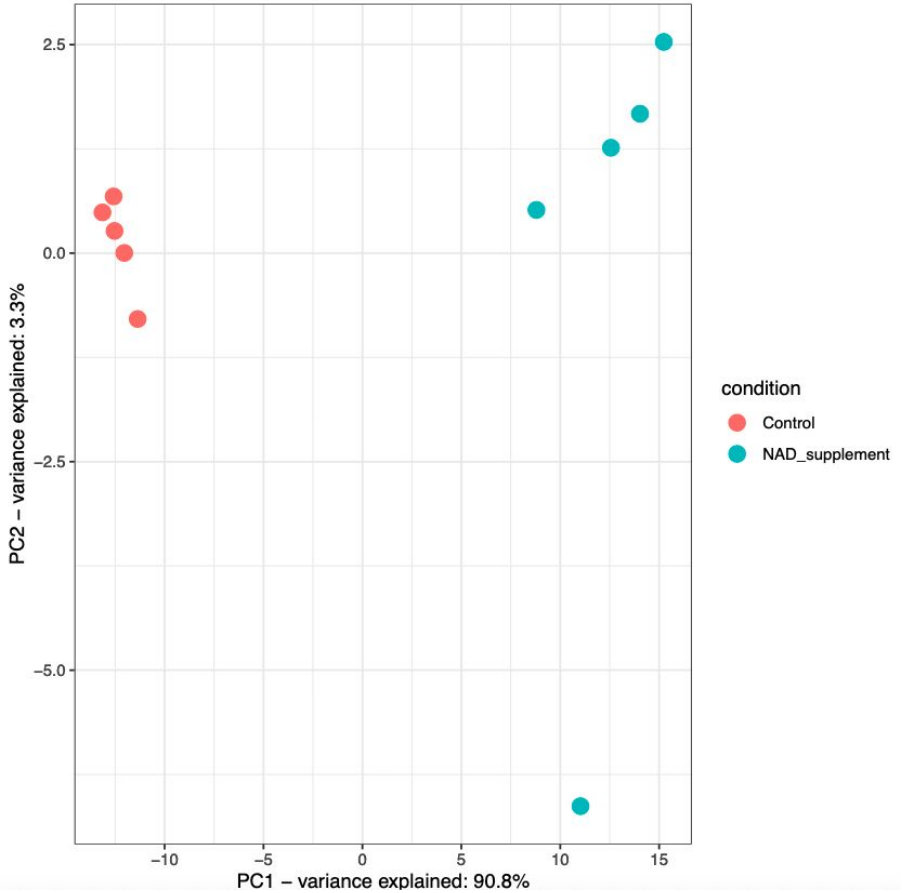
PCA plot

- Plot the PCA

```
> pdf("PCAplot.pdf")
> ggplot(ggPCA_out, aes(x=PC1,y=PC2,color=condition)) +
  geom_point(size=4) +
  labs(x = paste0("PC1 - variance explained: ", percentVar[1], "%"),
    y = paste0("PC2 - variance explained: ", percentVar[2], "%")) +
  theme_bw()
> dev.off()
```

- Use the ‘Files’ app in the portal to view the results

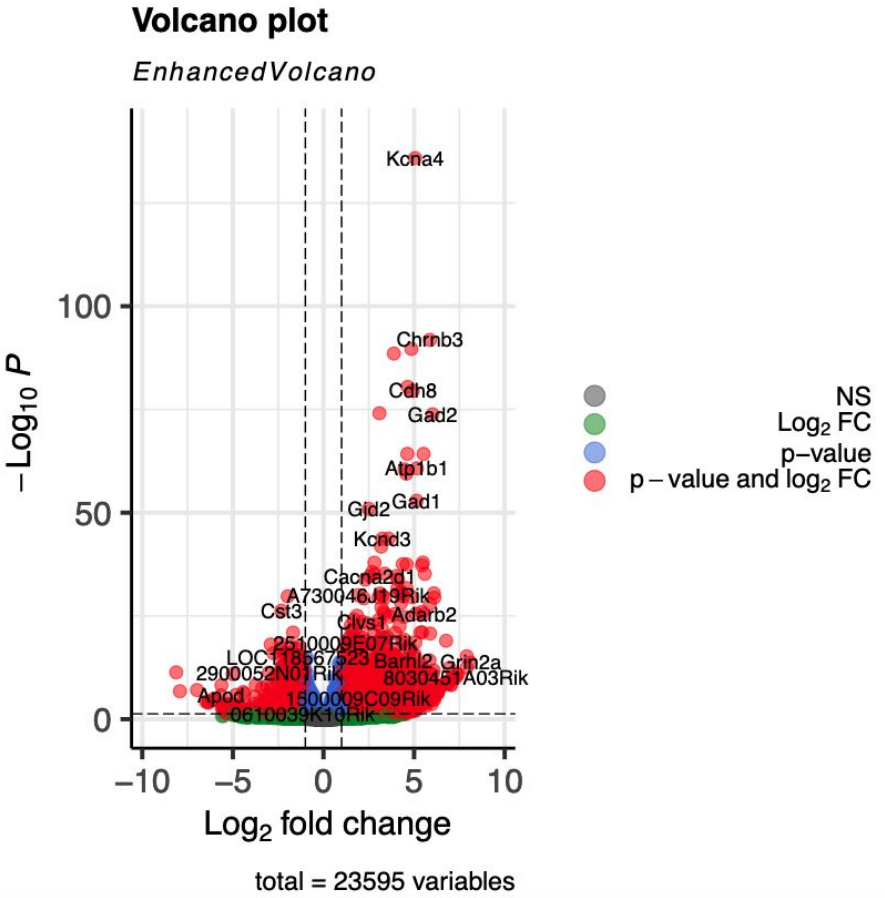
PCA plot


2.5
0.0
−2.5
−5.0
PC2 – variance explained: 3.3%
−10
−5
0
5
10
15
PC1 – variance explained: 90.8%
condition
Control
NAD_supplement

Volcano Plot

```
> pdf("Volcano_plot.pdf")
> EnhancedVolcano(res,
                  lab = rownames(res),
                  x = 'log2FoldChange',
                  y = 'padj',
                  pCutoff = 0.05,
                  FCcutoff = 1.0,
                  pointSize = 3.0,
                  labSize = 4.0,
                  colAlpha = 1/2,
                  drawConnectors = FALSE,
                  legendPosition = "right")
> dev.off()
```

Volcano Plot



Heatmap

- Reorder the results based on adjusted p-values
- Assign genes with adjusted p-values below 0.05 and absolute log2 fold changes >= 6.5 to the variable ‘sig’

```
> resorted_deresults <- res[order(res$padj),]
> sig <- resorted_deresults[!is.na(resorted_deresults$padj) &
                            resorted_deresults$padj < 0.05 &
                            abs(resorted_deresults$log2FoldChange) >=
6.5,]
```

Heatmap

- Assign the gene names from ‘sig’ to a new variable named ‘selected’
- We will use the list of gene names for the heatmap

```
> selected <- rownames(sig)
> selected
```

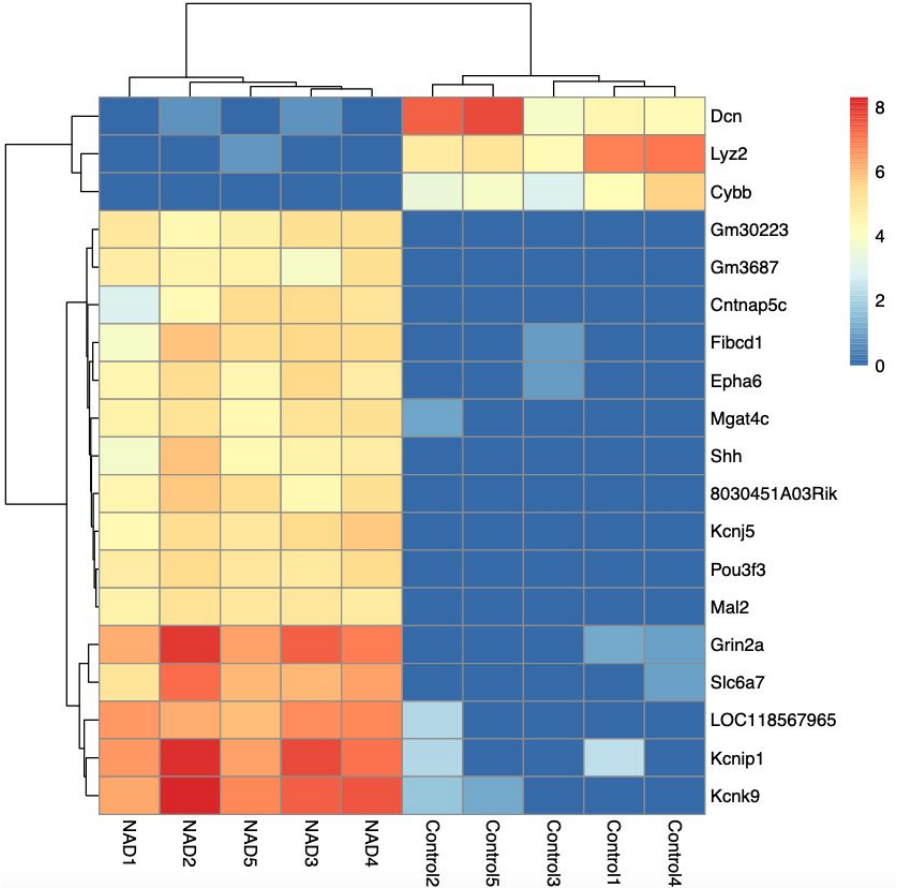
```
> selected
 [1] "Kcnip1"         "Kcnk9"          "Grin2a"         "Slc6a7"
 [5] "LOC118567965"   "Lyz2"           "Pou3f3"         "Kcnj5"
 [9] "Mal2"           "8030451A03Rik"  "Gm30223"        "Fibcd1"
[13] "Gm3687"         "Shh"            "Mgat4c"         "Cntnap5c"
[17] "Epha6"          "Cybb"           "Dcn"
>
```

Heatmap

- We need to normalize the data
- Then we can create the heatmap using the pheatmap package

```
> transformed_readcounts <- normTransform(dds)
> pdf("Heatmap.pdf")
> pheatmap(assay(transformed_readcounts)[selected,],
             cluster_rows = TRUE, show_rownames = TRUE,
             cluster_cols = TRUE,
             labels_col = colData(dds)$sampleName)
> dev.off()
```

Heatmap


Dcn
Lyz2
Cybb
Gm30223
Gm3687
Cntnap5c
Fibcd1
Epha6
Mgat4c
Shh
8030451A03Rik
Kcnj5
Pou3f3
Mal2
Grin2a
Slc6a7
LOC118567965
Kcnip1
Kcnk9
NAD1
NAD2
NAD5
NAD3
NAD4
Control2
Control5
Control3
Control1
Control4
8
6
4
2
0

Save your commands to file

```
> savehistory(file = "RNA_class_Rcommands.txt")
```