RNA-seq and Differential Expression

Texas A&M HPRC

Wes Brashear

October 10, 2025

High Performance
Research Computing
DIVISION OF RESEARCH

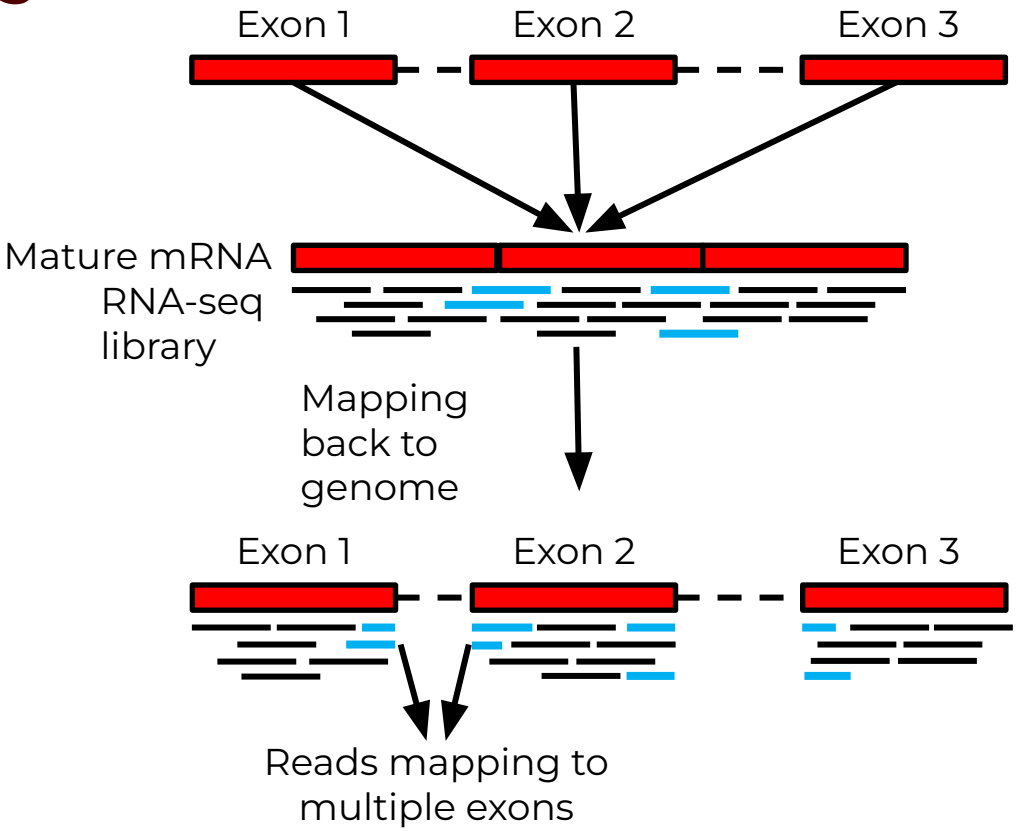
What does RNA-seq data provide?

- Measure gene expression
- Detect differences in expression between groups
- Annotate genomes
- Transcriptome assembly
- Nucleotide variant discovery
- Genome assembly scaffolding

RNA-seq Applications

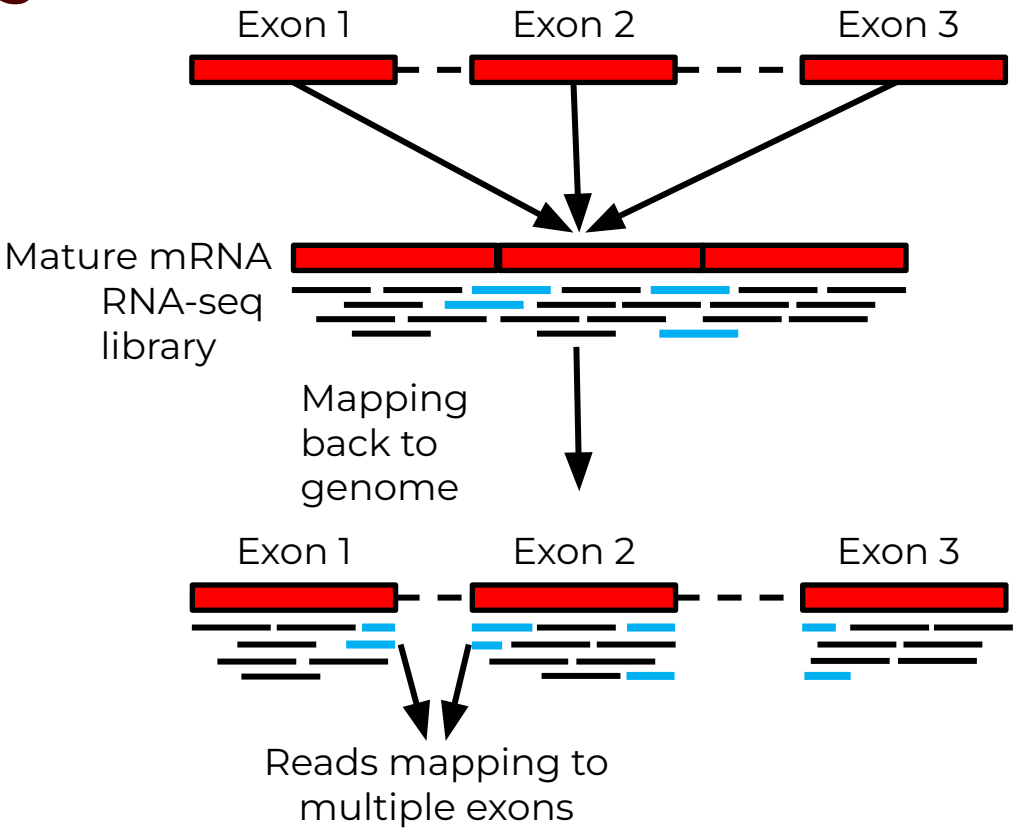
- Transcriptome Assembly
  - ***de novo***: Trinity, Oases, SOAPdenovo-Trans
  - **Reference-based**: Trinity, StringTie, Cufflinks

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



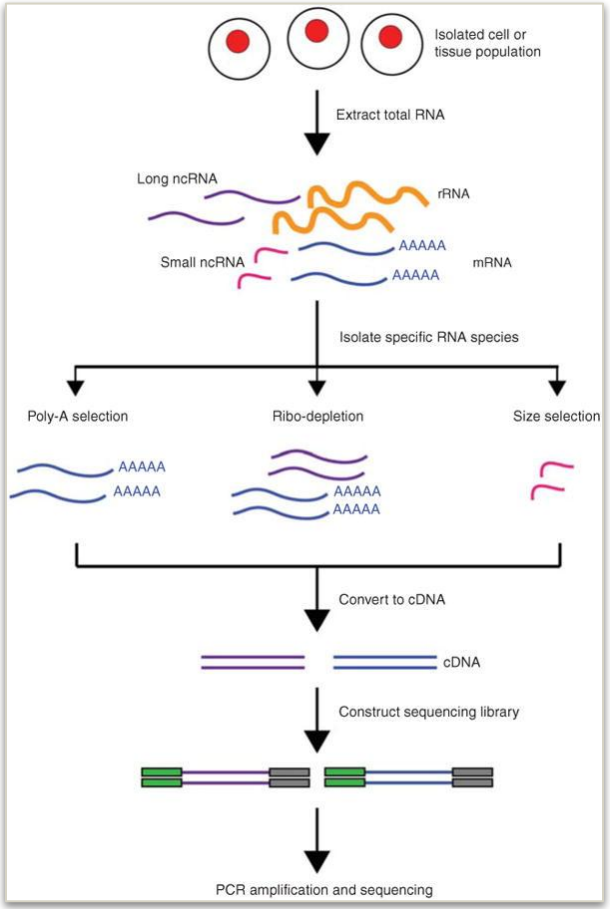
RNA-seq Applications

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



Types of RNA-seq Libraries

- Poly-A selection - enriches for mRNA
- Ribosomal depletion - removes rRNA, leaving mRNA, lncRNA, and pre-mRNA
- Size selection (smRNA)



Kukurba and Montgomery, 2016

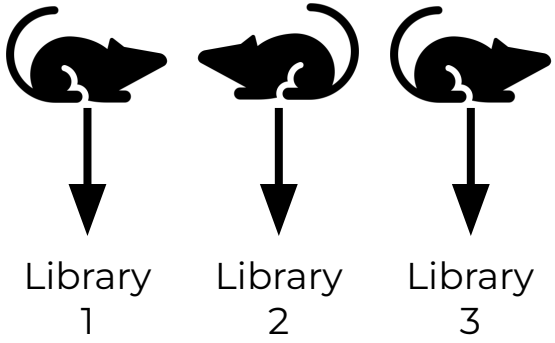
Experimental Design (for Differential Expression)

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

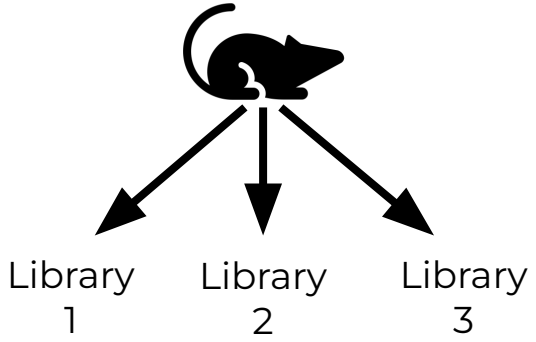
- Replicate Number
  - 3 replicates per condition minimum (will likely recover 20-40% of true DEGs)
  - Schurch et al. (2016) suggest 6 replicates per condition minimum, 12 replicates per condition optimal

Experimental Design (for Differential Expression)

- Biological Replicates
  - Independent samples from different populations or individuals



- Technical Replicates
  - Multiple libraries from the same individual

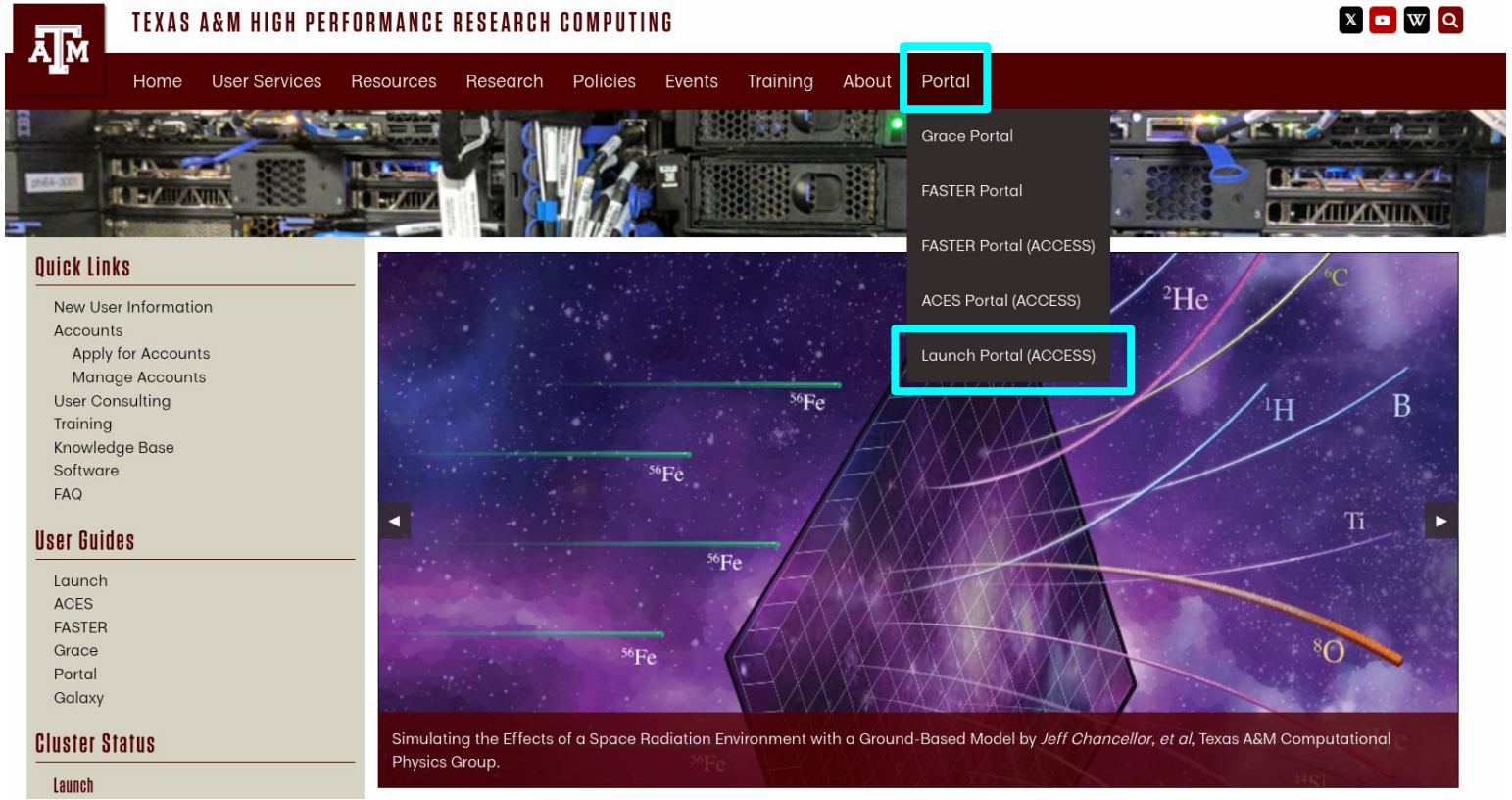


Experimental Design (for Differential Expression)

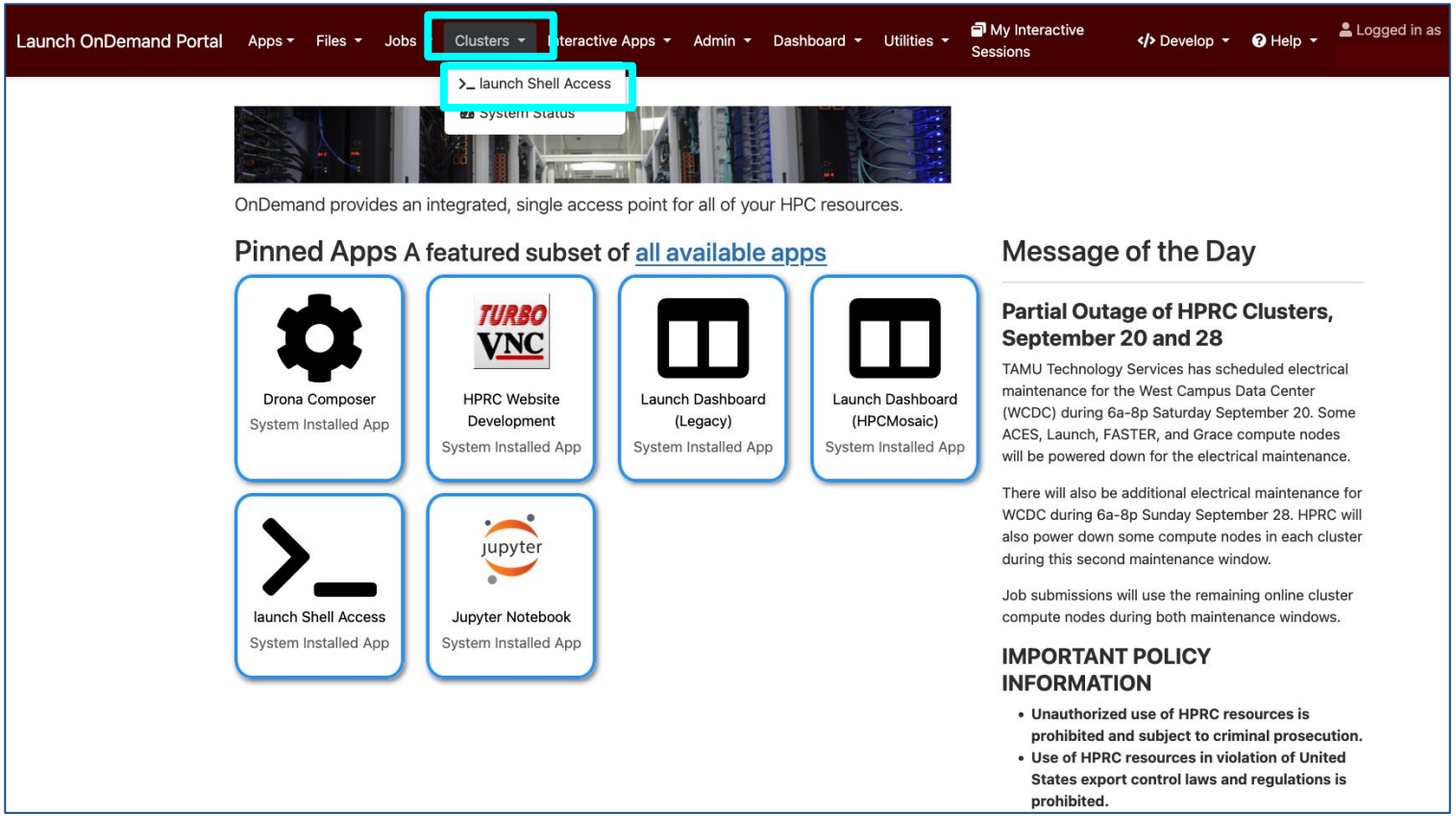
Replicates - Which to use?

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

Accessing the HPRC Launch Portal



Accessing Launch Shell in OOD Portal



Accessing Launch Shell in OOD Portal

```
********************************************************************************
This computer system and the data herein are available only for authorized
purposes by authorized users. Use for any other purpose is prohibited and may
result in disciplinary actions or criminal prosecution against the user. Usage
may be subject to security testing and monitoring. There is no expectation of
privacy on this system except as otherwise provided by applicable privacy laws.
Refer to University SAP 29.01.03.M0.02 Acceptable Use for more information.
********************************************************************************

Last login: Thu Sep 25 23:58:18 2025 from 10.74.0.6
Your current disk quotas are:
Disk                              Disk Usage      Limit    File Usage      Limit
/scratch/user/u.wb109972               1.8T       3.0T         85132     250000
  * Quota increase for /scratch/user/u.wb109972 will expire on Apr 1, 2026
/scratch/group/p.cis250369.000           4K       5.0T             1     500000
/scratch/group/p.agr250003.000           4K       5.0T             1     500000
/scratch/group/p.tra240020.000       691.5G       5.0T          2230     500000
/scratch/group/p.agr250005.000        87.8G       5.0T          5824     500000
/scratch/group/hprc                  900.9G       5.0T        133427     500000
  * Quota increase for /scratch/group/hprc will expire on Apr 1, 2026
Type 'showquota' to view these quotas again.
[u.wb109972@launch-login1 ~]$
```

Example Data

- Create a new directory in your scratch space

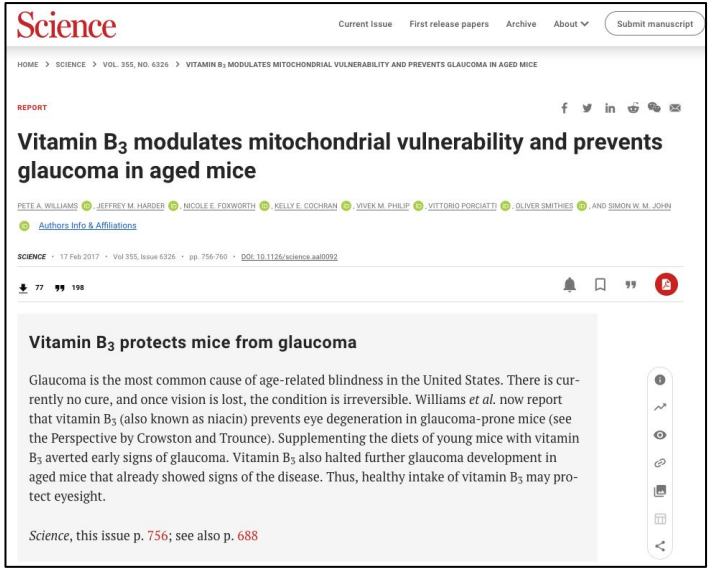
```
$ mkdir $SCRATCH/RNA_class
```

- 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/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

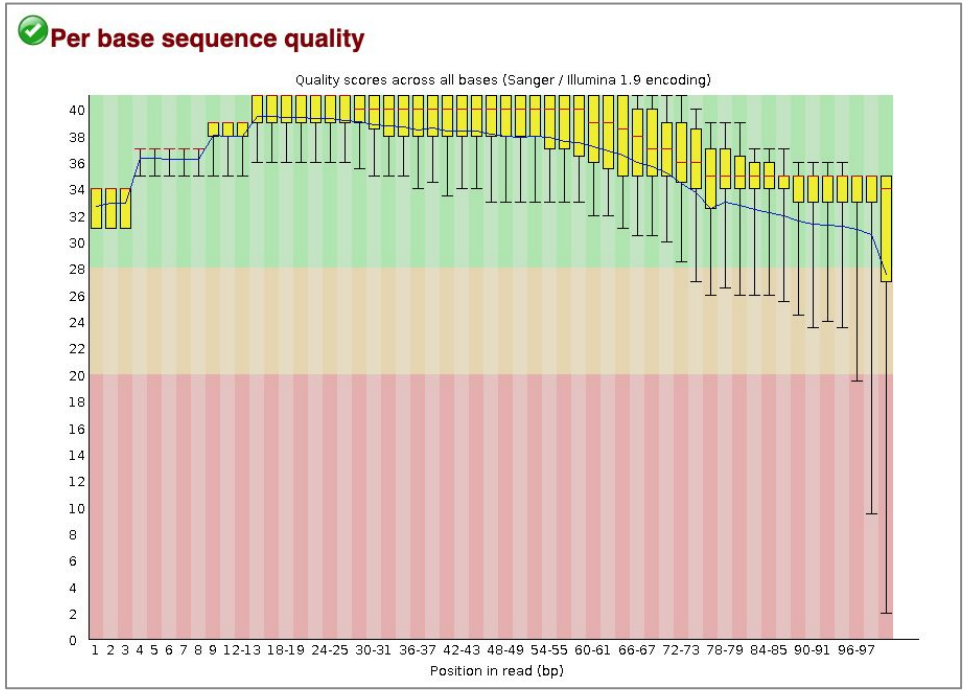
Vitamin B_3 protects mice from glaucoma

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 Launch:

```
$ module spider fastqc
```

```
$ module spider FastQC/0.12.1-Java-11
```

- Clear any previously loaded modules and load FastQC:

```
$ module purge
```

```
$ module load FastQC/0.12.1-Java-11
```

Running jobs on Launch

- 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/kb/Software/useful-tools/GCATemplates/

Quality Control

- Run FastQC on our example fastqs:

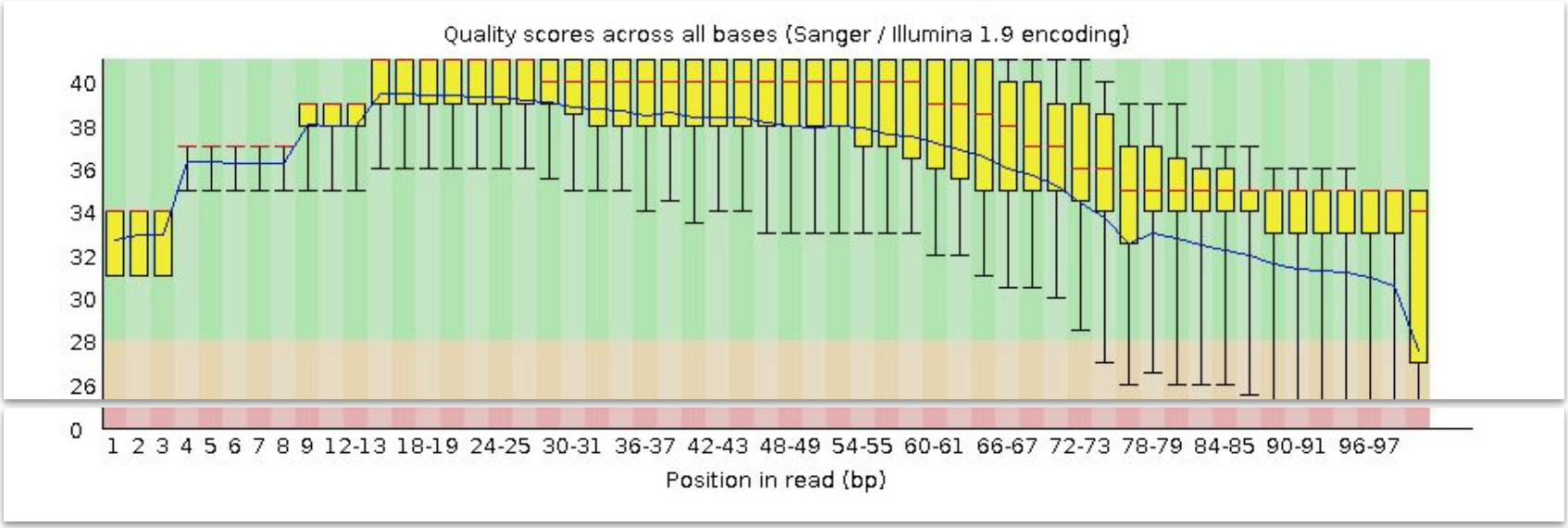
```
$ fastqc -t 2 -o . Control1_R1.fastq.gz Control1_R2.fastq.gz
```

- Go to “Files” tab in Launch portal and navigate to the RNA_class directory

- FastQC results saved as html files

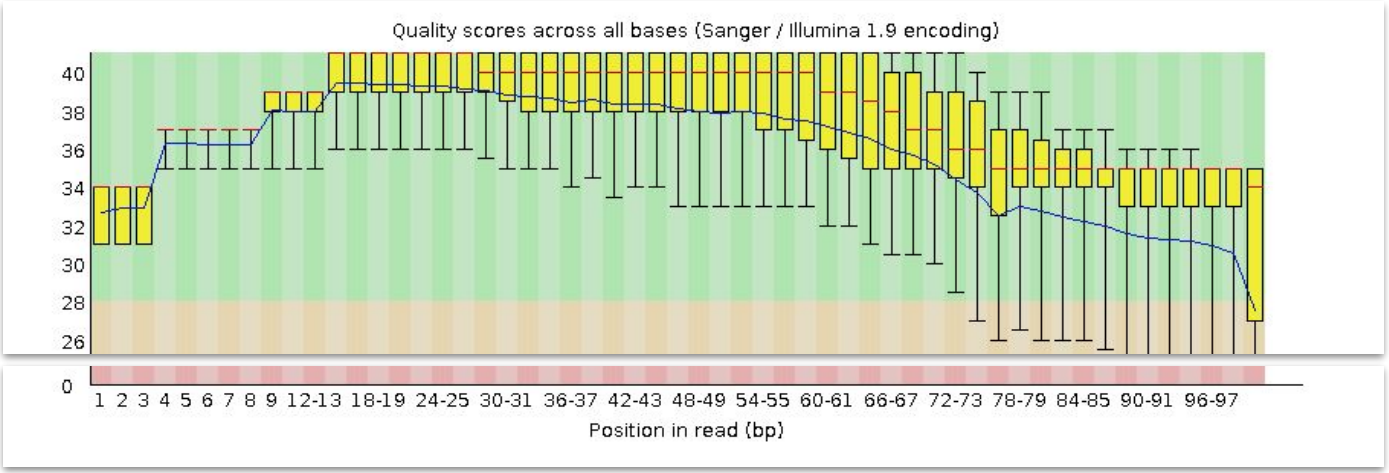
FASTQ Format

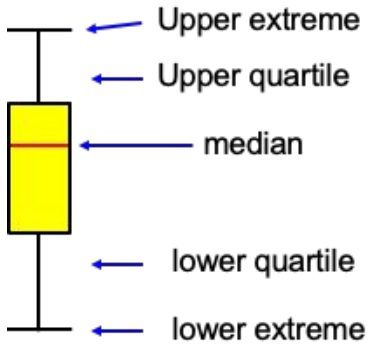
```
@ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
GATCGGAAGAGCACACGTCTGAACTCCAGTCACGATCAGATCTCGTATGCCGTCTTCTGCTTGAAAAAAAAAAAAAAAAACAAAAAACATAATGCCGTAAA
+ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
=:=4AD=B8A:+<A::1<:AE<C3*?F<B???<?:8:6?B*9BD;/638.=-'-.@7=).=A:6?DDDCBBBB9555&&)+((+2&&+(((((((&&&+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

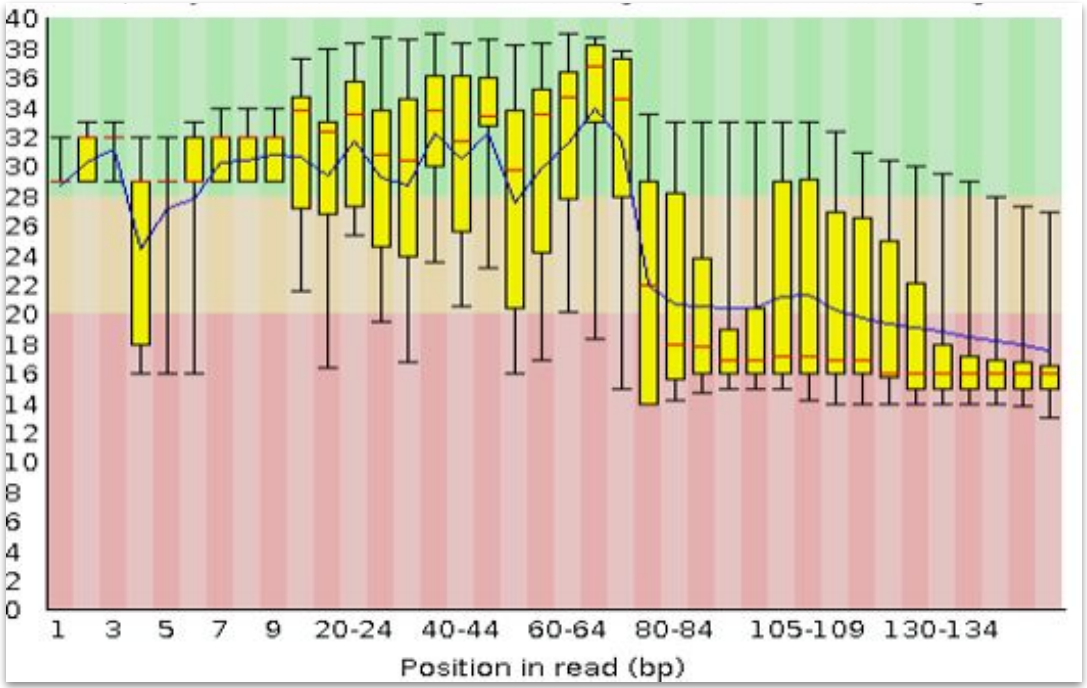
```
@ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
GATCGGAAGAGCACACGTCTGAACTCCAGTCACGATCAGATCTCGTATGCCGTCTTCTGCTTGAAAAAAAAAAAAAAAAACAAAAAACATAATGCCGTAAA
+ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
=:=4AD=B8A:+<A::1<:AE<C3*?F<B???<?:8:6?B*9BD;/638.=-'-.@7=).=A:6?DDDCBBBB9555&&)+((+2&&+((((((()&&&+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

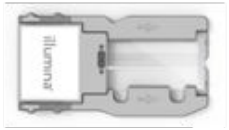
Example 1. Failed per base sequence quality - expired MiSeq kit

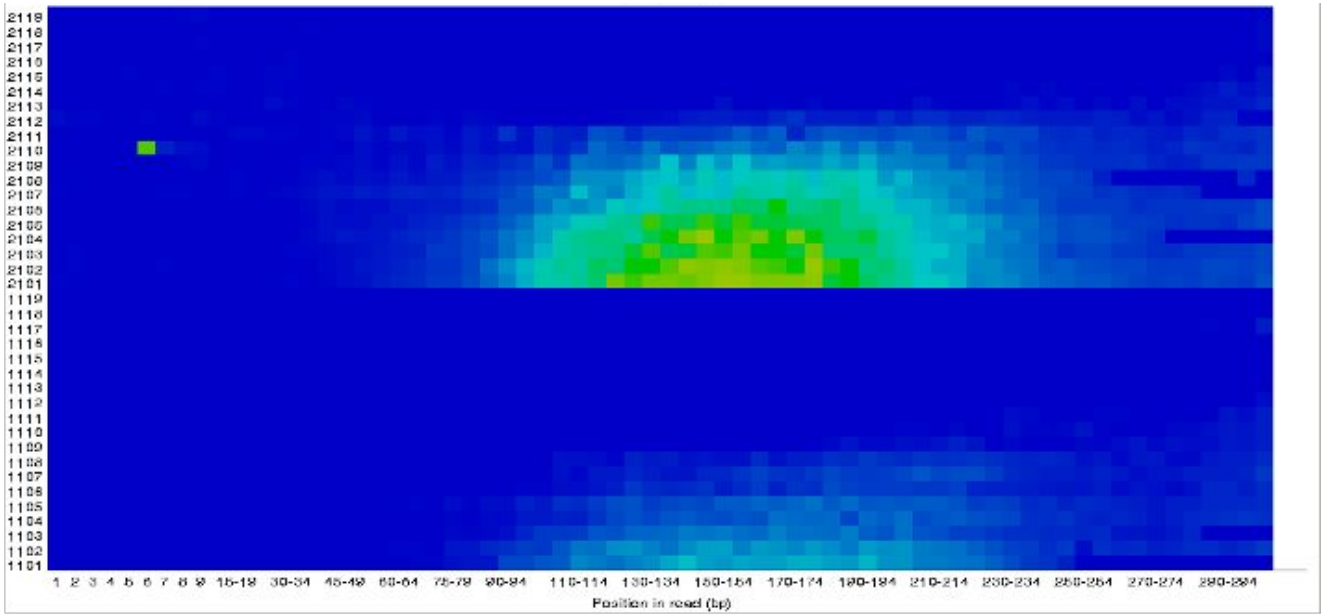


Failed QC Examples

Example 2. Faulty flowcell

MiSeq flowcell

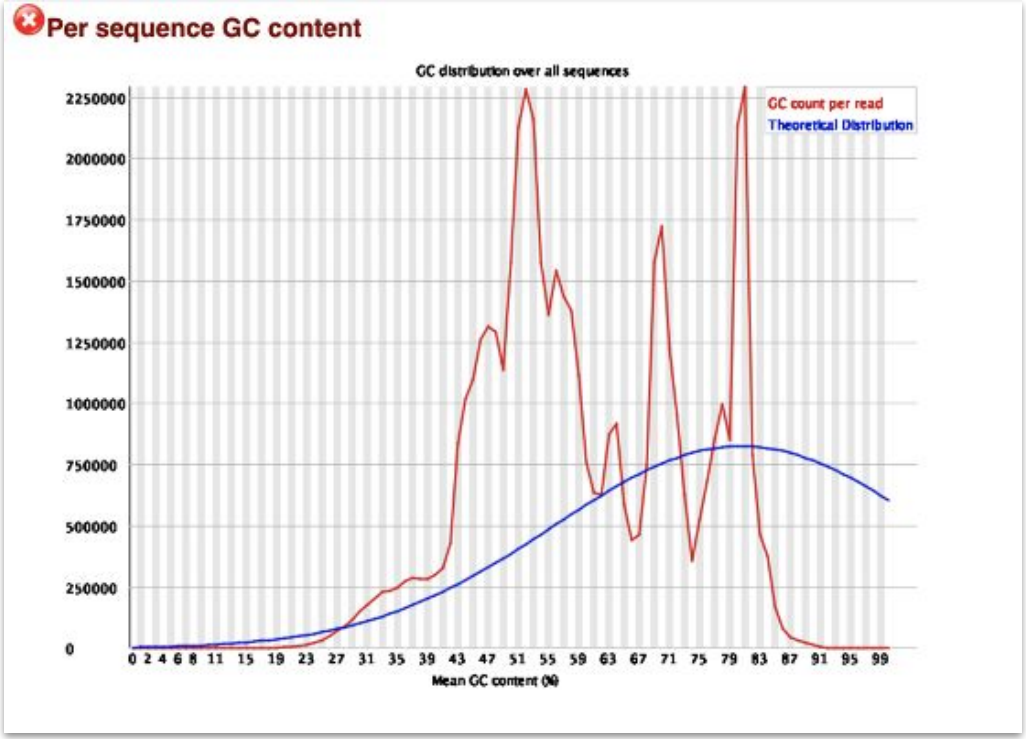




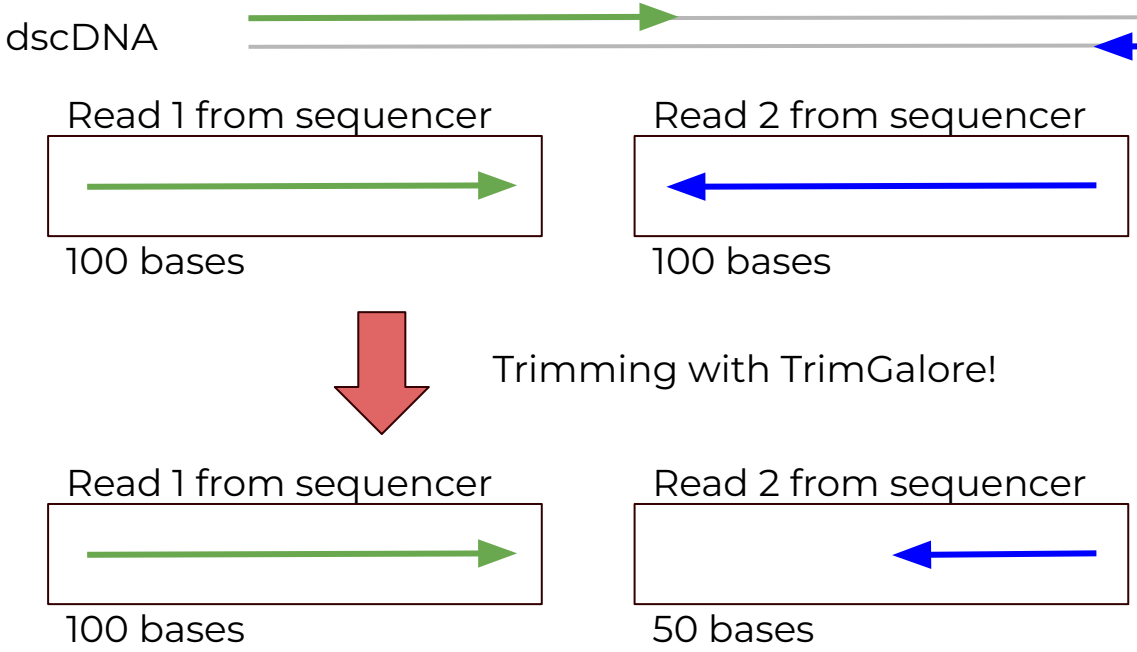
good quality — poor quality

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/12.3.0 Trim_Galore/0.6.10
```

- Run Trim_Galore!

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

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
  - HPRC maintains indexed genomes for popular aligners
  - 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/12.2.0 OpenMPI/4.1.4 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/training/mm39/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 -p 2 \
    -1 Control1_R1_val_1.fq.gz \
    -2 Control1_R2_val_2.fq.gz \
    -S Control1.sam
```

Aligning Reads to a Reference Genome

```
236499 reads; of these:
  236499 (100.00%) were paired; of these:
    30736 (13.00%) aligned concordantly 0 times
    197200 (83.38%) aligned concordantly exactly 1 time
    8563 (3.62%) aligned concordantly >1 times
    ----
    30736 pairs aligned concordantly 0 times; of these:
    3583 (11.66%) aligned discordantly 1 time
    ----
    27153 pairs aligned 0 times concordantly or discordantly; of these:
    54306 mates make up the pairs; of these:
    30660 (56.46%) aligned 0 times
    21188 (39.02%) aligned exactly 1 time
    2458 (4.53%) aligned >1 times
93.52% overall alignment rate
```

Processing Alignment Files

- Alignment files may need to be modified and/or converted before any downstream analyses:
  - Sorting (name or position/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/1.20
```

```
$ module load GCC/13.3.0 SAMtools/1.20
```

Processing Alignment Files

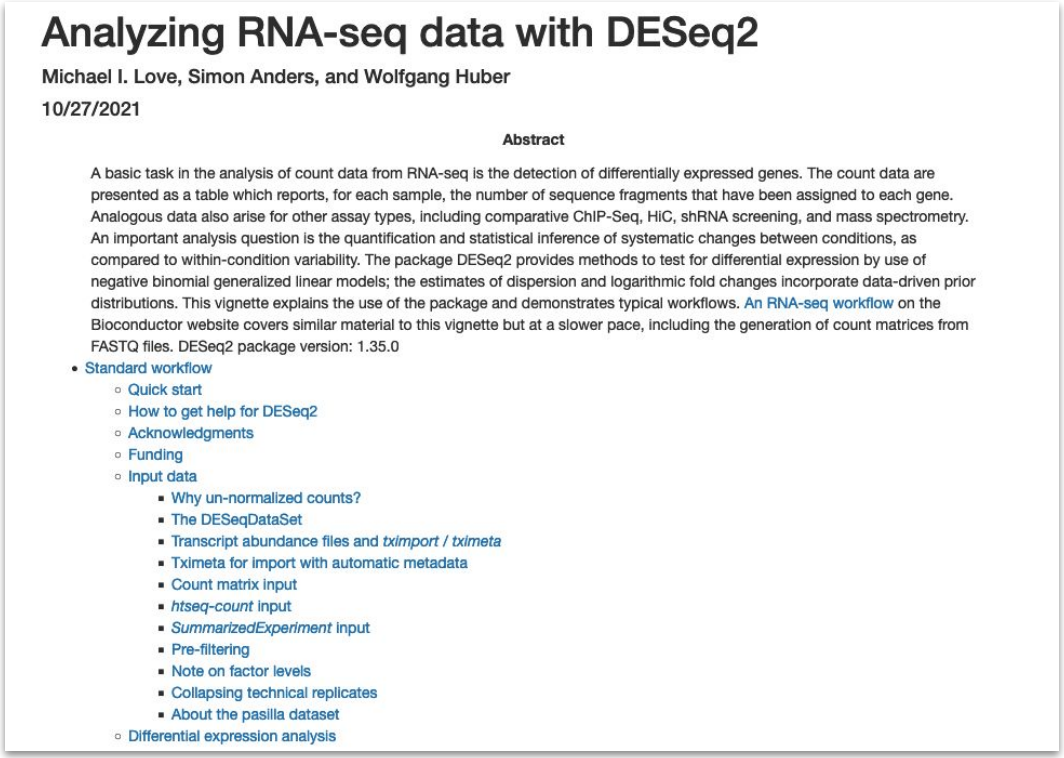
- Run SAMtools sort to convert and sort the alignment file in one step:

```
$ samtools sort --threads 2 -o
Control1_sorted.bam Control1.sam
```

- Index the new bam file:

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

Differential Expression Analysis with DESeq2



Analyzing RNA-seq data with DESeq2

Michael I. Love, Simon Anders, and Wolfgang Huber

10/27/2021

Abstract

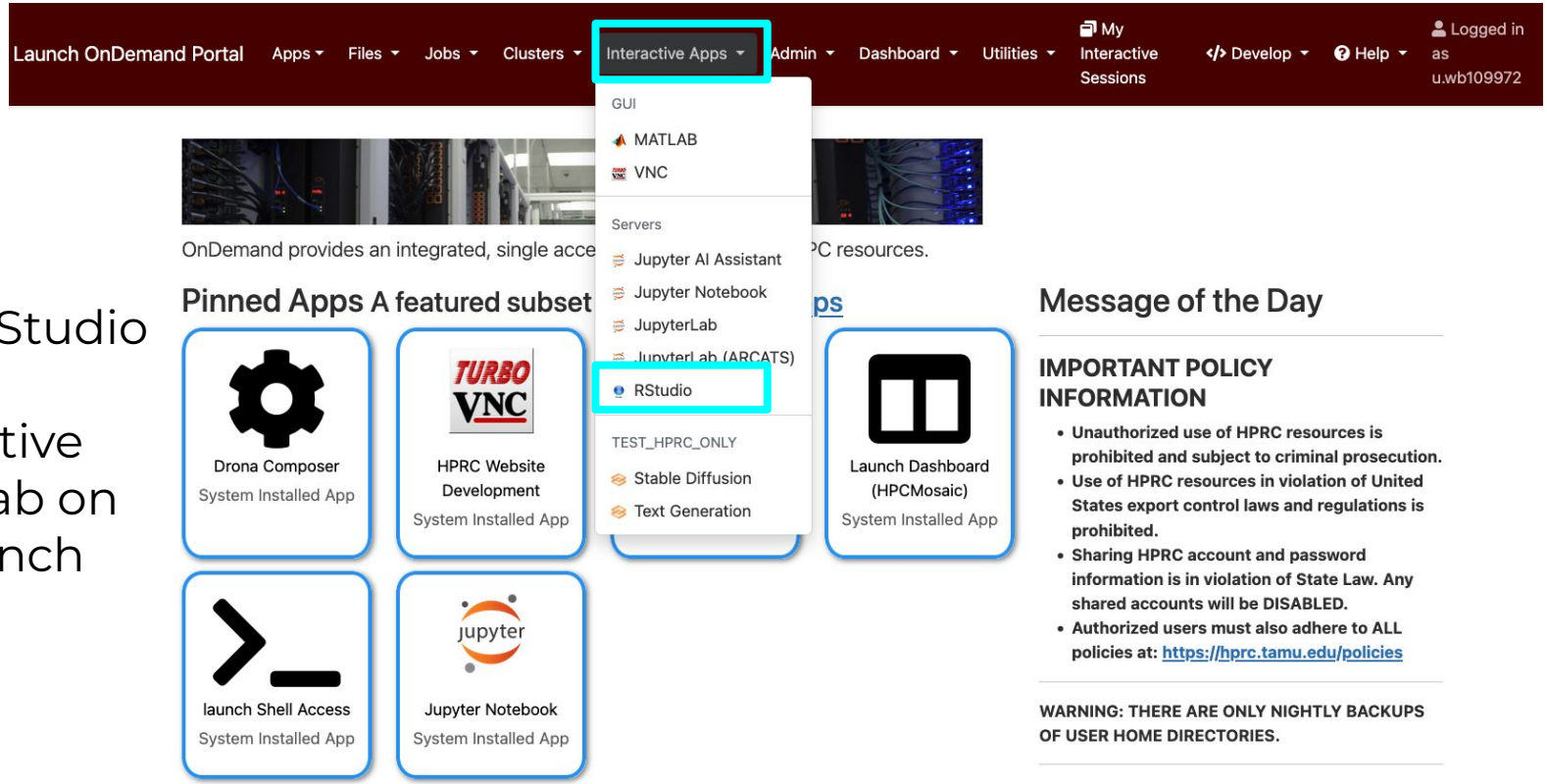
A basic task in the analysis of count data from RNA-seq is the detection of differentially expressed genes. The count data are presented as a table which reports, for each sample, the number of sequence fragments that have been assigned to each gene. Analogous data also arise for other assay types, including 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

- Standard workflow
  - Quick start
  - How to get help for DESeq2
  - Acknowledgments
  - Funding
  - Input data
    - Why un-normalized counts?
    - The DESeqDataSet
    - Transcript abundance files and *tximport* / *tximeta*
    - Tximeta for import with automatic metadata
    - Count matrix input
    - *htseq-count* input
    - *SummarizedExperiment* input
    - Pre-filtering
    - Note on factor levels
    - Collapsing technical replicates
    - About the pasilla dataset
  - Differential expression analysis

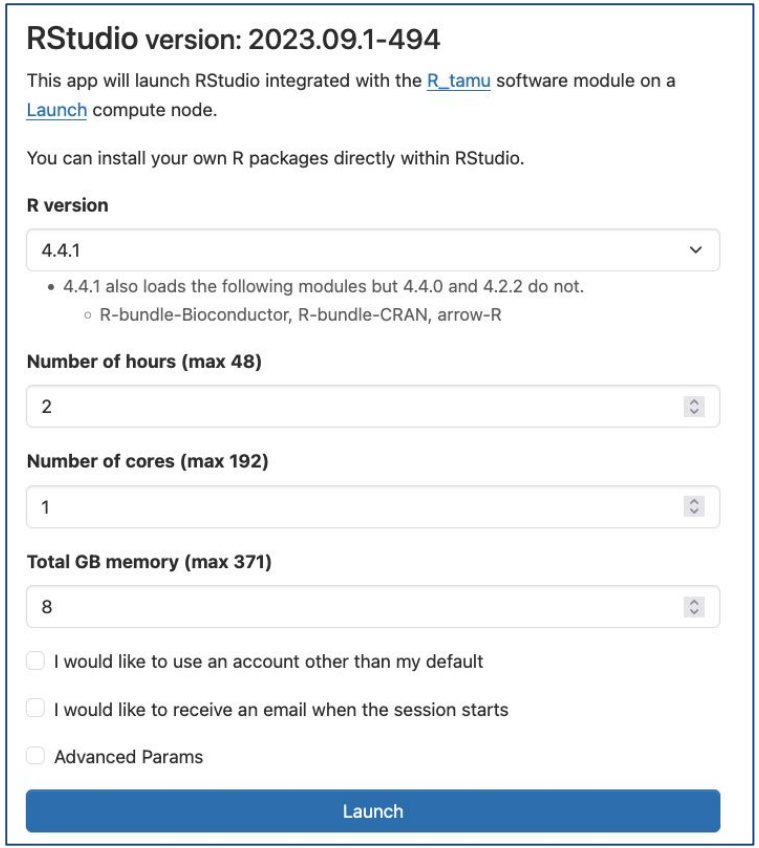
http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html

RStudio on Launch

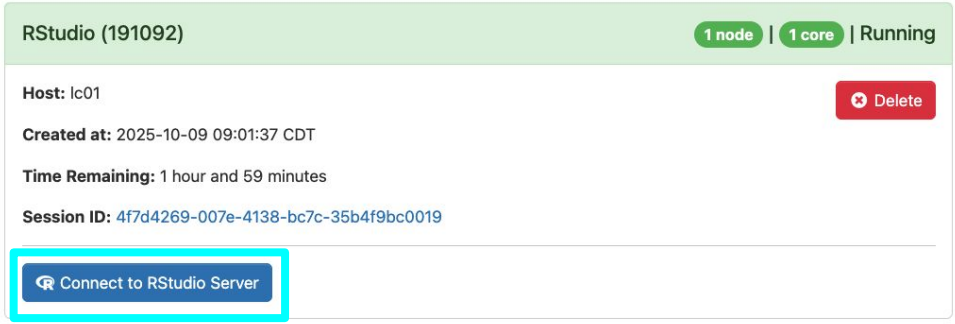
- Open RStudio in the “Interactive Apps” tab on the Launch portal



RStudio on Launch



- Select R version 4.4.1
- Set the number of hours to 2
- Set the number of cores to 1
- Set the Total GB memory to 8
- Click Launch Button
- Wait for the session to start
- Click “Connect to RStudio Server



Differential Expression Analysis

Open a new R script and set your working directory:

```
setwd("/scratch/user/username/RNA_class/R")
```

- Let's look at the contents of the directory and the sample table (in the console):

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

Differential Expression Analysis

- Load the required packages:

```
library("GenomicFeatures")
library("Rsamtools")
library("GenomicAlignments")
library("DESeq2")
library("ggplot2")
library("EnhancedVolcano")
library("pheatmap")
```

- Highlight this section of code in the script and click “Run”

Differential Expression Analysis

- This code was run previously to generate read counts (runtime ~ 40 minutes)
- You will need to run this for your own differential expression study

```
### Read in GFF using and extract genes using GenomicFeatures
mouseGFF <- makeTxDbFromGFF("GCF_000001635.27_GRCm39_genomic.gff",
format="gff")
exonsByGene <- exonsBy(mouseGFF, by="gene")
```

DO NOT RUN THESE COMMANDS!!!

Differential Expression Analysis

- This code was run previously to generate read counts (runtime ~ 40 minutes)
- You will need to run this for your own differential expression study

```
### Read in alignment files using Rsamtools
fls <- list.files(".", pattern=".bam$", full=TRUE)
bamLst <- BamFileList(fls, yieldSize = 100000)

### Count reads and generate summarized experiment using
GenomicAlignments
rdcnts <- summarizeOverlaps(exonsByGene, bamLst, mode="Union",
                            singleEnd=FALSE, ignore.strand=FALSE,
                            fragments=TRUE)
```

DO NOT RUN THESE COMMANDS!!!

Differential Expression Analysis

- This code was run previously to generate read counts (runtime ~ 40 minutes)
- You will need to run this for your own differential expression study

```
# Save the read counts (only needed if you wish to reload them later)

save(rdcnts, file = "read_counts.rda")
```

DO NOT RUN THESE COMMANDS!!!

Differential Expression Analysis

- Load in the previously saved read counts

```
load("read_counts.rda")
```

- You can see your environment now has the variable "rdcnts" to which is assigned the read counts that were previously generated

```
rdcnts
```

- Output:

```
class: RangedSummarizedExperiment
dim: 46333 8
metadata(0):
assays(1): counts
rownames(46333): 0610005C13Rik 0610006L08Rik ... n-TYgta9 n-Tcgca44
rowData names(0):
colnames(8): Control 1.bam Control 2.bam ... NAD 3.bam NAD 4.bam
```

Differential Expression Analysis

- Associate metadata with read counts:

```
### Get metadata associated with experiment (e.g. control vs. treatment)
sampleInfo <- read.csv("sampleTable.csv")
sampleInfo <- DataFrame(sampleInfo)

###Check that sample info and summarized experiment are in the same
order
colnames(rdcnts)
sampleInfo$sample

### Add metadata to summarized experiment
colData(rdcnts) <- cbind(colData(rdcnts), sampleInfo)
rdcnts
```

Differential Expression Analysis

- Output:

```
class: RangedSummarizedExperiment
dim: 46333 8
metadata(0):
assays(1): counts
rownames(46333): 0610005C13Rik 0610006L08Rik ... n-TYgta9 n-Tcgca44
rowData names(0):
colnames(8): Control_1.bam Control_2.bam ... NAD_3.bam NAD_4.bam
colData names(2): sample condition
```

Differential Expression Analysis

- Create the dds object, filter lowly expressed genes, and run the differential expression analysis:

```
### Build DESeqDataSet
dedata <- DESeqDataSet(rdcnts, design = ~ condition)

# Filter genes with low read counts
keep <- rowSums(counts(dedata)) >= 10
dedata <- dedata[keep,]

# Run DE Analysis and get the results
dea <- DESeq(dedata)
deresults <- results(dea, contrast=c("condition","NAD+","Control"))
deresults
```

Differential Expression Analysis

- Output:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
...                 ...            ...       ...       ...       ...       ...
n-TVcac2        4.61166      -2.473339  1.315784 -1.879745 0.0601428  0.249185
n-TWcca1        8.32216       0.599863  0.847017  0.708207 0.4788166  0.731300
n-TWcca5        3.97718      -0.964683  0.932859 -1.034115 0.3010825  0.591389
n-TYgta1        3.41003      -0.127466  1.018863 -0.125106 0.9004393  0.960367
n-TYgta5        1.40084      -3.201069  2.133385 -1.500465 0.1334941        NA
```

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
                baseMean log2FoldChange     lfcSE      stat    pvalue      padj
               <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Mean of normalized counts for all samples

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
                baseMean log2FoldChange     lfcSE      stat    pvalue      padj
               <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik    9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik  437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik    9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik   12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Log2 fold change: NAD supplement vs Control

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Log fold change standard error

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Wald statistic: NAD supplement vs Control

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

Wald test p value
(unadjusted)

Differential Expression Analysis

DESeq Results Explained:

```
log2 fold change (MLE): condition NAD+ vs Control
Wald test p-value: condition NAD. vs Control
DataFrame with 26679 rows and 6 columns
               baseMean log2FoldChange     lfcSE      stat    pvalue      padj
              <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
0610005C13Rik   9.98560     -0.4474544  0.765978 -0.584161  0.559112  0.781709
0610009B22Rik 437.17107      0.0394277  0.257005  0.153412  0.878073  0.950227
0610009E02Rik   9.91211      0.0800339  0.796742  0.100451  0.919986  0.970706
0610009L18Rik  12.08492      0.4734402  0.638896  0.741029  0.458676  0.718705
0610010F05Rik 533.81405      0.3834811  0.295820  1.296333  0.194861  0.477140
```

BH corrected p-values
(corrected for multiple testing)

Differential Expression Analysis

- How many genes are differentially expressed?

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

- Collect all the DEGs and write them to file:

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

PCA (Principal Component Analysis)

- Log transform the results and calculate the row variance

```
logTran <- rlog(dea)
rv <- rowVars(assay(logTran))
```

- Create a list of genes with the greatest variance:

```
select <- order(rv, decreasing = TRUE)[1:100]
```

PCA plot

- Run the principal component analysis (PCA)

```
PCA <- prcomp(t(assay(logTran)[select, ]))
summary(PCA)
```

- Output:

```
Importance of components:
                          PC1     PC2    PC3     PC4     PC5     PC6     PC7       PC8
Standard deviation     10.2855 3.62332 2.4524 1.95125 1.41120 0.94034 0.91944 3.426e-15
Proportion of Variance  0.7986 0.09911 0.0454 0.02874 0.01503 0.00668 0.00638 0.000e+00
Cumulative Proportion   0.7986 0.89776 0.9432 0.97191 0.98694 0.99362 1.00000 1.000e+00
```

PCA plot

- Set up the PCA for ggplot2

```
ggPCA_out <- as.data.frame(PCA$x)
ggPCA_out <- cbind(ggPCA_out, sampleInfo)
head(ggPCA_out)
```

- Output:

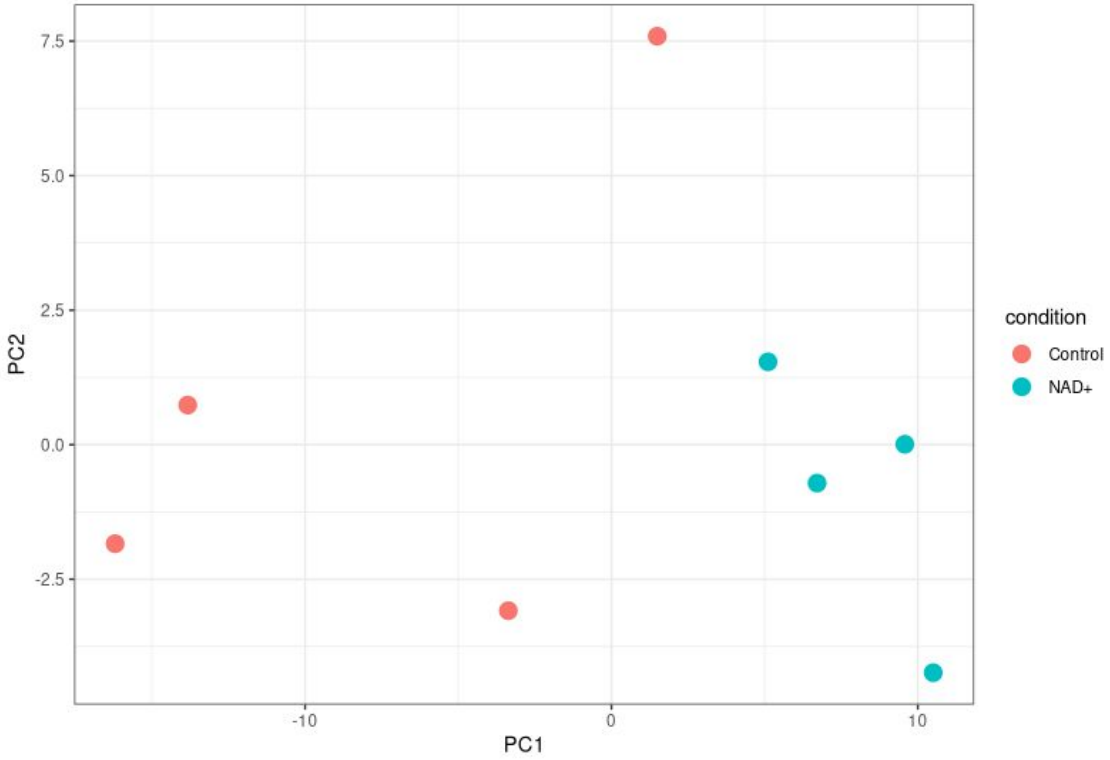
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	sample	condition
Control_1.bam	-13.831986	0.7365321	-1.5921315	-1.026889	2.6391440	-0.1557725	-0.209471735	3.719259e-15	Control 1	Control
Control_2.bam	-16.199870	-1.8398356	-0.1905942	-1.093955	-2.3782932	-0.1180327	0.421495197	3.085138e-15	Control 2	Control
Control_3.bam	-3.366288	-3.0849304	2.9935568	3.680902	0.5251263	0.3059604	-0.004739673	3.307655e-15	Control 3	Control
Control_4.bam	1.494677	7.5909002	-1.2581734	1.971045	-0.7243558	-0.3807520	0.056693950	3.100875e-15	Control 4	Control
NAD_1.bam	5.110172	1.5401130	0.9138034	-1.528954	-0.1487299	2.0516708	-0.382566885	2.963949e-15	NAD+ 1	NAD+
NAD_2.bam	6.712872	-0.7138124	1.8365213	-1.311407	-0.3233619	-1.1536478	-1.616245692	3.071854e-15	NAD+ 2	NAD+

PCA plot

- Plot the PCA

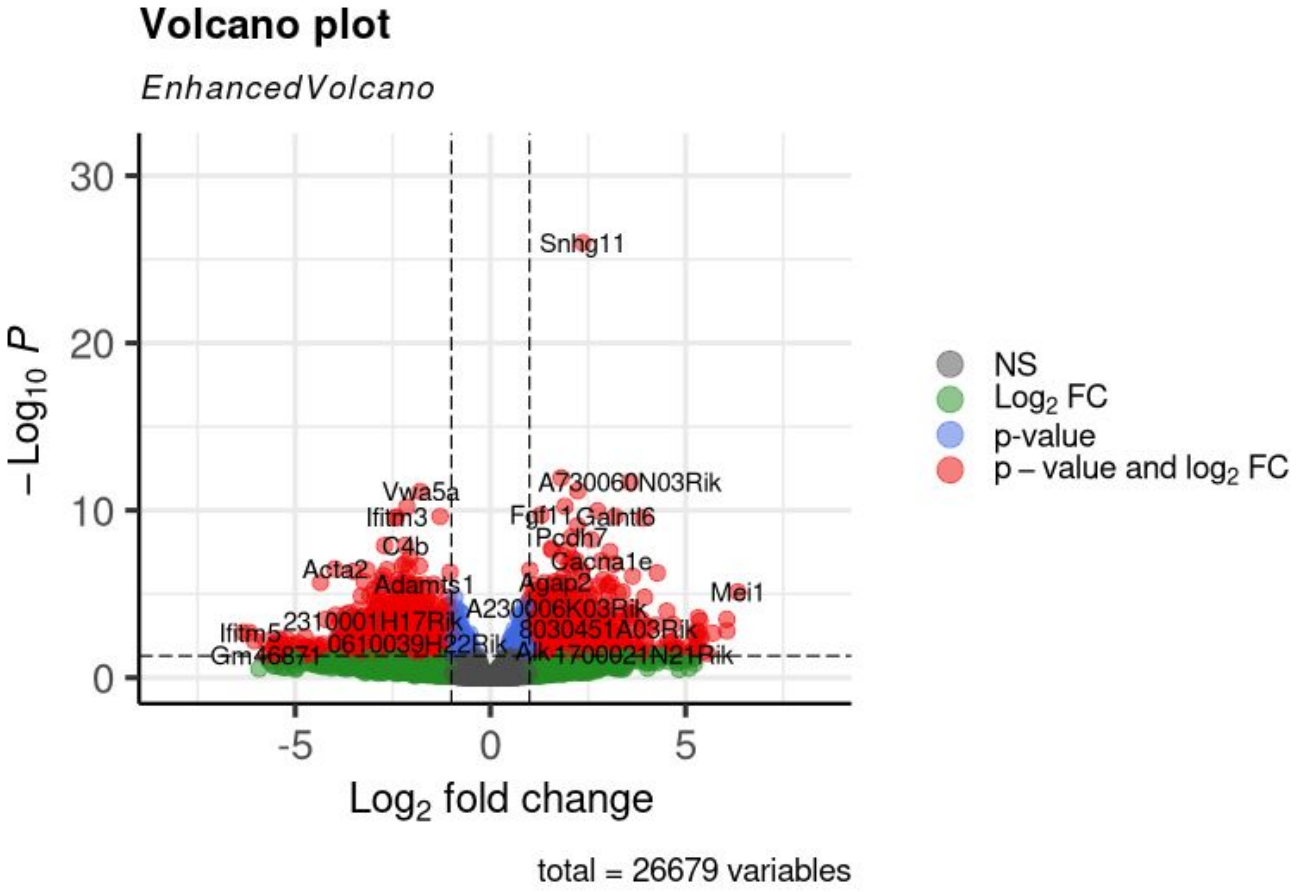
```
ggplot(ggPCA_out, aes(x=PC1,y=PC2,color=condition)) +
    geom_point(size=4) +
        theme_bw()
```

PCA plot



Volcano Plot

```
EnhancedVolcano(deresults,
                lab = rownames(deresults),
                x = 'log2FoldChange',
                y = 'padj',
                pCutoff = 0.05,
                FCcutoff = 1.0,
                pointSize = 3.0,
                labSize = 4.0,
                colAlpha = 1/2,
                drawConnectors = FALSE,
                legendPosition = "right")
```


Volcano plot
EnhancedVolcano
30
20
10
0
−Log10 P
Snhg11
A730060N03Rik
Vwa5a
Ifitm3
Fgf11
Galntl6
Pcdh7
C4b
Gacha1e
Acta2
Adamts1
Agap2
Mei1
A230006K03Rik
2310001H17Rik
8030451A03Rik
Ifitm5
0610039H22Rik
Alk
1700021N21Rik
Gm46871
-5
0
5
Log2 fold change
total = 26679 variables
NS
Log2 FC
p-value
p – value and log2 FC

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

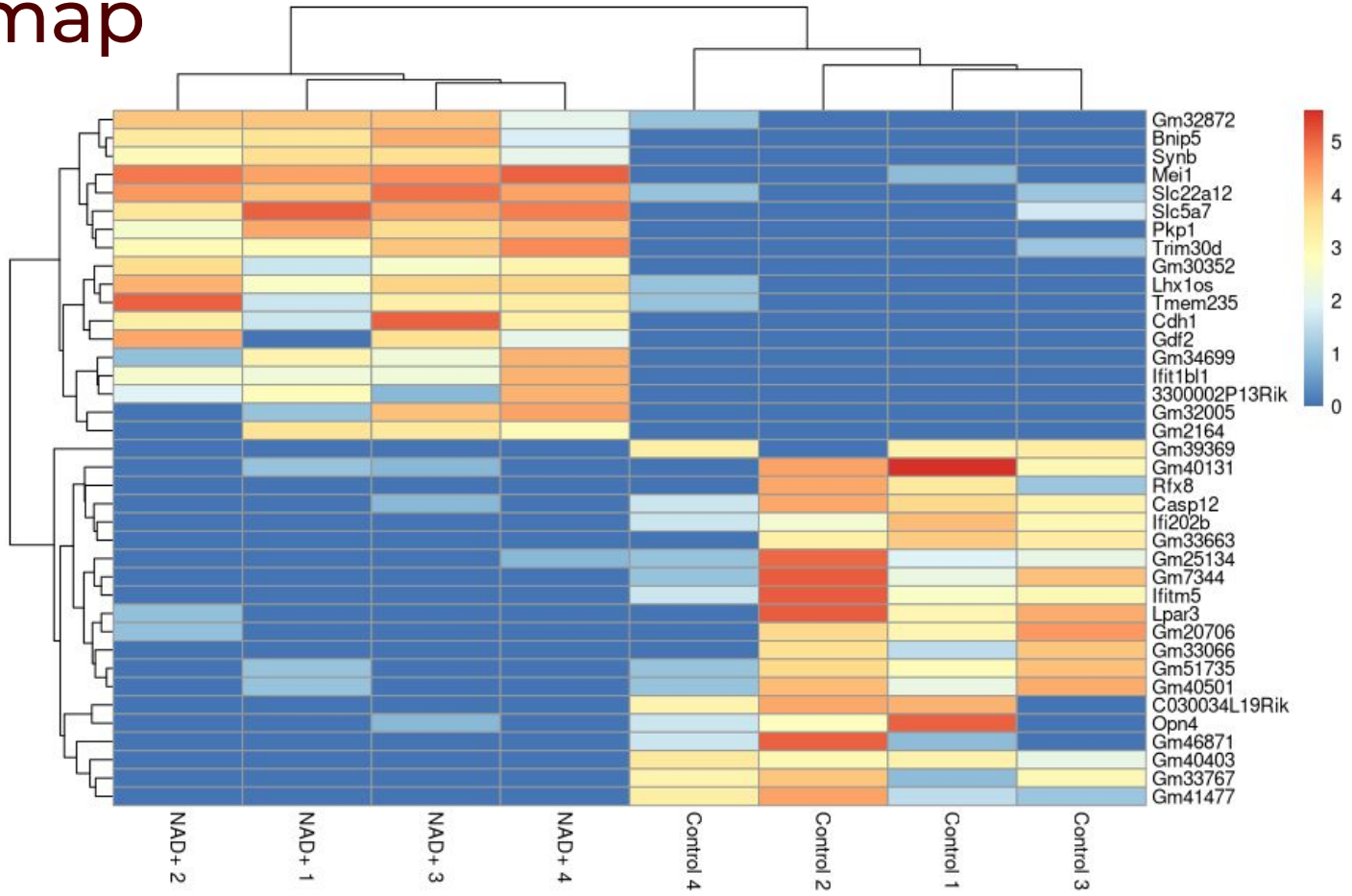
```
> selected
 [1] "Mei1"         "Slc22a12"      "Pkp1"          "Slc5a7"        "Lhx1os"        "Cdh1"          "Trim30d"
 [8] "Gm7344"       "Ifitm5"        "Bnip5"         "Gm32872"       "Synb"          "Casp12"        "Tmem235"
[15] "Gm40403"      "Lpar3"         "C030034L19Rik" "Ifi202b"       "Gm40131"       "Ifit1bl1"      "Gm33767"
[22] "Gm51735"      "Gm40501"       "Gm30352"       "Gm41477"       "Gm20706"       "Gm34699"       "Gm33663"
[29] "Gm25134"      "3300002P13Rik" "Gdf2"          "Opn4"          "Gm39369"       "Gm32005"       "Gm33066"
[36] "Gm2164"       "Rfx8"          "Gm46871"
```

Heatmap

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

```
transformed_readcounts <- normTransform(dea)
pheatmap(assay(transformed_readcounts)[selected,],
             cluster_rows = TRUE, show_rownames = TRUE,
             cluster_cols = TRUE,
             labels_col = colData(dea)$sample)
```

Heatmap


Gm32872
Bnip5
Synb
Mei1
Slc22a12
Slc5a7
Pkp1
Trim30d
Gm30352
Lhx1os
Tmem235
Cdh1
Gdf2
Gm34699
Ifit1bl1
3300002P13Rik
Gm32005
Gm2164
Gm39369
Gm40131
Rfx8
Casp12
Ifi202b
Gm33663
Gm25134
Gm7344
Ifitm5
Lpar3
Gm20706
Gm33066
Gm51735
Gm40501
C030034L19Rik
Opn4
Gm46871
Gm40403
Gm33767
Gm41477
5
4
3
2
1
0
NAD+ 2
NAD+ 1
NAD+ 3
NAD+ 4
Control 4
Control 2
Control 1
Control 3

HIGH PERFORMANCE RESEARCH COMPUTING
TEXAS A&M UNIVERSITY

https://hprc.tamu.edu

HPRC Helpdesk:

help@hprc.tamu.edu
Phone: 979-845-0219

Take our short course survey!

https://u.tamu.edu/hprc_shortcourse_survey

Help us help you. Please include details in your request for support, such as, Cluster (FASTER, Grace, ACES, Launch), NetID (UserID), Job information (Job id(s), Location of your jobfile, input/output files, Application, Module(s) loaded, Error messages, etc), and Steps you have taken, so we can reproduce the problem.