# Practical RNA-seq analysis

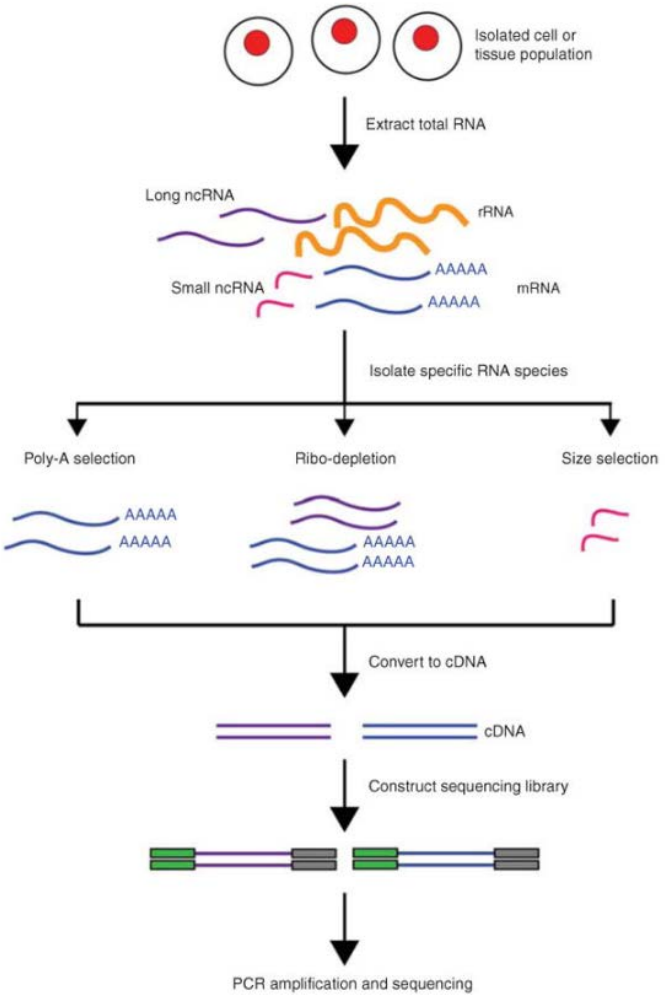
Inma Barrasa
Bioinformatics and Research Computing (BaRC)
Whitehead Institute

Feb 13th 2020

http://barc.wi.mit.edu/hot_topics/

WHITEHEAD INSTITUTE

# Overview of RNA-seq

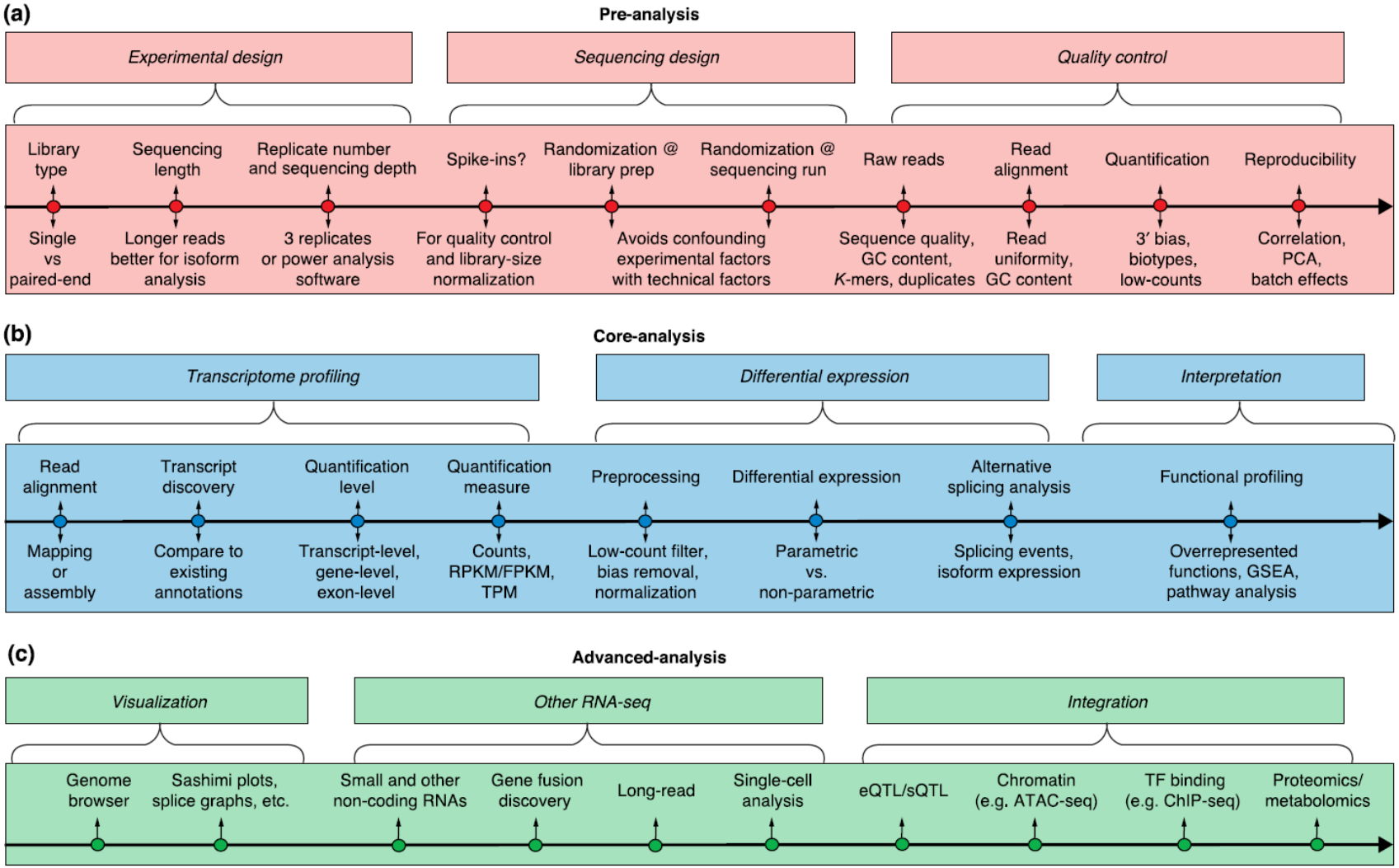



*RNA Sequencing and Analysis*
*K. R. Kukurba and S. B. Montgomery*
*Cold Spring Harb Protoc. 2015 Nov; 2015(11): 951–969.*

# Why do RNA-seq?

- RNA-seq includes experiments to
  - **Profile abundance of mRNA and other RNAs**
  - **Identify "differentially expressed" genes**
  - Identify alternated spliced transcript isoforms
  - Assemble transcriptome
  - Identify variants in transcribed regions of genome

# RNA-seq Analysis Overview




Conesa, A., et al. *A survey of best practices for RNA-seq data analysis* Genome Biology (2016)

# Outline

- Experimental design*
- Quality control
- Sequence preparation*
- Mapping spliced reads
- Counting gene levels
- Normalization and identifying "differentially expressed" genes
- Creating figures and summaries

*not included in the hands-on exercises

# Hands-on exercises

- All data is in
  /nfs/BaRC_Public/Hot_Topics/RNA-seq_2020
- Create directories on tak
- Link to data files on BaRC_Public

- See handout for series of commands (step 0)

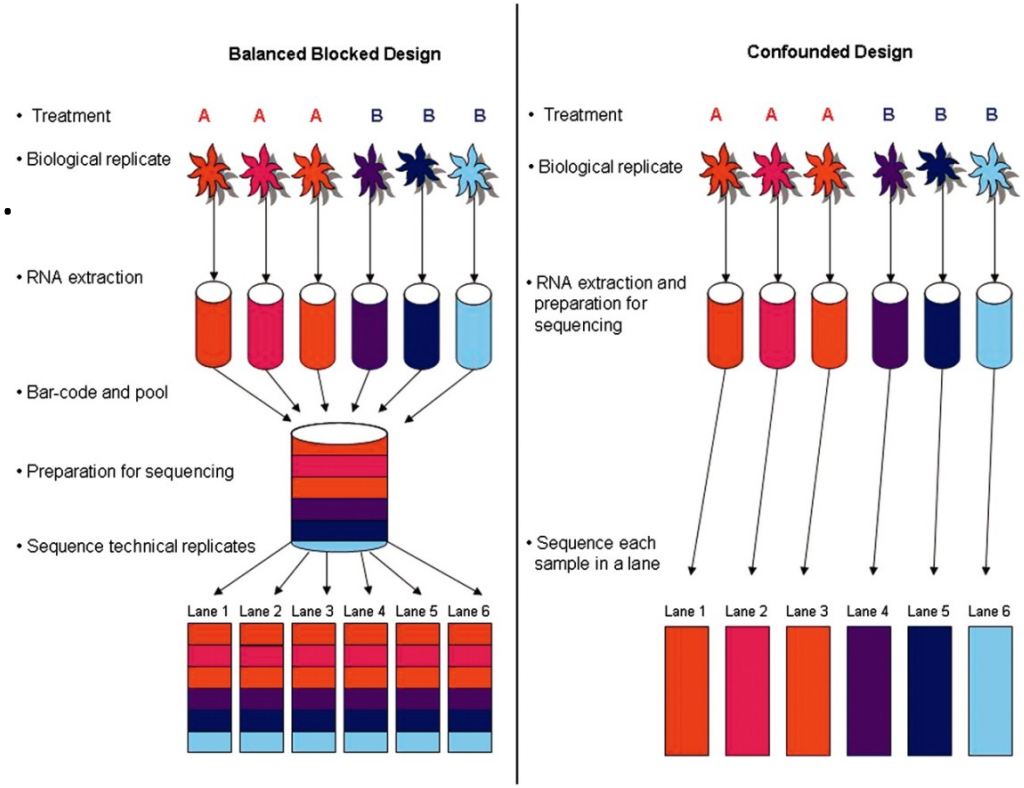
- Commands can be copied from file
  RNA-seq_2020_commands.txt

# Experimental Design

- Replication is essential if results with confidence are desired.
- With the combination of high numbers of reads per sample and multiplexing, the number of Illumina lanes can be much fewer than number of samples.
- Lots of details to think about:
  - Has someone already done an experiment like this?
  - Total RNA or poly(A) RNA or …
  - Number of samples?
  - Read length?
  - Paired or unpaired reads?
  - Number of reads?
  - What reference genome to use?
  - Stranded or unstranded?
  - What reference transcriptome to use?

# Experimental Design

- Lots of data typically cannot make up for a poor experimental design.
- Look out for bias and confounding.
- Short-read sequencing requires an effectively designed experiment.
- See BaRC about reducing batch effects




Auer, PL and Doerge, RW *Statistical design and analysis of RNA sequencing data* Genetics (2010)

# QC Before Alignment

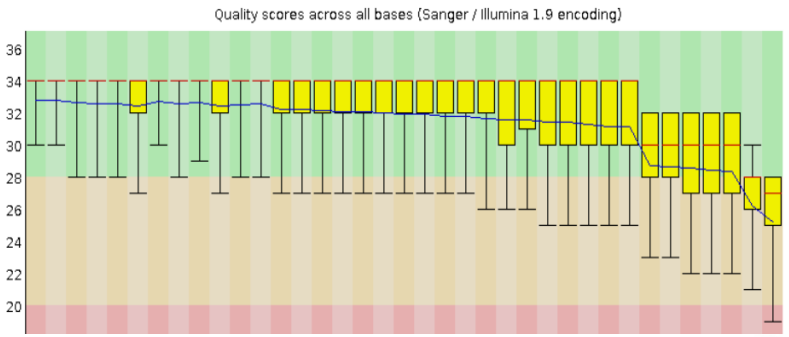
- FastQC, use mulitQC to view
- Check quality of file of raw reads (fastqc_report.html)
- Respond to QC analysis:
  - Filter poor-quality reads
  - Trim poor-quality positions
  - Trim adapter and/or other vector
- Check quality of file of modified reads
- See Hot Topics on quality control Jan 2017
- See handout for fastqc command (step 1)

```
@WIGTC-HISEQ3:1:1212:18183:1997#TTAGGC/1
NCCACAACAGGGCACGGTGCGGAATAGAGAACTATCCCTT
+WIGTC-HISEQ3:1:1212:18183:1997#TTAGGC/1
B[[[[aeaeeaeeeeeeeeeeeeeeeeeeeeeeeeeeee
```

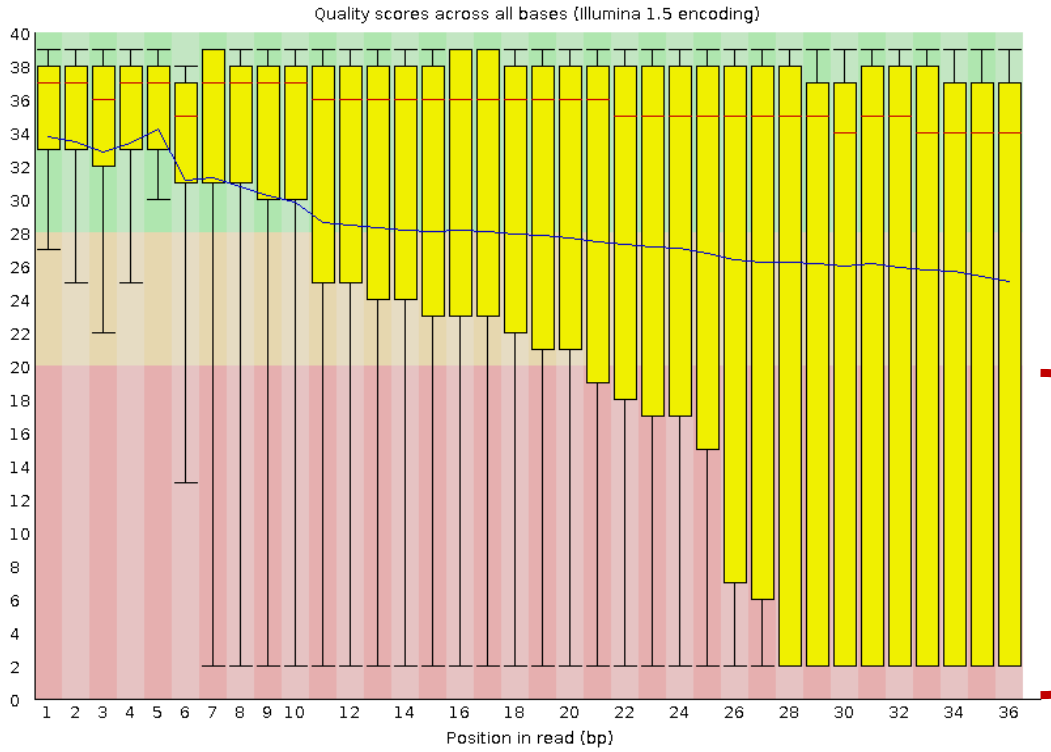
**Basic Statistics**

| Measure | Value |
|---|---|
| Filename | NA07357_CEU_1.fastq |
| File type | Conventional base calls |
| Encoding | Sanger / Illumina 1.9 |
| Total Sequences | 7940176 |
| Filtered Sequences | 0 |
| Sequence length | 37 |
| %GC | 56 |

**Per base sequence quality**



# FastQC: per base sequence quality




very good quality calls

reasonable quality

poor quality

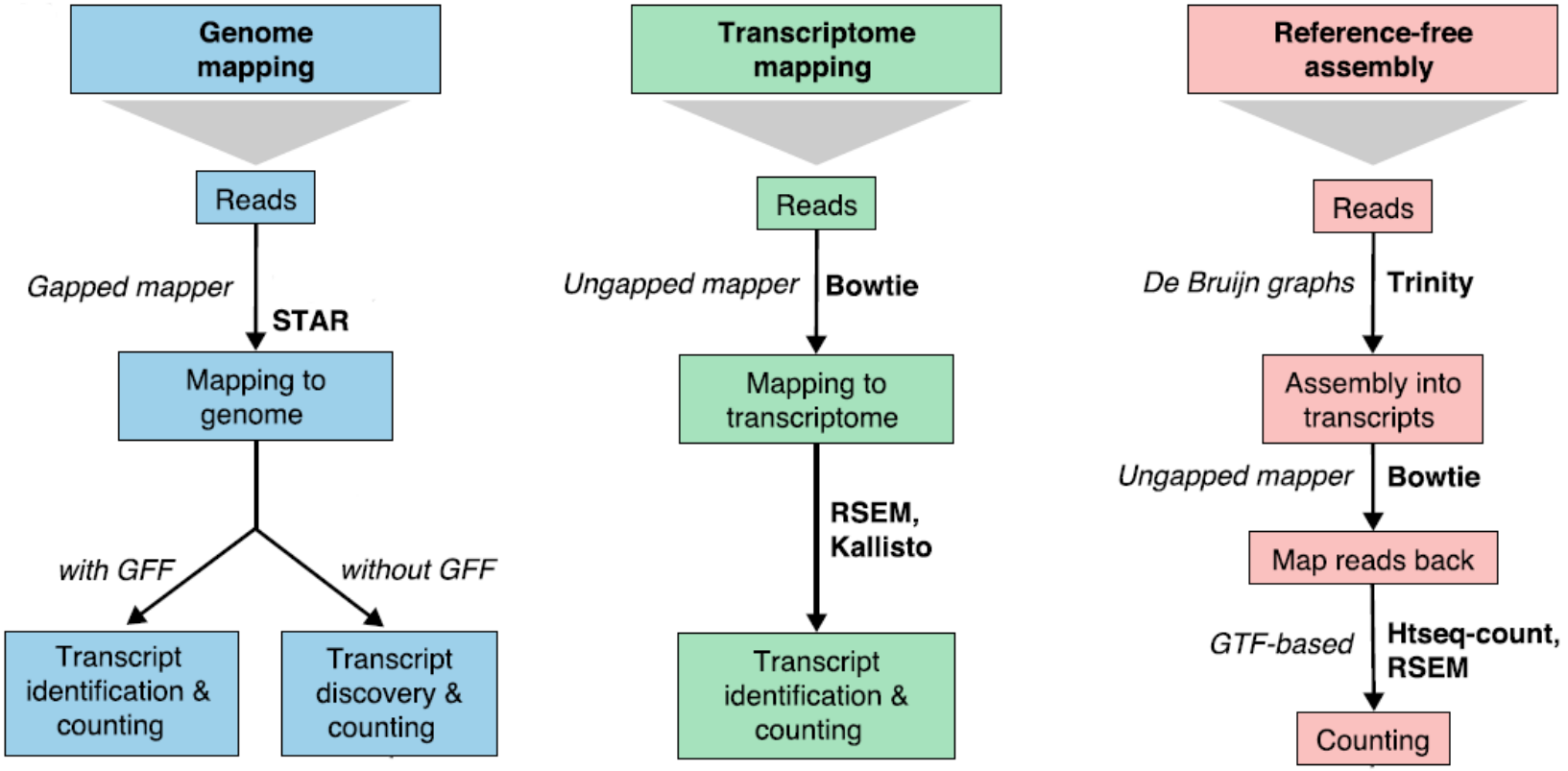
Red: median blue: mean yellow: 25%, 75% whiskers: 10%, 90%

Quality = 10 => error rate = 10% => base call has 90% confidence
Quality = 20 => error rate = 1% => base call has 99% confidence
Quality = 30 => error rate = 0.1% => base call has 99.9% confidence

# Responding to Quality Issues

- Method 1:
  - Drop all poor-quality reads
  - Trim poor-quality bases
  - Map only good-quality bases
- Method 2:
  - Keep all reads as is
  - Map as many as possible
  - Current mappers incorporate the read quality score into the mapping quality score

# Mapping Considerations




Adapted from Conesa, A., et al. *A survey of best practices for RNA-seq data analysis* Genome Biology (2016)

# RNA-Seq Genome Mapping

- Reads can be mapped with a **splice-aware alignment tool** such as STAR (recommended)
- The ideal tool should map to best hit(s), whether to continuous or spliced genome segments
- Look at alignment statistics and mapped reads in a browser (and re-map if needed).

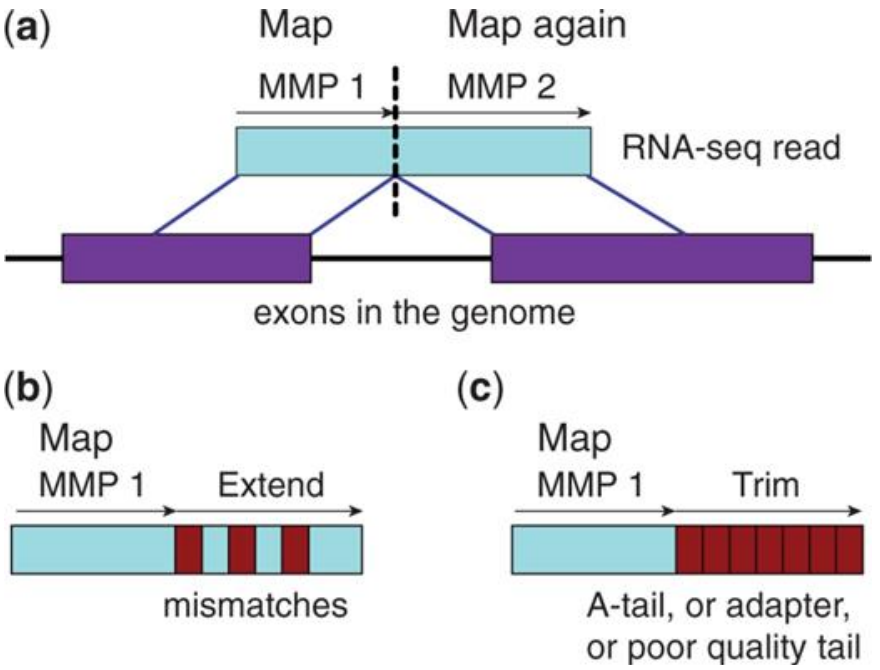
# Mapping considerations

- Type of **quality score encoding**.
- Use all or just **canonical** chromosomes?
- Include known splice junctions (in GTF file).
- Do you want to look for novel splice junctions?
- How short of a sub-read should map to an exon boundary?
- How long are your introns? Specify maximum **intron length** allowed.

# STAR Aligner

I. Sequential search for Maximal Mappable Prefix (MMP)

II. Stitch together all the seeds that were aligned to the genome from I)




Dobin, A., et al. *STAR: ultrafast universal RNA-seq aligner* Bioinformatics (2012)

# Alignment with STAR

- Create genome index using *genomeGenerate,* also see /nfs/genomes

- Run alignment, e.g.

STAR --genomeDir /path/to/GenomeDir --readFilesIn /path/to/read1.fq.gz /path/to/read2.fq.gz --sjdbScore 2 --outFileNamePrefix whateverPrefix --runThreadN 8 --readFilesCommand zcat --outSAMtype BAM SortedByCoordinate

| | |
|---|---|
| **--runMode <alignReads, genomeGenerate>** | "alignReads" does the actual mapping. "genomeGenerate" generates the genomeDir required for mapping (default = alignReads). |
| **--genomeDir </path/to/GenomeDir>** | Specifies the path to the directory used for storing the genome information created in the genomeGenerate step. |
| **--readFilesIn <read1.fastq read2.fastq>** | Specifies the fastq files containing the reads, can be single-end or paired-end. |
| **--sjdbScore <n>** | Provides extra alignment score for alignments that cross database junctions (default = 2). |
| **--runThreadN <n>** | Specifies the number of threads to use. |
| **--readFilesCommand <cmd>** | Specifies the command to uncompress compressed fastq files. For gzipped files (*.gz) use --readFilesCommand zcat. |
| **--outSAMtype <BAM sortingMode>** | Specifies the type of BAM file to create. Options: 'BAM Unsorted', 'BAM SortedByCoordinate', 'BAM Unsorted SortedByCoordinate' (to create both unsorted and sorted BAMs) |

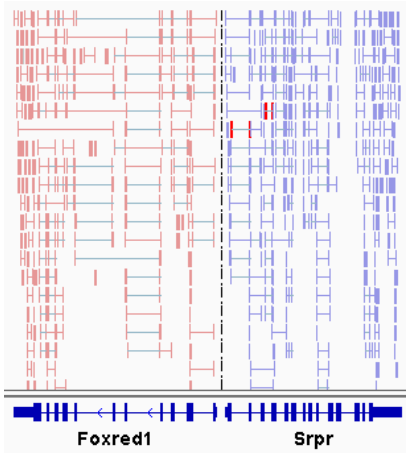
See handout for STAR command (step 2)

# Hands on

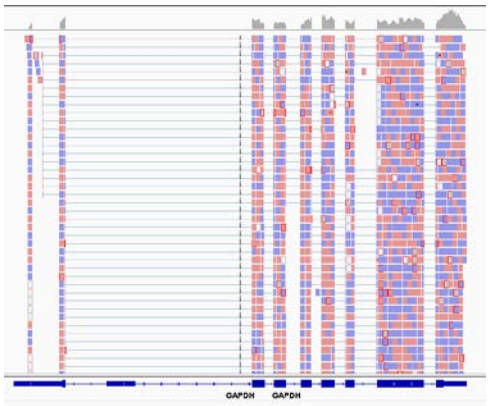
- How does "gene expression" (really: transcript levels) differ between European and Africans?
- European samples (n=2) Montgomery et al., 2010
  - 37nt paired-end reads
- African samples (n=2) Pickrell et al., 2010
  - 46nt single-end reads
- Sample data contains about 10% of total reads
- Is this a good design?

# QC after Alignment

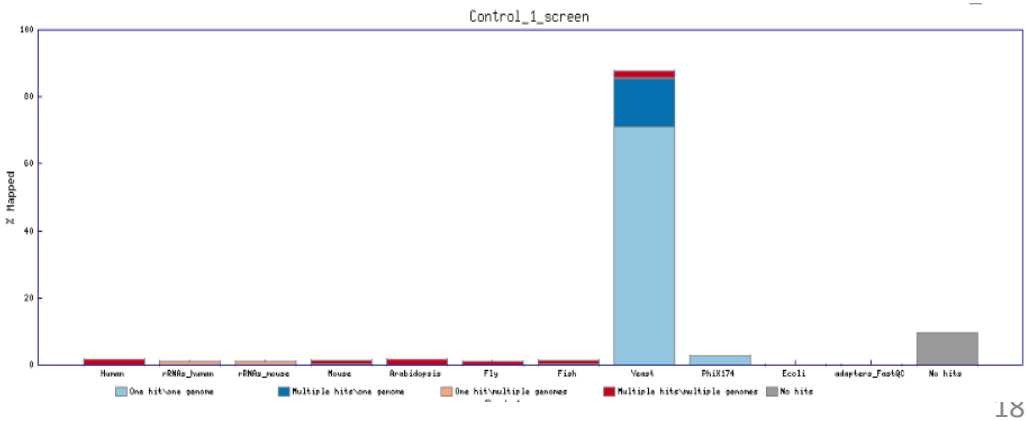
- Confirm that reads are stranded or unstranded
  - Run infer_experiment.py (from **RseQC** package)
  - Look at BAM reads in genome browser
- Contamination?
  - **FastQ Screen**




*stranded*

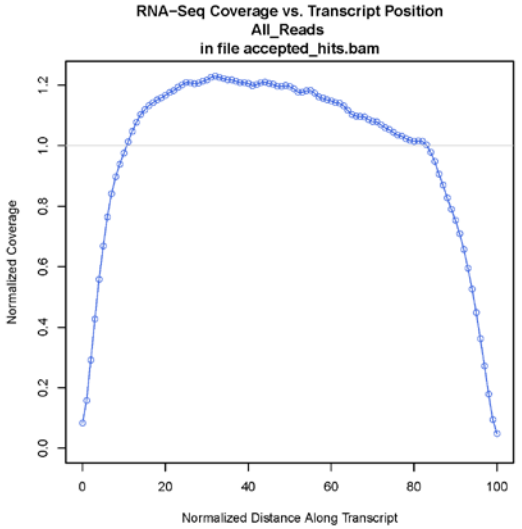


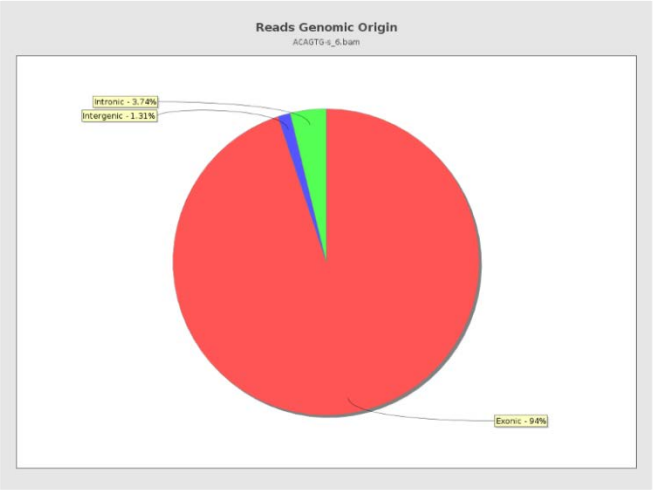

*unstranded*



# QC after Alignment

- Do reads cover the length of a typical transcript, or is there 3' or 5' bias?
  - Run **Picard** tool: CollectRnaSeqMetrics
- What fraction of reads map to annotated exons?
  - Run '**qualimap** rnaseq'
- See BaRC SOPs for commands

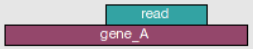
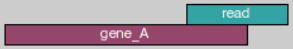
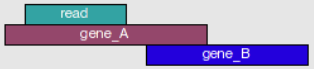


# Counting RNA-seq Features

- What features are of interest? Gene, transcript, and/or exon counts?
- How do we want to treat a read maps to multiple places?
- What happens if a read overlaps with multiple features?
- Does the direction of a read need to agree with the direction of the feature? Is RNA stranded, reversed strand or unstranded.

htseq-count "modes"

| | union | intersection_strict | intersection_nonempty |
|---|---|---|---|
|  | gene_A | gene_A | gene_A |
|  | gene_A | no_feature | gene_A |
|  | gene_A | no_feature | gene_A |
|  | gene_A | gene_A | gene_A |
|  | gene_A | gene_A | gene_A |
|  | ambiguous | gene_A | gene_A |
|  | ambiguous | ambiguous | ambiguous |

# Counting Methods

- featureCounts (recommended)

  bioinf.wehi.edu.au/featureCounts/

  – Output is raw counts

- htseq-count

  htseq.readthedocs.io/en/master/count.html

  – Output is raw counts

- Cufflinks

  cole-trapnell-lab.github.io/cufflinks/

  – Output is FPKM and related statistics

- Bedtools (intersectBed; coverageBed)

  bedtools.readthedocs.io/

  Output is raw counts (but may need post-processing)

# Running featureCounts

Count reads mapping to the specified gene models:

- Usage:

```
featureCounts [options] -a <annotation_file> -o
<output_file> input_file1 [input_file2] ...
```

- Example:

```
#single-end reads (unstranded)
 featureCounts -a gene_anotations.gtf -o
 MySample.featureCounts.txt MySample.bam
#paired-end reads (forward stranded)
 featureCounts –p -s 1 -a gene_anotations.gtf –o
 MySample.featureCounts.txt MySample.sorted.bam
```

# Running featureCounts: Options

| Option | Description |
|---|---|
| --minOverlap | Minimum number of overlapping bases in a read that is required for read assignment. 1 by default. |
| --fracOverlap | Minimum fraction of overlapping bases in a read that is required for read assignment. |
| -M | Multi-mapping reads will also be counted. |
| -O | Assign reads to all their overlapping meta-features (or features if -f is specified). |
| --fraction | Assign fractional counts to features. |
| -s | Perform strand-specific read counting.<br>Acceptable values:<br>0 (unstranded), 1 (stranded) and<br>2 (reversely stranded).<br>0 by default. |

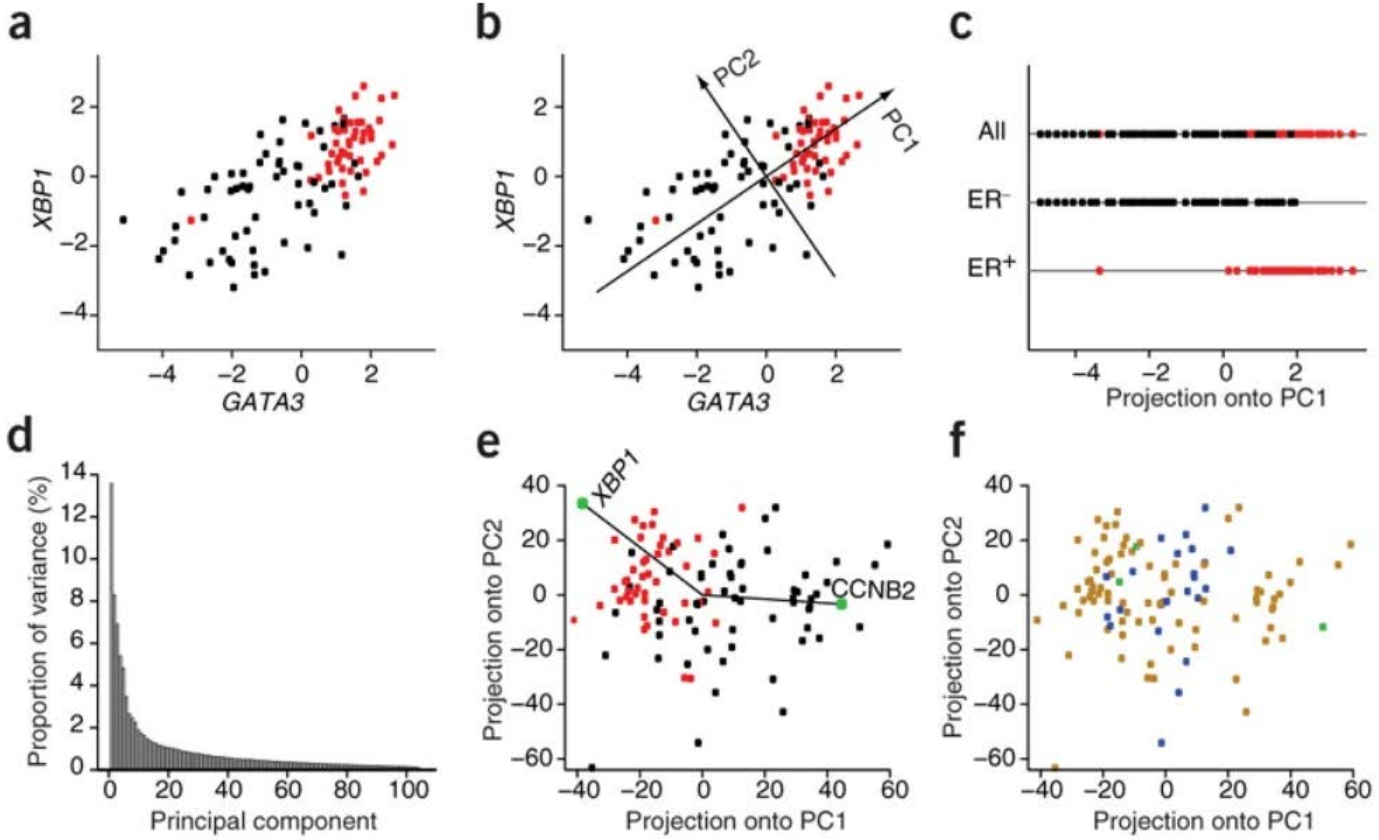
See handout for featureCounts commands (step 3)

# Normalization

- Raw counts cannot be compared directly
- Correct for sequencing depth (i.e. library size)
  - CPM: counts per million
  - DESeq: Relative Log Expression (RLE)
  - edgeR: Trimmed Mean M-values (TMM)
  - FPKM*: fragments per kilobase per million mapped reads
  - TPM*: transcripts per million

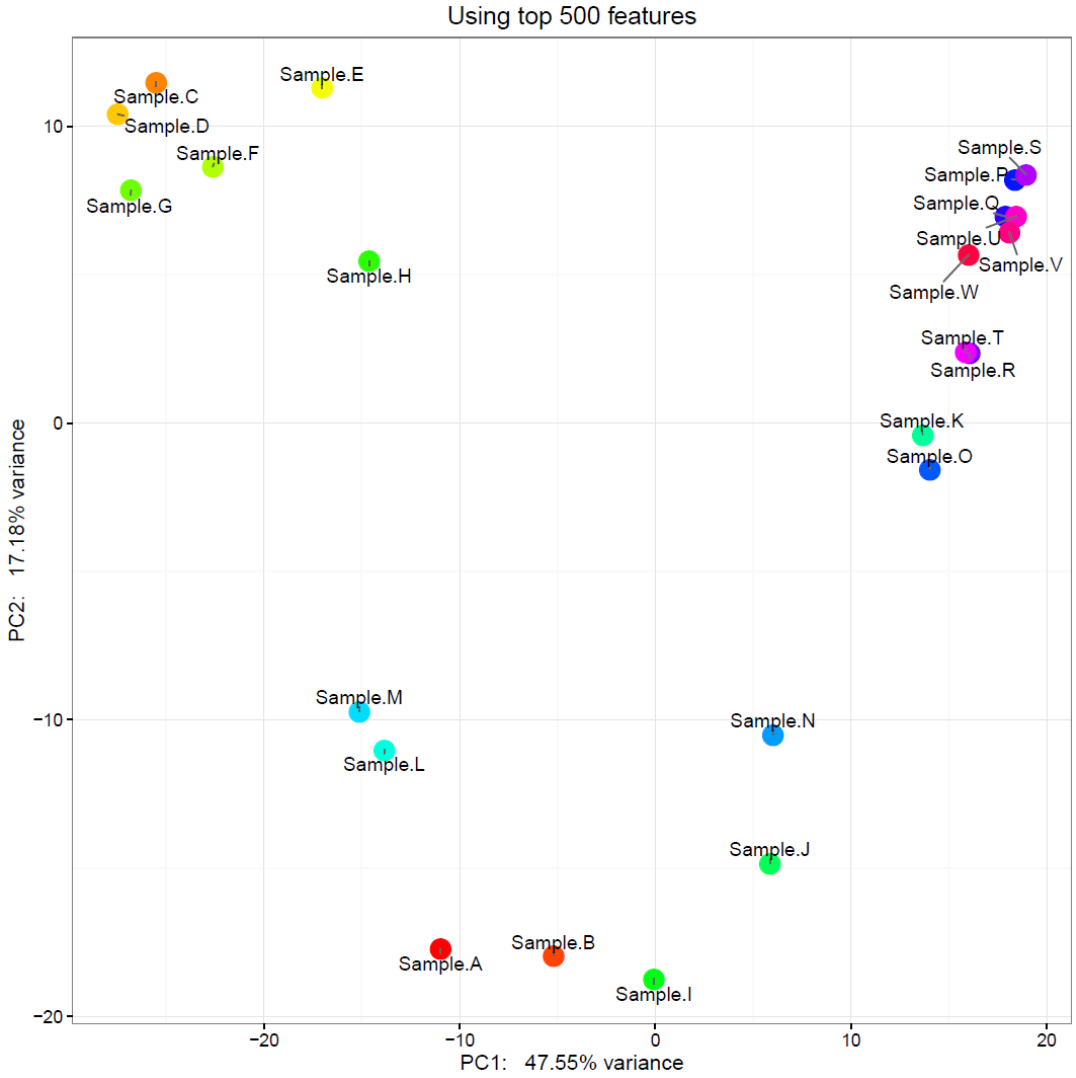
*can be used to compare across genes or transcripts

# Using Principal Components Analysis to explore your data




Ringnér, M. What is principal component analysis?. Nat Biotechnol 26, 303–304 (2008). https://doi.org/10.1038/nbt0308-303

# Using PCA to explore your data




Are there batch effects on the data?

Talk to BaRC if you want to learn how to remove batch effects.

# Differential Expression Methods

- Count-based methods (recommended)
  - Input is matrix of raw counts
  - DESeq2 (R package) -- recommended
  - edgeR (R package)
  - Typically used to compare gene counts
- Accounting for batch effects on count-based methods
  - DESeq2: `dds = DESeqDataSet(se, design = ~ batch + condition)`
  - edgeR: `design = model.matrix(~Batch+Treatment)`

  See http://barcwiki.wi.mit.edu/wiki/SOPs/rna-seq-diff-expressions
- Cufflinks (cuffdiff)
  - Inputs are mapped reads (SAM format)
  - Typically used to compare gene and transcript counts

- See handout for DESeq2 commands (step 4)

# Interpreting DESeq2 output

| Gene ID (from GTF file) | Mean norm counts | Log2 (fold change) | logFC std error | Wald statistic | Raw p-value | FDR p-value | Raw counts | | | | Normalized counts = raw / (size factor) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Feature.ID | baseMean | log2(YRI/CEU) | lfcSE | stat | pvalue | padj | CEU_NA07357 | CEU_NA11881 | YRI_NA18502 | YRI_NA19200 | CEU_NA07357.norm | CEU_NA11881.norm | YRI_NA18502.norm | YRI_NA19200.norm |
| ENSG00000251705 | 114.15 | -3.48 | 0.46 | -7.60 | 2.90E-14 | 4.48E-11 | 197 | 250 | 8 | 14 | 172.84 | 257.56 | 5.77 | 20.42 |
| ENSG00000236552 | 66.39 | -3.86 | 0.52 | -7.47 | 8.29E-14 | 1.07E-10 | 180 | 96 | 2 | 5 | 157.93 | 98.9 | 1.44 | 7.29 |
| ENSG00000226958 | 2073.59 | -2.45 | 0.38 | -6.52 | 6.88E-11 | 7.58E-08 | 3324 | 4148 | 407 | 556 | 2916.42 | 4273.49 | 293.32 | 811.14 |
| ENSG00000064886 | 54.20 | 3.31 | 0.54 | 6.09 | 1.11E-09 | 1.05E-06 | 9 | 1 | 84 | 101 | 7.9 | 1.03 | 60.54 | 147.35 |
| ENSG00000198786 | 2006.17 | -1.95 | 0.32 | -6.08 | 1.23E-09 | 1.05E-06 | 4601 | 2391 | 834 | 633 | 4036.84 | 2463.33 | 601.05 | 923.47 |
| ENSG00000100292 | 58.87 | -2.86 | 0.47 | -6.04 | 1.54E-09 | 1.19E-06 | 131 | 98 | 13 | 7 | 114.94 | 100.96 | 9.37 | 10.21 |

```
sizeFactors (from DESeq2):

CEU_NA07357 CEU_NA11881 YRI_NA18502 YRI_NA19200
  1.1397535   0.9706359   1.3875763   0.6854579
```

# Differential Expression Issues

- Given that statistics are
  - based on complex models
  - influenced by even more complex biology

  The p-values may not be accurate but can be very effective at ranking genes

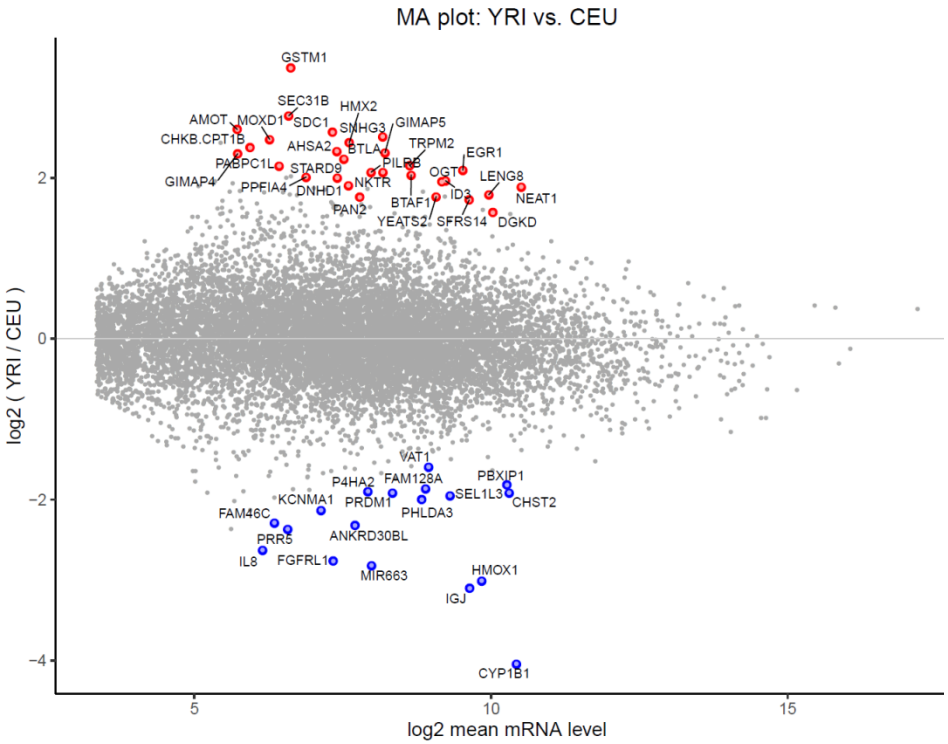
- Statistics don't work very well when one sample has no counts.

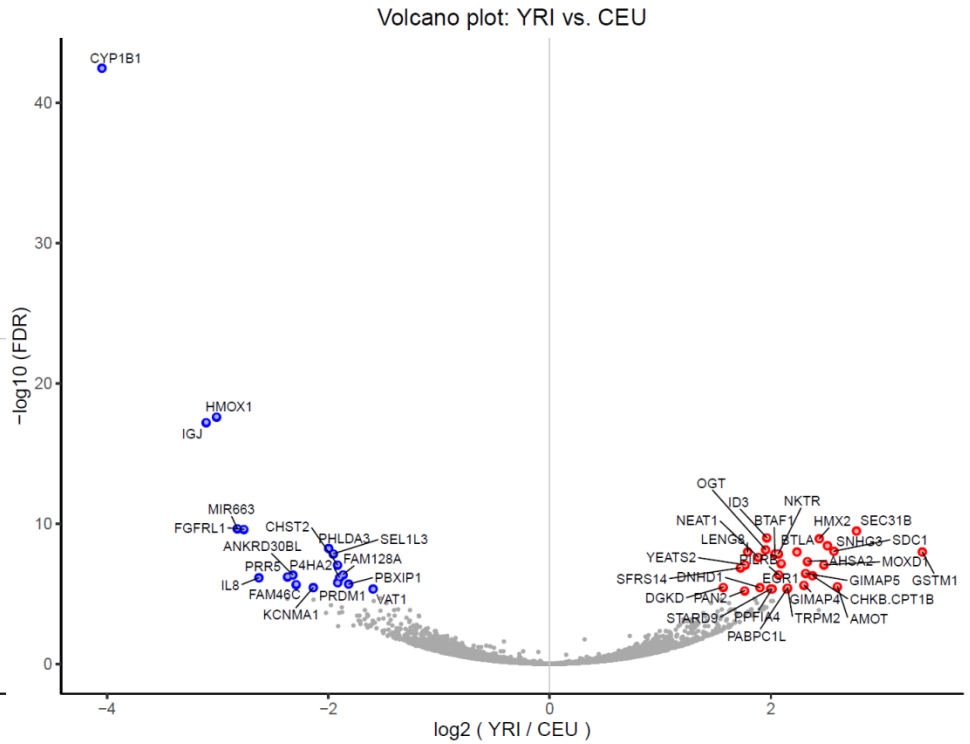
- You have to choose appropriate thresholds.

# Presenting Results

- What do you want to show?
- All-gene scatterplots can be helpful to
  - See level and fold-change ranges
  - Identify sensible thresholds
  - Hint at data or analysis problems
- Heatmaps are useful if many conditions are being compared but only for gene subsets
- Output normalized read counts with same method used for DE statistics
- Whenever one gene is especially important, look at the mapped reads in a genome browser

# MA (ratio-intensity) and volcano plots



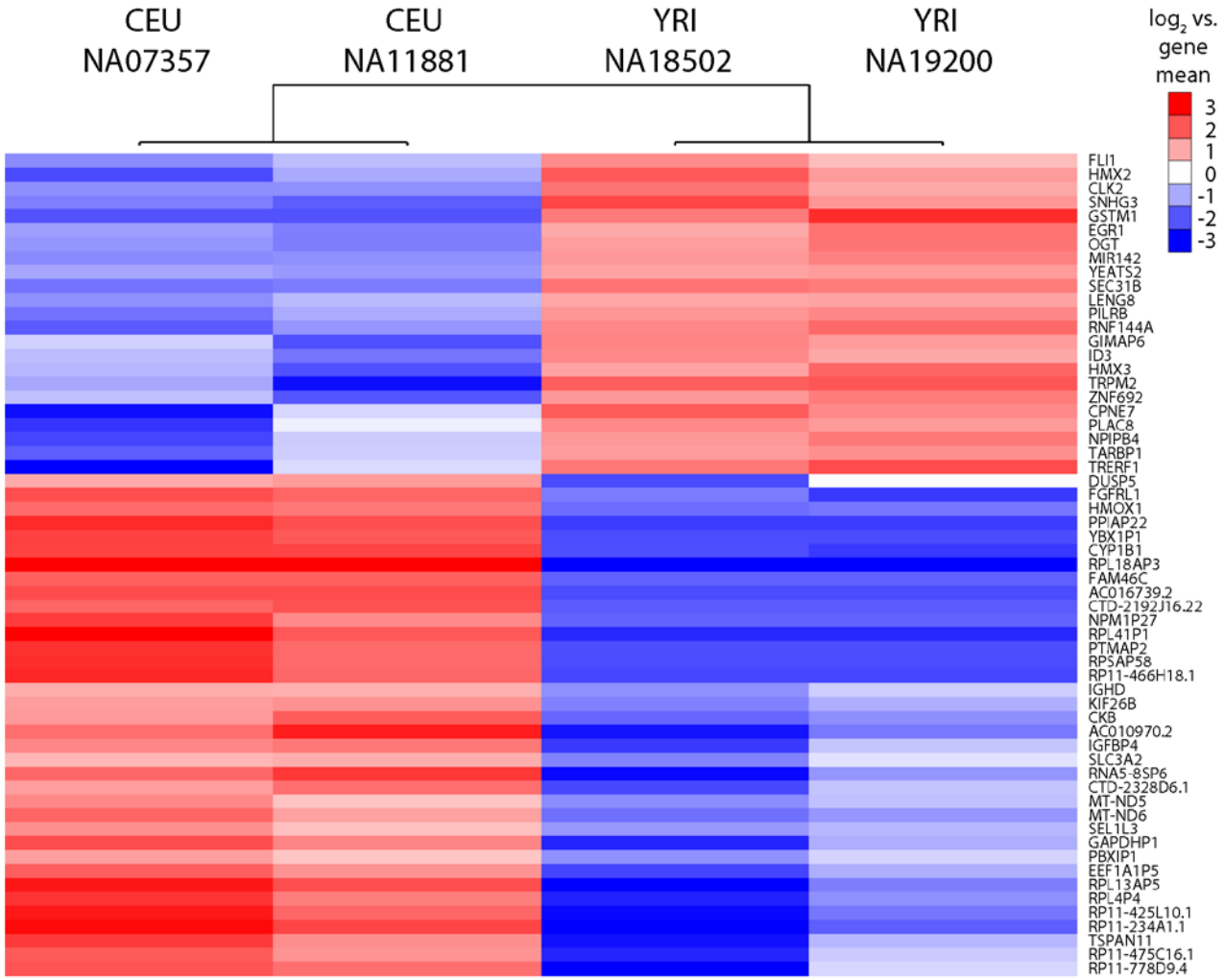




MA (ratio-intensity) plot

Volcano plot

See handout commands step 5

# Heatmap example: genes with FDR < 0.2




Excel:
- Add pseudocounts

Cluster 3.0:
- Log-transform
- Mean center
- Cluster

Java TreeView:
- Visualize
- Export

Illustrator
- Assemble

# Heatmap

- https://software.broadinstitute.org/morpheus/

# Public RNA-seq Datasets

- NCBI GEO
- EBI ArrayExpress
- Broad (e.g. CCLE, GTEx)
- Recount2
- ENCODE
- TCGA
- /nfs/BaRC_datasets

# Summary

- Experimental design
- Quality control (fastqc)
- Sequence preparation
- Mapping spliced reads (STAR)
- Counting gene levels (featureCounts)
- Normalization and identifying "differentially expressed" genes (DESeq2 R package)
- Creating figures and summaries
- Save your commands!

# Resources

- Previous Hot Topics (http://jura.wi.mit.edu/bio/education/hot_topics/)
- An introduction to R and Bioconductor: A BaRC Short Course
- BaRC SOPs (http://barcwiki.wi.mit.edu/wiki/SOPs)
- Online software manuals
  - STAR, featureCounts, DESeq2, etc.
- Various datasets: **`/nfs/BaRC_datasets`**
- Genome index and GTF files are in **`/nfs/genomes`**

# Upcoming Hot Topics

- Single cell RNA-seq: February 27th
- Excel tips and tricks: March 12th
- Dimensionality reduction: March