Statistical Inference for RNA-seq - Part I

Keegan Korthauer

9 February 2022

with slide contributions from Paul Pavlidis

Learning objectives (next two lectures)

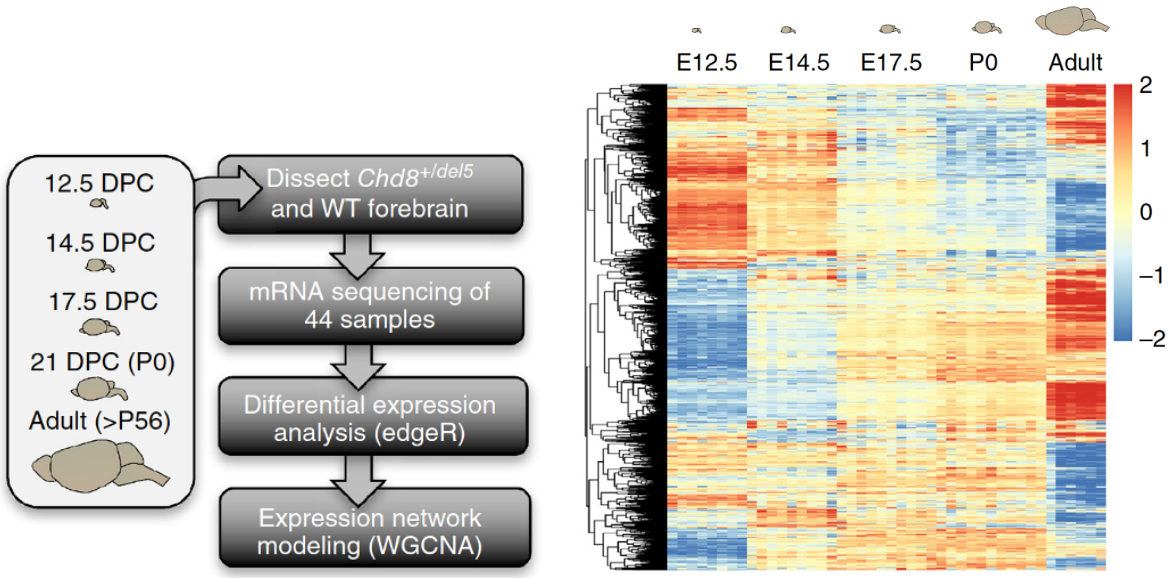
- Understand *why* and *when* between and within sample normalization are needed
- Apply common between and within sample normalization approaches to RNA-seq counts
- Understand why the *count nature* of RNA-seq data requires modification to the Differential Expression approaches applied to microarray data (e.g. `limma`)
- Apply models such as **limma-trend, limma-voom, `DESeq2` and `edgeR`** for inference of Differential Expression

Additional resources

- Companion notes for this lecture with greater detail can be found here
- For all of the specific methods we discuss, refer to the Bioconductor pages (vignettes, reference manuals) for the most current and thorough details on implementation

Recall the CHD8 RNA-seq experiment

- Gompers et al. (Nature Neuroscience 2017) analyzed 26 Chd8+/del5 and 18 WT littermates
  - Tested for differential expression across 11,936 genes accounting for sex, developmental stage and sequencing batch
- We'll use this dataset throughout this lecture to illustrate RNA-seq analysis



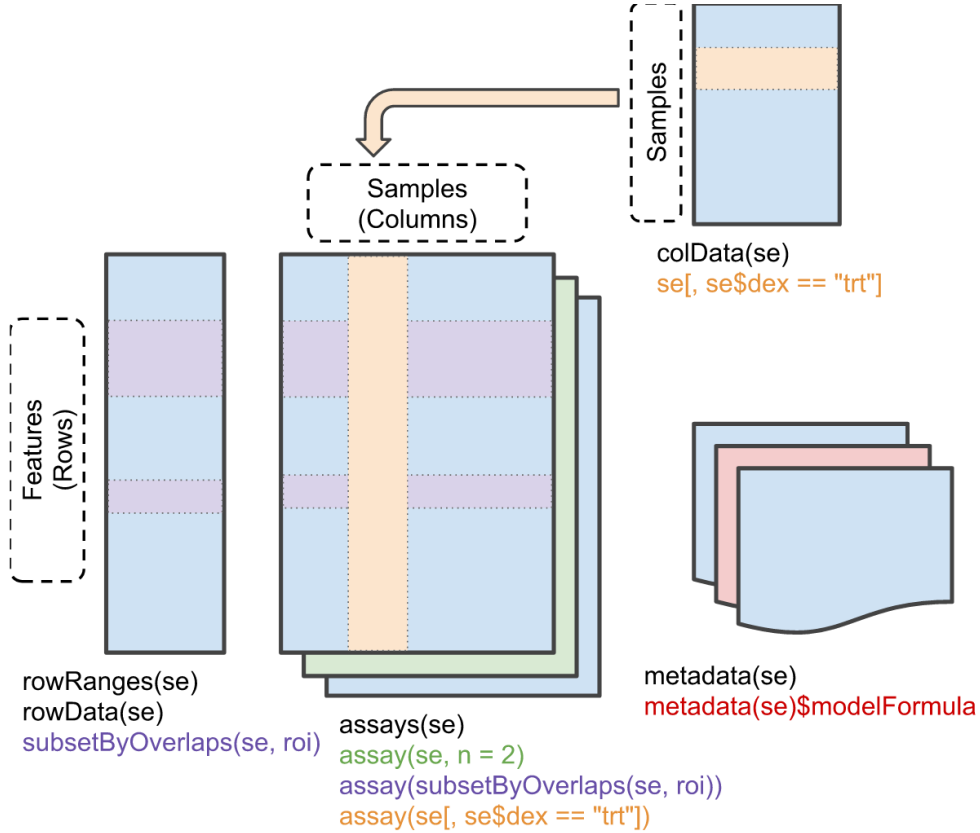
Figures from Gompers et al. (2017) paper

SummarizedExperiment object

Recall SummarizedExperiment: A special object format that is designed to contain data & metadata

```
sumexp
```

```
## class: SummarizedExperiment
## dim: 20962 44
## metadata(0):
## assays(1): counts
## rownames(20962): 0610005C13Rik 0610007P14Rik
... Zzef1 Zzz3
## rowData names(0):
## colnames(44): Sample_ANAN001A
Sample_ANAN001B ... Chd8.adult.S29
##   Chd8.adult.S31
## colData names(7): DPC Sex ... FeatureCounts
Sample
```



Anatomy of a SummarizedExperiment object

A look inside our `SummarizedExperiment` object

Peek at the first few rows/columns of the `counts` slot of our `SummarizedExperiment`:

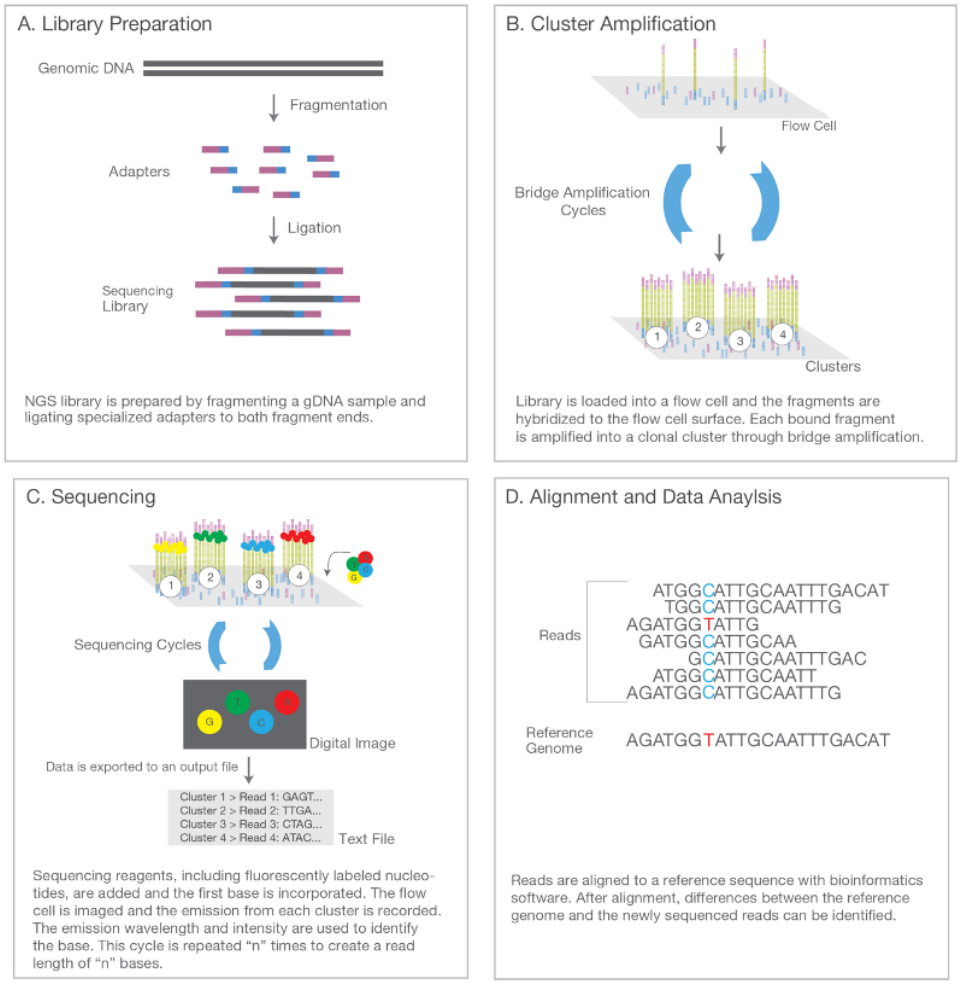
```
assays(sumexp)$counts[1:10, 1:4]
```

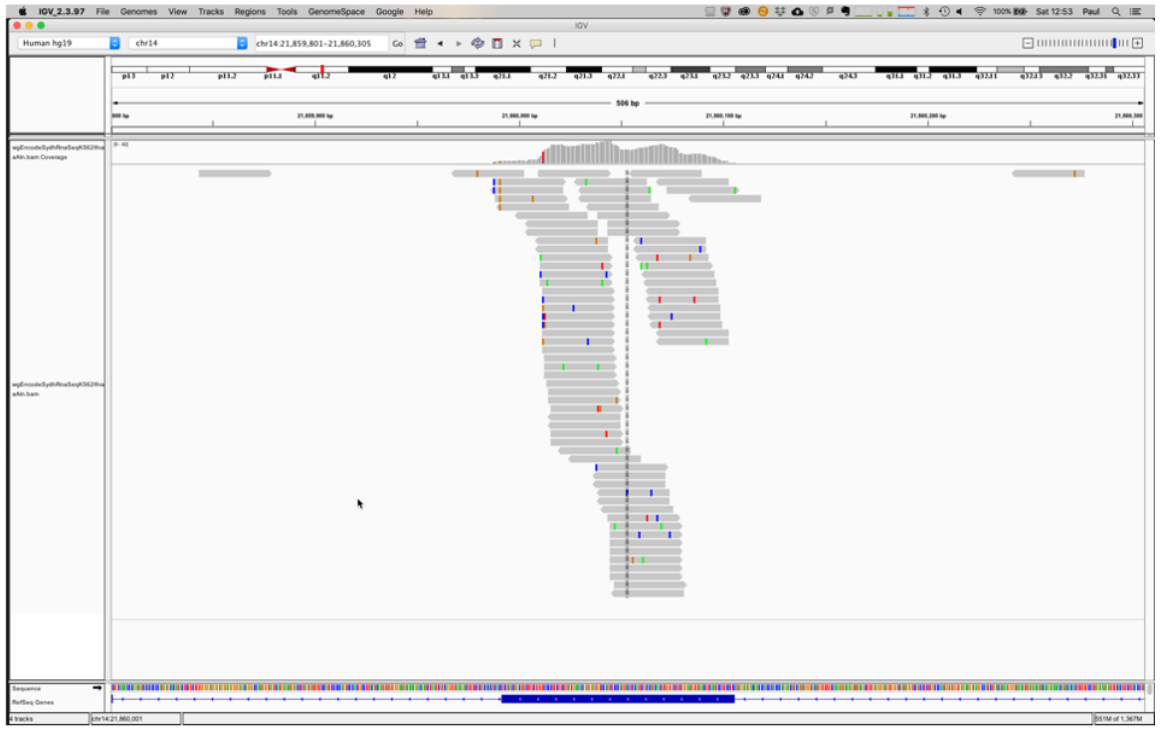
```
##                Sample_ANAN001A Sample_ANAN001B Sample_ANAN001C Sample_ANAN001D
## 0610005C13Rik               20              24              20               8
## 0610007P14Rik             1714            1796            1970            1996
## 0610009B22Rik              578             866             790             858
## 0610009L18Rik               50              82              38              64
## 0610009O20Rik             2580            2964            2942            3084
## 0610010B08Rik                0              10               0               2
## 0610010F05Rik             4516            6374            4860            5868
## 0610010K14Rik             2176            2358            2128            2746
## 0610011F06Rik              762             890             976            1030
## 0610012G03Rik              922            1176            1138            1358
```

Now we have count data

- In the exploratory data analysis lecture, we worked with transformed values (specifically "logRPKM" - more on this later) - these were continuous
- Now we will work with the raw RNA-seq counts (discrete)
- These counts represent the number of reads mapping to each feature (gene or transcript) - here we have gene counts
- Seminar 6 explores how to obtain read counts from alignment (BAM or SAM) files

Recall where these counts came from





Millions of short (~100bp) reads, each assigned to a gene

Review what we learned from EDA

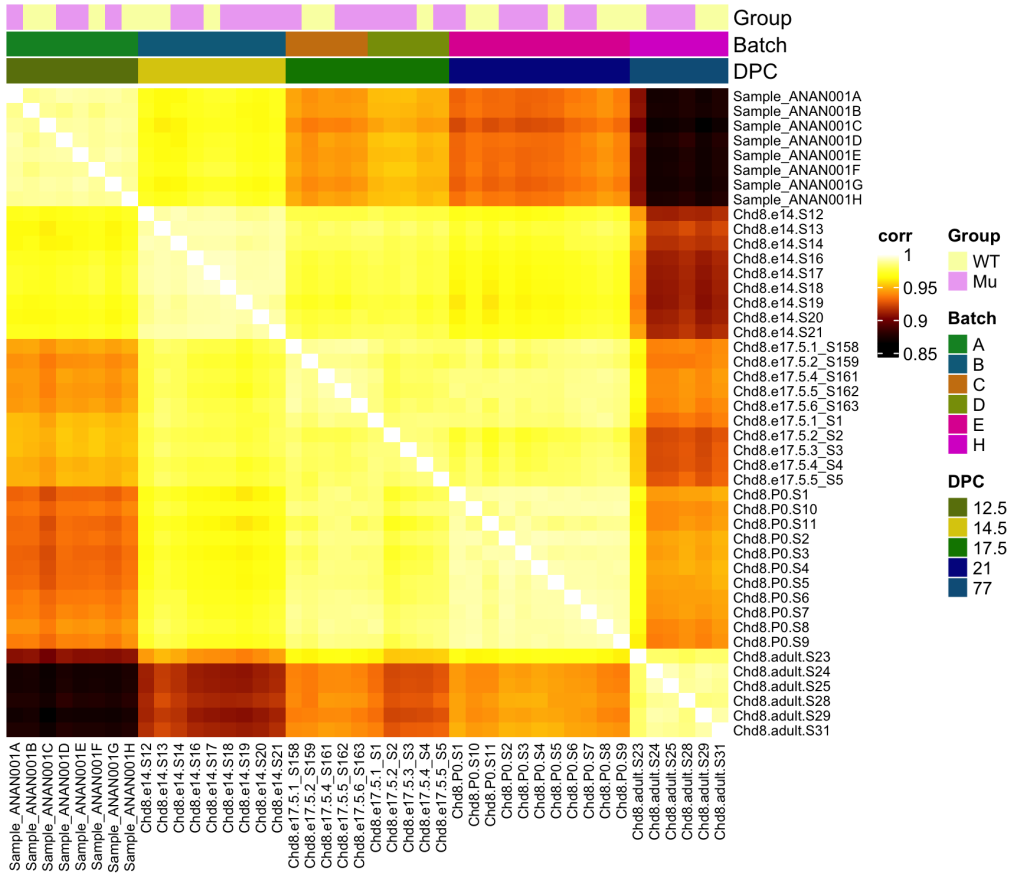
Batch (sequencing run) and DPC (days post conception) are confounded

```
table(colData(sumexp)$SeqRun, colData(sumexp)$DPC)
```

```
##
##     12.5 14.5 17.5 21 77
##   A    8    0    0  0  0
##   B    0    9    0  0  0
##   C    0    0    5  0  0
##   D    0    0    5  0  0
##   E    0    0    0 11  0
##   H    0    0    0  0  6
```

Review what we learned from EDA

- Batch (sequencing run) and DPC (days post conception) are also major sources of variation
- One sample was a potential minor outlier
- Note that only RPKM values provided in GEO; raw counts obtained directly from authors (see companion notes)
- Also found that sex was mislabeled for some samples



Differential expression analysis on Chd8 data

- Main variable of interest: Group (Genotype: Chd8 mutant vs WT)
- We'd like to fit a model for each gene so we can test for Group effect, and adjust for:
  - Sex (M vs F, 2 level factor)
  - DPC (days post conception, 5 level factor)

Differential expression analysis on Chd8 data

- Main variable of interest: Group (Genotype: Chd8 mutant vs WT)
- We'd like to fit a model for each gene so we can test for Group effect, and adjust for:
  - Sex (M vs F, 2 level factor)
  - DPC (days post conception, 5 level factor)
- Using what we learned in previous lectures, we can formulate this model as

$$Y_i = \theta + \tau_{Mut}x_{i,Mut} + \tau_F x_{i,F} + \tau_{D14.5}x_{i,D14.5} + \tau_{D17.5}x_{i,D17.5} + \tau_{D21}x_{i,D21} + \tau_{D77}x_{i,D77} + \epsilon_i$$

Differential expression analysis on Chd8 data

- Main variable of interest: Group (Genotype: Chd8 mutant vs WT)
- We'd like to fit a model for each gene so we can test for Group effect, and adjust for:
  - Sex (M vs F, 2 level factor)
  - DPC (days post conception, 5 level factor)
- Using what we learned in previous lectures, we can formulate this model as

$$Y_i = \theta + \tau_{Mut}x_{i,Mut} + \tau_F x_{i,F} + \tau_{D14.5}x_{i,D14.5} + \tau_{D17.5}x_{i,D17.5} + \tau_{D21}x_{i,D21} + \tau_{D77}x_{i,D77} + \epsilon_i$$

$$x_{i,Mut} = \begin{cases} 1 \text{ if sample } i \text{ is Mutant} \\ 0 \text{ otherwise} \end{cases}, \quad x_{i,F} = \begin{cases} 1 \text{ if sample } i \text{ is Female} \\ 0 \text{ otherwise} \end{cases}, \quad x_{i,D\#} = \begin{cases} 1 \text{ if sample } i \text{ is DPC\#} \\ 0 \text{ otherwise} \end{cases}$$

where $D\# \in \{D14.5, D17.5, D21, D77\}$

Differential expression analysis on Chd8 data

- Our model has no interaction term (though we could add one if we wish)
- $p = 7$ parameters to estimate in our model: $\theta, \tau_{Mut}, \tau_F, \tau_{D14.5}, \tau_{D17.5}, \tau_{D21}, \text{ and } \tau_{D77}$
- $n = 44$ samples total, so our model has $n - p = 44 - 7 = 37$ degrees of freedom
- How can we test whether there is differential expression between Chd8 Mut and WT using our model?

Differential expression analysis on Chd8 data

- Our model has no interaction term (though we could add one if we wish)
- $p = 7$ parameters to estimate in our model: $\theta, \tau_{Mut}, \tau_F, \tau_{D14.5}, \tau_{D17.5}, \tau_{D21},$ and τ_{D77}
- $n = 44$ samples total, so our model has $n - p = 44 - 7 = 37$ degrees of freedom
- How can we test whether there is differential expression between Chd8 Mut and WT using our model?
- Recall that since this is an additive model, the parameters represent main effects

Design matrix in R

```
modm <- model.matrix(~ Sex + Group + DPC, data = colData(sumexp))
head(modm, 10)
```

```
##                 (Intercept) SexF GroupMu DPC14.5 DPC17.5 DPC21 DPC77
## Sample_ANAN001A           1    1       1       0       0     0     0
## Sample_ANAN001B           1    0       0       0       0     0     0
## Sample_ANAN001C           1    0       0       0       0     0     0
## Sample_ANAN001D           1    1       1       0       0     0     0
## Sample_ANAN001E           1    1       1       0       0     0     0
## Sample_ANAN001F           1    1       0       0       0     0     0
## Sample_ANAN001G           1    1       1       0       0     0     0
## Sample_ANAN001H           1    1       0       0       0     0     0
## Chd8.e14.S12              1    0       0       1       0     0     0
## Chd8.e14.S13              1    1       0       1       0     0     0
```

Are we ready to fit the model?

Might start with the `limma` approach on the raw counts, but...

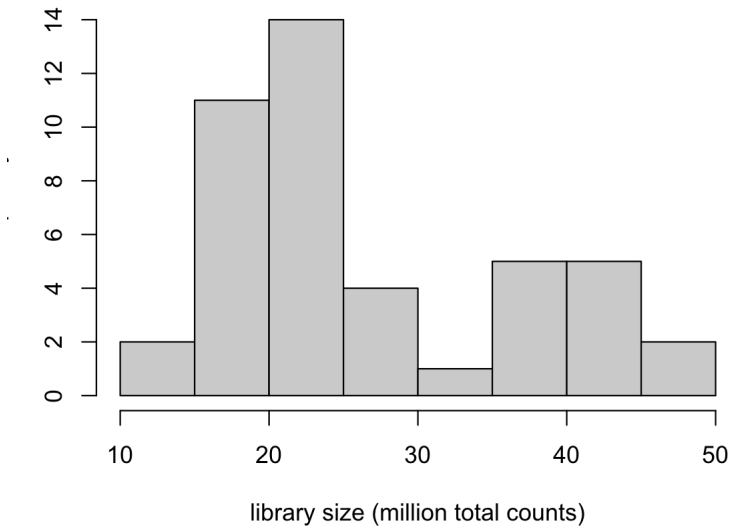
Are we ready to fit the model?

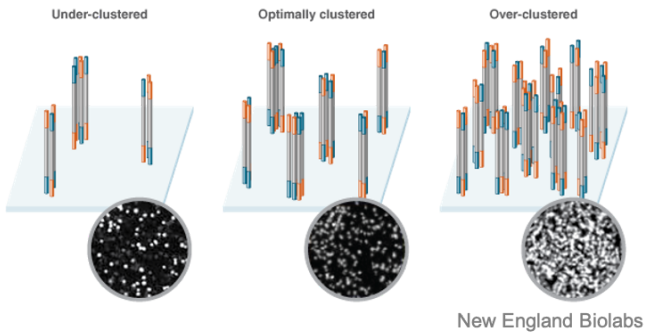
Might start with the `limma` approach on the raw counts, but...

Not so fast - we have to consider additional sources of variation!

Library size (sequencing depth)

- Library size: Total number of read counts per sample
- Ideally this would be the same for all samples, but it isn't
- Number of reads per sample depends on factors like how many samples were multiplexed and how evenly, cluster density, RNA quality, etc.



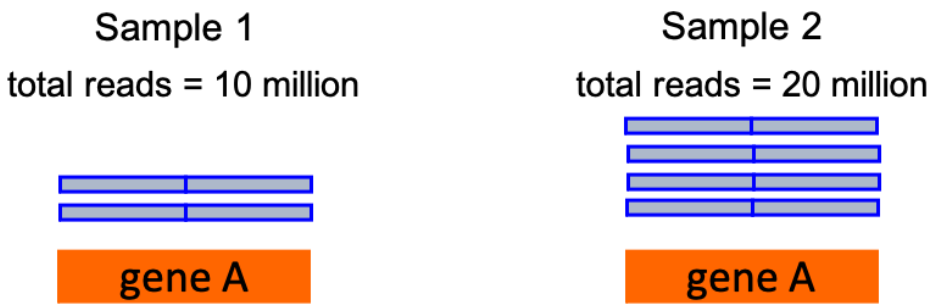


Why does library size matter?

- We want to compare gene counts between samples
- Intuition: if we sequence one group of samples twice as much, gene counts in that sample look roughly twice as large even if there's no DE!

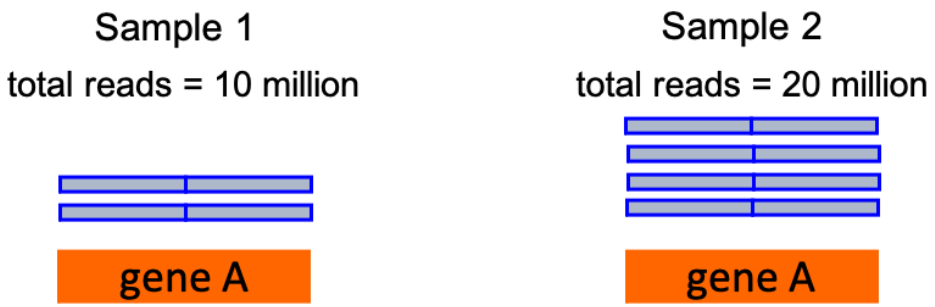
Why does library size matter?

- We want to compare gene counts between samples
- Intuition: if we sequence one group of samples twice as much, gene counts in that sample look roughly twice as large even if there's no DE!
- Between samples - higher sequencing depths leads to higher gene/transcript read counts



Why does library size matter?

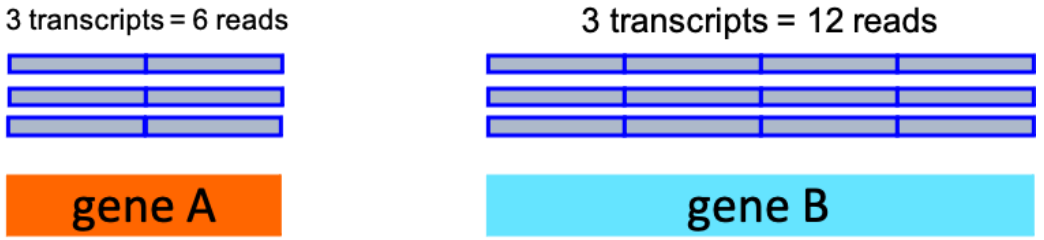
- We want to compare gene counts between samples
- Intuition: if we sequence one group of samples twice as much, gene counts in that sample look roughly twice as large even if there's no DE!
- Between samples - higher sequencing depths leads to higher gene/transcript read counts



- You may come across (older) papers in the literature where data was down-sampled to make library sizes the same (not recommended)

Within-sample comparisons

- Other factors of variation come into play if we also want to compare counts between genes within sample (less common)
- At the same expression level, longer genes/transcripts have more read counts



How can we make fair between- and within-sample comparisons?

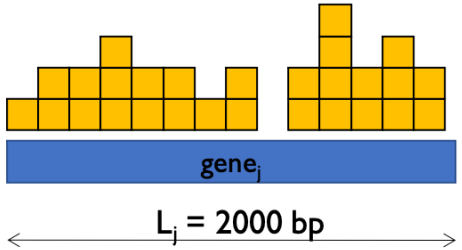
- Normalized expression units: expression values adjusted for factors like library size, gene length
  - e.g.RPKM/FPKM, TPM, CPM
  - useful for visualization / clustering
- Normalization factors: scalar values representing relative library size of each sample
  - e.g. TMM, DESeq size factors
  - useful to include in models of raw counts to adjust for library size

How can we make fair between- and within-sample comparisons?

- Normalized expression units: expression values adjusted for factors like library size, gene length
  - e.g.RPKM/FPKM, TPM, CPM
  - useful for visualization / clustering
- Normalization factors: scalar values representing relative library size of each sample
  - e.g. TMM, DESeq size factors
  - useful to include in models of raw counts to adjust for library size
- For analysis (e.g. DE) it is ideal to start with raw counts
  - raw counts required for many methods
  - can always compute normalized values from raw counts (but not vice versa)

Normalized expression units

- RPKM/FPKM: reads/fragments per kb of exon per million mapped reads



$R_{ij} = 28$ reads in gene j, sample i

$\Sigma_j R_{ij} = 11$ million reads in sample i

$$RPKM_{ij} = \frac{R_{ij}}{\frac{L_j}{10^3}\frac{\Sigma_j R_{ij}}{10^6}} = \frac{28}{\frac{2000}{10^3}\frac{1.1\times10^7}{10^6}} = 1.27$$

- RPKM is the more appropriate term for paired-end data

Normalized expression units, continued

- TPM: Transcripts per million

$$TPM_{ij} = \frac{R_{ij}}{L_j} \frac{10^6}{\Sigma_j R_{ij}/L_j} = \frac{FPKM_{ij}}{\Sigma_j FPKM_{ij}/10^6}$$

Normalized expression units, continued

- TPM: Transcripts per million

$$TPM_{ij} = \frac{R_{ij}}{L_j} \frac{10^6}{\Sigma_j R_{ij}/L_j} = \frac{FPKM_{ij}}{\Sigma_j FPKM_{ij}/10^6}$$

- CPM: Counts per million

$$CPM_{ij} = \frac{R_{ij}}{\Sigma_j R_{ij}/10^6}$$

Normalized expression units, continued

- TPM: Transcripts per million

$$TPM_{ij} = \frac{R_{ij}}{L_j} \frac{10^6}{\Sigma_j R_{ij}/L_j} = \frac{FPKM_{ij}}{\Sigma_j FPKM_{ij}/10^6}$$

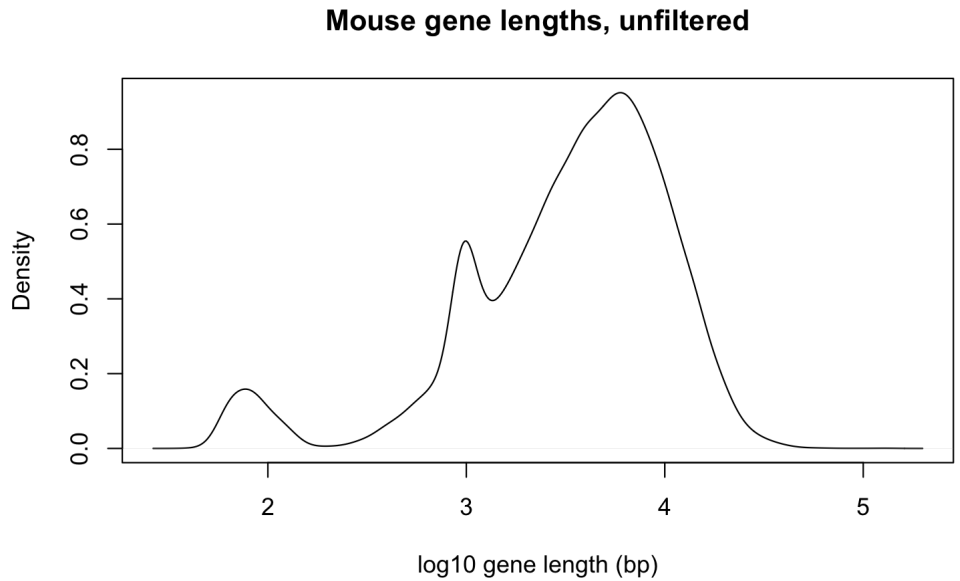
- CPM: Counts per million

$$CPM_{ij} = \frac{R_{ij}}{\Sigma_j R_{ij}/10^6}$$

- See this useful blog post on relationship between these units
- Which of these measures are between-sample normalization measures? Within-sample? Both?

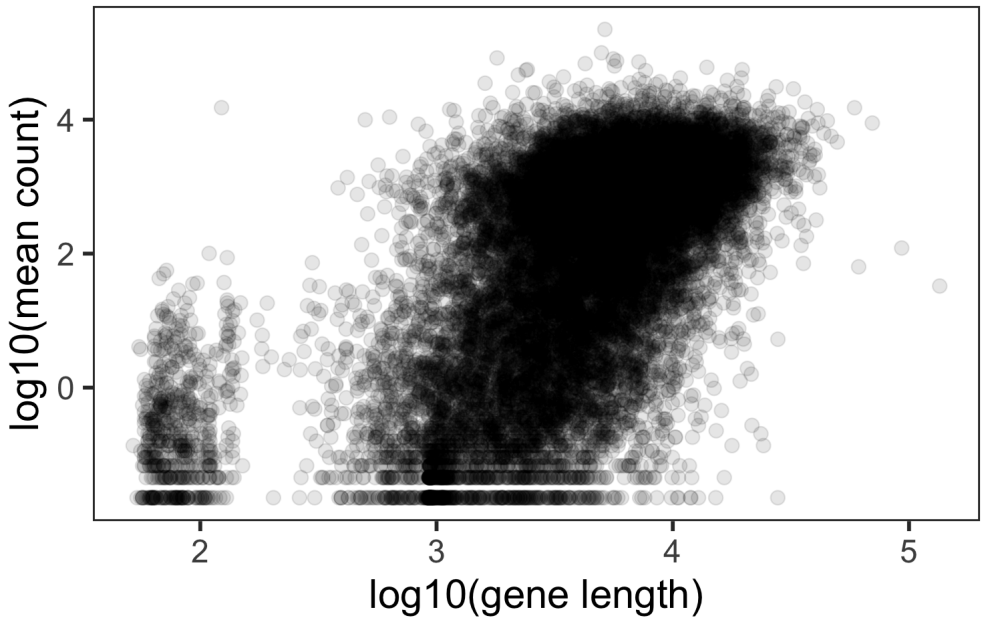
How much does "gene length" vary?

- Really we mean "total effective length of transcript used in assigning reads to genes"
- If all genes are same lengths, FPKM won't do anything interesting
- In mouse, "gene length" varies > 3 orders of magnitude, but mostly ~2.5Kb - 4.3Kb
- Your organism may vary



How does gene length relate to counts?

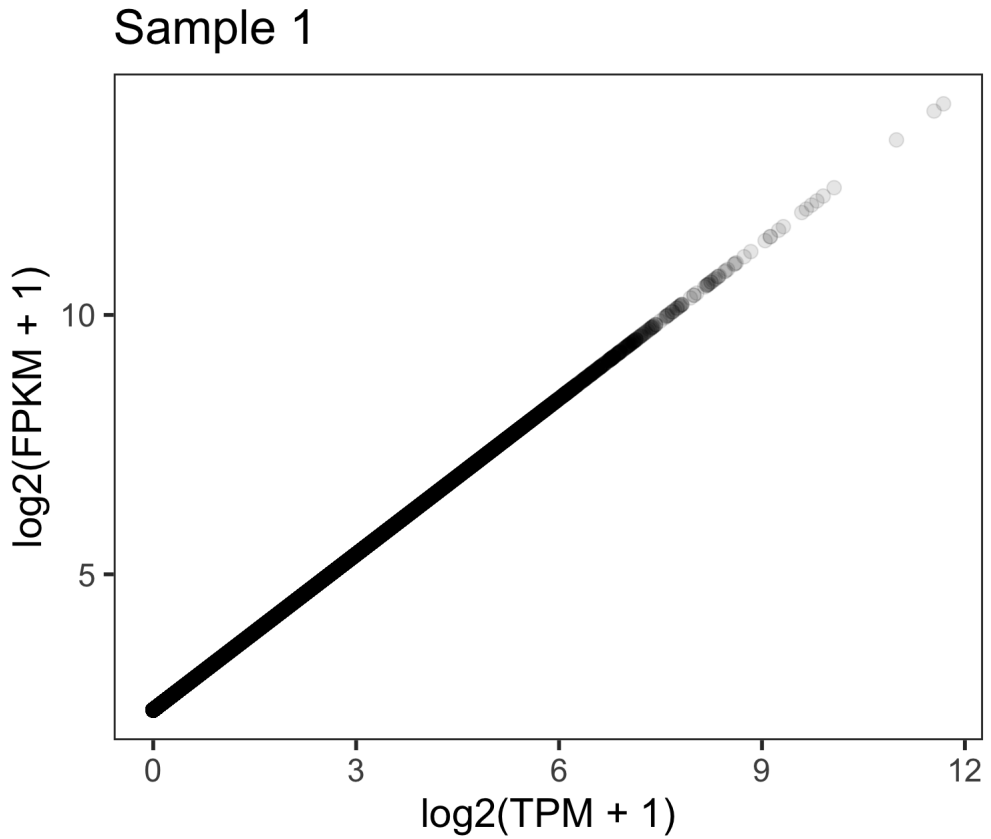
- If all genes were actually expressed at same level (RNA molecules per cell), we'd expect a perfect relation
- Of course genes are not all expressed at the same level, so we expect the effect of length to be less obvious
- Rank correlation between length and mean expression in our example data is ~0.59



FPKM vs TPM

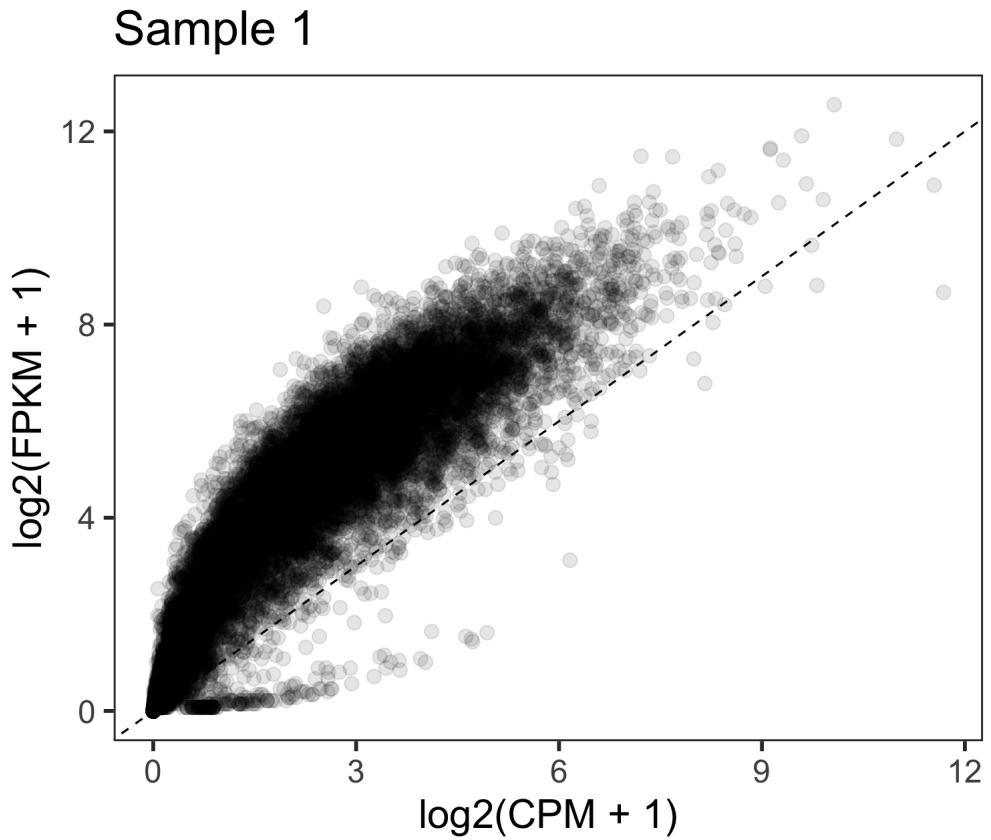
These metrics both enable comparison of expression levels of different genes within sample.

Any doubt about "gene length" will be propagated to both measures.



FPKM vs CPM

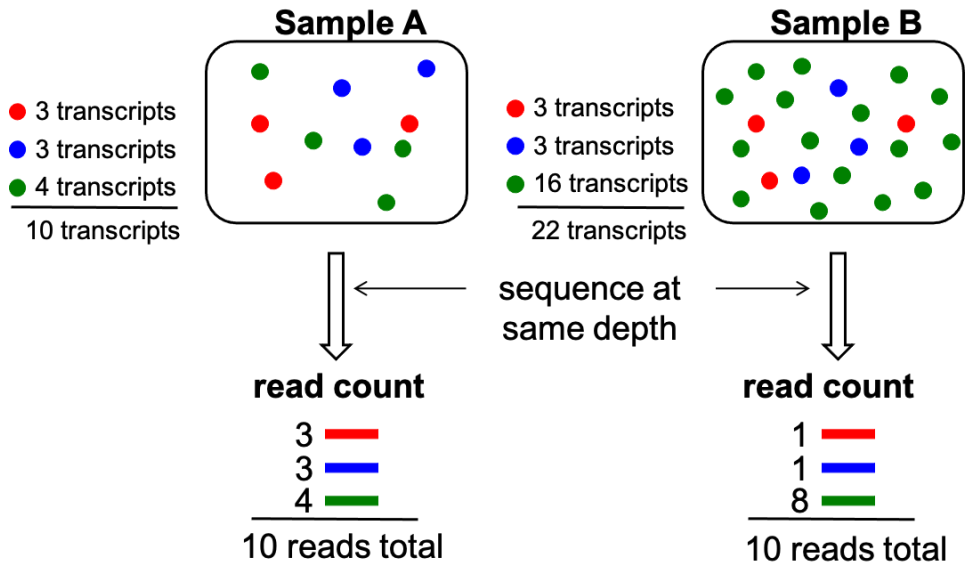
If we're comparing samples to each other, there's no important difference between FPKM/TPM and CPM so long as we assume "effective gene length" is constant across samples



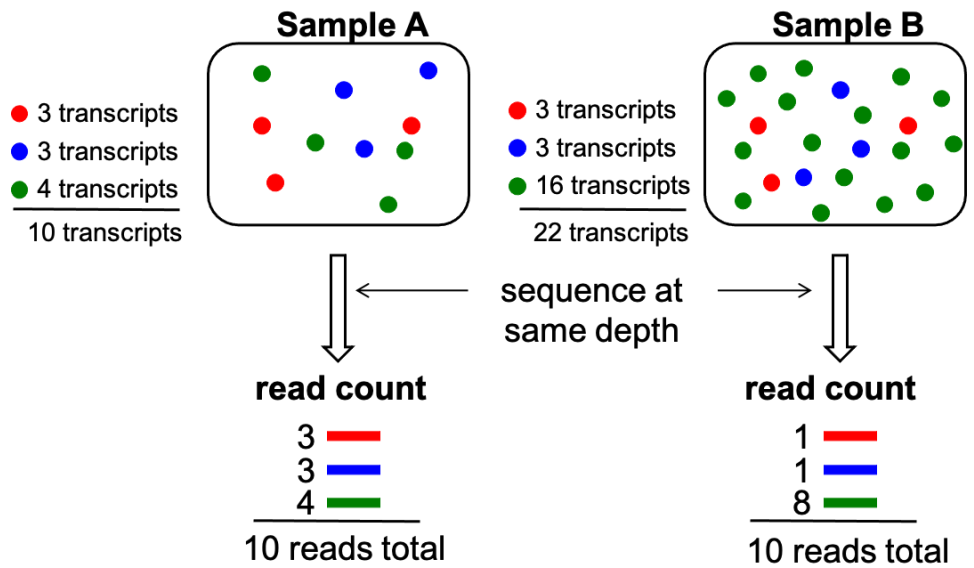
Between-sample normalization

- Computing FPKM, CPM or TPM largely corrects for differences in library size
- However, there is a complication: "Sequence space"
  - Finite number of reads implies that observing reads for one gene decreases ability to observe reads for other genes
  - This is a fundamental difference from microarrays, where each spot is essentially independent
- This isn't a major problem unless there are large differences in composition between samples, but should be inspected
  - Normalization factors are generally robust to this

Effect of sequence space



Effect of sequence space

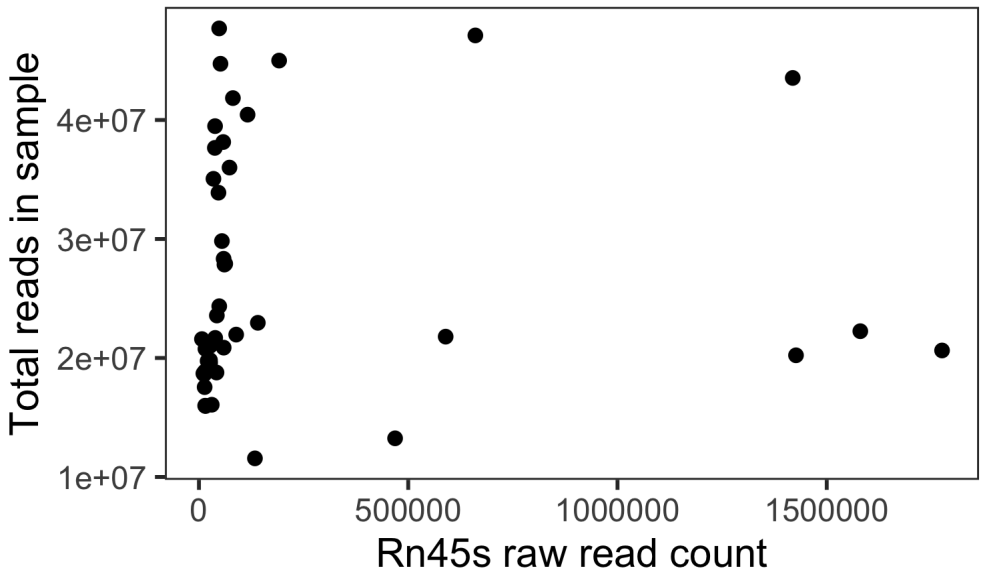


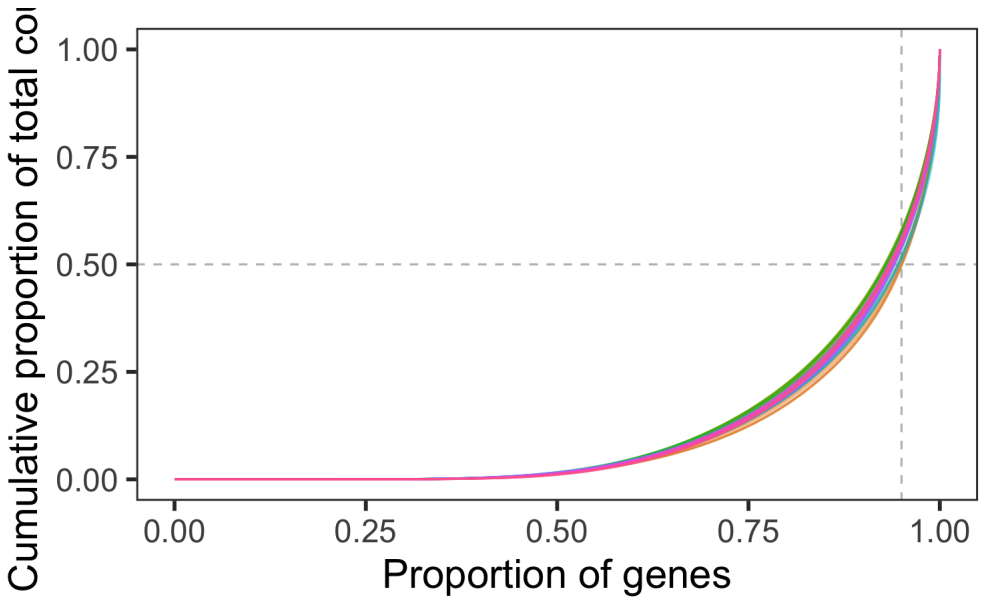
- By CPM or FPKM, red and blue gene appear be down-regulated in sample B (green gene really is diff ex)
- Adjusting expression levels in Sample B by a factor of 3 would be needed

See Robinson and Oshlack (2010)

Sequence space in our example data

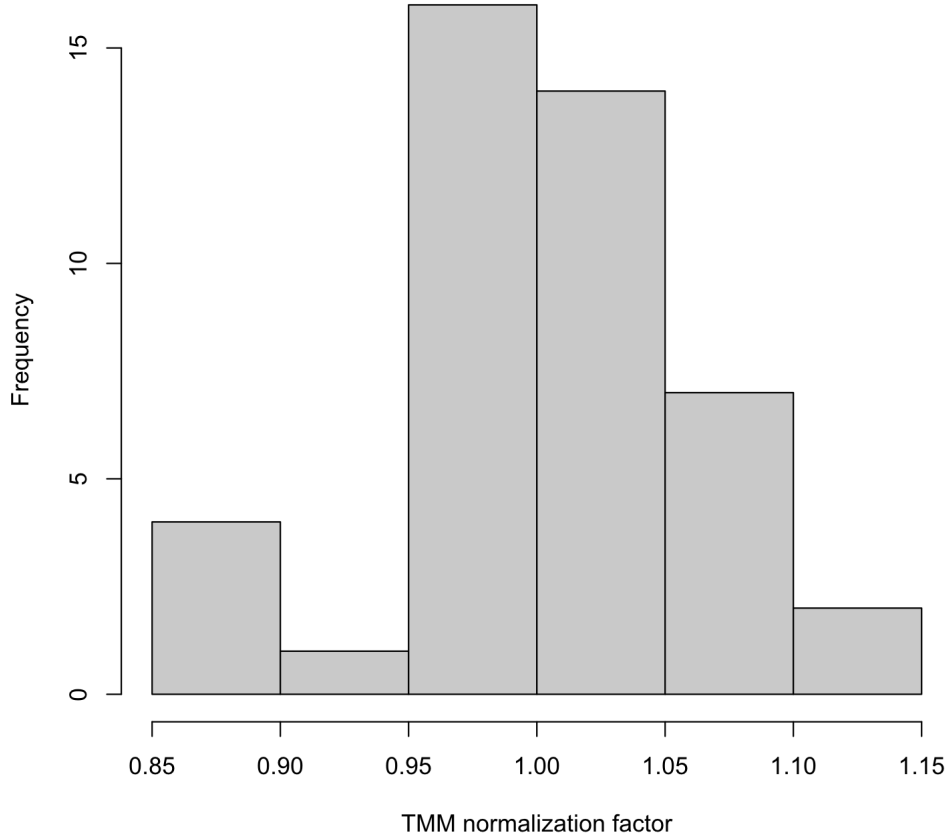
- One gene with $> 100,000$ mean reads per sample ($> 5\%$ of data in some samples): Rn45s
- Overall, 5% of the genes take up ~50% of the space in this data set, but this is reasonably consistent across the samples
- Side note: Rn45s is potentially a contaminant - a ribosomal RNA that should have been removed during sample prep, which involved poly-A selection





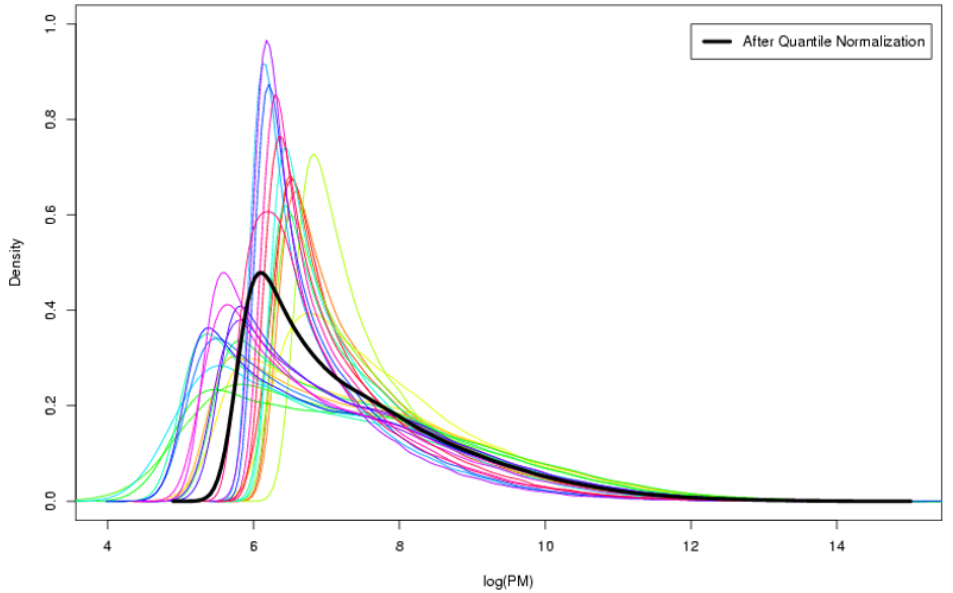
Normalization factors

- Estimate effective library size, accounting for "sequence space"
- Not used as a direct data adjustment, but included in a statistical model
- Example: TMM - "trimmed means of M-values"
    - M-values: per-gene ratios of counts among samples
    - Trimmed: extreme values are ignored
    - Values adjusted to have product = 1
    - Assumes that no more than half of genes are DE
    - Calculate with `edgeR::calcNormFactors`



Alternative: Quantile normalization

- Essentially: rank transformation
- Effect: Force all samples to have the same distribution of expression values
- Not typically applied in RNA-seq analyses unless there still exist large differences in distributions among samples after standard normalization procedures, and reason to believe they aren't biological
- Algorithm:
  - Rank transform (sample-by-sample)
  - Replace each rank with the mean (across samples) of the observed value at that rank



Bolstad et al. Bioinformatics. 2003 Jan 22;19(2):185-93

Preprocessing: filtering lowly expressed genes

- Common step which can be beneficial for a few reasons:
  - Genes with very low mean expression across samples may be uninteresting
  - Fitting models on a smaller number of genes can be faster
  - May obtain a more 'well-behaved' association between mean and variance, which might affect some methods (e.g. Voom)
- No universal threshold - depends on library size of dataset, and possibly mean-variance trend
- Original study: keep genes with > 2 samples that have CPM greater > 10

```
assays(sumexp)$cpm <- cpm(counts, log = FALSE, normalized.lib.sizes = FALSE)
keep <- which(rowSums(assays(sumexp)$cpm > 10) > 2)
length(keep)
```

```
## [1] 12021
```

```
sumexp <- sumexp[keep,]
```

Differential expression: Why we need new methods

- Goal: accurate p-values for our hypothesis tests
  - Accurate: "Uniform under the null"
  - Properties relied upon for inference from t-statistics won't hold for count data
- Perhaps most important: Heteroskedasticity and Overdispersion
  - Strong mean-variance relationship expected with count data
  - Biological variance over and above binomial sampling variance

Properties of expression data: counts

NOTE: We are focused on the distribution of expression values for a gene across technical or biological replicates - for this discussion we care less about comparing two genes within a sample

Microarray

- Signal is fundamentally counts (deep down: photon detection)
- But values are averaged across pixels and counts are high
- Never really have zero: background
- "Continuous-like"

Sequencing

- Unit of measurement is the read; no such thing as 0.2 read
- Counts of reads start at 0
- As counts get high, the distinction with microarrays should decrease

Statistics of counts: Binomial

- Number of reads observed for gene g in a given sample is a random variable
- Say RNA for gene g is present "in the cell" at about 1 out of every 1,000,000 molecules
  - Abundance $a_g = 1/1,000,000 = 1 \times 10^{-6}$ ("probability of success")
- If we randomly pick $R_i = \Sigma_g R_{ig} = 1,000,000$ molecules ("reads" = "trials"), how many gene g RNAs will we see? $E(R_{ig}|R_i) = ?$

Statistics of counts: Binomial

- Number of reads observed for gene g in a given sample is a random variable
- Say RNA for gene g is present "in the cell" at about 1 out of every 1,000,000 molecules
  - Abundance $a_g = 1/1,000,000 = 1 \times 10^{-6}$ ("probability of success")
- If we randomly pick $R_i = \Sigma_g R_{ig} = 1,000,000$ molecules ("reads" = "trials"), how many gene g RNAs will we see? $E(R_{ig}|R_i) = ?$
- But could get 0, 2, 3, 4, ... etc just by chance: this is a Binomial distribution
  - probability distribution of the number of successes in n trials, each with probability of success p is $(Binomial(n, p))$
  - mean = np
  - In our example $R_{ig} \sim Binomial(R_i, a_g)$ where $n = R_i$ and $p = a_g$

Statistics of counts: Poisson

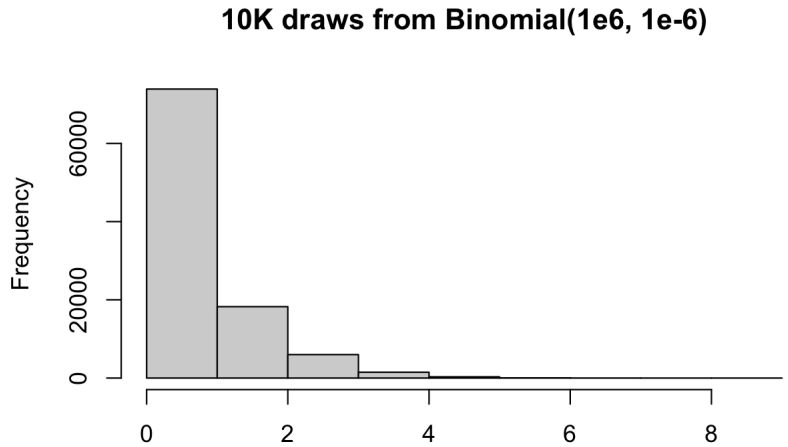
- Poisson distribution counts discrete occurrences along a continuous interval of time/space
  - parameterized by a rate parameter λ
  - key difference from Binomial: number of events can be infinitely large
- For count data, the variance is a function of the mean (*very* different from a normal)
  - Binomial: mean $= np$, variance $= np(1-p)$
  - Poisson: mean = variance = λ

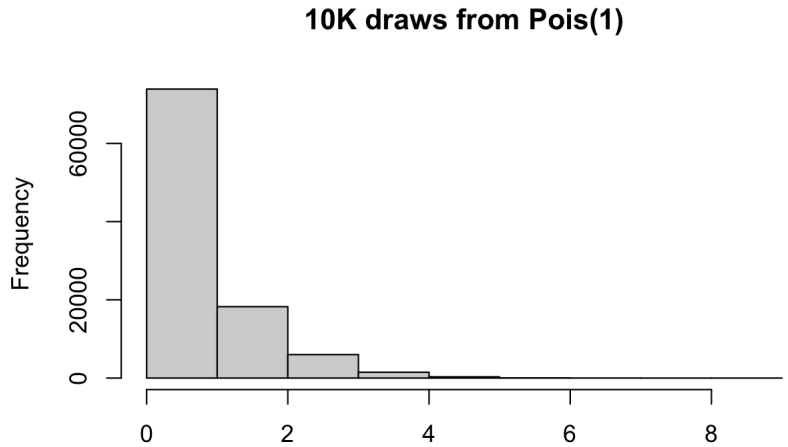
Statistics of counts: approximations

- Binomial approximation of Poisson: for large n and small np (rule of thumb: $n > 20$ & $np < 5$)
  - Approximately $R_{ig} \sim Poisson(R_i a_g)$
- Binomial approximation of Normal: For large np (rule of thumb: np & $n(1-p) > 5$)
  - Approximately $R_{ig} \sim Normal(R_i a_g,\ R_i a_g(1-a_g))$

Statistics of counts: approximations

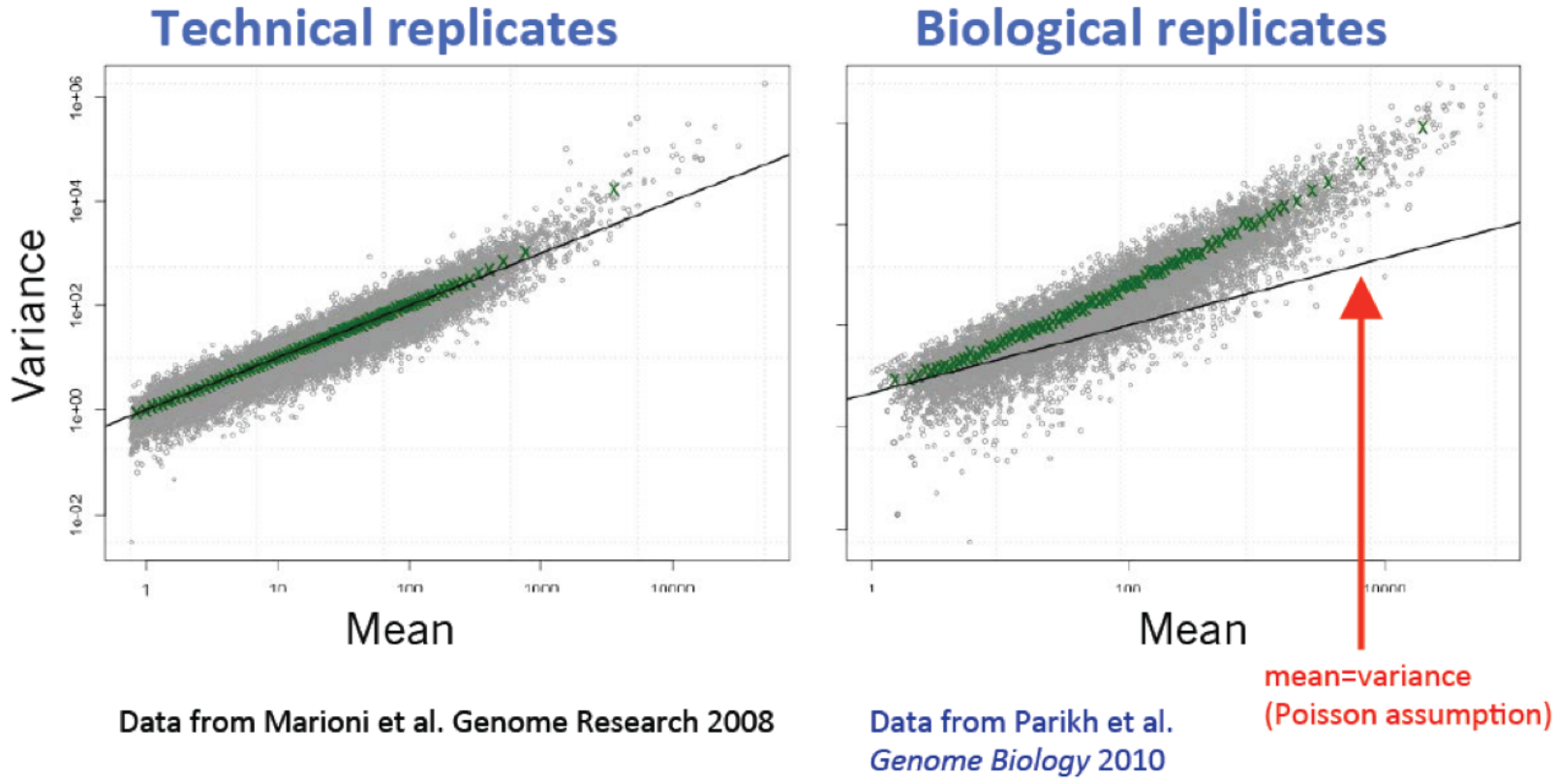
- Binomial approximation of Poisson: for large n and small np (rule of thumb: $n > 20$ & $np < 5$)
  - Approximately $R_{ig} \sim Poisson(R_i a_g)$
- Binomial approximation of Normal: For large np (rule of thumb: np & $n(1-p) > 5$)
  - Approximately $R_{ig} \sim Normal(R_i a_g,\ R_i a_g(1-a_g))$





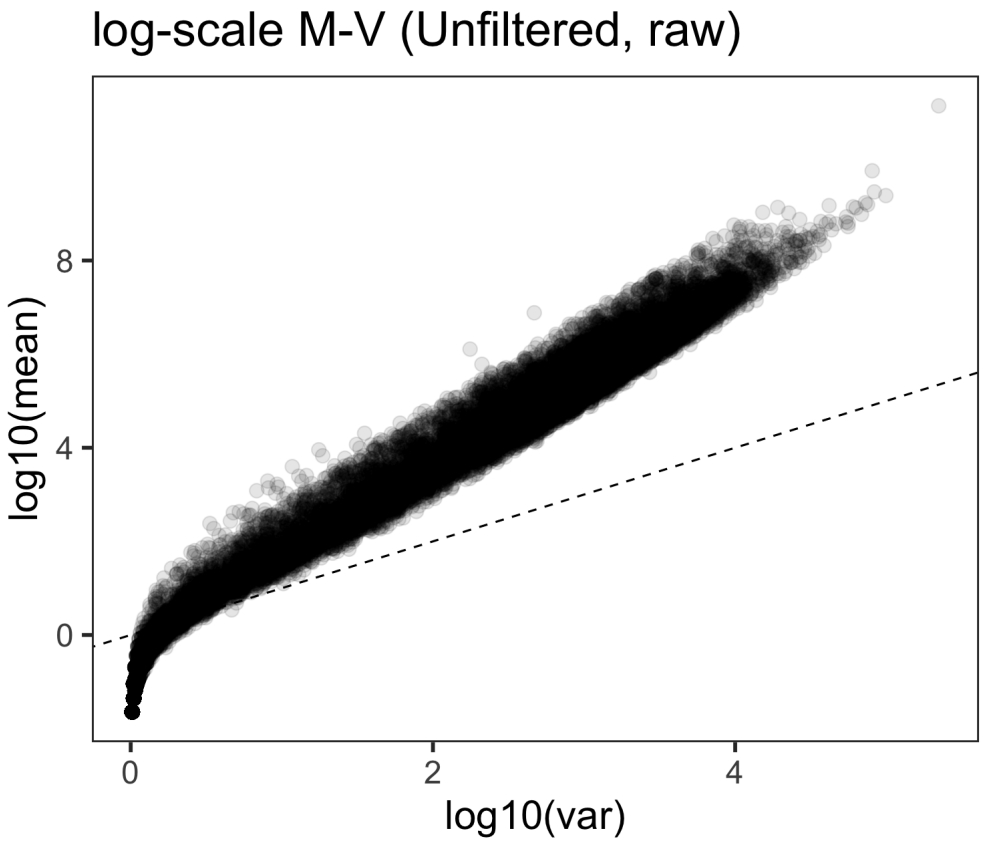
Overdispersion

Poisson OK for technical replicates, but does not capture biological variability



Impact of heteroskedasticity

- OLS: assume all errors have the same variance (within gene)
- If not true, higher variance observations get more weight in minimization of error than they should (since less precise)
  - Standard errors of parameter estimates will be poor estimates
  - Recall: $t = \frac{\hat{\beta}}{\hat{se}(\hat{\beta})}$
  - ...So p-values will also be wrong - in case of positive relationship, too small

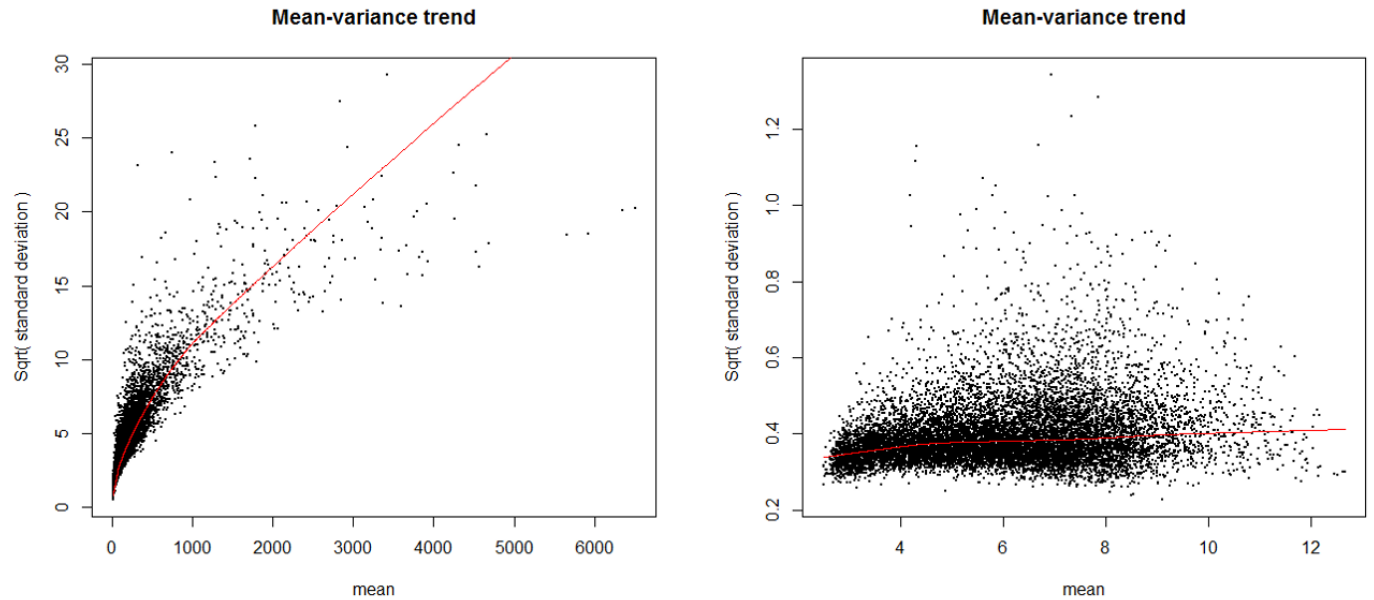


Options for DE analysis on counts

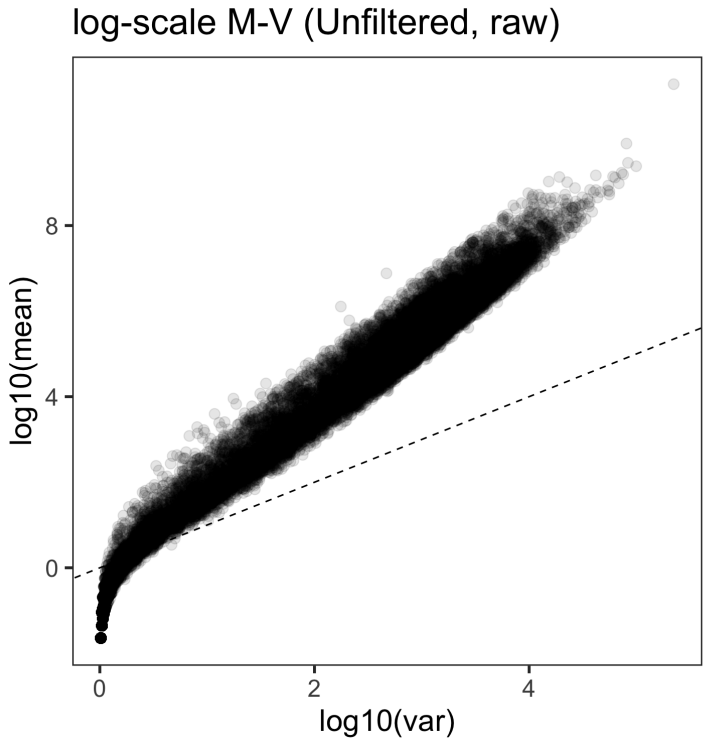
- Summary of the problem: Count data is expected to violate both normality and constant variance assumptions
- Even microarray data usually has some mean-variance relation!
- Possibilities for coping:
  - Use a non-parametric test (e.g. SAMseq - based on Wilcoxon; will not discuss further - lower power)
  - Make adjustments and model as usual
  - Use a model specific for count data

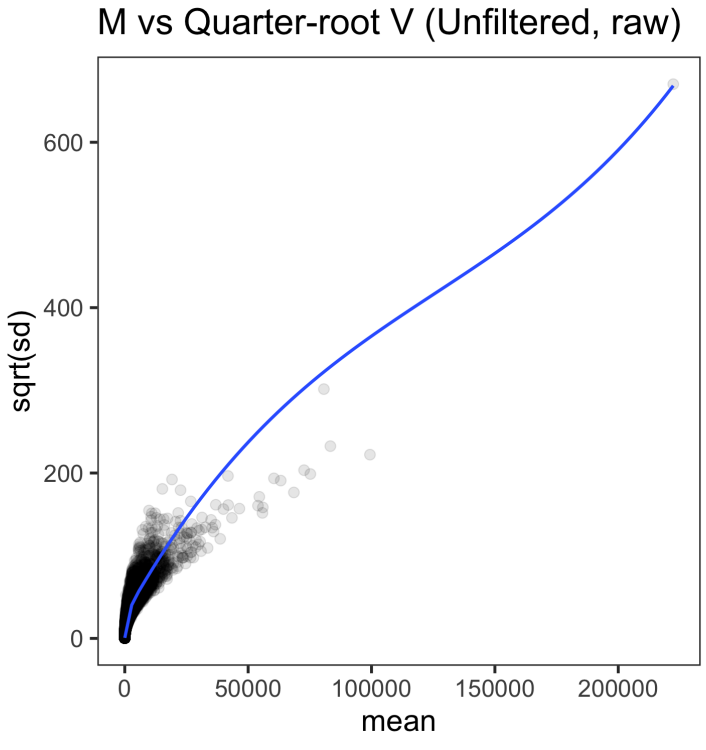
Transformation can help

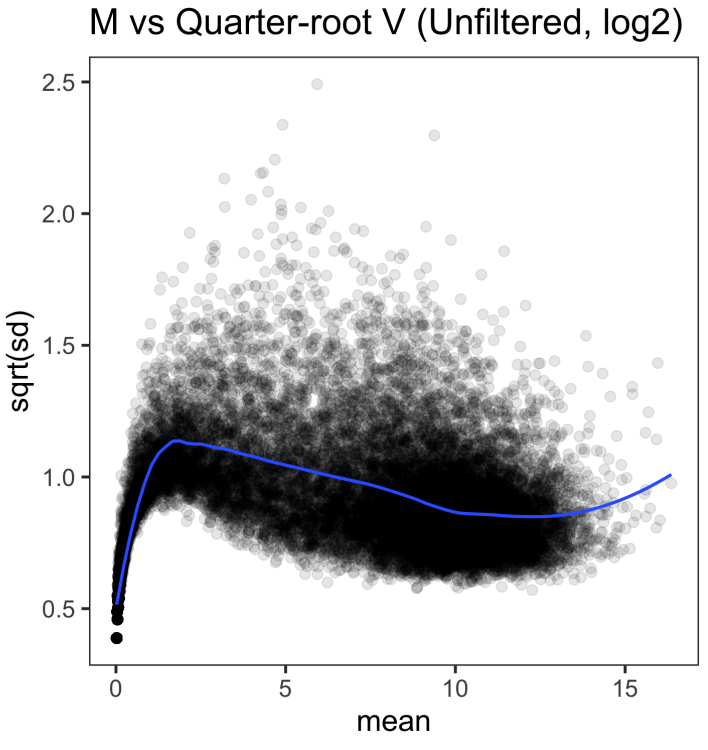
- For microarray data, taking logs is often deemed sufficient to reduce M-V trends
- We'll use plots like this which are mean vs $\sqrt{sd}$ (quarter root variance) instead of mean vs variance (you'll see why later on)
- Behaviour of the "photoreceptor" microarray data set (raw on left, log-tranformed on right):



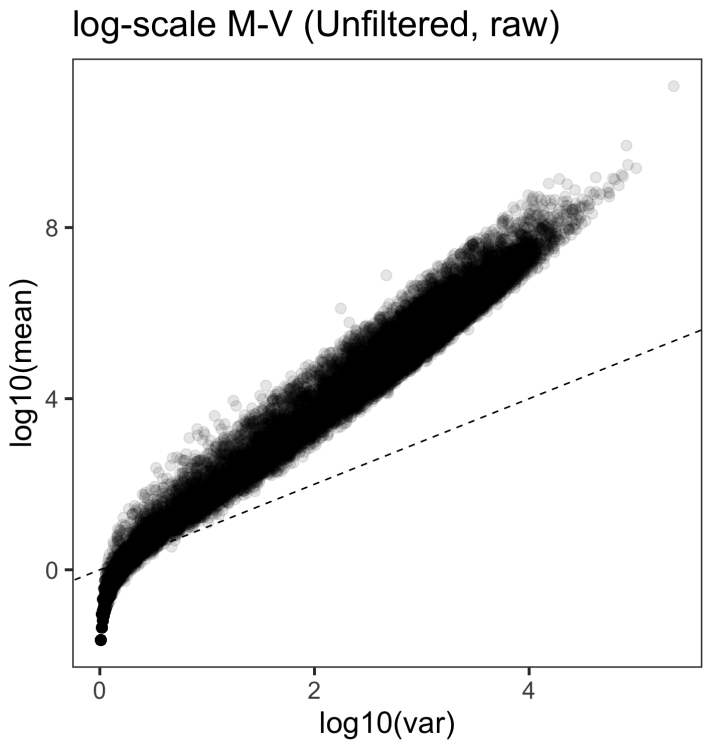
Chd8 data & effect of log transform

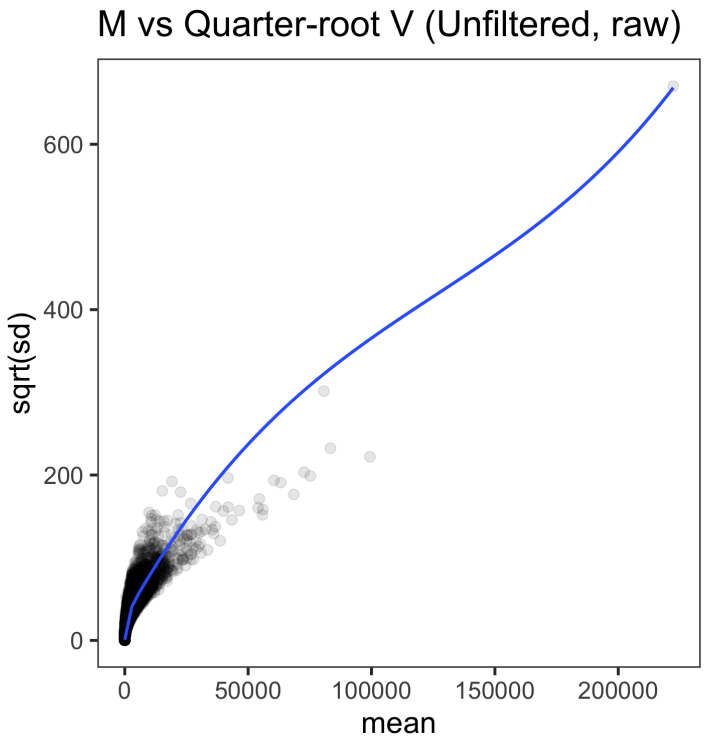


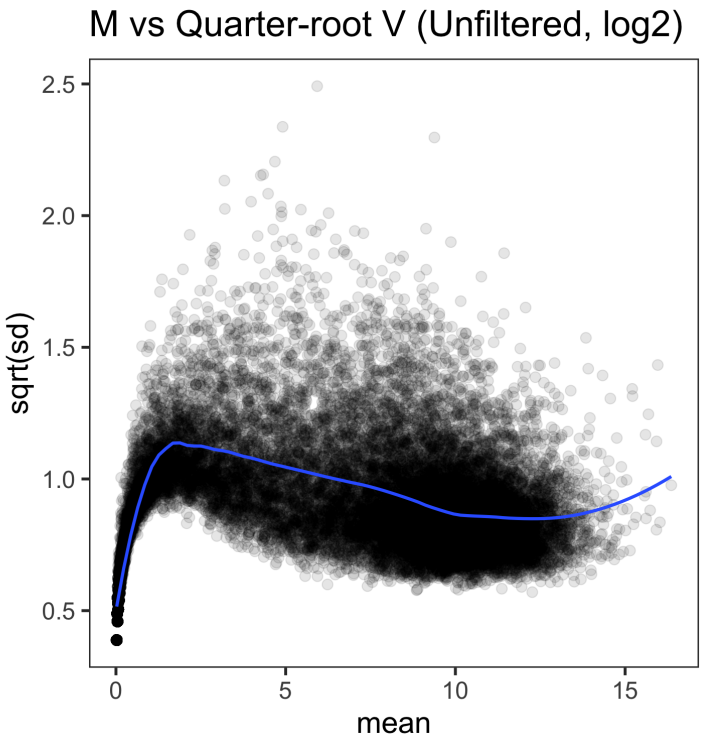




Chd8 data & effect of log transform

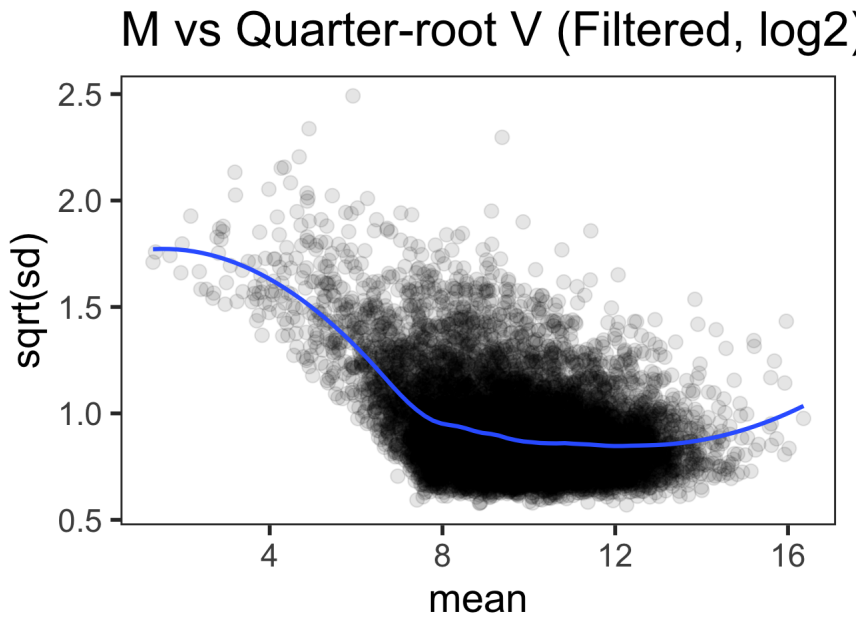




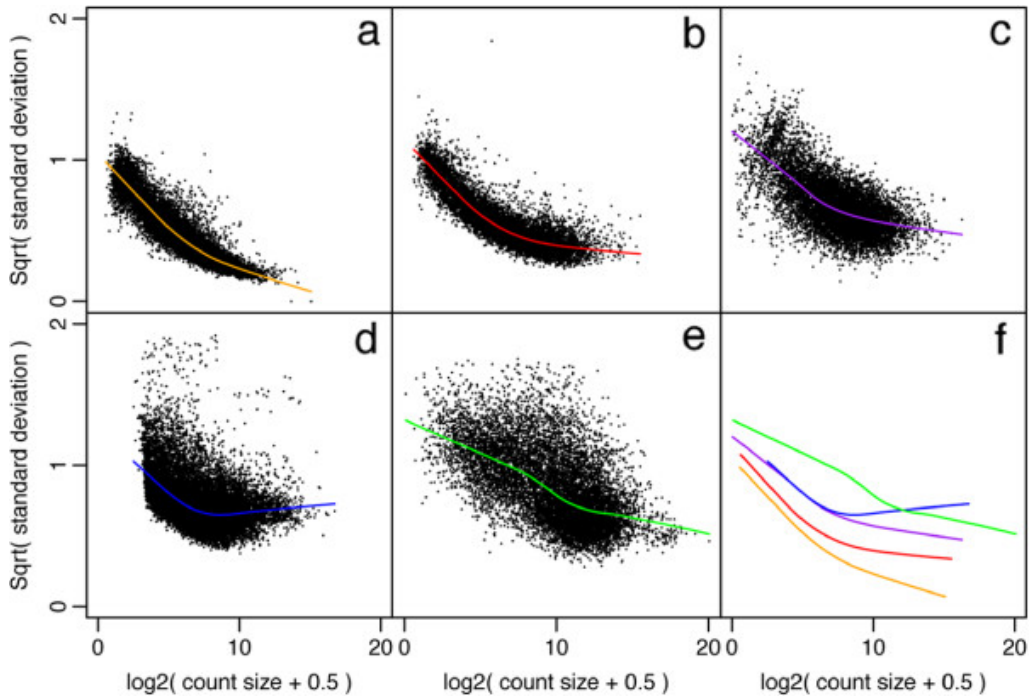


For RNA-seq data, log-transformation doesn't reliably improve the trends

Mean variance trends in various RNA-seq datasets



Chd8 dataset (Filtered to remove lowly expressed genes, log2-transformed)



Panels (a)-(e) represent datasets with increasing expected biological variability

Source: Law et al. 2014

Next time

How do we handle these M-V relationships in our analysis?