

Bulk RNA-seq analysis

Charlotte Soneson

Friedrich Miescher Institute for Biomedical Research &
SIB Swiss Institute of Bioinformatics



Swiss Institute of
Bioinformatics



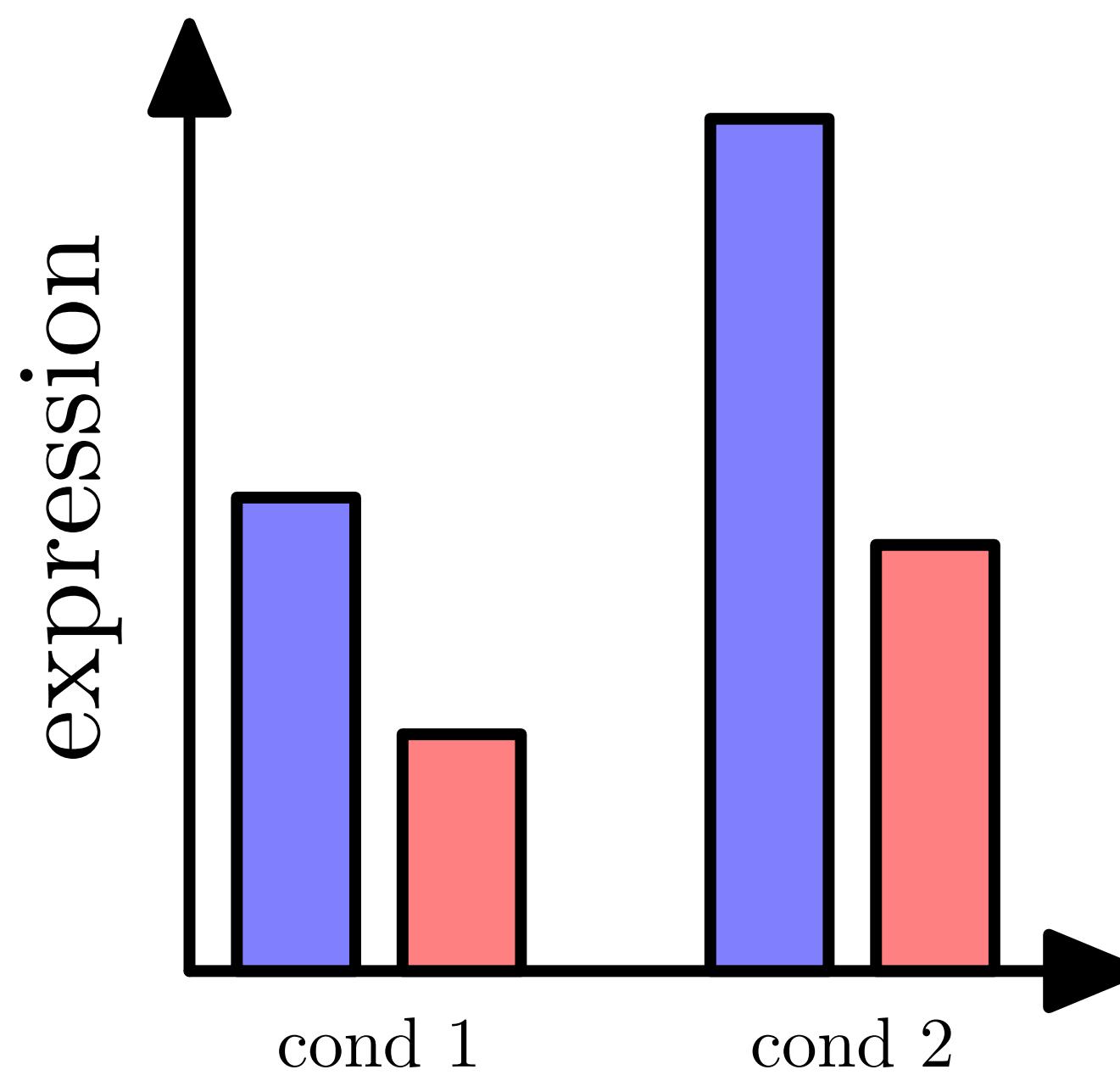
Friedrich Miescher Institute
for Biomedical Research

Differential analysis types for RNA-seq

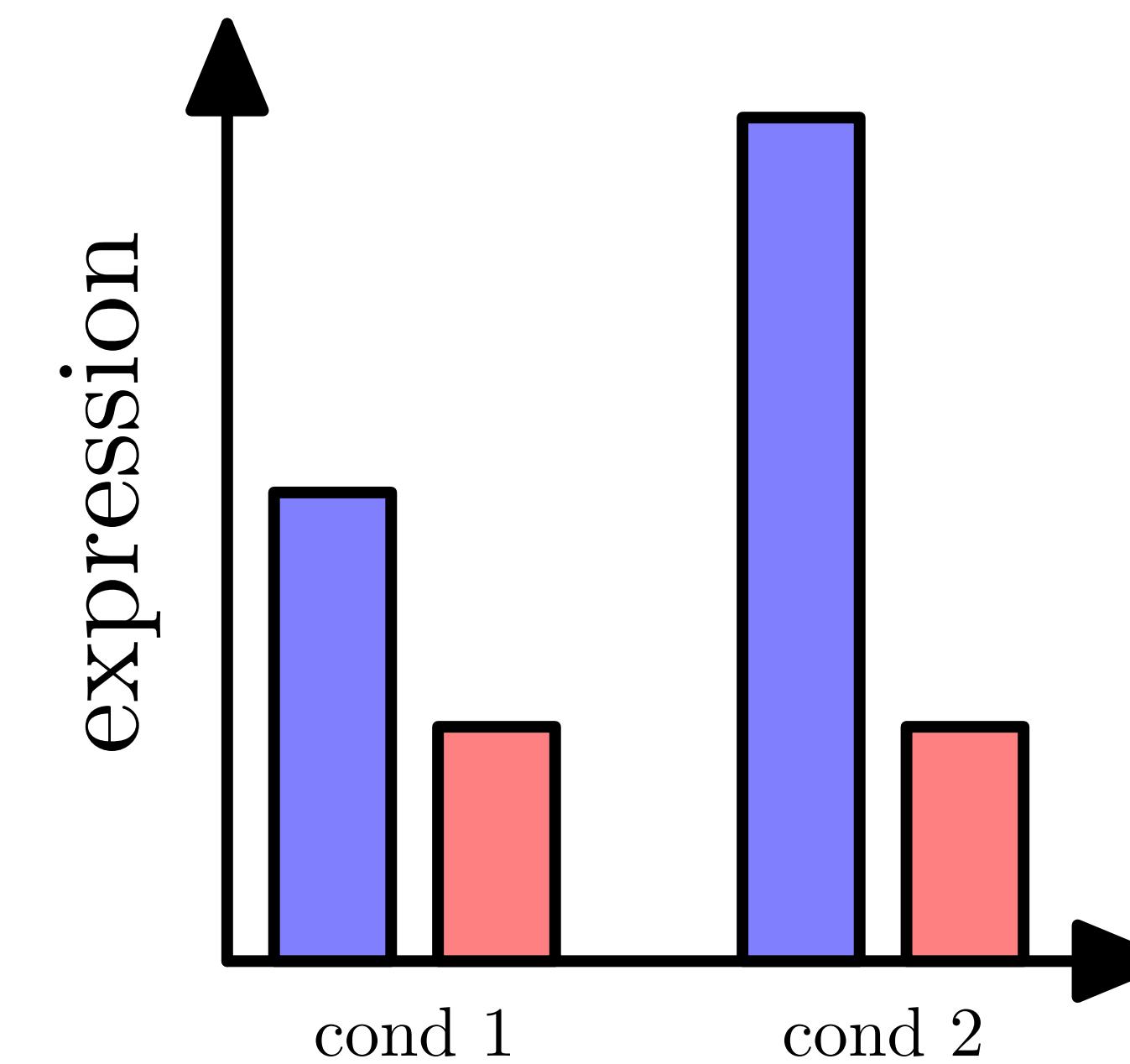
- Does the total output of a gene change between conditions? **DGE**
- Does the expression of individual transcripts change? **DTE**
- Does any isoform of a given gene change? **DTE+G**
- Does the isoform composition for a given gene change? **DTU/DIU/DEU**
- (Does *anything* change? GDE*)
 - need **different** abundance quantification of transcriptomic features (genes, transcripts, exons)

Differential analysis types for RNA-seq

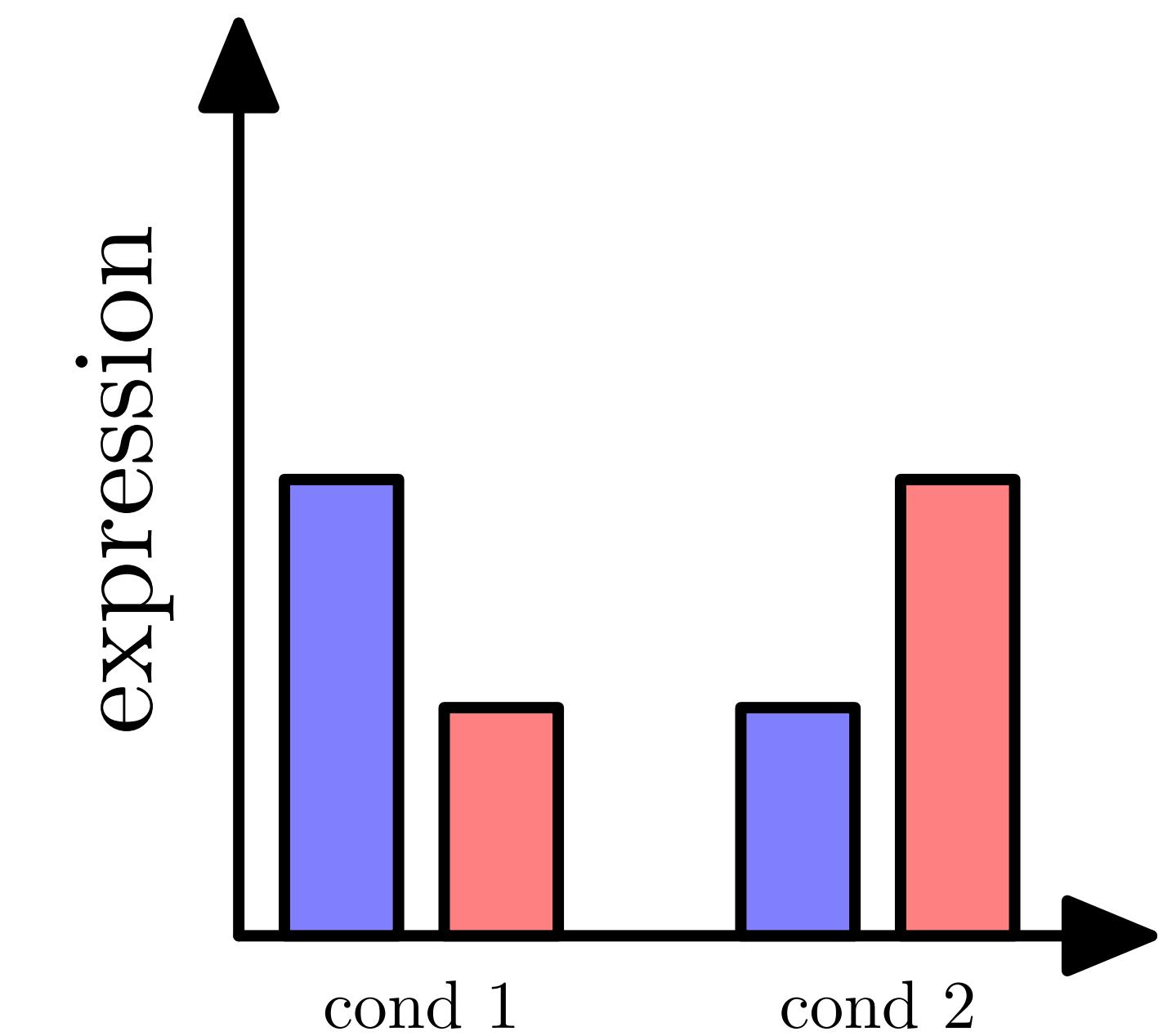
DGE
(also DTE)



DTE
(also DGE, DTU)



DTU
(also DTE)



isoform A



isoform B

Differential expression analysis

- Input: expression/abundance matrix
(features x samples) + grouping/sample annotation

	SRR1039508	SRR1039509	SRR1039512	SRR1039513	SRR1039516	SRR1039517	SRR1039520	SRR1039521
ENSG000000000003	693	451	887	416	1148	1069	774	581
ENSG000000000005	0	0	0	0	0	0	0	0
ENSG000000000419	466	515	623	364	590	794	419	510
ENSG000000000457	326	274	372	223	356	450	308	297
ENSG000000000460	91	75	61	48	110	95	100	82
ENSG000000000938	0	0	2	0	1	0	0	0

- Output: result table (one line per feature)

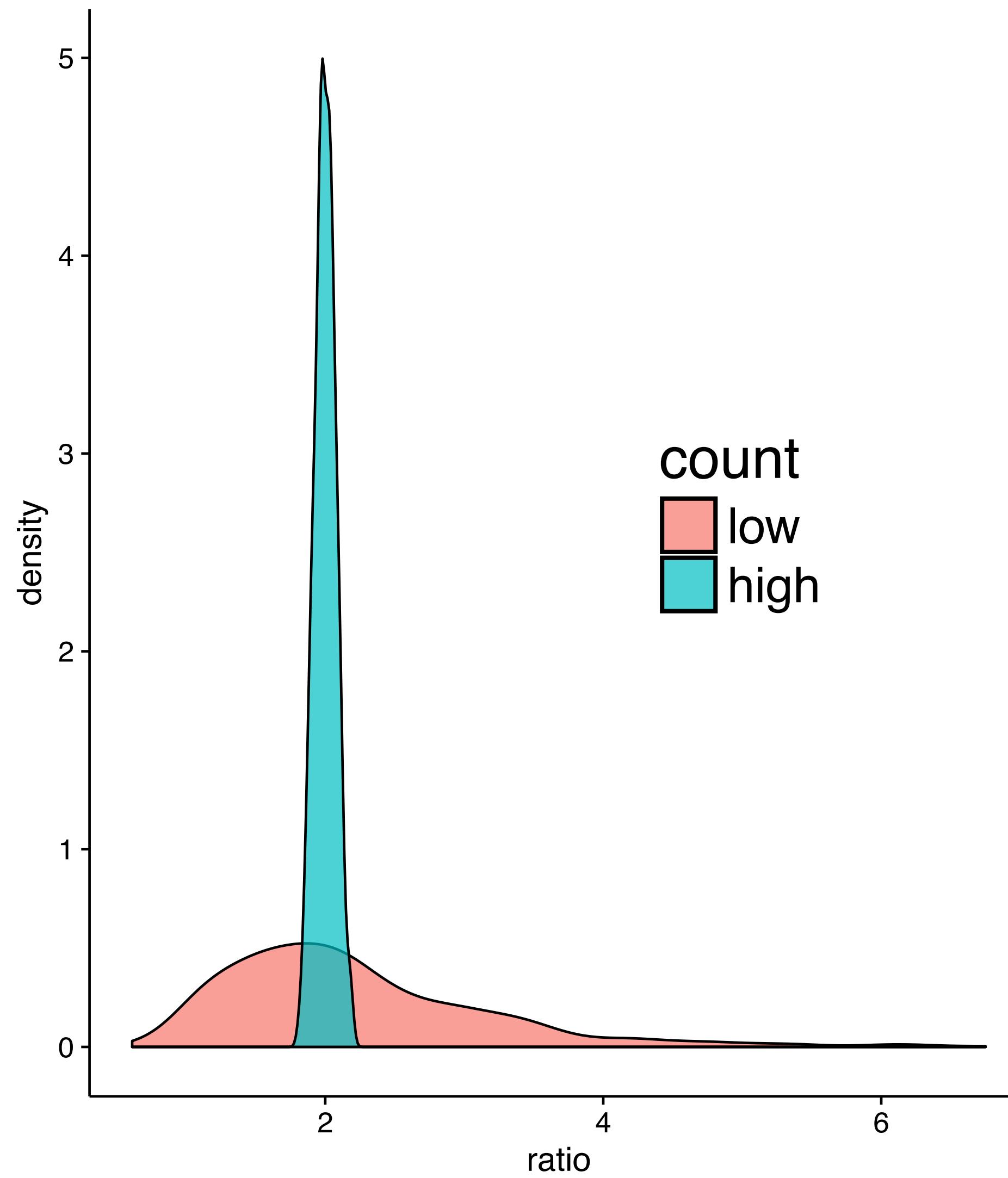
	logFC	logCPM	LR	PValue	FDR
ENSG0000109906	-5.882117	4.120149	924.1622	5.486794e-203	3.493826e-198
ENSG0000165995	-3.236681	4.603028	576.1025	2.641667e-127	8.410672e-123
ENSG0000189221	-3.316900	6.718559	562.9594	1.909251e-124	4.052512e-120
ENSG0000120129	-2.952536	7.255438	506.3838	3.881506e-112	6.179067e-108
ENSG0000196136	-3.225084	6.911908	463.2175	9.587512e-103	1.221008e-98
ENSG0000101347	-3.759902	9.290645	449.9697	7.323427e-100	7.772231e-96
ENSG0000211445	-3.755609	9.102440	433.4656	2.861624e-96	2.603138e-92
ENSG0000162692	3.616656	4.551120	402.0266	1.994189e-89	1.587300e-85
ENSG0000171819	-5.705289	3.474697	389.3431	1.150502e-86	8.140055e-83
ENSG0000152583	-4.364255	5.491013	376.1995	8.363745e-84	5.325782e-80

Differential expression analysis - input

	SRR1039508	SRR1039509	SRR1039512	SRR1039513	SRR1039516	SRR1039517	SRR1039520	SRR1039521
ENSG000000000003	693	451	887	416	1148	1069	774	581
ENSG000000000005	0	0	0	0	0	0	0	0
ENSG000000000419	466	515	623	364	590	794	419	510
ENSG000000000457	326	274	372	223	356	450	308	297
ENSG000000000460	91	75	61	48	110	95	100	82
ENSG000000000938	0	0	2	0	1	0	0	0

- **Most** RNA-seq methods (e.g., edgeR, DESeq2, voom) need **raw counts** (or equivalent) as input
- **Don't** provide these methods with (e.g.) RPKMs, FPKMs, TPMs, CPMs, log-transformed counts, normalized counts, ...
- Read documentation carefully!

Why not only relative abundances?



- Ex: ratio between two Poisson distributed variables
- Low:
mean = 20 vs mean = 10
- High:
mean = 2000 vs mean = 1000

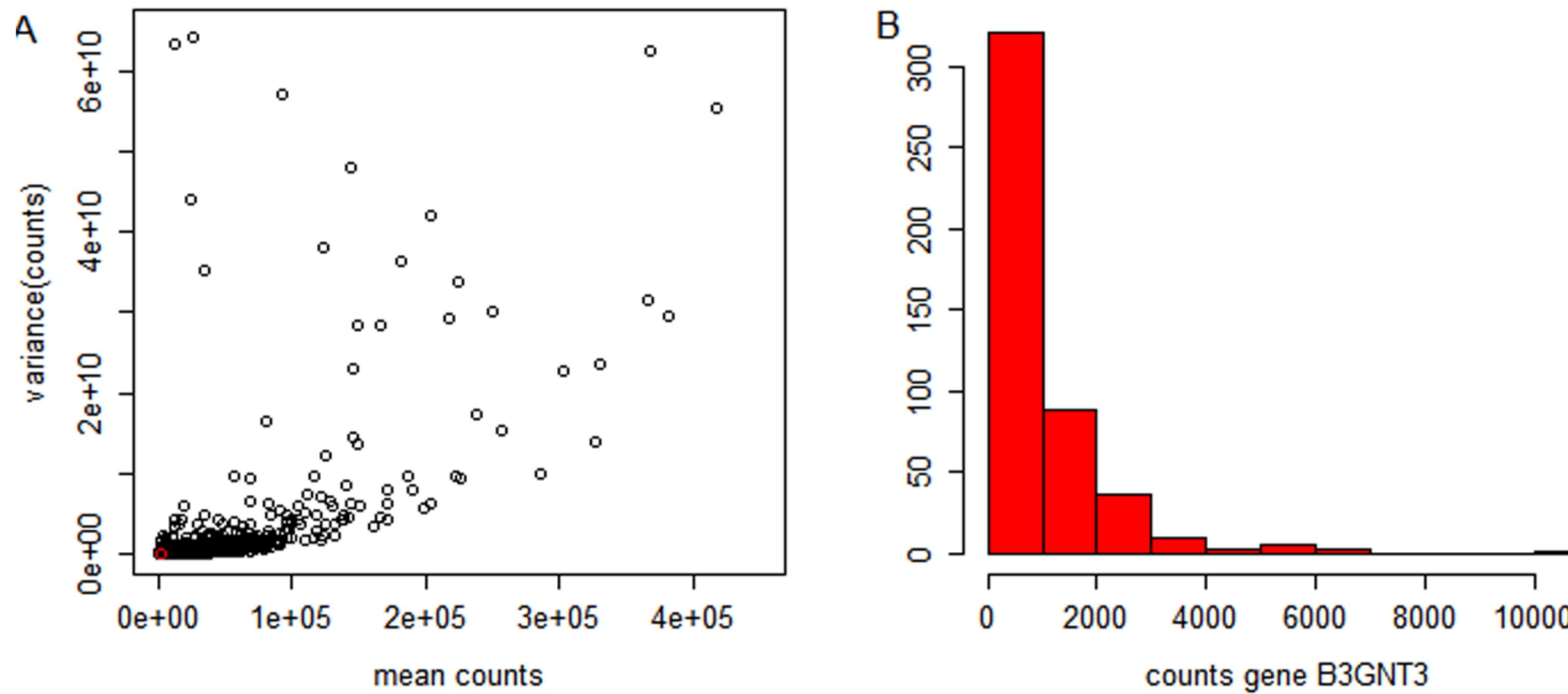
Challenges for RNA-seq data analysis

- Choice of statistical distribution
- Normalization between samples
- Few samples -> difficult to estimate parameters (e.g., variance)
- High dimensionality (many genes) -> many tests

Challenges for RNA-seq data analysis

- **Choice of statistical distribution**
- Normalization between samples
- Few samples -> difficult to estimate parameters (e.g., variance)
- High dimensionality (many genes) -> many tests

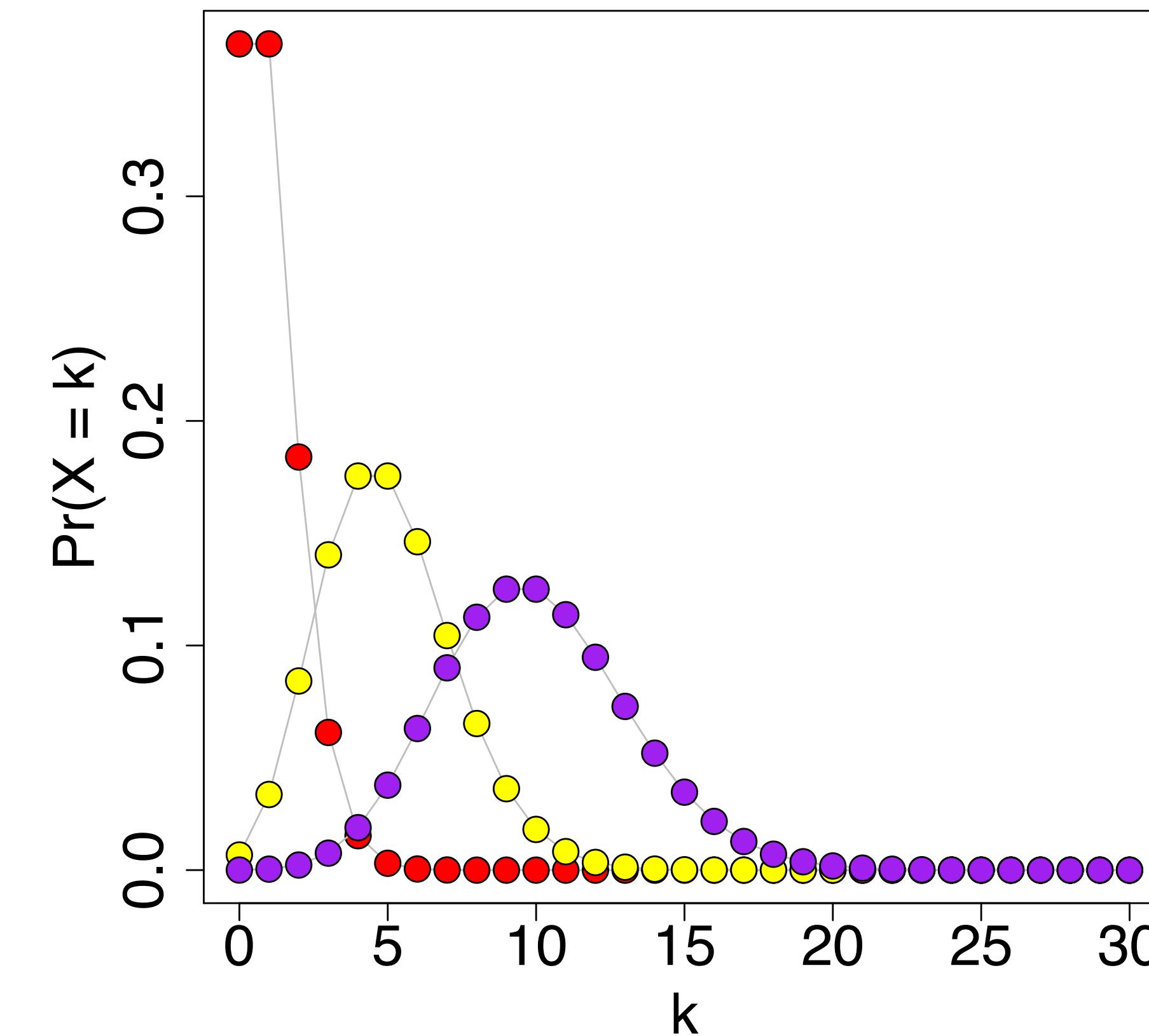
Characteristics of RNA-seq data



- Variance depends on the mean count
- Counts are non-negative and often highly skewed

Modeling counts - the Poisson distribution

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$



Modeling counts - the Poisson distribution

- A famous use of the Poisson distribution was given by von Bortkiewicz (1898) in *Das Gesetz der kleiner Zahlen*
- He studied the number of soldiers in the Prussian army who got kicked by horses, over a number of years and corps



# horsekicks (k)	# obs	fraction
0	109	0,545
1	65	0,325
2	22	0,11
3	3	0,015
4	1	0,005

Modeling counts - the Poisson distribution

- A famous use of the Poisson distribution was given by von Bortkiewicz (1898) in *Das Gesetz der kleiner Zahlen*
- He studied the number of soldiers in the Prussian army who got kicked by horses, over a number of years and corps



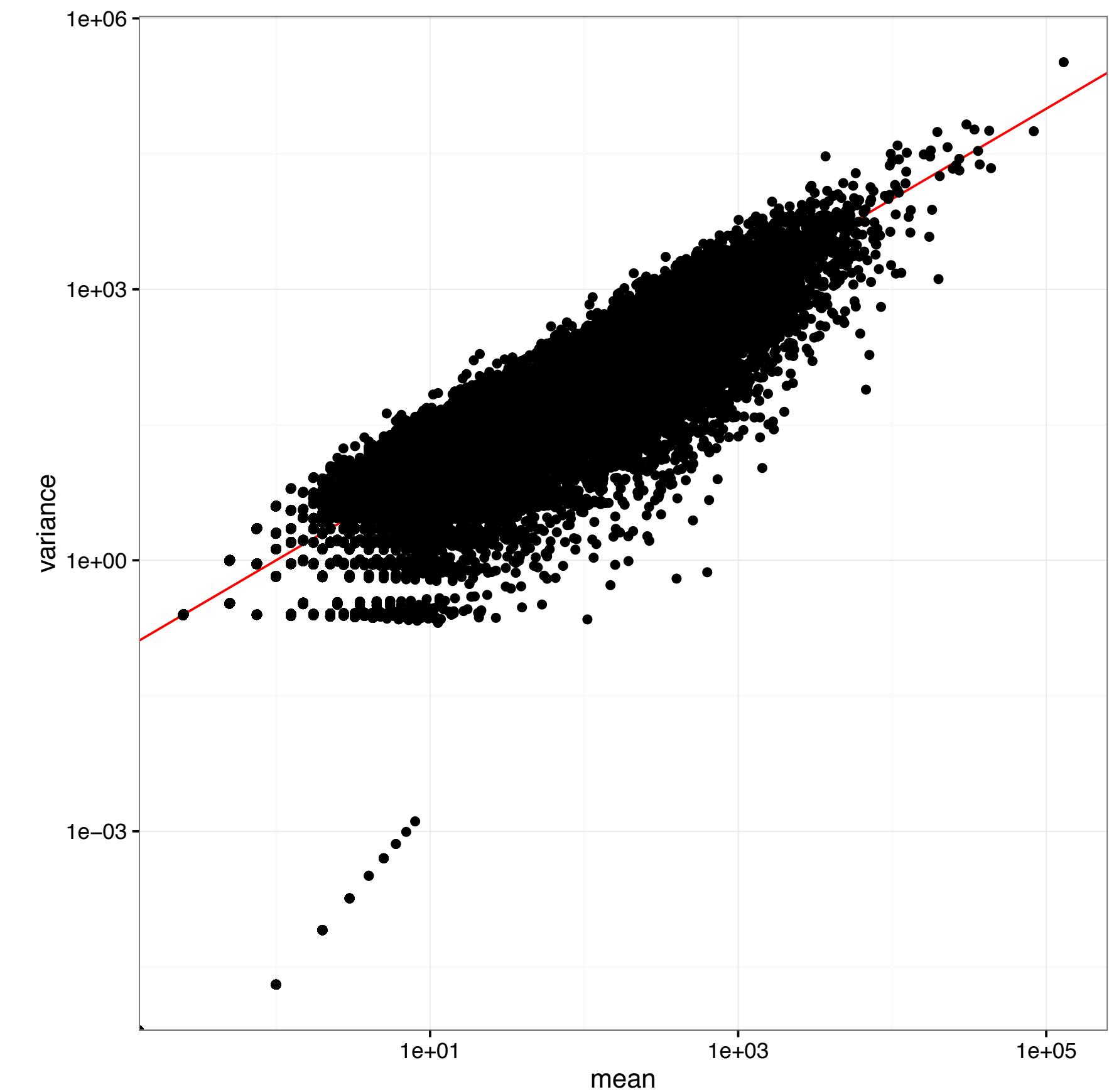
# horsekicks (k)	# obs	fraction	$\frac{0,61^k}{k!} e^{-0,61}$
0	109	0,545	0,543
1	65	0,325	0,331
2	22	0,11	0,101
3	3	0,015	0,0206
4	1	0,005	0,00313

The Poisson distribution for RNA-seq counts

- For RNA-seq data:
 - reads ~ soldiers
 - mapping to gene A ~ being kicked by a horse
- Assumes that the probability of a read mapping to gene A is the same for all samples within a class

Modeling counts

- **Poisson distribution**
 - Quantifies sampling variability
 - $\text{var}(X) = \mu$
 - Represents technical replicates well
(mRNA proportions are identical across samples)

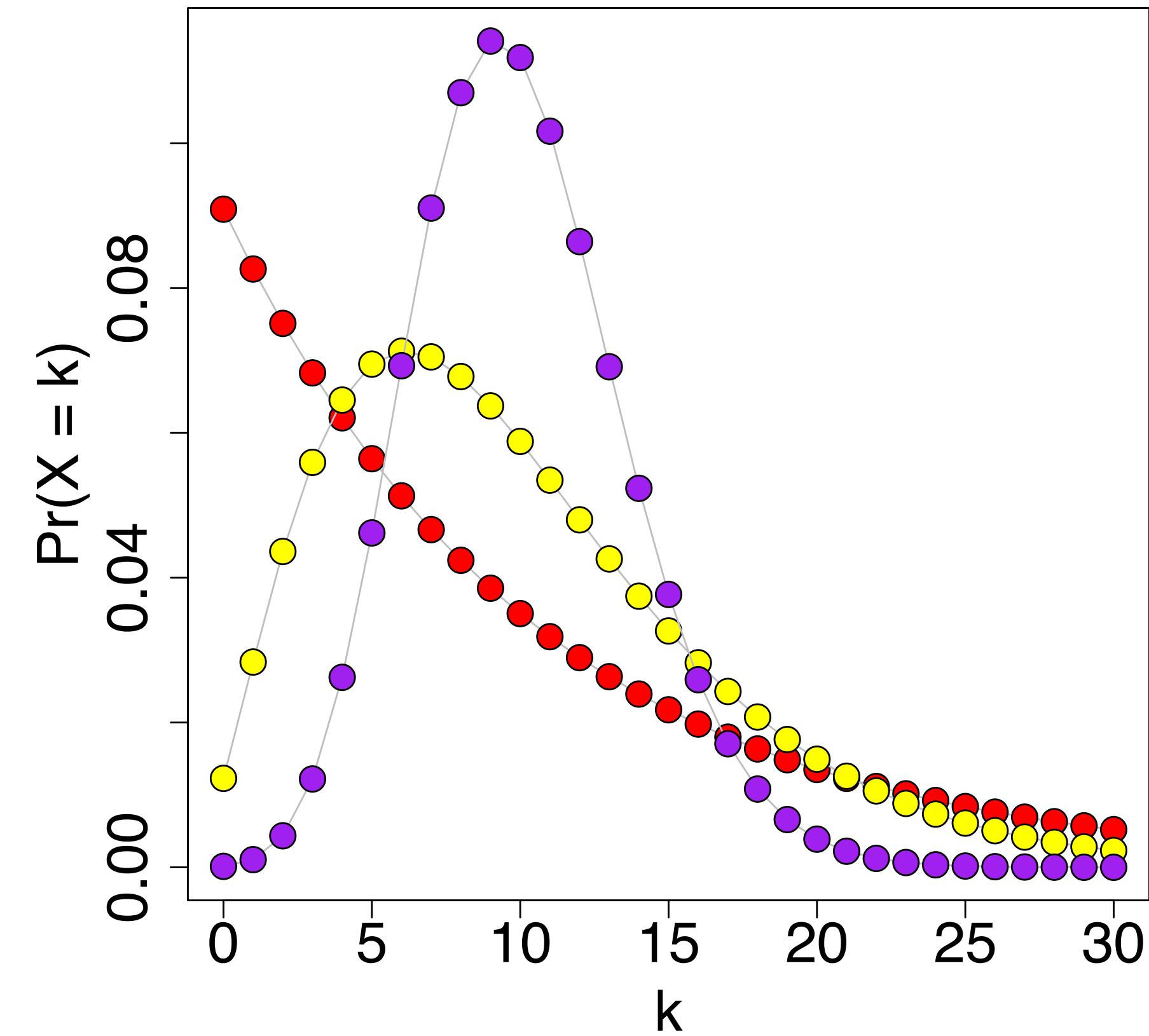


Example from SEQC data, same sample sequenced across multiple lanes

Modeling counts - the Negative Binomial distribution

$$P(X = k) = \binom{k + r - 1}{k} \cdot (1 - p)^r p^k$$

Generalizes the
Poisson distribution

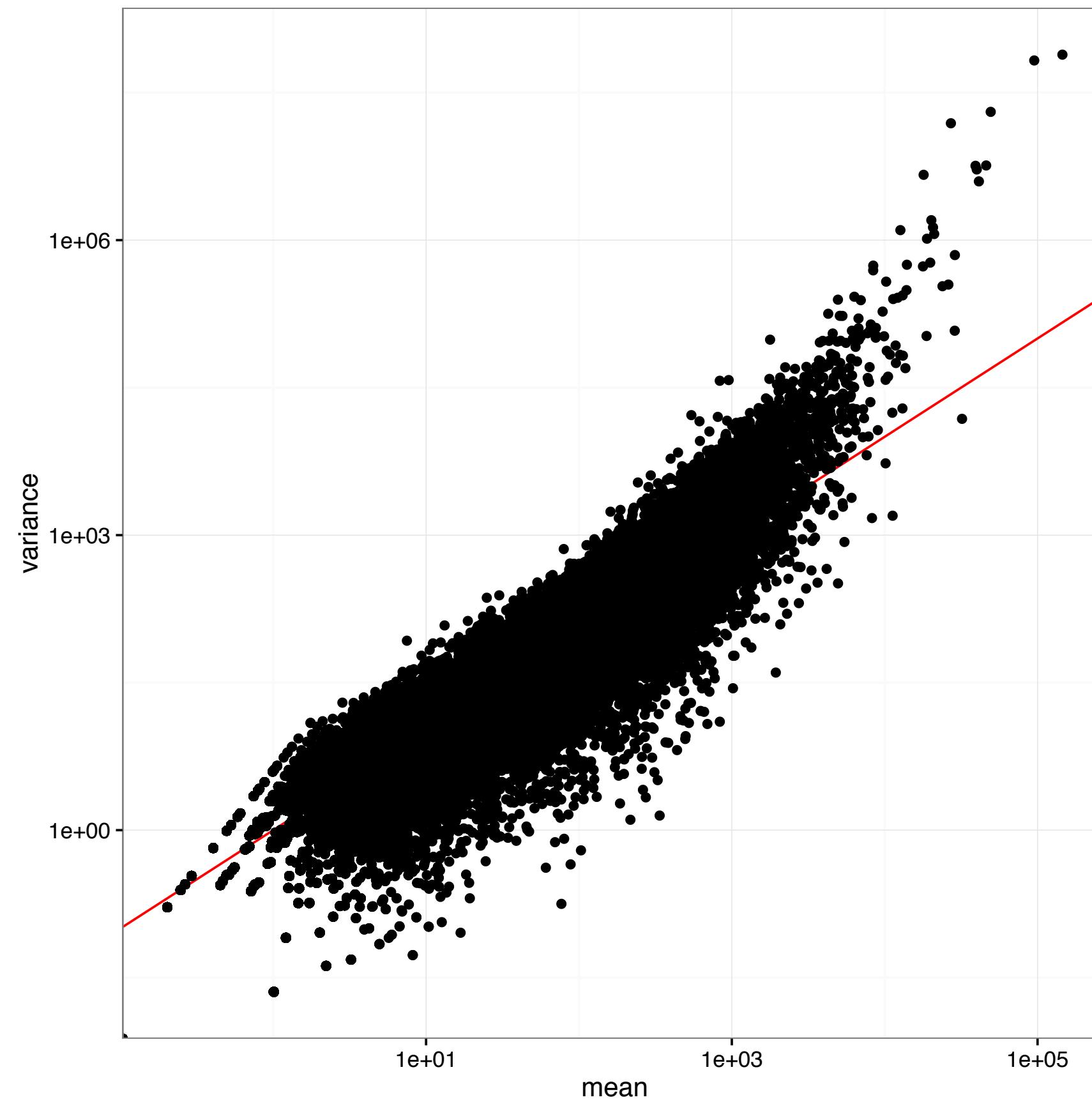


Modeling counts

- **Negative binomial distribution**

- $var(X) = \mu + \theta\mu^2$
- θ = dispersion
- $\sqrt{\theta}$ = "biological coefficient of variation"
- Allows mRNA proportions to vary across samples
- Captures variability across biological replicates better

Example from SEQC data, replicates of the same RNA mix



With count data...

- *linear* modeling (and thus t-tests, ANOVA, etc) is no longer suitable for inference
- *Generalized linear models* to the rescue!

A crash course on GLMs

- A GLM consists of three parts:
 - A *distribution*, specifying the conditional distribution of the response Y given the predictor values
 - A *linear predictor*

$$\eta = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- A *link function* g , linking the conditional expected value of Y to η :
$$g(E[Y|X]) = \eta$$

The linear model is a GLM

- A GLM consists of three parts:
 - A *distribution*, specifying the conditional distribution of the response Y given the predictor values (**Gaussian**)
 - A *linear predictor*

$$\eta = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- A *link function* g , linking the conditional expected value of Y to η :
 $g(E[Y|X]) = \eta$ (**Identity function**)

Other commonly used GLMs

- Logistic regression - binary response
 - Binomial distribution
 - logit link function
- Loglinear regression - count response
 - Poisson distribution
 - log link function

GLMs for RNA-seq

- Negative Binomial distribution
- Log link function
- Implemented e.g. in edgeR and DESeq2

GLMs vs transformation

- The link function in the GLM transforms the *mean*, not the observed values
- Thus, we can transform the systematic part without changing the assumptions on the random part
- By transforming the response (the observed values), we change also the random part (e.g., the association between mean and variance)

voom

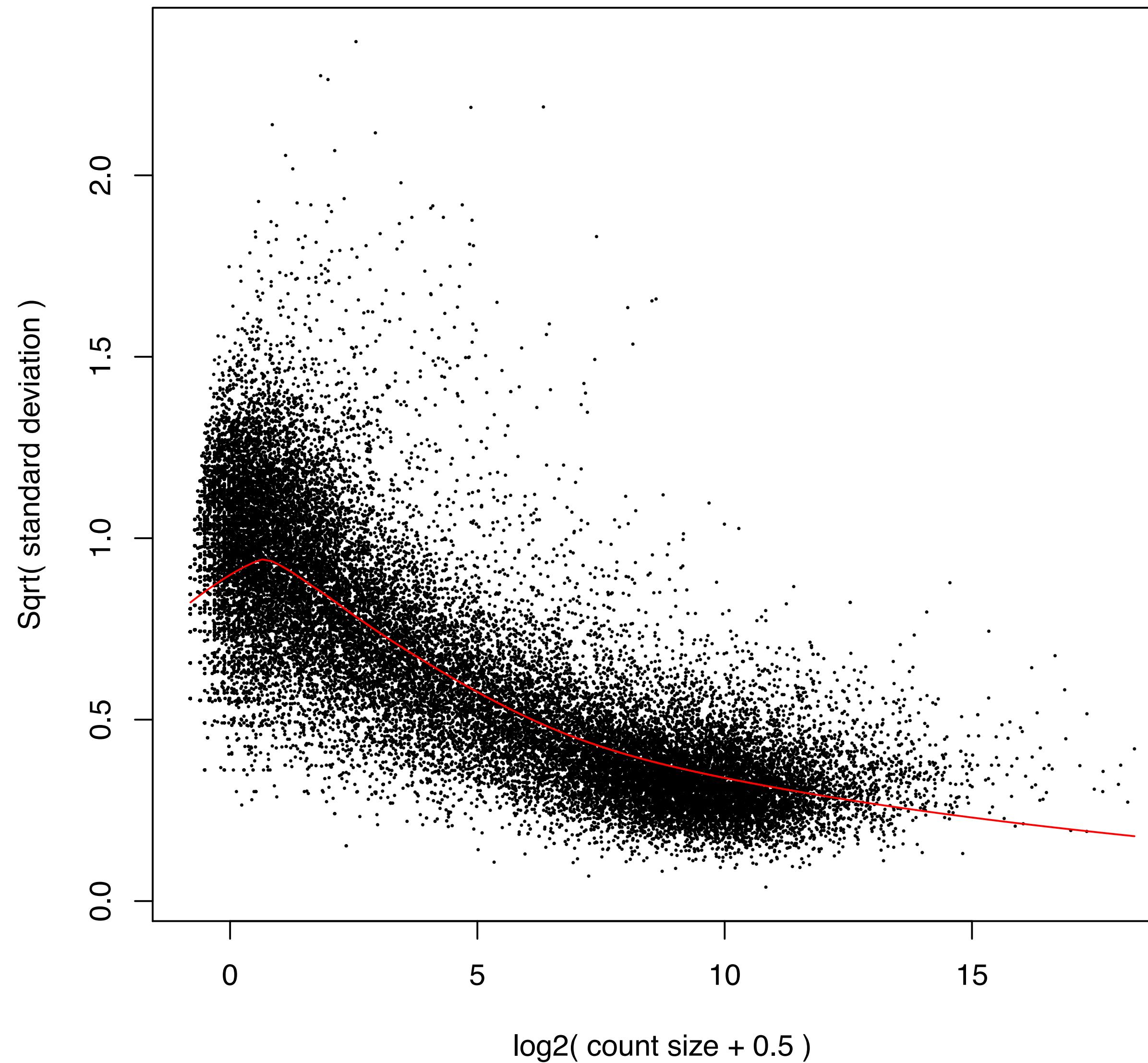
- Instead of modeling the counts, we can **transform them** to a suitable scale and model them with a normal distribution ("microarray-like").
- **voom** (part of the limma package) calculates logCPM values

$$y_{gi} = \log_2 \left(\frac{r_{gi} + 0.5}{R_i + 1.0} \times 10^6 \right)$$

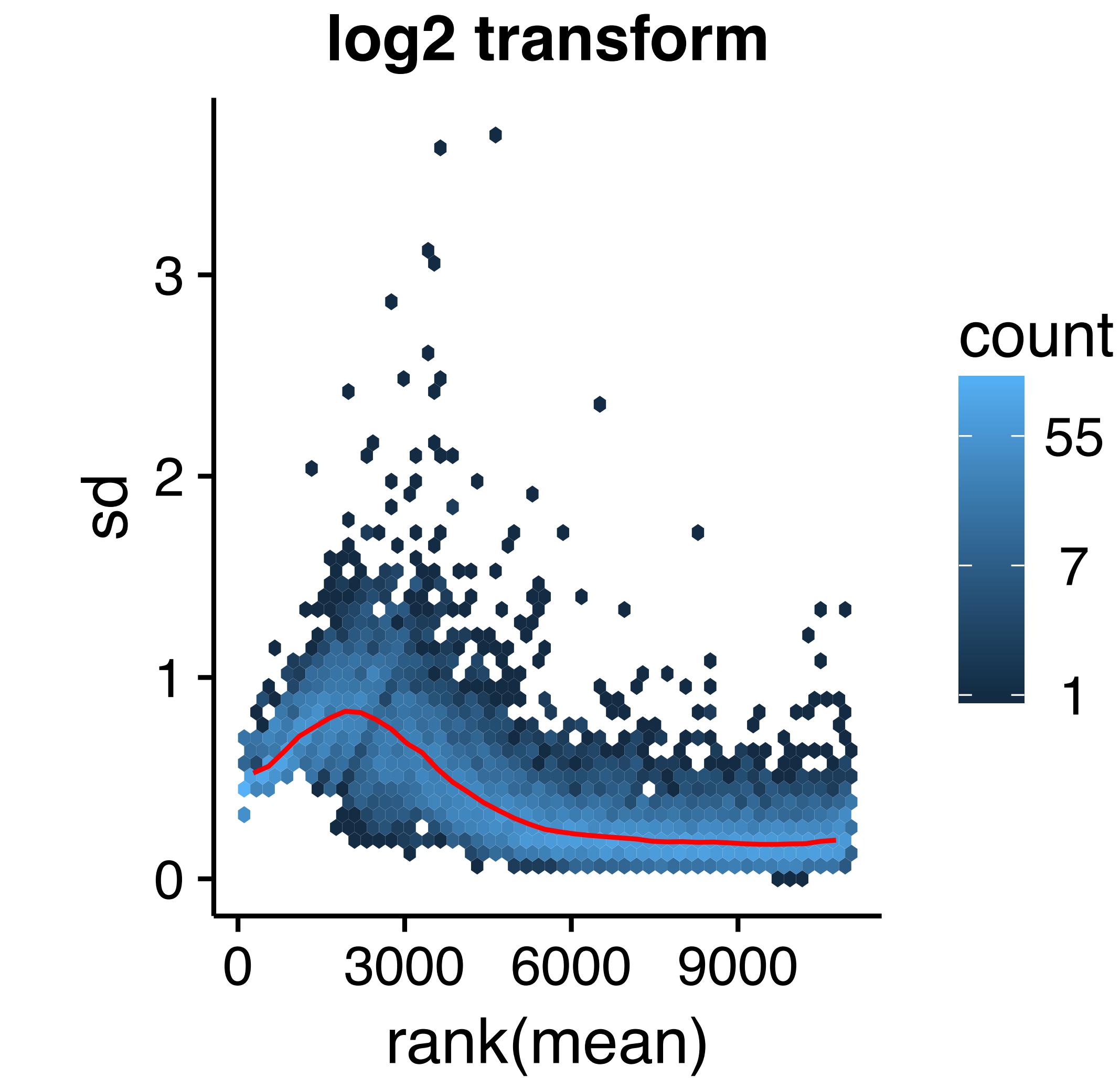
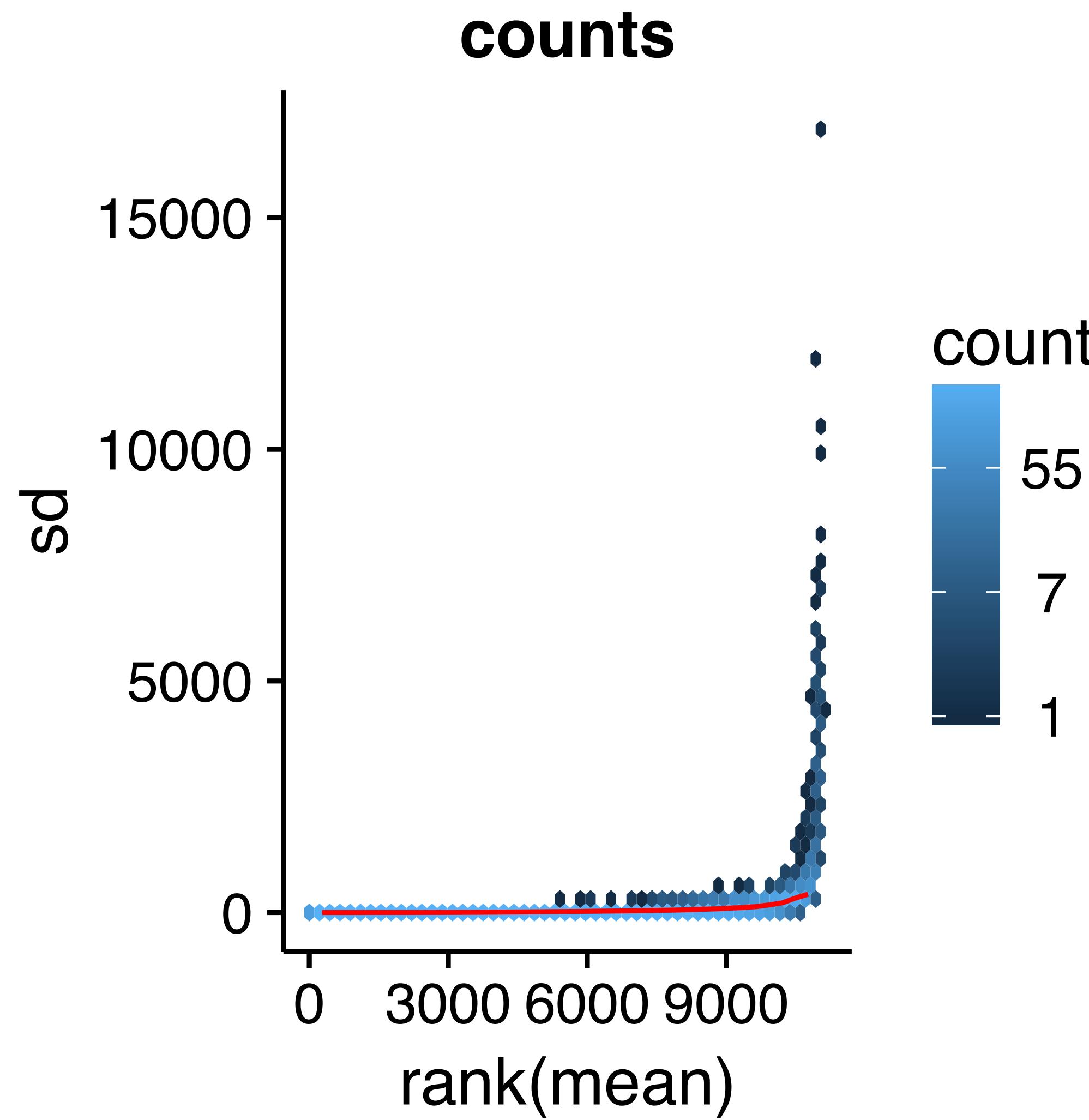
- Transformed data is heteroskedastic (variance depends on mean) - use weighted least squares

voom - mean/variance relationship

voom: Mean–variance trend

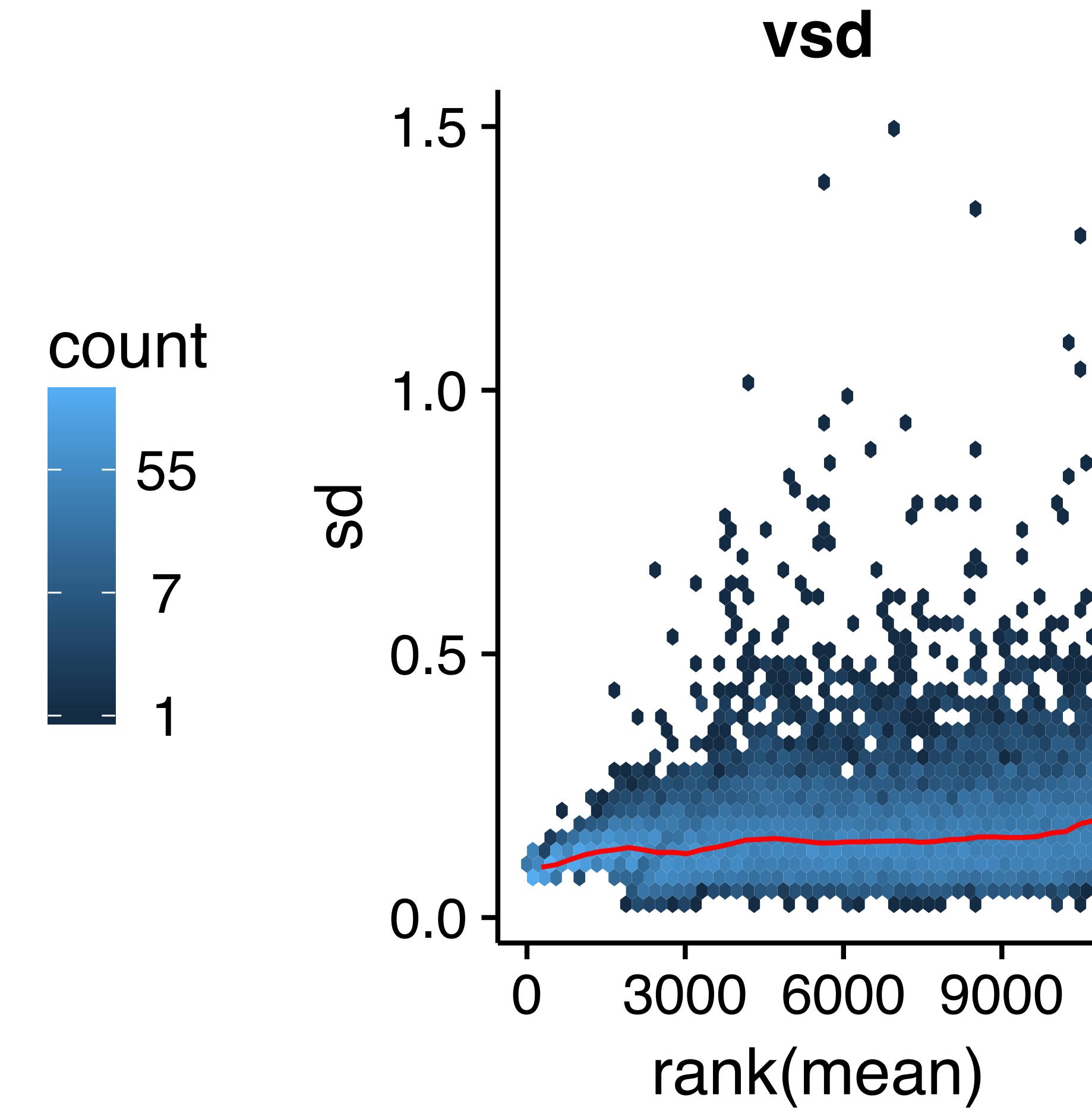
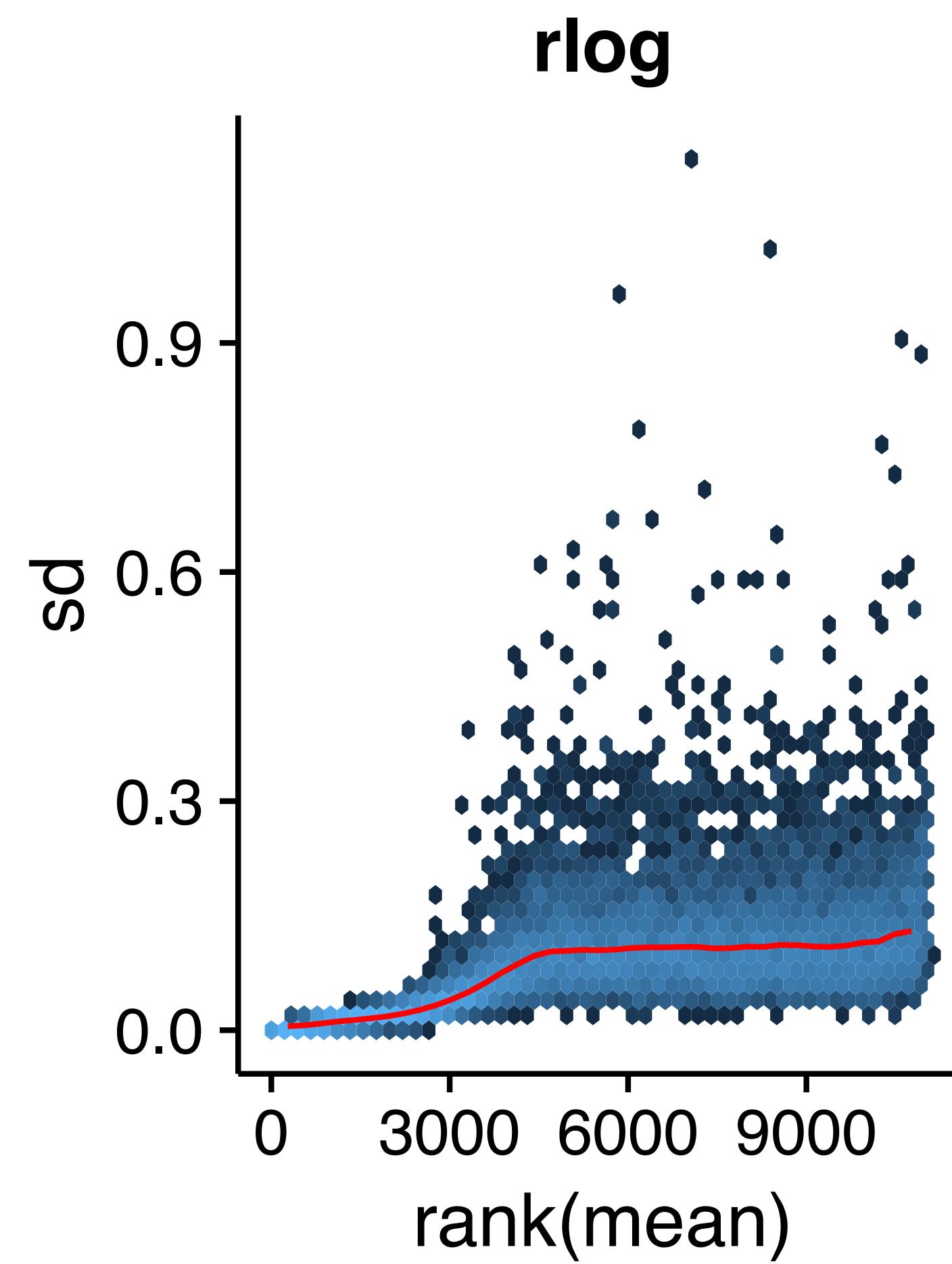


Data transformations - log



Data transformations - DESeq2

- Two approaches: rlog, variance stabilizing transformation
- Aim: remove dependence of variance on mean after transformation



Nonparametric tests

- Do not assume a specific distribution for the data
- Significance is inferred via permutations
- Perform well for highly replicated experiments
- Lack power for experiments with few replicates
- Resistant to outliers

Challenges for RNA-seq data analysis

- Choice of statistical distribution
- **Normalization between samples**
- Few samples -> difficult to estimate parameters (e.g., variance)
- High dimensionality (many genes) -> many tests

Normalization

- Observed counts depend on:
 - abundance
 - gene length
 - sequencing depth
 - sequencing biases
 - ...
- “As-is”, not directly comparable across samples

Normalization

$$C_{ij} \sim NB(\mu_{ij} = s_{ij}q_{ij}, \theta_i)$$

raw count for gene i in sample j

normalization factor

relative abundance

dispersion

The diagram illustrates the components of a negative binomial distribution. The mean μ_{ij} is shown as the product of a normalization factor s_{ij} and relative abundance q_{ij} . The dispersion parameter θ_i is indicated by an arrow pointing towards the mean term.

- s_{ij} is a normalization factor (or offset) in the model
- counts are not explicitly scaled
 - important exception: voom/limma (followed by explicit modeling of mean-variance association)

Simple example - offsets

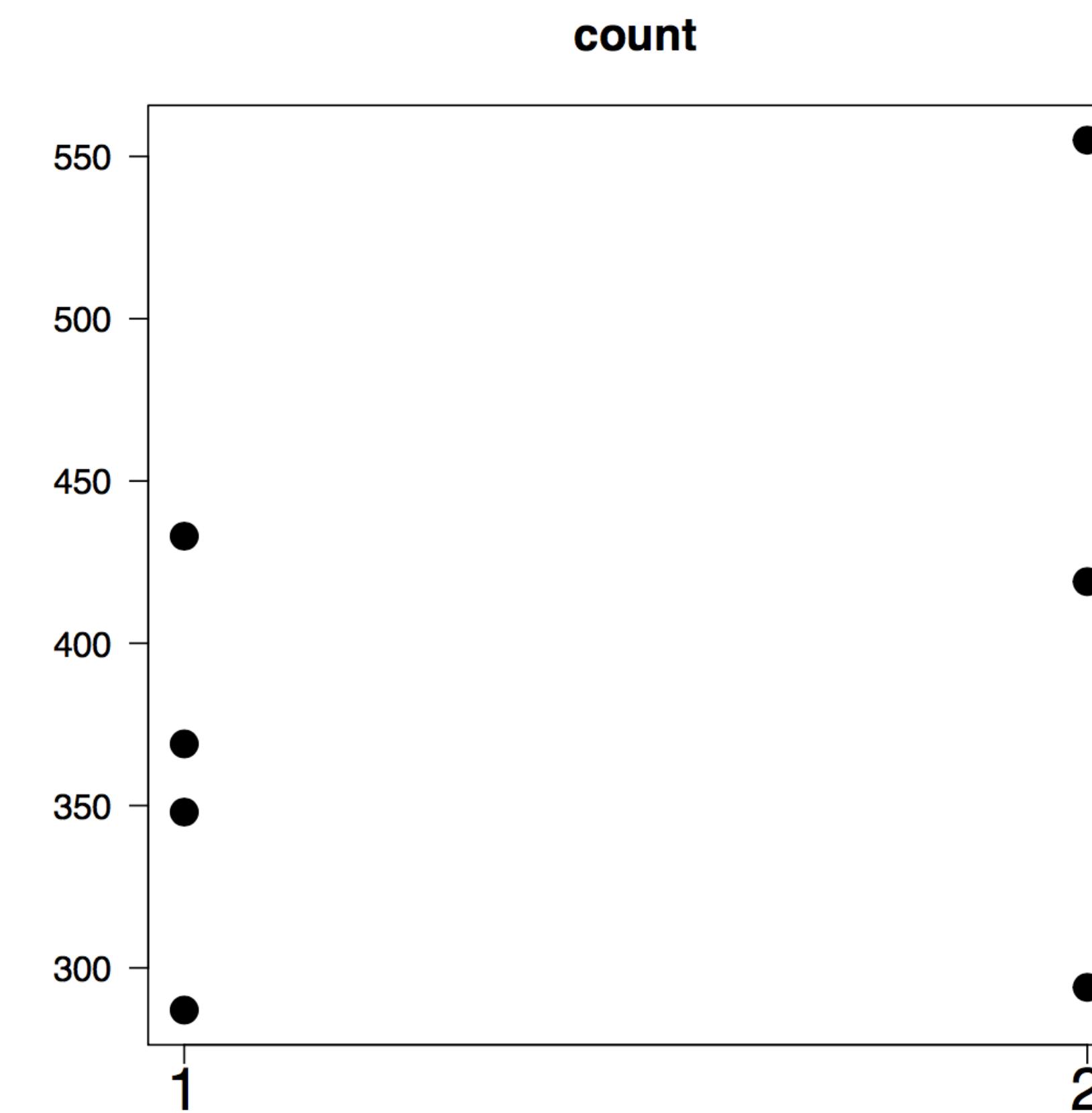
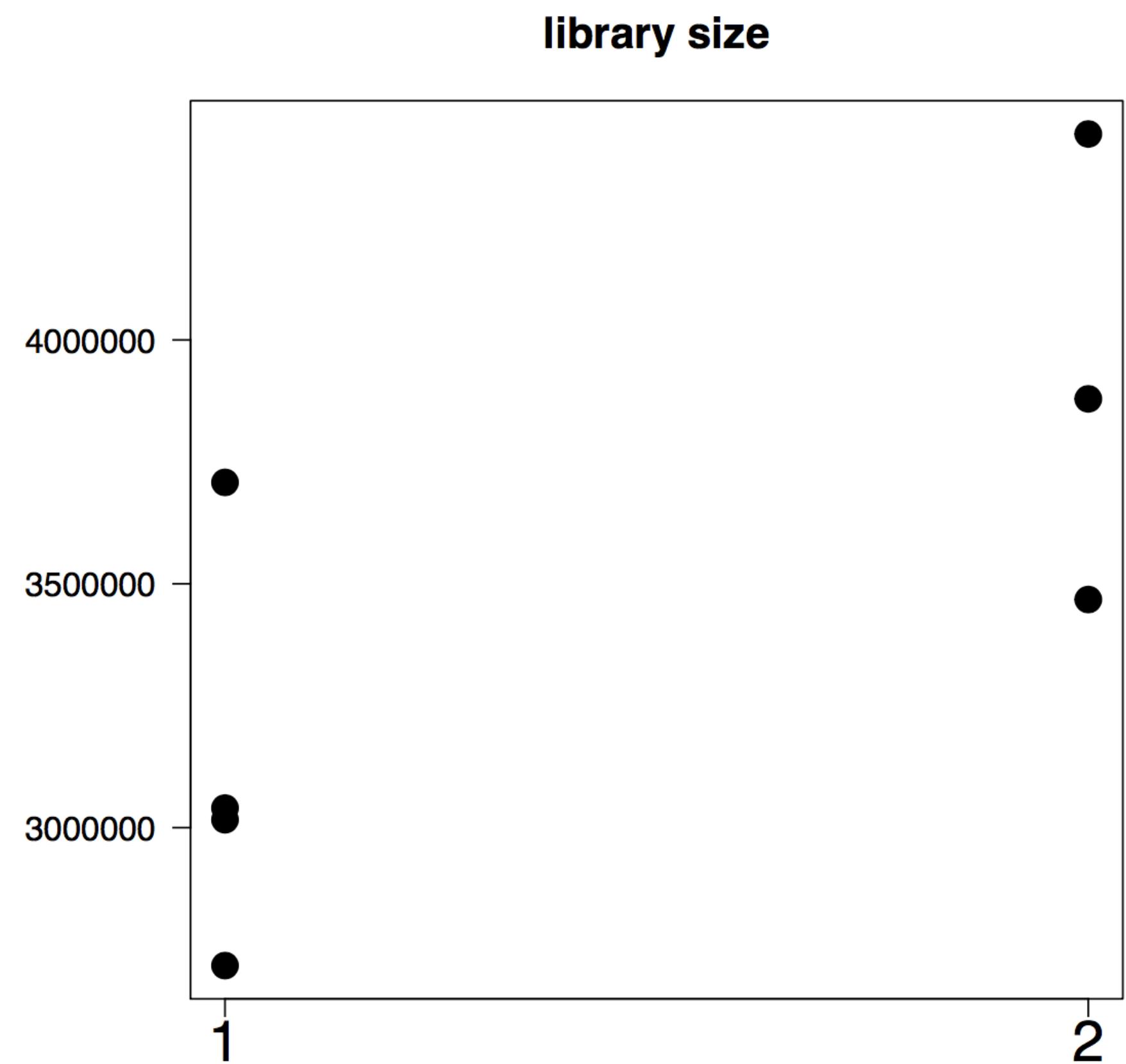
- Assume that we have RNA-seq reads for one gene. Is the gene differentially expressed?

```
count.data <- data.frame(counts = c(369, 287, 348, 433, 555, 294, 419),  
                           cond = c("1", "1", "1", "1", "2", "2", "2"))  
glm.pois <- glm(counts ~ cond, family = poisson, data = count.data)  
coefficients(summary(glm.pois))
```

	##	Estimate	Std. Error	z value	Pr(> z)
## (Intercept)	5.8840	0.02638	223.050	0.000e+00	
## cond2	0.1626	0.03853	4.219	2.451e-05	

Simple example - offsets

- Relate counts to library sizes



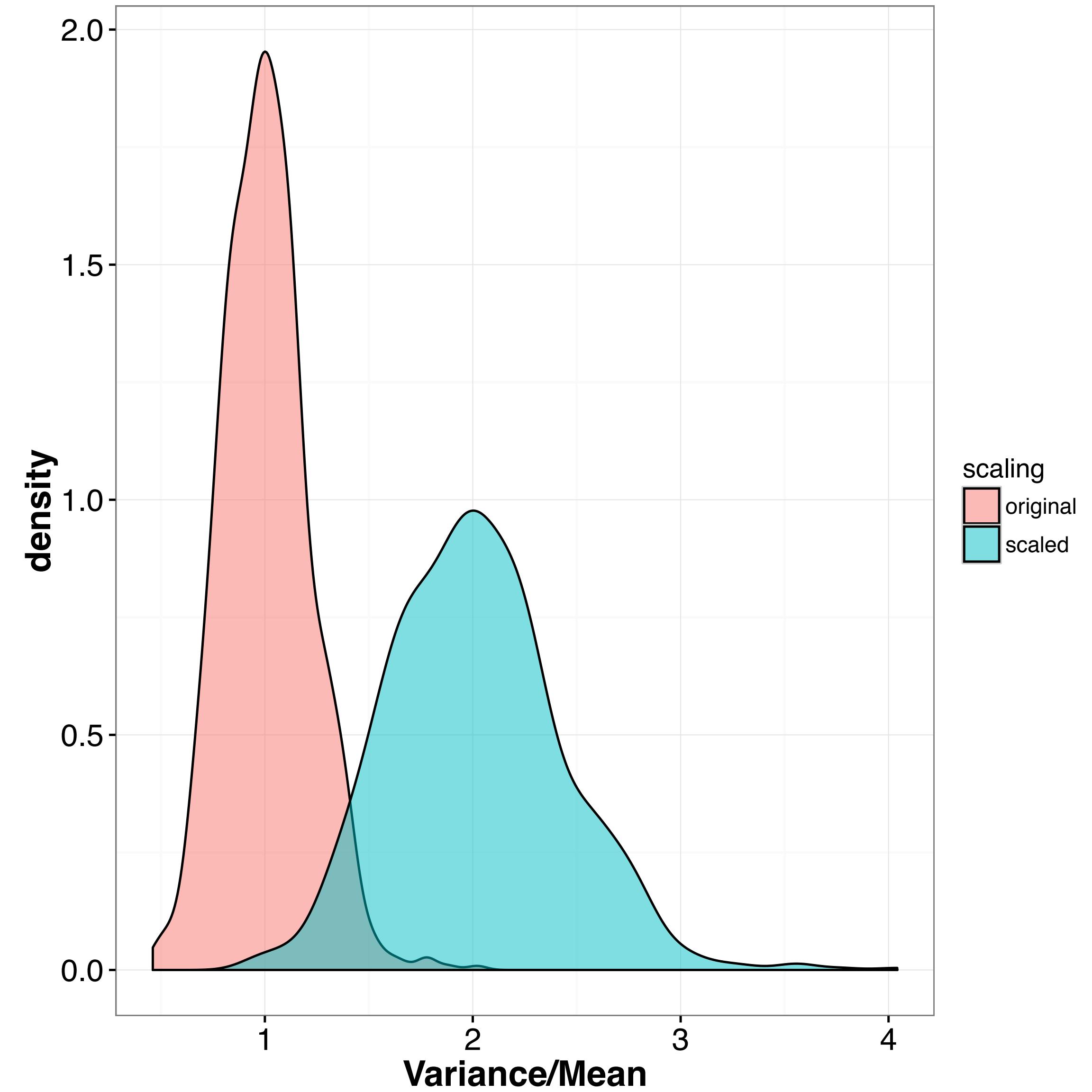
Simple example - offsets

- Incorporate library size as offset

```
count.data$lib.size <- c(3040296, 2717092, 3016179, 3707895,  
                         4422272, 3467730, 3879114)  
glm.pois <- glm(counts ~ cond + offset(log(lib.size)), family = poisson,  
                  data = count.data)  
coefficients(summary(glm.pois))  
  
##             Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -9.06944   0.02638 -343.802 0.00000  
## cond2       -0.06635   0.03853   -1.722 0.08506
```

Why offset rather than scaling?

- Variance/Mean for Poisson distributed variable, before as well as after multiplying the values with 2.



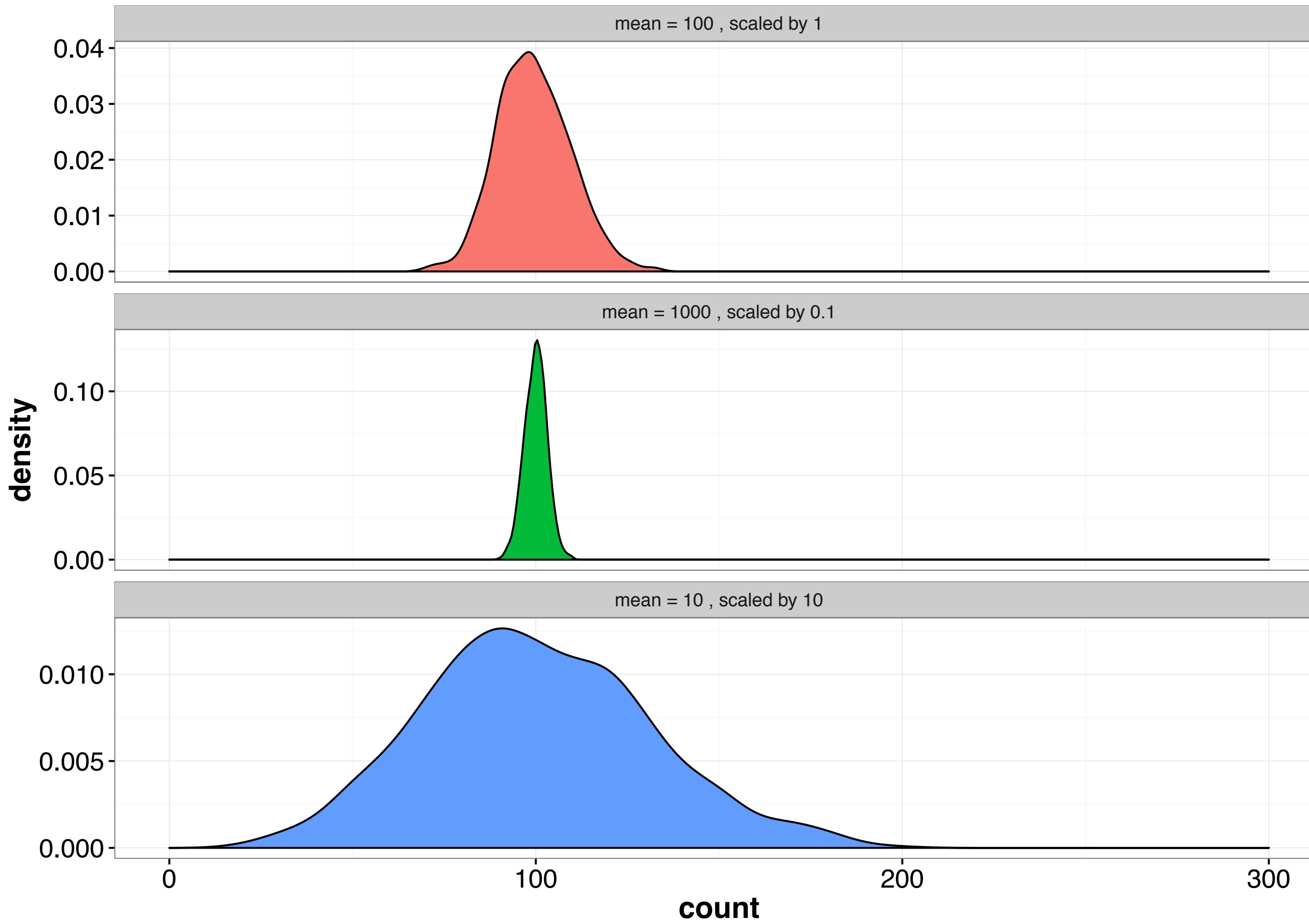
Why offset rather than scaling?

- Poisson distributed variables with different means, scaled to have mean = 100

Raw count
mean = 100

Raw count
mean = 1000,
scaled by 0.1

Raw count
mean = 10,
scaled by 10



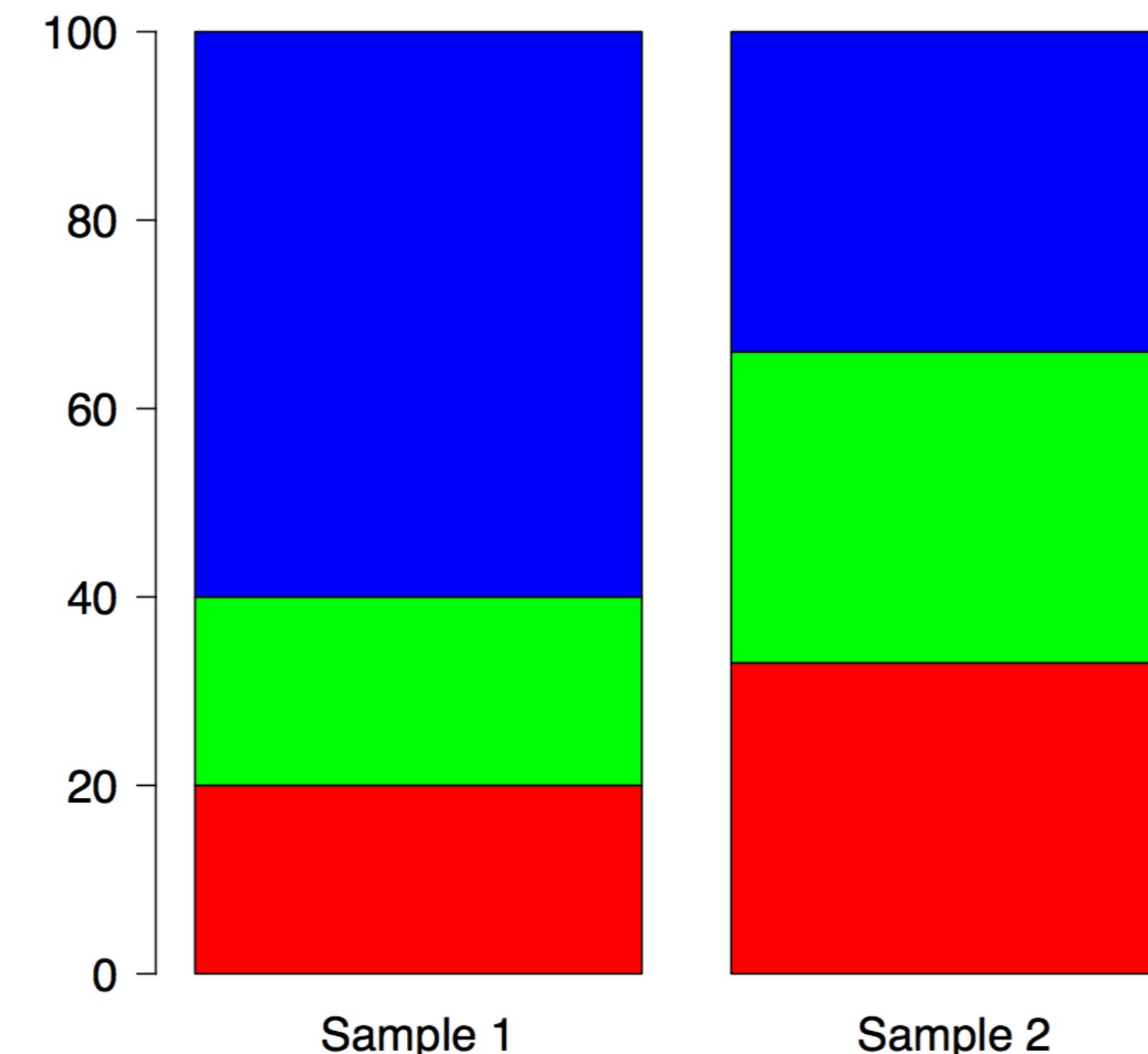
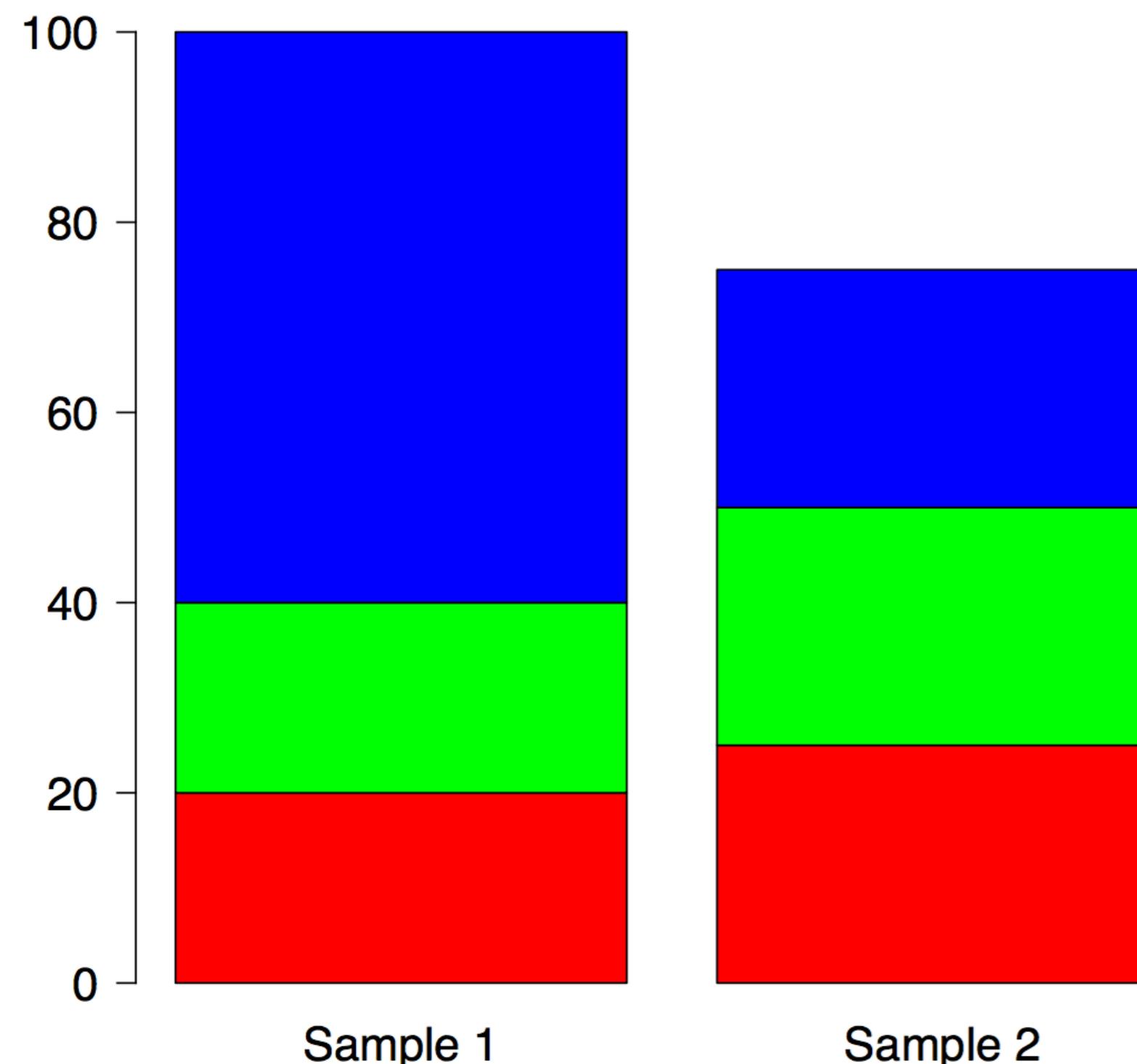
How to calculate normalization factors?

- Attempt 1: **total count** (library size)
 - Define a reference sample (one of the observed samples or a “pseudo-sample”) - gives a “target library size”
 - Normalization factor for sample j is defined by

$$\frac{\text{total count in sample } j}{\text{total count in reference sample}}$$

The influence of RNA composition

- Observed counts are relative
- High counts for some genes are “compensated” by low counts for other genes

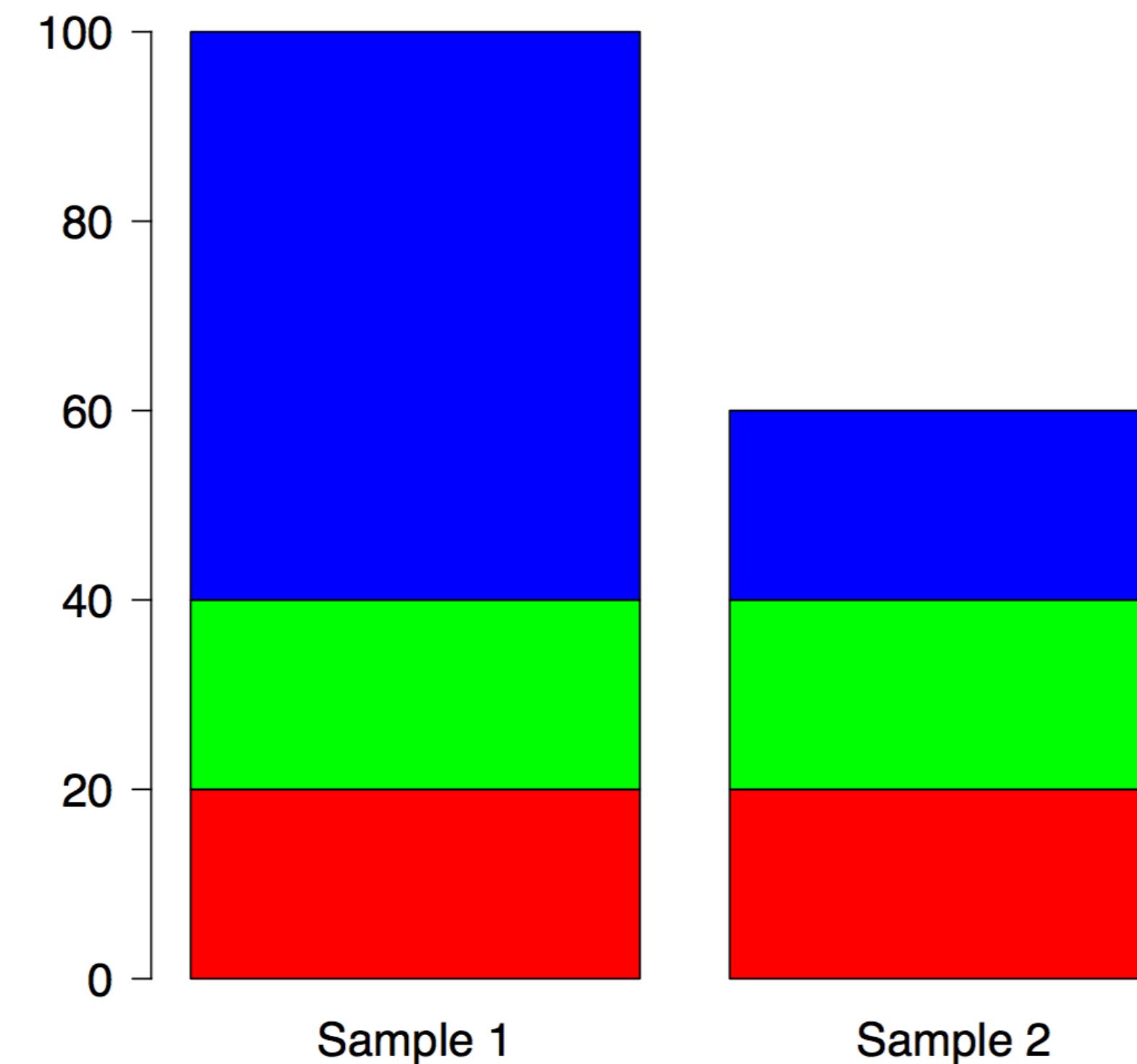
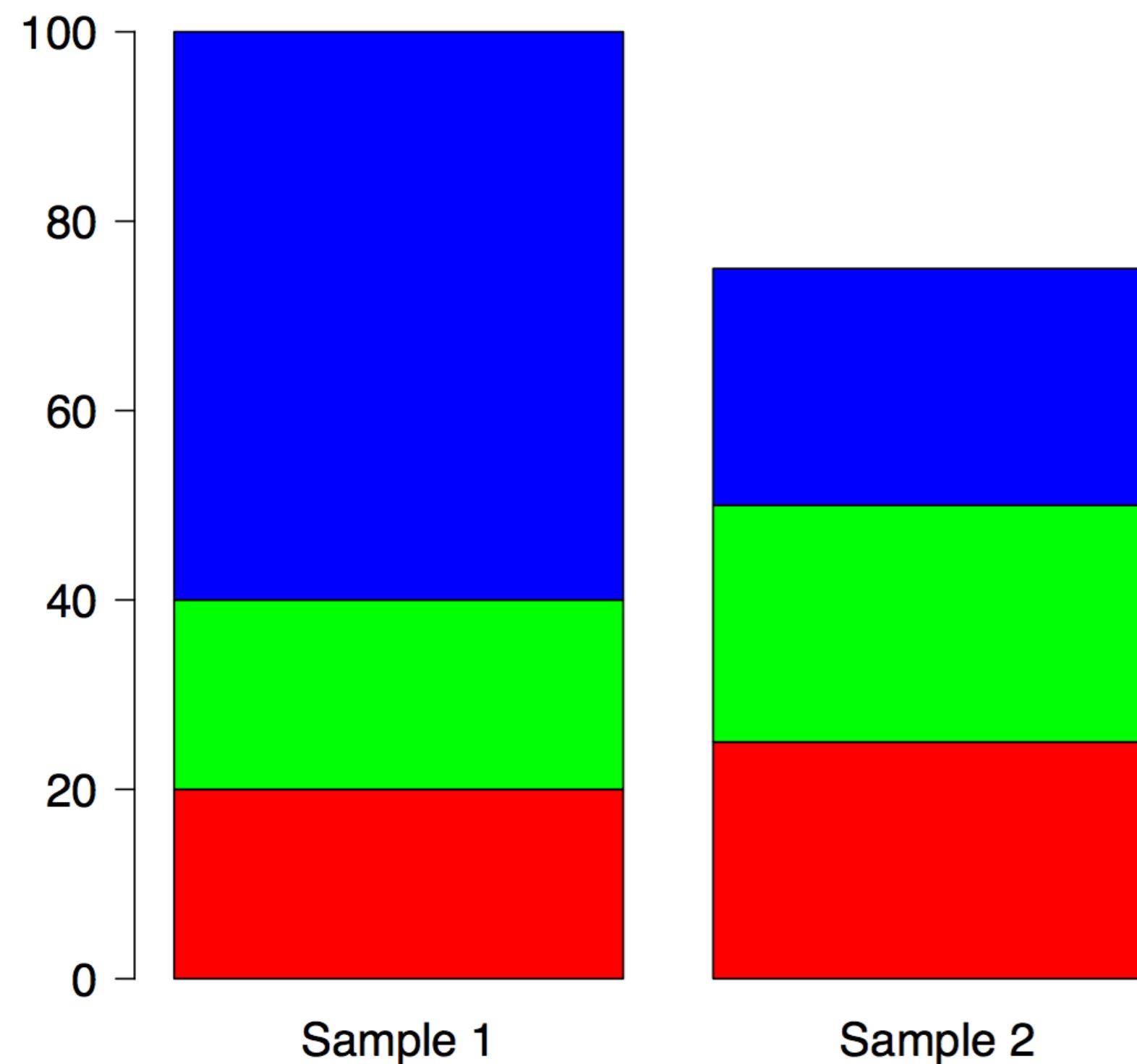


How to calculate normalization factors?

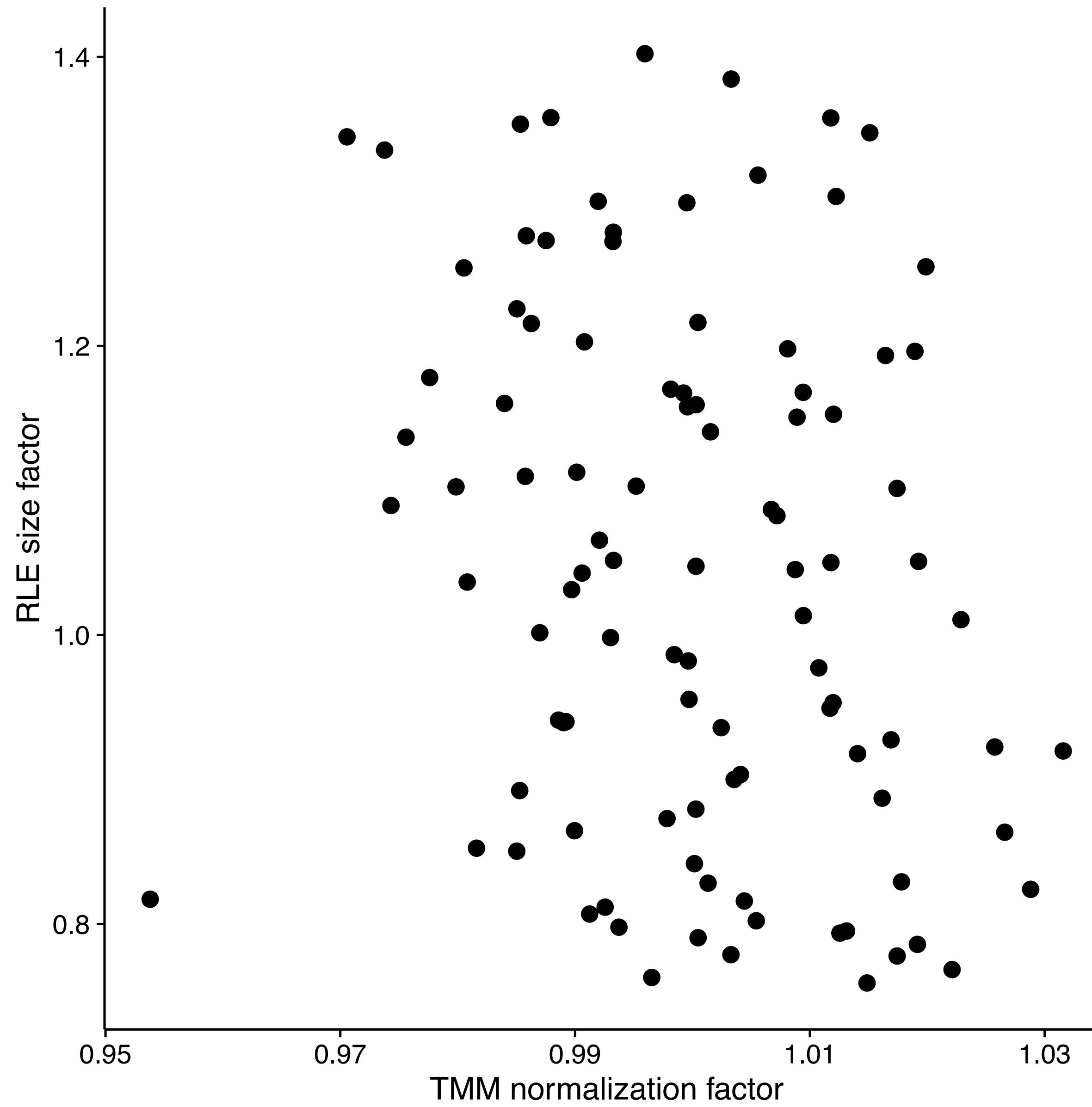
- Attempt 2: total count (library size) * compensation for differences in composition
- Idea: use only non-differentially expressed genes to compute the normalization factor
- Implemented by both edgeR (TMM) and DESeq2 (median count ratio)
- Both these methods assume that most genes are not differentially expressed

How to calculate normalization factors?

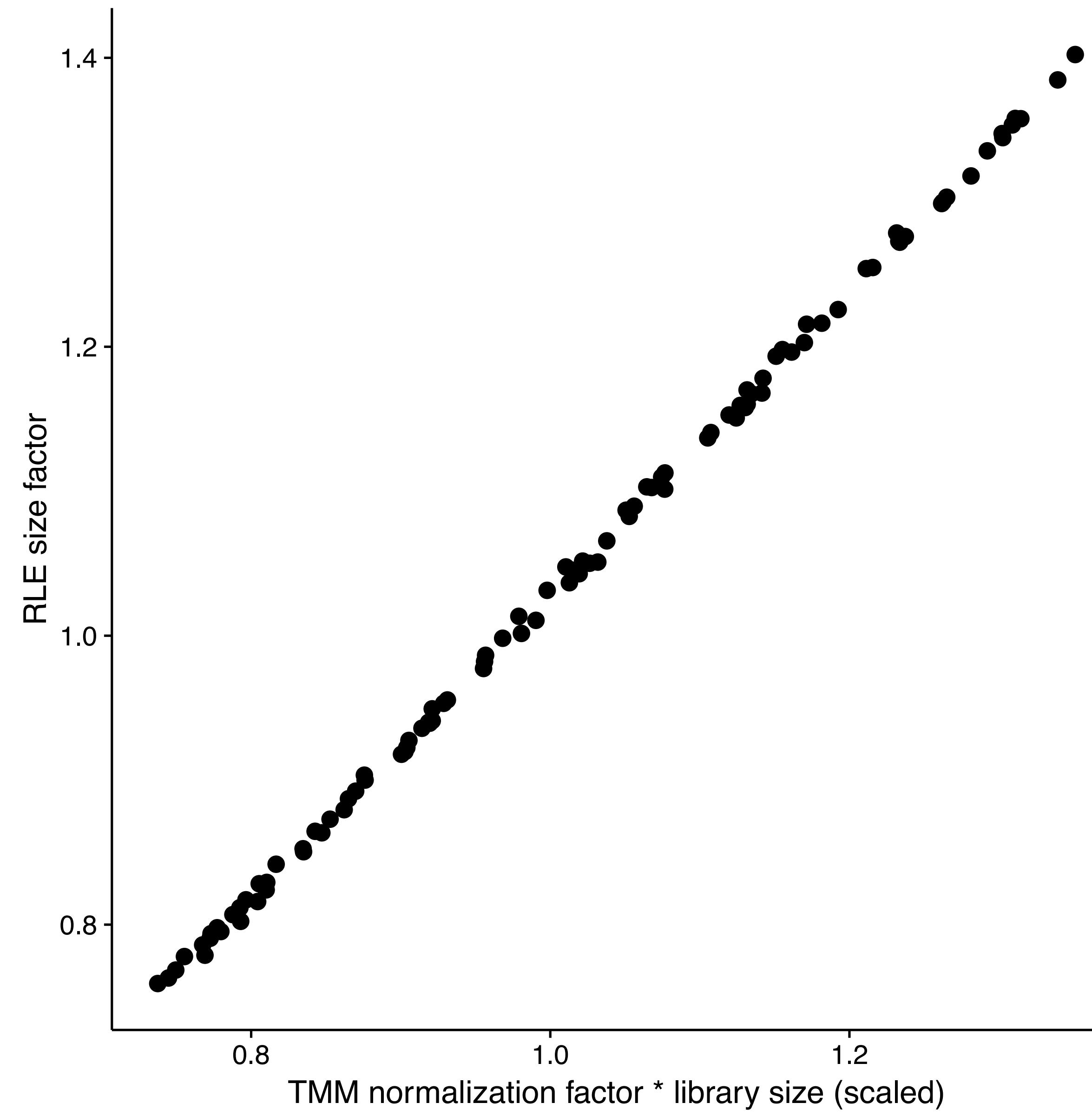
- Attempt 2: total count (library size) * compensation for differences in composition



“Normalization factors” (edgeR) vs “size factors” (DESeq2)



"Effective library sizes" (edgeR) vs "size factors" (DESeq2)

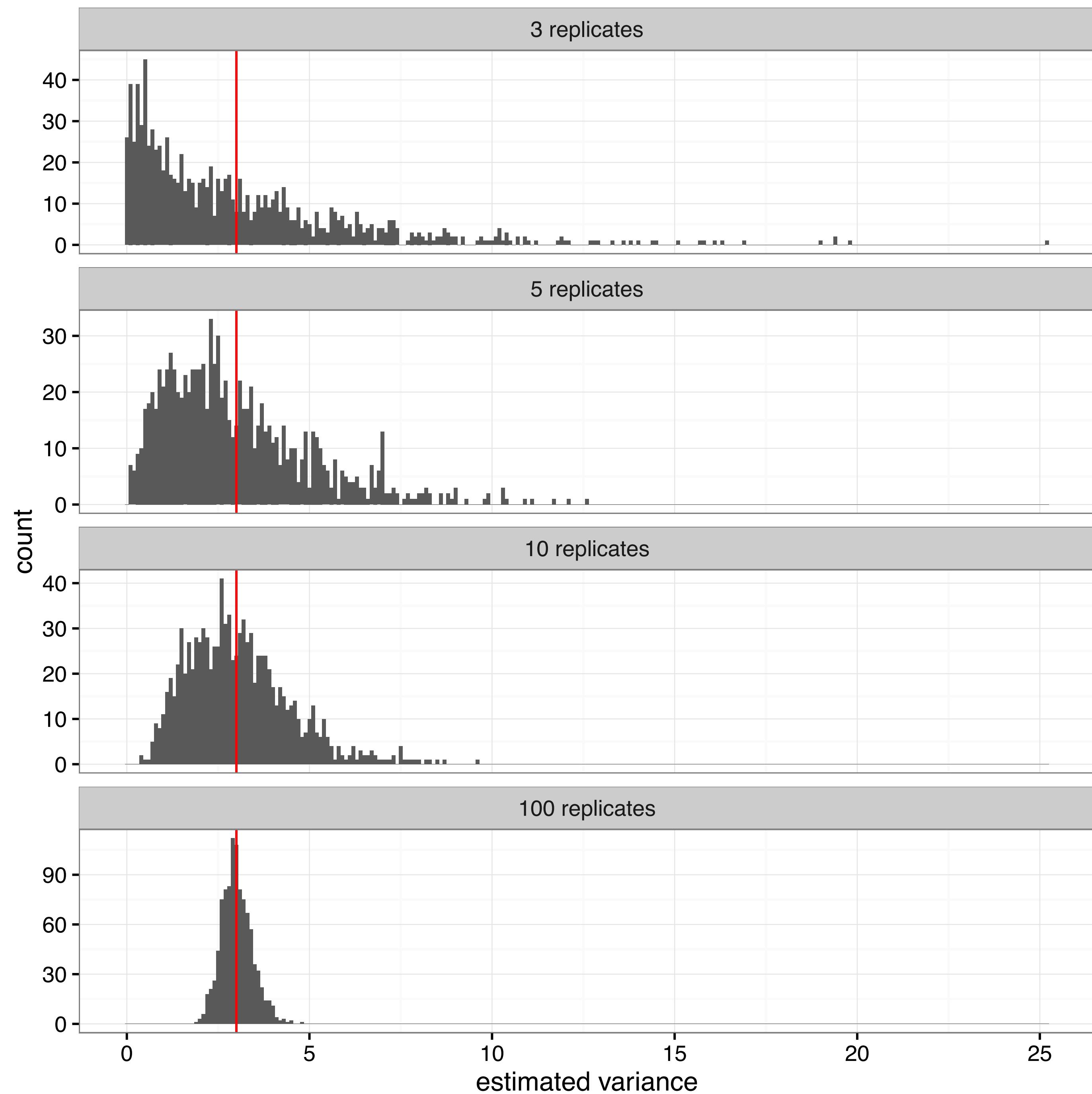


Challenges for RNA-seq data analysis

- Choice of statistical distribution
- Normalization between samples
- **Few samples -> difficult to estimate parameters (e.g., variance)**
- High dimensionality (many genes) -> many tests

Example:
estimate variance of
normally distributed
variable

True value = 3

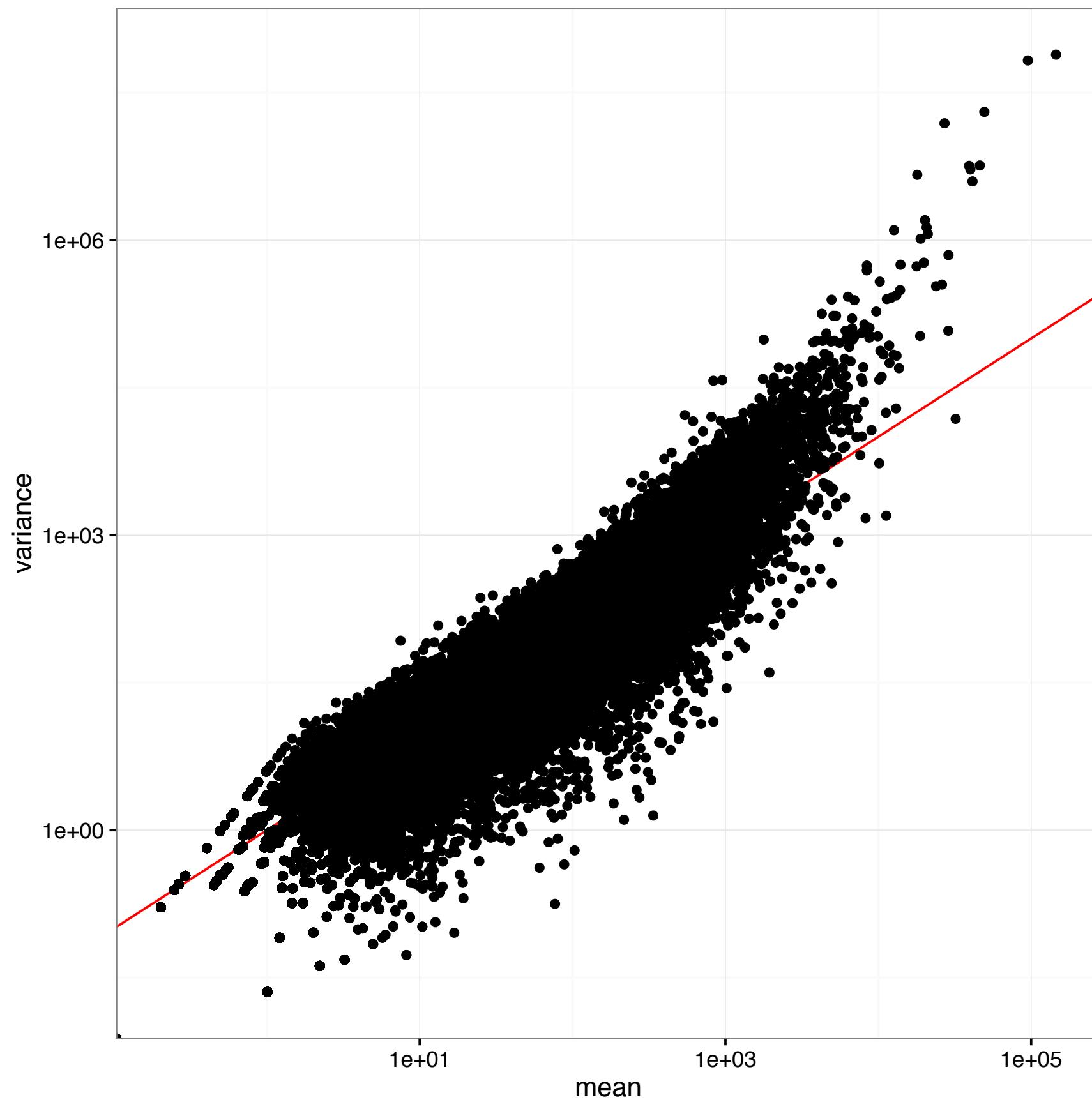


Modeling counts

- **Negative binomial distribution**

- $\text{var}(X) = \mu + \theta\mu^2$
- θ = dispersion
- $\sqrt{\theta}$ = "biological coefficient of variation"
- Allows mRNA proportions to vary across samples
- Captures variability across biological replicates better

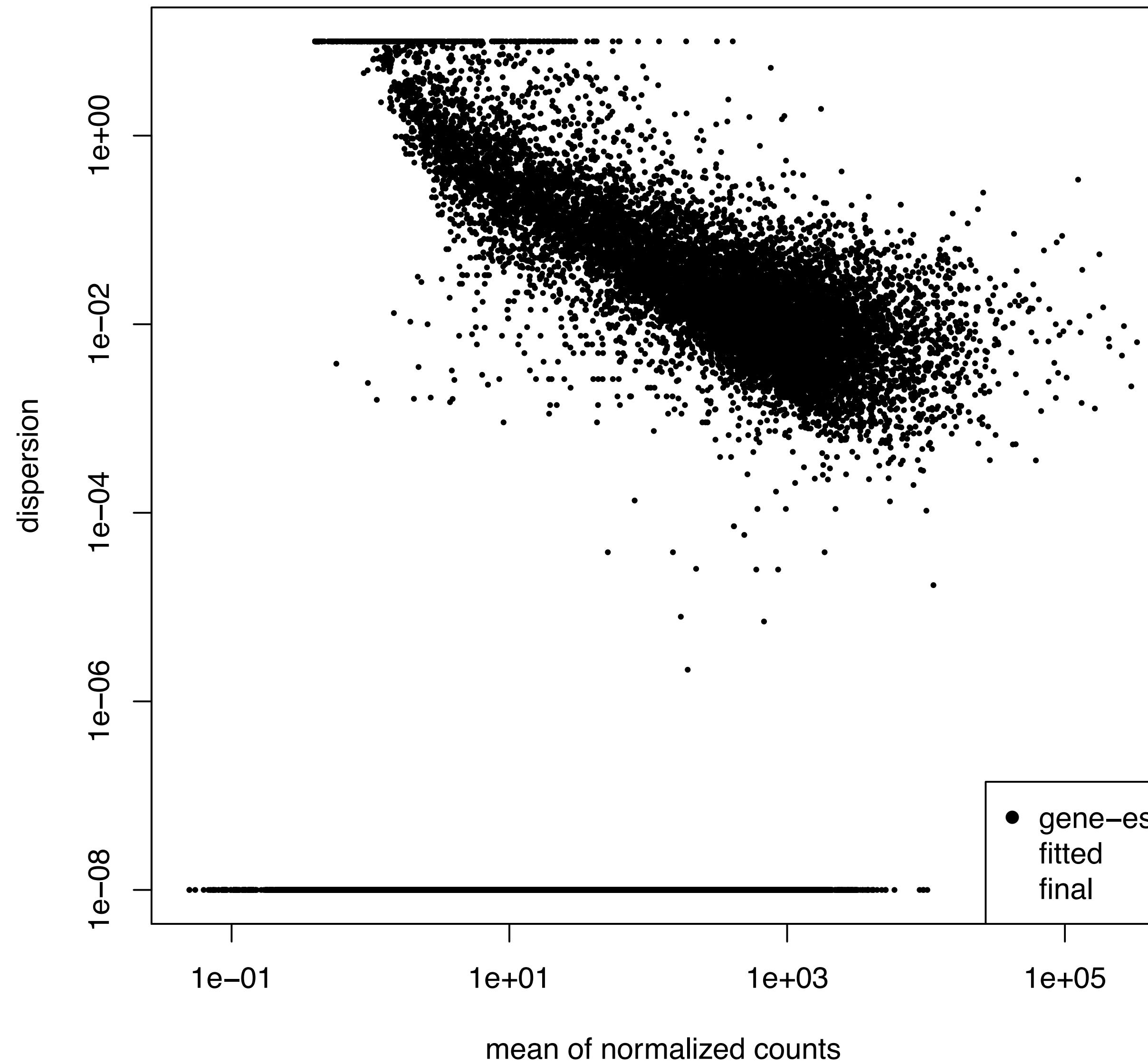
Example from SEQC data, replicates of the same RNA mix



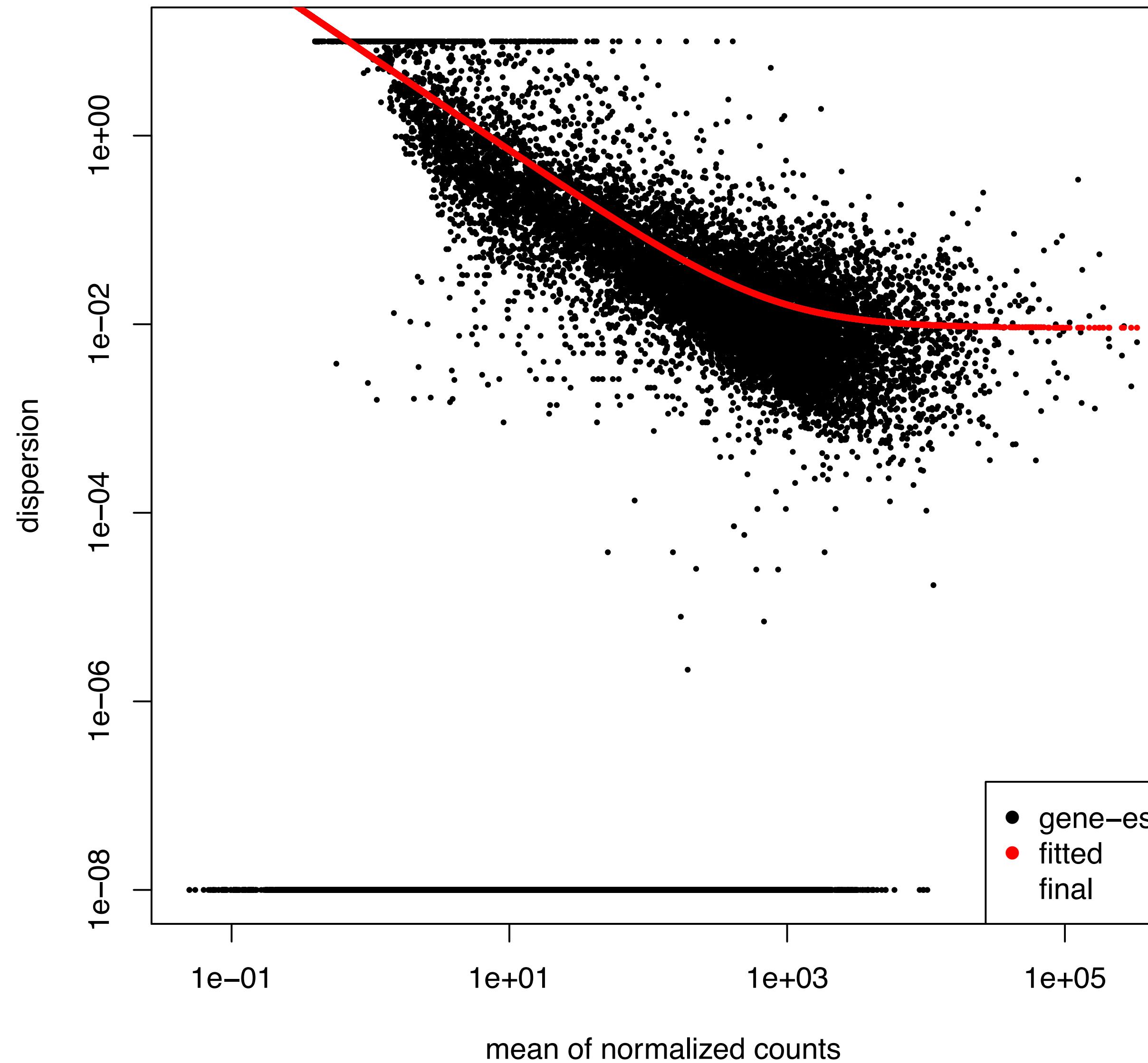
Shrinkage dispersion estimation

- Take advantage of the large number of genes
- Shrink the gene-wise estimates towards a center value defined by the observed distribution of dispersions across
 - all genes (“common” dispersion estimate)
 - genes with similar expression (“trended” dispersion estimate)

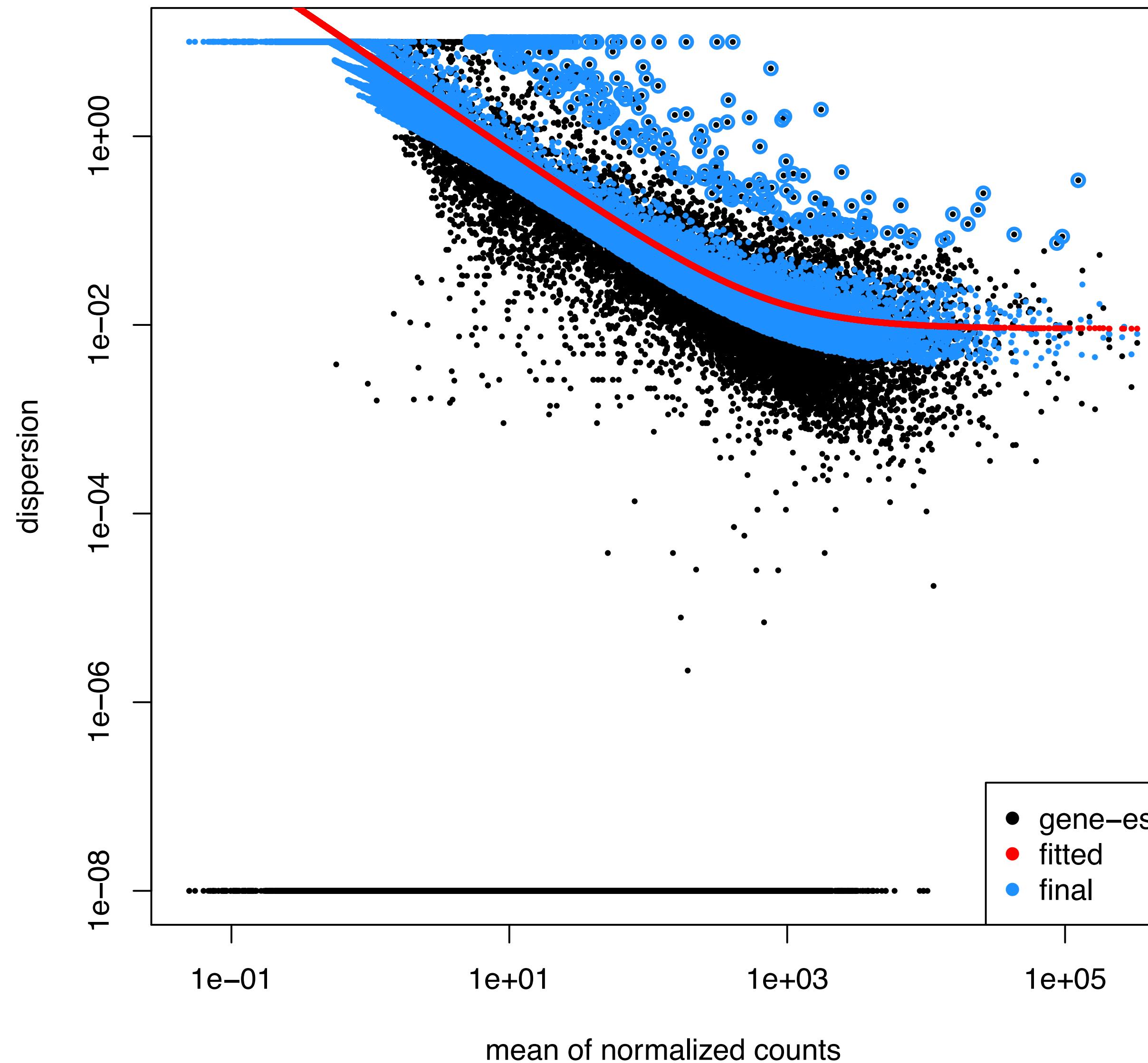
Shrinkage dispersion estimation



Shrinkage dispersion estimation



Shrinkage dispersion estimation



Challenges for RNA-seq data analysis

- Choice of statistical distribution
- Normalization between samples
- Few samples -> difficult to estimate parameters (e.g., variance)
- **High dimensionality (many genes) -> many tests**

Independent filtering

- Idea: filter out genes that have little chance of showing significance (**without** looking at the test results!!!)
- Improves detection power for remaining genes (fewer tests - less strict correction for multiple testing)
- For RNA-seq, typically filter based on expression

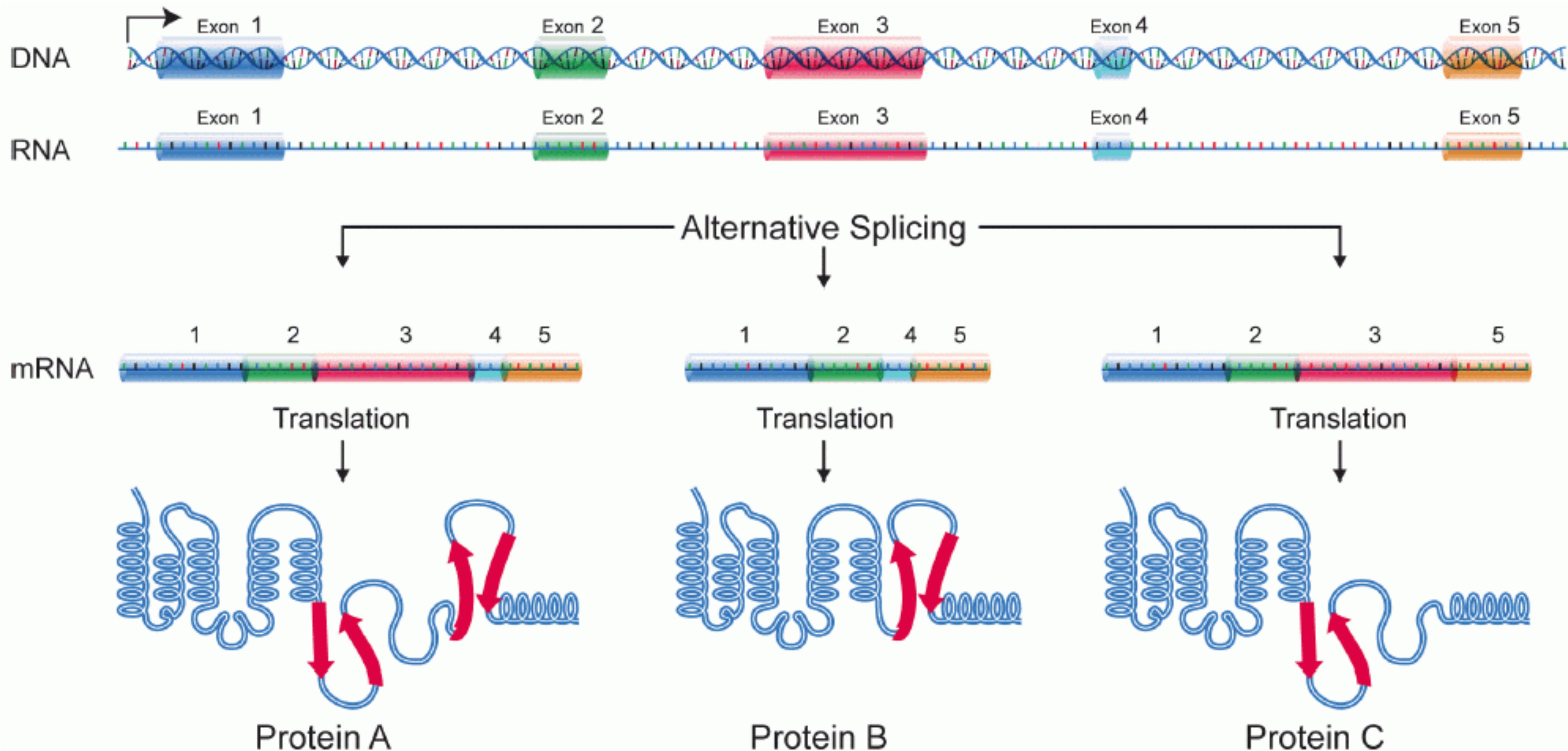
Independent filtering

- DESeq2:
 - filters based on the average normalized counts, using an optimized threshold.
 - p-values for excluded genes are set to NA in results
- edgeR:
 - manual filtering before applying test
 - all remaining genes are tested, and get a p-value

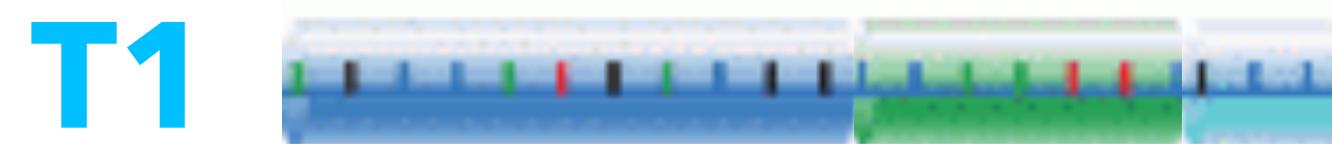
Testing against a threshold

- By default, the null hypothesis is that the log-fold change (β) between conditions is 0
- Both edgeR and DESeq2 can test more general null hypothesis, e.g. $|\beta| \leq 1$
- Useful if very small fold changes are not of interest
- Note that this is **not** the same as setting both a p-value and a fold change threshold on the regular test results!

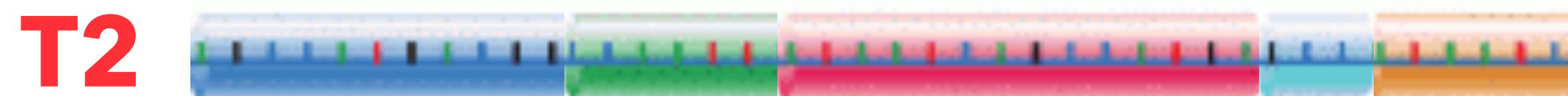
Making use of the transcript abundances



Impact of differential isoform usage on gene-level counts



length = \mathbf{L}

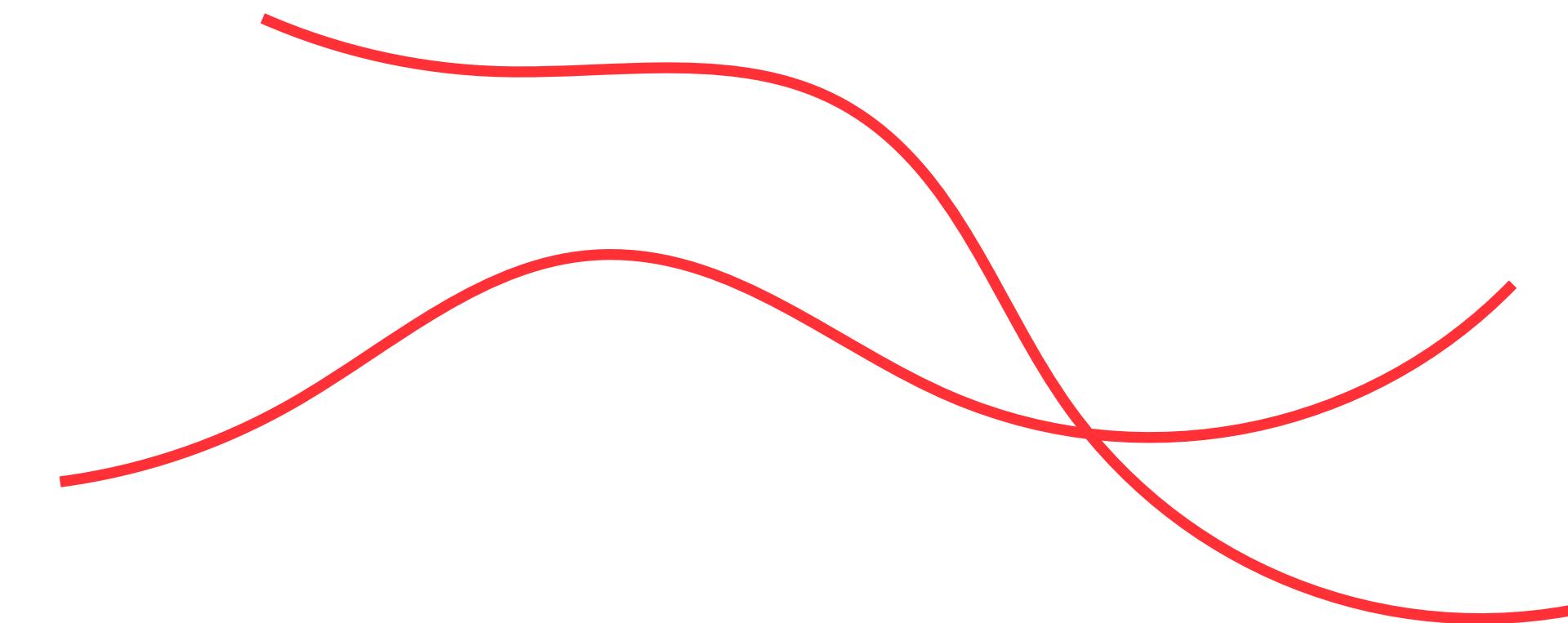


length = $\mathbf{2L}$

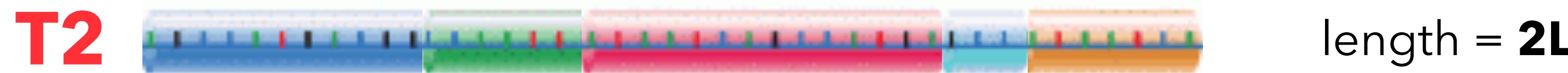
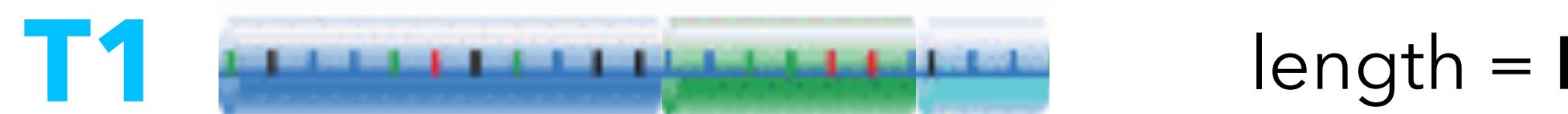
sample 1



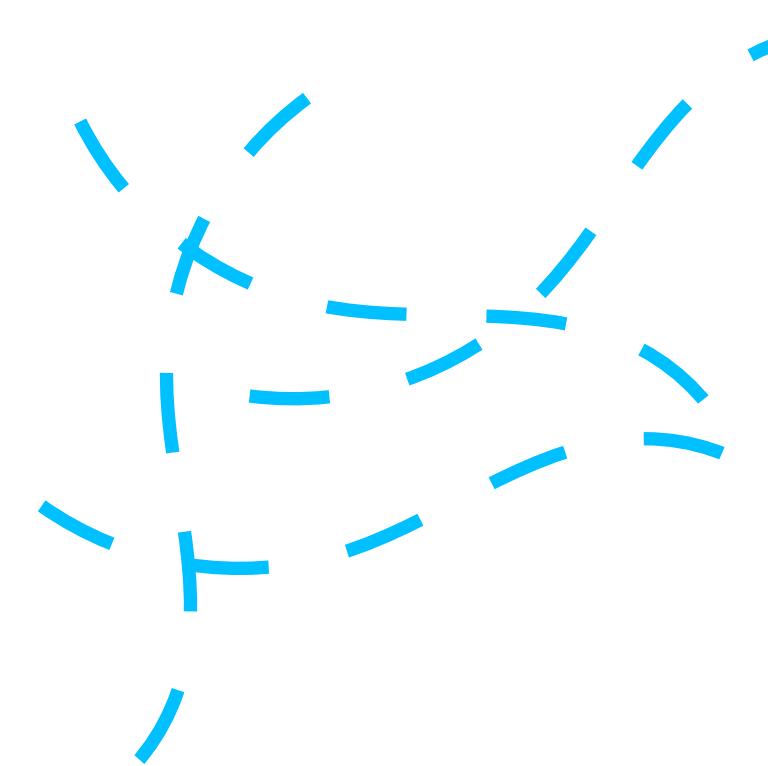
sample 2



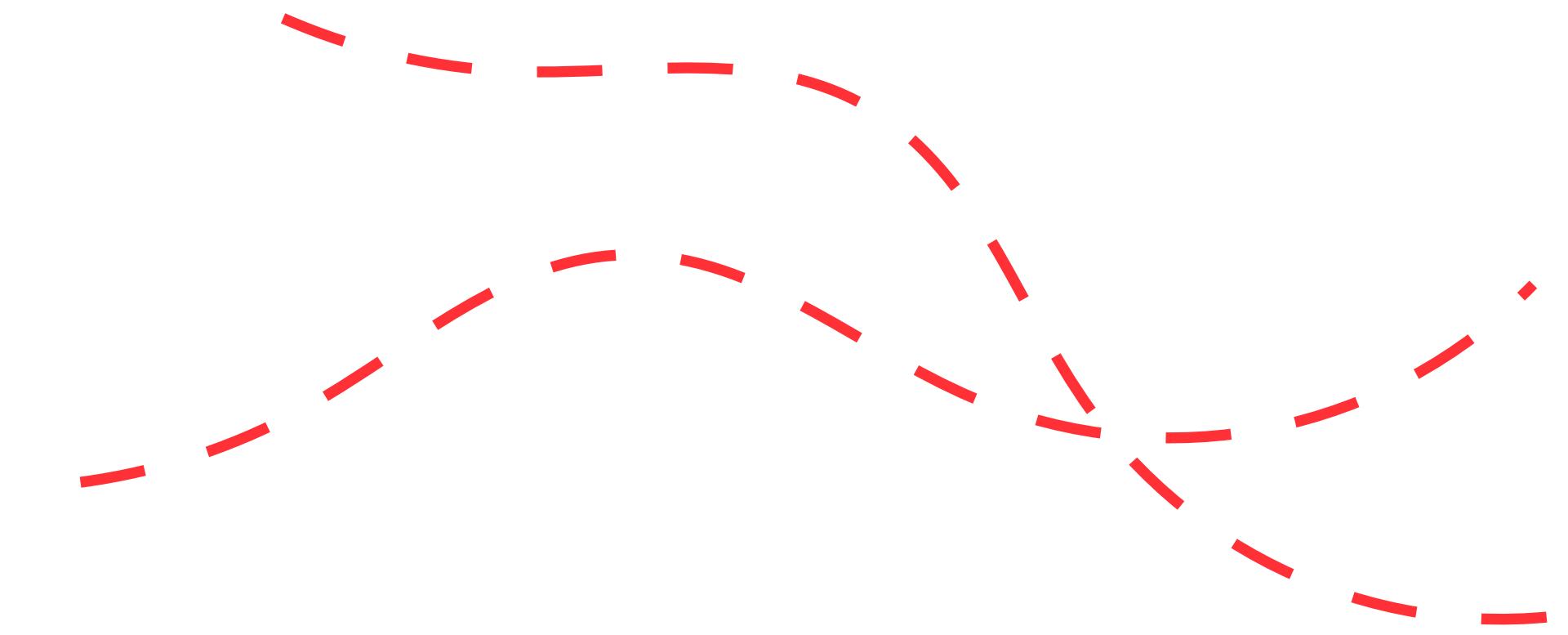
Impact of differential isoform usage on gene-level counts



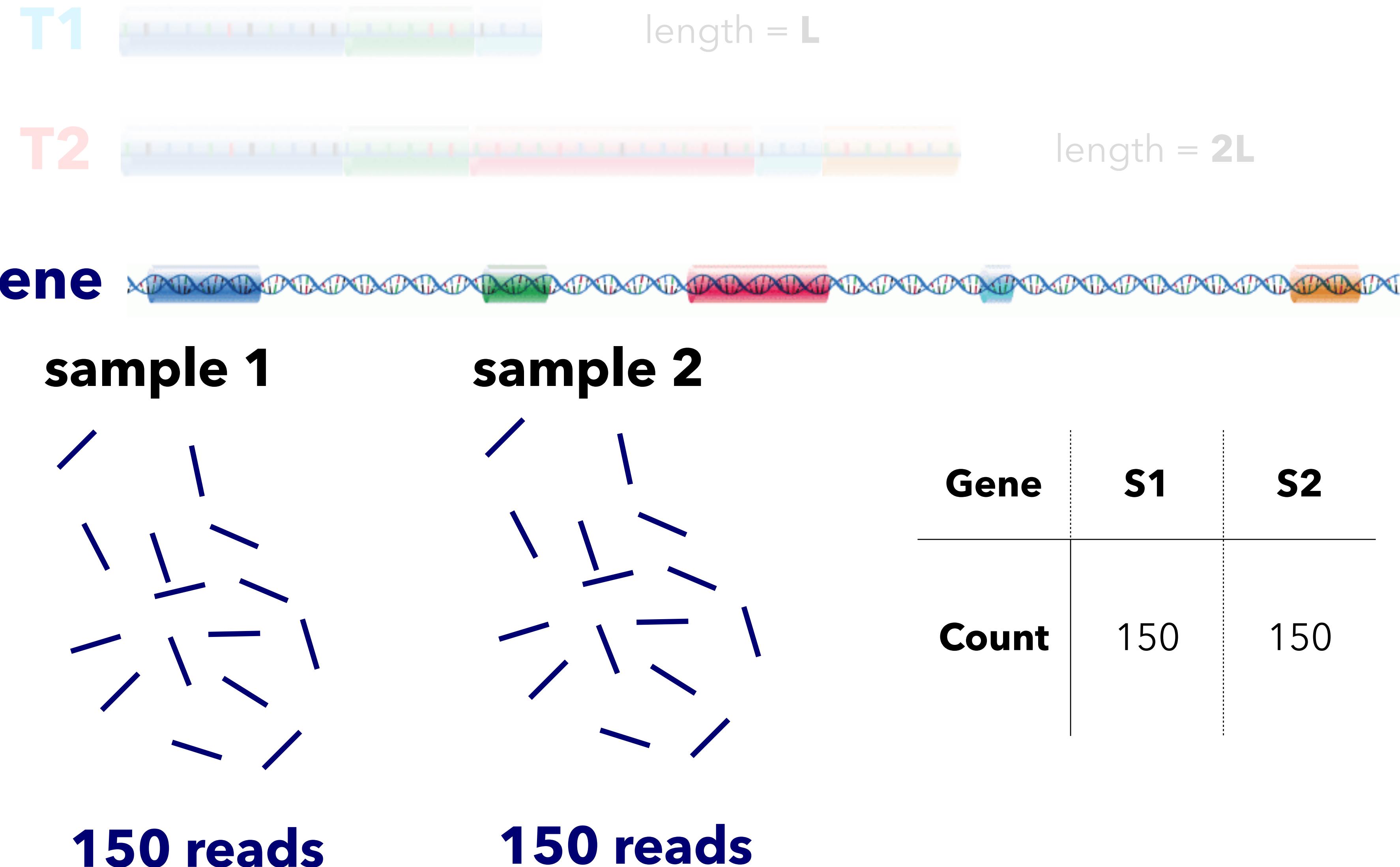
sample 1



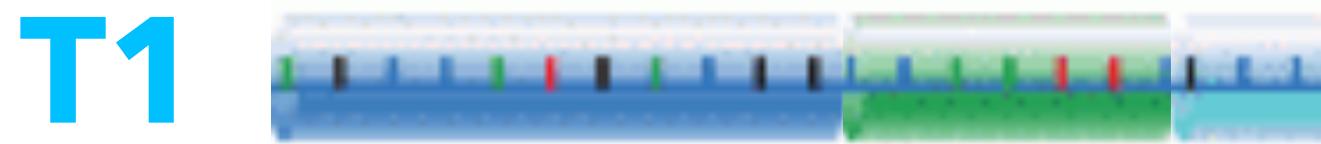
sample 2



Impact of differential isoform usage on gene-level counts



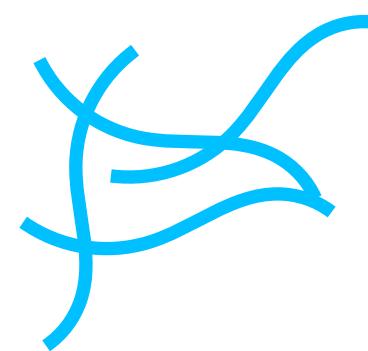
Average transcript lengths



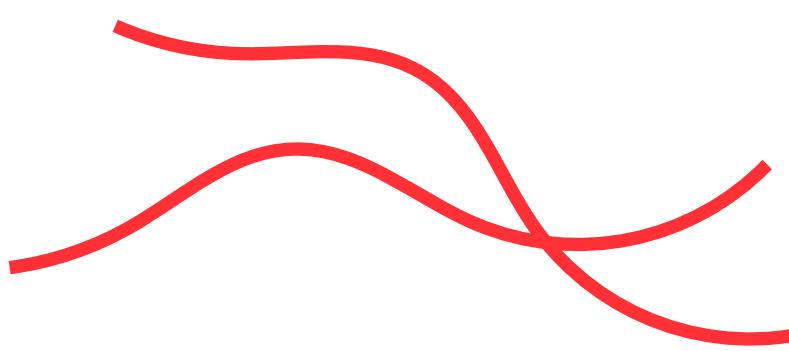
length = **L**



length = **2L**

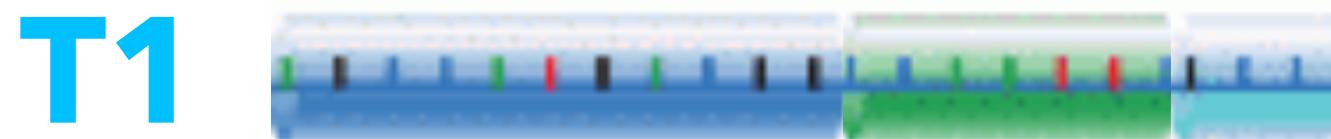


$$ATL_{g1} = 1 \cdot L + 0 \cdot 2L = L$$



$$ATL_{g2} = 0 \cdot L + 1 \cdot 2L = 2L$$

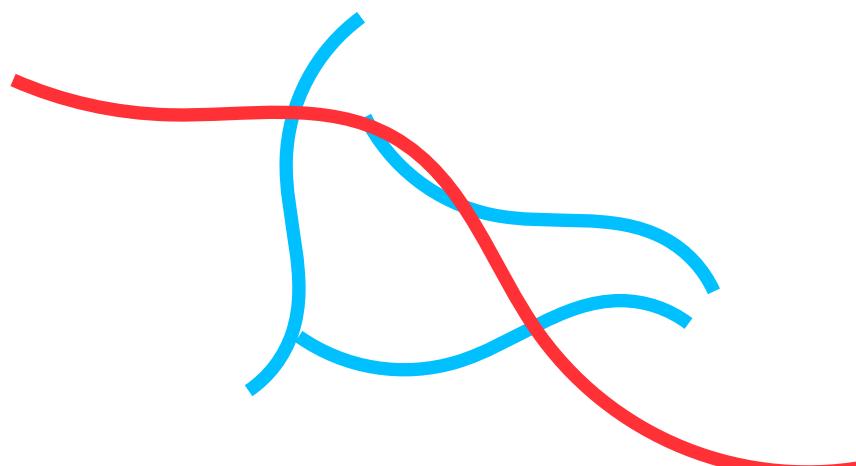
Average transcript lengths



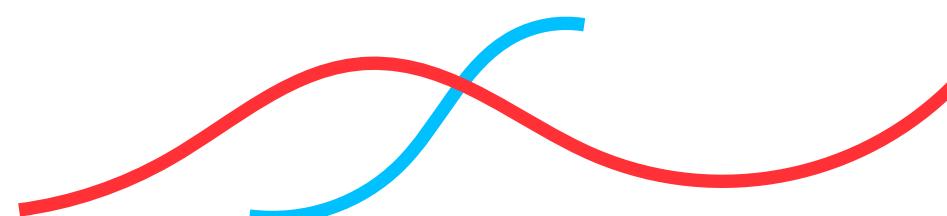
length = **L**



length = **2L**

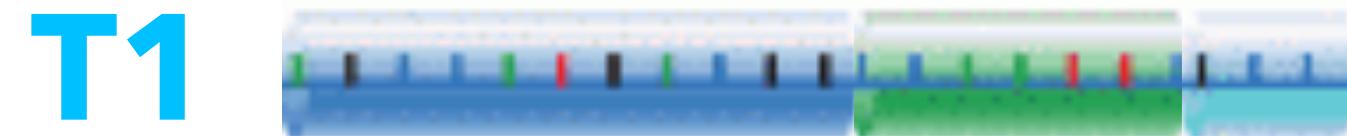


$$ATL_{g1} = 0.75 \cdot L + 0.25 \cdot 2L = 1.25L$$

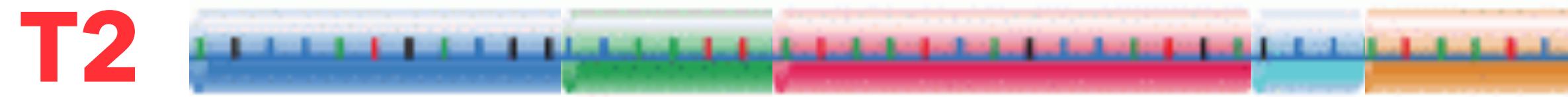


$$ATL_{g2} = 0.5 \cdot L + 0.5 \cdot 2L = 1.5L$$

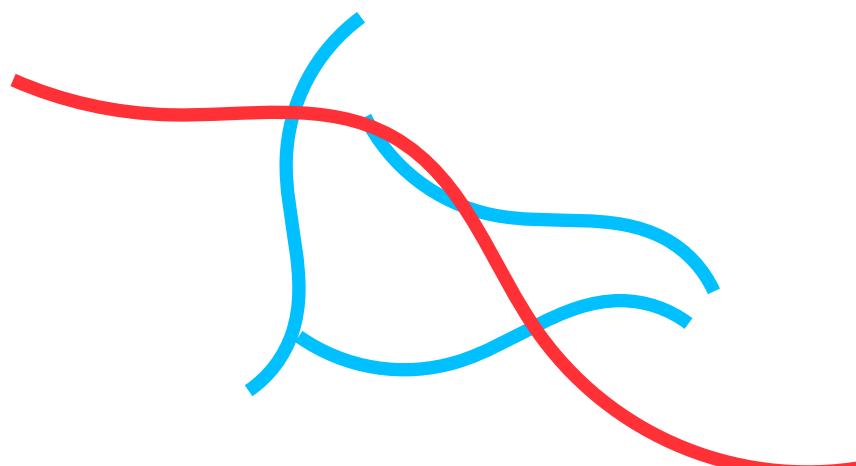
Average transcript lengths



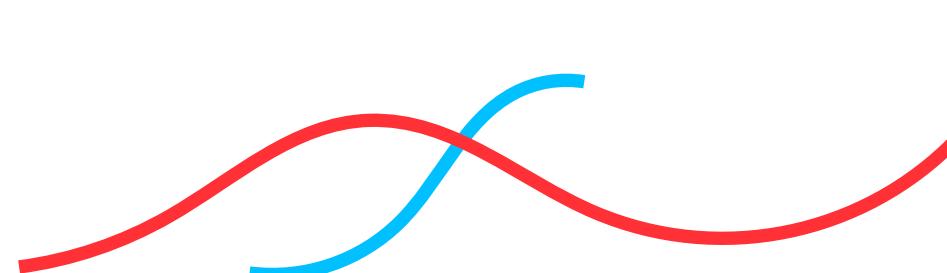
length = **L**



length = **2L**



$$ATL_{g1} = 0.75 \cdot L + 0.25 \cdot 2L = 1.25L$$



$$ATL_{g2} = 0.5 \cdot L + 0.5 \cdot 2L = 1.5L$$

weights obtained from transcript TPM estimates

Offsets (“scaling factors”)

$$C_{ij} \sim NB(\mu_{ij} = s_{ij}q_{ij}, \theta_i)$$

raw count for gene i in sample j

scaling factor

relative abundance

dispersion

The diagram illustrates the components of a negative binomial distribution. It shows the raw count for gene i in sample j as C_{ij} , which follows a negative binomial distribution (NB) with parameters μ_{ij} , $s_{ij}q_{ij}$, and θ_i . The parameter θ_i is labeled as dispersion. The parameter $s_{ij}q_{ij}$ is labeled as the scaling factor. The parameter μ_{ij} is labeled as relative abundance.

- Extend scaling factor for given sample and gene to include the **average length of the transcripts** from the gene that are present in the sample

Offsets ("average transcript lengths")

- Similar to correction factors for library size, but sample- **and** gene-specific
- Transcript abundance levels (TPMs) can be obtained from (e.g.) Salmon or kallisto
- Average transcript length for gene g in sample s :

$$ATL_{gs} = \sum_{i \in g} \theta_{is} \bar{\ell}_{is}, \quad \sum_{i \in g} \theta_{is} = 1$$

$\bar{\ell}_{is}$ = effective length of isoform i (in sample s)

θ_{is} = relative abundance of isoform i in sample s