

# Data Preprocessing

---

Workshop on RNA-Seq

**Roy Francis and Nima Rafati**

NBIS, SciLifeLab

# Raw data

- Raw count table

##	DSSd00_1	DSSd00_2	DSSd00_3	DSSd07_1	DSSd07_2	DSSd07_3
## ENSMUSG000000102693	0	0	0	0	0	0
## ENSMUSG000000064842	0	0	0	0	0	0
## ENSMUSG000000051951	0	1	2	0	3	2
## ENSMUSG000000102851	0	0	0	0	0	0
## ENSMUSG000000103377	0	0	0	0	0	0
## ENSMUSG000000104017	0	0	0	0	0	0

- Metadata

##	SampleName	SampleID	No	Model	Day	Group	Replicate
## DSSd00_1	DSSd00_1	KI_PC1606_01	1	DSS	0	day00	1
## DSSd00_2	DSSd00_2	KI_PC1606_02	2	DSS	0	day00	2
## DSSd00_3	DSSd00_3	KI_PC1606_03	3	DSS	0	day00	3
## DSSd07_1	DSSd07_1	KI_PC1606_13	13	DSS	7	day07	1
## DSSd07_2	DSSd07_2	KI_PC1606_14	14	DSS	7	day07	2
## DSSd07_3	DSSd07_3	KI_PC1606_15	15	DSS	7	day07	3

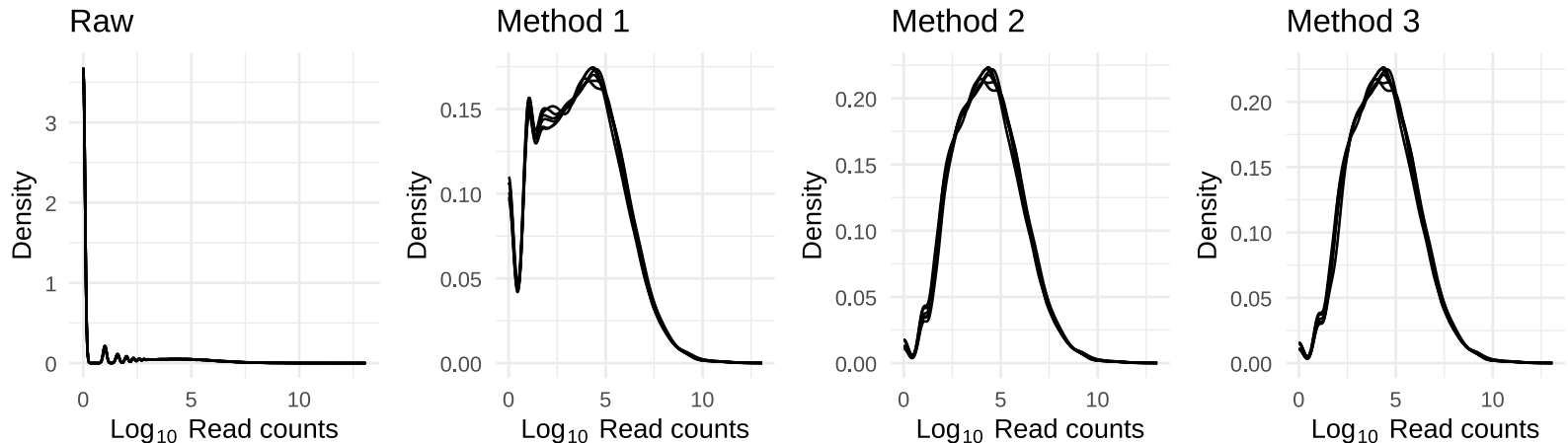
# Filtering

- Remove genes and samples with low counts

```
cf1 <- cr[rowSums(cr>0) >= 3, ] # Keep rows/genes that have at least one read
cf2 <- cr[rowSums(cr>2) >= 3, ] # Keep rows/genes that have at least three reads
cf3 <- cr[rowSums(edgeR::cpm(cr)>5) >= 3, ] # need at least three samples to have cpm >
```

count/read per million (cpm/rpm): a normalized value for sequencing depth.

- Inspect distribution



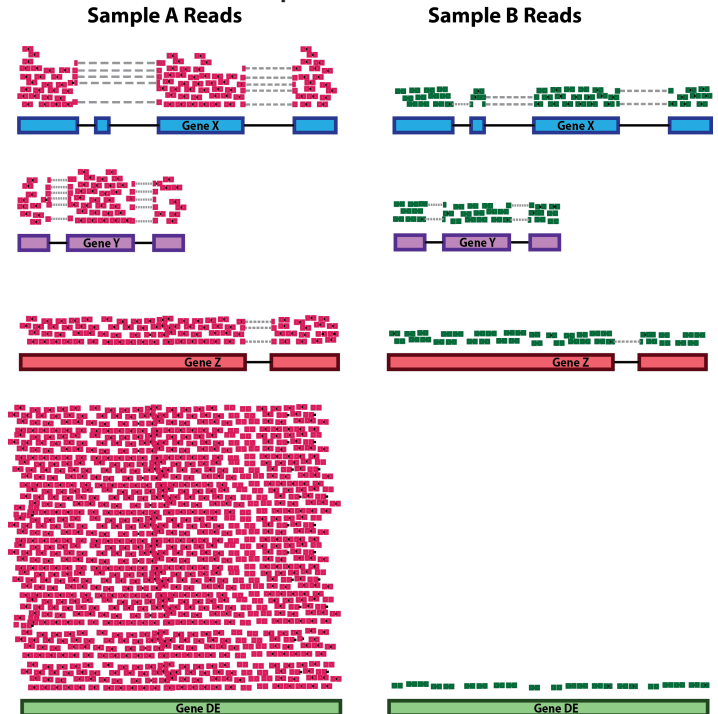
- Inspect the number of rows (genes) available after filtering

```
## Raw: 55487, Method 1: 16099, Method 2: 12656, Method 3: 12496
```

# Normalisation

- Removing technical biases in sequencing data (e.g. sequencing depth and gene length)
- Make counts comparable across samples

- Control for compositional bias



```
##      A B A_tc B_tc
## x  20 6 0.33 0.38
## y  25 6 0.42 0.38
## z  15 4 0.25 0.25
```

```
##      A B A_tc B_tc
## x   20 6 0.12 0.33
## y   25 6 0.16 0.33
## z   15 4 0.09 0.22
## de 100 2 0.62 0.11
```

# Normalisation

- Make counts comparable across features (genes). It can be useful for gene to gene comparisons.

## Sample A Reads



##	counts	gene_length	norm_counts
## x	50	10	5
## y	25	5	5

- Bring counts to a human-friendly scale

## Normalisation by library size

- Assumes total expression is the same under different experimental conditions
- Methods include TC, RPKM, FPKM, TPM
- RPKM, FPKM and TPM control for sequencing depth and gene length
- Total number of RPKM/FPKM normalized counts for each sample will be different, therefore, you cannot compare the normalized counts for each gene equally between samples.
- TPM is proportional to RPKM and enables better comparison between samples because total per sample sums to equal value

##	A	B	len	A_rpm	B_rpm	A_rpk	B_rpk	A_rpk	B_rpk	A_tpm	B_tpm
## x	20	6	2000	408163	222222	204081.5	111111.0	10.00	3.0	493827	153846
## y	25	6	4000	510204	222222	127551.0	55555.5	6.25	1.5	308642	76923
## z	4	15	1000	81633	555556	81633.0	555556.0	4.00	15.0	197531	769231
## sum	49	27	7000	1000000	1000000	413265.5	722222.5	20.25	19.5	1000000	1000000

rpm = cpm.

## Normalisation by distribution

- Assumes technical effects are same for DE and non-DE genes
- Assumes number of over and under-expressed genes are roughly same across conditions
- Corrects for compositional bias
- Methods include Q, UQ, M, RLE, TMM, MRN
- `edgeR::calcNormFactors()` implements TMM, TMMwsp, RLE & UQ
- `DESeq2::estimateSizeFactors()` implements relative log expression (RLE)
- Does not correct for gene length
- `geTMM` is gene length corrected TMM

```
##      A  B  len  ref A_ratio B_ratio      A_mrn      B_mrn
## x 20  6 2000 10.95    1.83    0.55 10.928962 10.90909
## y 25  6 4000 12.25    2.04    0.49 13.661202 10.90909
## z  4 15 1000  7.75    0.52    1.94  2.185792 27.27273
```

## Normalisation by testing

- A more robust version of normalisation by distribution
- A set of non-DE genes are detected through hypothesis testing
- Tolerates a larger difference in number of over and under expressed genes between conditions
- Methods include PoissonSeq, DEGES

## Normalisation using Controls

- Assumes controls are not affected by experimental condition and technical effects are similar to all other genes
- Useful in conditions with global shift in expression
- Controls could be house-keeping genes or spike-ins
- Methods include RUV, CLS

## Stabilizing variance

- Variance is stabilised across the range of mean values
- Methods include VST, RLOG, VOOM
- For use in exploratory analyses. Not for DE.
- `vst()` and `rlog()` functions from *DESeq2*
- `voom()` function from *Limma* converts data to normal distribution



## Recommendations

- Most tools use a mix of many different normalisations
- For DGE using DGE R packages (DESeq2, edgeR, Limma etc), use **raw counts**
- For visualisation (PCA, clustering, heatmaps etc), use VST or RLOG
- For own analysis with gene length correction, use TPM (maybe geTMM?)
- Custom solutions: spike-ins/house-keeping genes



# Thank you. Questions?

R version 4.1.3 (2022-03-10)

Platform: x86\_64-pc-linux-gnu (64-bit)

OS: Ubuntu 18.04.6 LTS

Built on : 🏠 14-Mar-2023 at 🕒 22:22:02

2023 • SciLifeLab • NBIS