

RNA-seq differential expression analysis

Dario Righelli

19/09/2019

What we will cover

We will cover differential expression analysis of RNA-seq data in R/Bioconductor.

We will start from a matrix of gene-level read counts.

We will cover the two most popular packages, DESeq2 and edgeR.

I will also show you how to deal with unwanted variation using the RUVSeq package.

What we will not cover

We will not talk about the preprocessing of RNA-seq data, i.e., what we do to obtain the gene-level read counts.

These steps are usually done with stand-alone software outside R.

We will not talk about isoform-level analysis and alternative splicing.

We will focus on gene-level differential expression.

Where to find these slides

https://github.com/drighelli/rnaseq_meetup

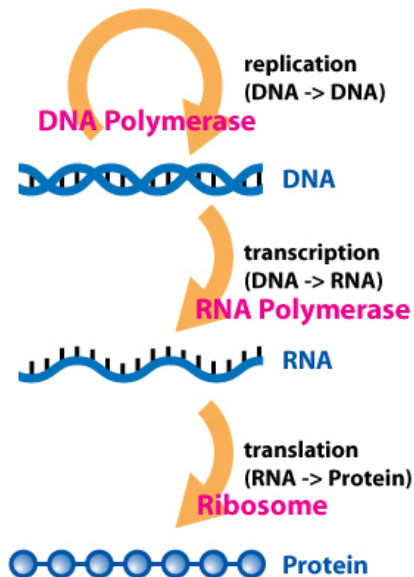
Where to find additional resources

- ▶ The edgeR user guide
<https://bioconductor.org/packages/edgeR>
- ▶ The DESeq2 vignette
<https://bioconductor.org/packages/DESeq2>
- ▶ The F1000 Research Bioconductor gateway
<https://f1000research.com/gateways/bioconductor>
- ▶ <https://support.bioconductor.org>

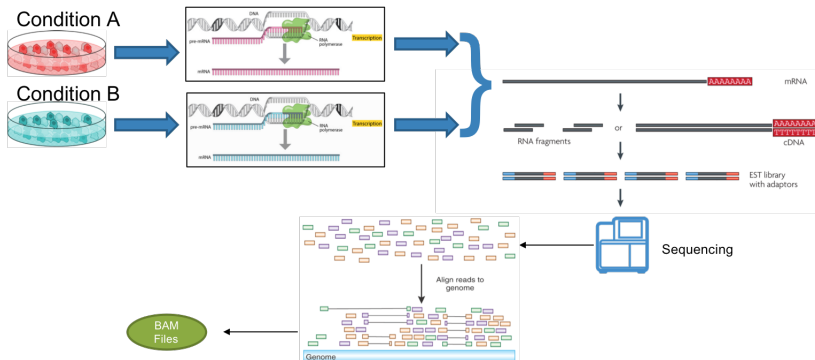
Where to find me

- ▶ Istituto per le Applicazioni del Calcolo “M. Picone” - Consiglio Nazionale delle Ricerche, via P. Castellino 111, Napoli
- ▶ email: d.righelli@na.iac.cnr.it
- ▶ github: <https://github.com/drighelli>

From RNA to gene-level read counts



From RNA to gene-level read counts



From RNA to gene-level read counts

##		CC3	CC5	CC6	CC7	CC8	FC3	FC5	FC6	FC7	FC8	RT3
##	ENSMUSG000000000001	2034	2232	1253	2024	1510	994	1703	1796	1502	2145	1600
##	ENSMUSG0000000000028	81	93	77	91	85	106	81	84	70	95	121
##	ENSMUSG0000000000037	52	59	28	52	36	12	40	34	41	56	26
##	ENSMUSG0000000000049	15	32	15	18	14	49	10	18	11	24	21
##	ENSMUSG0000000000056	3125	3256	2175	3283	2553	1638	2276	2900	2223	3179	2504
##	ENSMUSG0000000000058	1412	1324	819	1243	668	446	821	815	786	1646	817
##		RT5	RT6	RT7	RT8							
##	ENSMUSG0000000000001	1734	1834	1982	1316							
##	ENSMUSG0000000000028	92	102	102	60							
##	ENSMUSG0000000000037	44	46	40	45							
##	ENSMUSG0000000000049	22	17	11	9							
##	ENSMUSG0000000000056	3045	3106	3441	1940							
##	ENSMUSG0000000000058	945	1031	1170	990							

The Poisson Model

When statisticians see counts, they immediately think about Simeon Poisson.



The Poisson Model

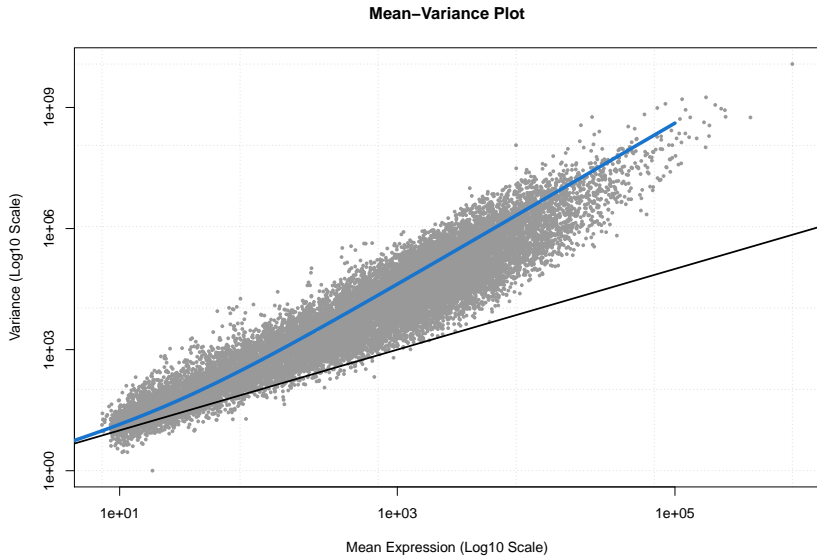
The Poisson distribution naturally arises from binomial calculations, with a large number of trials and a small probability.

It has a rather stringent assumption: **the variance is equal to the mean!**

$$\text{Var}(Y_{ij}) = \mu_{ij}$$

In real datasets the variance is greater than the mean, a condition known as **overdispersion**.

A real example



The Negative Binomial Model

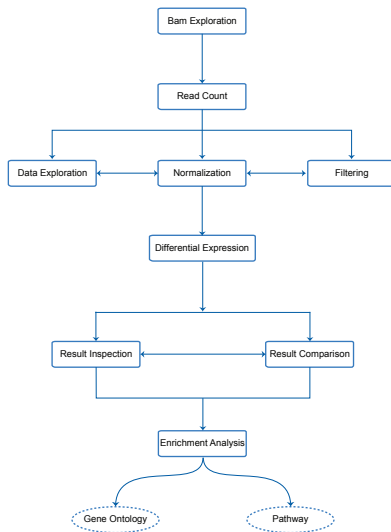
A generalization of the Poisson model is the negative binomial, that assumes that the variance is a quadratic function of the mean.

$$\text{Var}(Y_{ij}) = \mu_{ij} + \phi_j \mu_{ij}^2$$

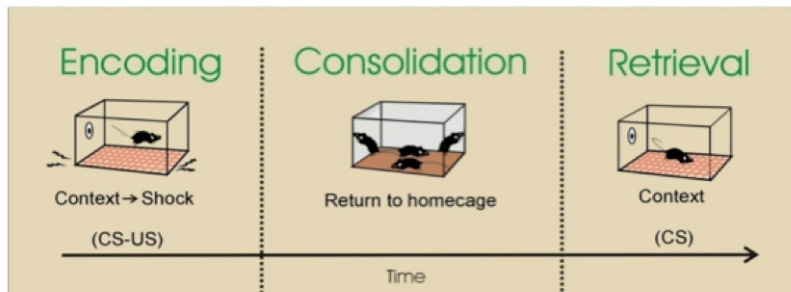
where ϕ is called the **dispersion parameter**.

Both edgeR and DESeq2 assume that the data is distributed as a negative binomial.

A typical analysis workflow



An example dataset



	Dissections						Dissections	
	10 am	10:30 am	4 pm	10 pm	10 am		10 am	10:30 am
Fear Conditioning		FC30	FC4	FC12	FC24	Test		RT30
Control		CC30	CC4	CC12				

An example dataset

- ▶ C57BL/6J adult male mice (2 months of age).
- ▶ Five animals per group: fear conditioning (FC), memory retrieval (RT), and controls (CC).
- ▶ Illumina 100bp paired-end reads mapped to the mouse genome (mm9) using GMAP/GSNAP.
- ▶ Ensembl (release 65) gene counts obtained using HTSeq.