We are here

1. RNA-Seq
   - Counting
   - Normalization
   - Practical Session

RNA-Seq: Expression value estimation

RNA-Seq

Genome

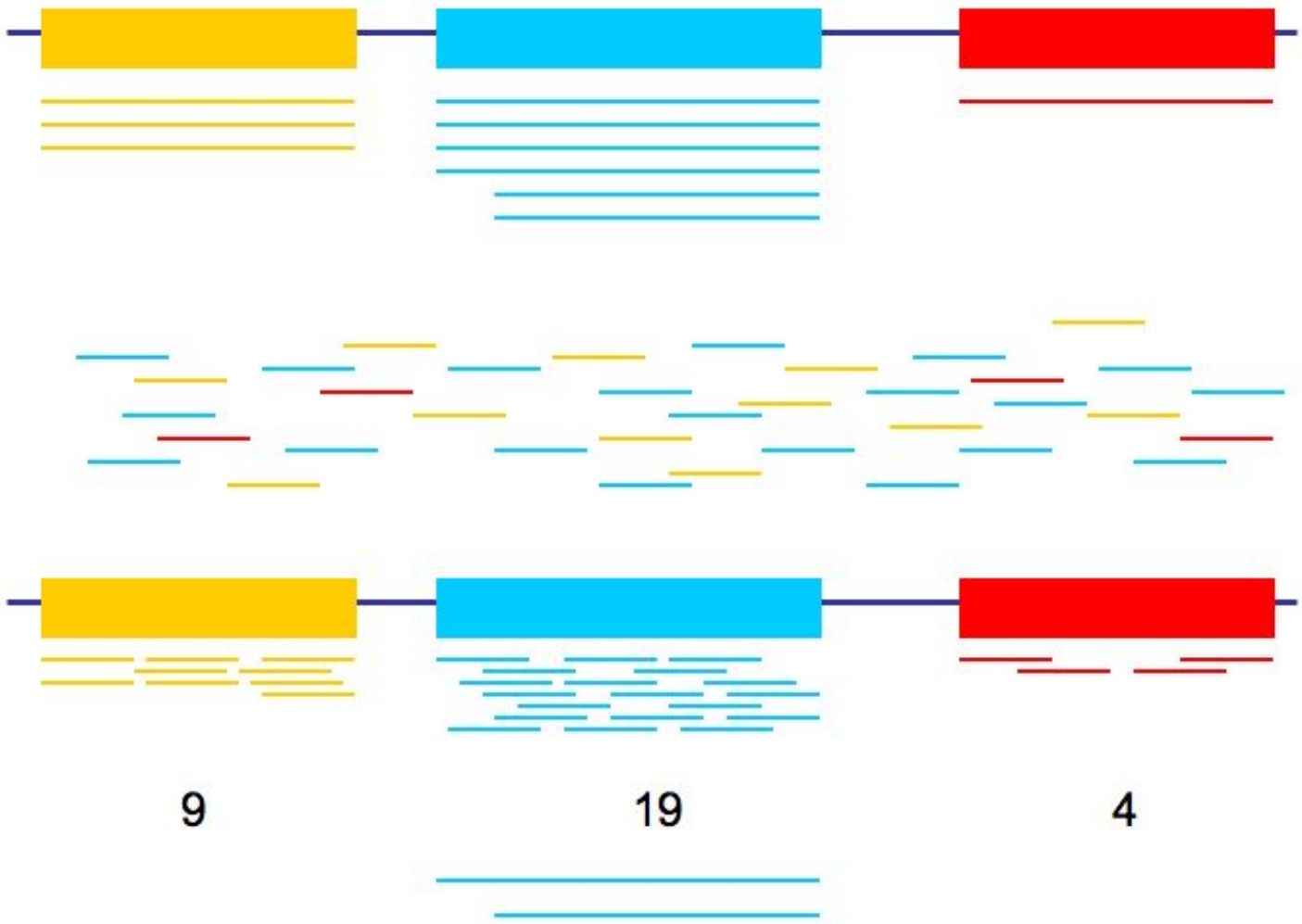
mRNA

Reads

Mapping

Expression count

Altern. Splicing



RNA-Seq: From reads to expression values

- RNA-Seq experiments result in large numbers of reads
- Reads are mapped against a reference genome
- We are interested in the expression strength of each gene.
- We could just count the reads for each gene.

Assumption

Expression strength $\sim$ number of reads

This requires

- Expression strength $=$ number of transcripts
- Number of transcripts $\overset{?}{\sim}$ number of reads

... but it is not so simple.

RNA-Seq: From reads to expression values

Counting using HTSeq (Anders *et al.*, HTSeq – A Python framework to work with high-throughput sequencing data Bioinformatics 2014):

RNA-Seq: From reads to expression values

Counting using HTSeq (Anders *et al.*, HTSeq – A Python framework to work with high-throughput sequencing data Bioinformatics 2014):

Given a file with aligned sequencing reads and a list of genomic features, the next step is to estimate the expression value of each feature.

$\Rightarrow$ first count how many reads map to each feature.

Feature: an interval (i.e., a range of positions) on a chromosome or a union of such intervals.

Example: a feature is a gene, and a gene is the union of all its exons.

RNA-Seq: From reads to expression values

Counting using HTSeq:

- Takes a gtf / gff file with gene model annotations and SAM/BAM file as input

RNA-Seq: From reads to expression values

Counting using HTSeq:

- Takes a gtf / gff file with gene model annotations and SAM/BAM file as input
- Counting: For each position i in the read, a set $S(i)$ is defined as the set of all features overlapping position i.
  It then considers (in the union mode) the union of all sets $S(i)$. If a read overlaps more than one feature (e.g. two genes), then it is not counted. If the union is empty, the read is counted as no_feature'.

RNA-Seq: From reads to expression values

Counting using HTSeq:

- Takes a gtf / gff file with gene model annotations and SAM/BAM file as input
- Counting: For each position i in the read, a set $S(i)$ is defined as the set of all features overlapping position i.
  It then considers (in the union mode) the union of all sets $S(i)$. If a read overlaps more than one feature (e.g. two genes), then it is not counted. If the union is empty, the read is counted as no_feature'.
- Outputs a table with counts for each feature as well as some statistics

Further common file formats

- Wiggle files are used to represent the coverage data (ie., continuous data
  Example:

```
variableStep chrom=chr2
300701 12.5
300702 12.5
300703 12.5
300704 12.5
300705 12.5
```

RNA-Seq: Sources of bias

Data from RNA-Seq experiments contains several sources of bias:

- Sample bias
  - Amount of RNA in the sample — v_{sample}
  - Number of reads produced by sequencing — N_{reads}

RNA-Seq: Sources of bias

Data from RNA-Seq experiments contains several sources of bias:

- Sample bias
  - Amount of RNA in the sample — v_{sample}
  - Number of reads produced by sequencing — N_{reads}
- Gene bias
  - longer genes produce more reads per transcript — $\ell(G)$

RNA-Seq: Sources of bias

Data from RNA-Seq experiments contains several sources of bias:

- Sample bias
  - Amount of RNA in the sample — v_{sample}
  - Number of reads produced by sequencing — N_{reads}
- Gene bias
  - longer genes produce more reads per transcript — $\ell(G)$
  - PCR efficiency bias — $e_{\text{PCR}}(G)$

RNA-Seq: Sources of bias

Data from RNA-Seq experiments contains several sources of bias:

- Sample bias
  - Amount of RNA in the sample — v_{sample}
  - Number of reads produced by sequencing — N_{reads}
- Gene bias
  - longer genes produce more reads per transcript — $\ell(G)$
  - PCR efficiency bias — $e_{\text{PCR}}(G)$
  - Sequence-dependent sequencing bias — $s_{\text{technology}}(G)$

RNA-Seq: Sources of bias

Data from RNA-Seq experiments contains several sources of bias:

- Sample bias
  - Amount of RNA in the sample — v_{sample}
  - Number of reads produced by sequencing — N_{reads}
- Gene bias
  - longer genes produce more reads per transcript — $\ell(G)$
  - PCR efficiency bias — $e_{\text{PCR}}(G)$
  - Sequence-dependent sequencing bias — $s_{\text{technology}}(G)$

$\Rightarrow$ The number of reads per gene G is a combination of these factors:

$$\text{reads}(G) \sim v_{\text{sample}} \cdot N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

The goal of RNA-Seq is to find the true expression strength of each gene.

RNA-Seq: Normalizing data to reduce bias (I)

$$\text{reads}(G) \sim v_{\text{sample}} \cdot N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

The influence of v_{sample} is two-fold:

1. Smaller sample volumes result in fewer reads overall.

$\rightarrow$ we can treat v_{sample} as a part of N_{reads} during normalization.

RNA-Seq: Normalizing data to reduce bias (I)

$$\text{reads}(G) \sim v_{\text{sample}} \cdot N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

The influence of v_{sample} is two-fold:

1. Smaller sample volumes result in fewer reads overall.

$\rightarrow$ we can treat v_{sample} as a part of N_{reads} during normalization.

2. If sample volume is too small, low-expressed genes are not represented by any reads at all, and are missed.

$\rightarrow$ This problem can not be solved by data normalization. The sample volume has to be chosen correctly during wet-lab work!

RNA-Seq: Normalizing data to reduce bias (II)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

Since we know the N_{reads} for each sample, we can easily correct for differences between samples.

For two samples A, B:

$$\text{Expression}'(G_A) \sim \text{Expression}(G_A) : N_{\text{reads}}(A)$$

$$\text{Expression}'(G_B) \sim \text{Expression}(G_B) : N_{\text{reads}}(B)$$

And differential expression can then be computed as

$$\frac{\text{Expression}'(G_A)}{\text{Expression}'(G_B)}$$

RNA-Seq: Normalizing data to reduce bias (III)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

This is known as the RPM (reads per million) measure:

Reads per million — **RPM**

$$\text{rpm}(G) = \text{reads}(G) \cdot \frac{10^6}{N_{\text{reads}}}$$

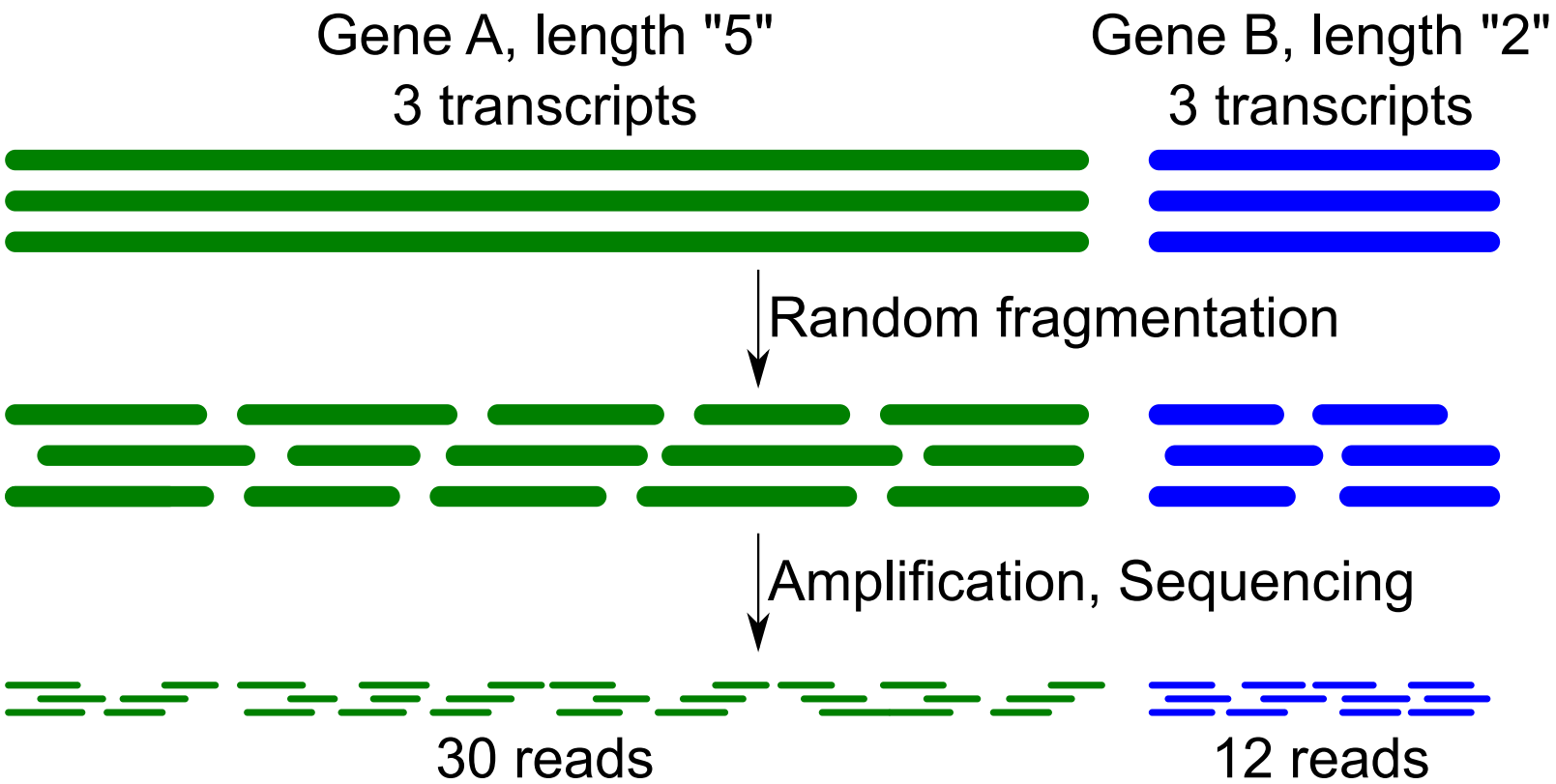
$\text{rpm}(G)$ is the expression of G relative to the total expression in the sample.

However, this is not without problems, as we will see in the next slide.

RNA-Seq: Normalizing data to reduce bias (IV)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

The influence of $\ell(G)$ is straight-forward:



RNA-Seq: Normalizing data to reduce bias (V)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

Since we know $\ell(G)$, we can correct for it. This leads to a new measure:

Reads per million per kilobase of exon model — **RPKM**

$$\text{rpkm}(G) = \text{reads}(G) \cdot \frac{10^6}{N_{\text{reads}}} \cdot \frac{1}{\ell(G)}$$

where $\ell(G)$ is the number of bases in all exons of G.

If a gene has several transcript variants (due to alternative splicing), each variant has its own (possibly different) rpkm value.

RNA-Seq: Normalizing data to reduce bias (VI)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot e_{\text{PCR}}(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

Sequence amplification by PCR is not unbiased.

Factors that influence amplification:

- GC rich sequences
- Sequences that form secondary structures
- ...

- We will treat PCR bias as part of technology-dependent bias.

RNA-Seq: Normalizing data to reduce bias (VII)

$$\text{reads}(G) \sim N_{\text{reads}} \cdot \ell(G) \cdot s_{\text{technology}}(G) \cdot \text{Expression}(G)$$

Different technologies have different biases.

- Illumina sequencing prefers longer reads

→ long genes with small changes appear more interesting than short genes with strong changes.
There is no good solution for this problem.
One could take a random region of e.g. 250 bp from each gene.

- GC bias: Expression of low-GC regions can appear twice as strong
Usually GC content of genes is not very different in one organism.

→ New third generation sequencing technologies eliminate the PCR step.

RNA-Seq: Normalization measures

So far we have discussed

- **RPM** – correcting for sample volume and sequencing depth
- **RPKM** – also correcting for transcript length

We mention a further measure, a variant of RPKM:

Fragments per kilobase exon model per million reads — **FPKM**

FPKM is used by *Cufflinks*. Built on a statistical model, it reflects the likelihood that one would observe the fragments in the experiment, given the proposed abundances on the transcripts.

Quantile normalization: Motivation

RPM, RPKM and FPKM normalize by the total number of reads, N_{reads}.

This is based on two assumptions:

The **percentage** of reads assigned to a gene G
reflects its expression strength.

and

A change in that percentage between two samples
reflects a change in expression strength.

But: N_{reads} is dominated by a small number of highly-expressed genes.

50% of all reads are due to 5% of the expressed genes
in a typical human sample.

Quantile normalization: Example 1

Consider two sequenced samples A and B.

- Both contain the same number of reads.
- Our simple organism has three genes: X, Y, and Z.
- Gene expression is unchanged except for one highly-expressed gene
- How is this reflected in the percentages?

	Reads			Percentages		
Gene	A	B	interpretation	A	B	interpretation
X	10	10	no change	10	5	**50% down**
Y	40	40	no change	40	20	**50% down**
Z	50	150	300% up	50	75	**50% up**
	100	200				

Quantile normalization: Example 2

Consider two sequenced samples A and B with the same number of reads.

- This organism has four genes: X, Y, and $\hat{X}$, $\hat{Y}$.
- $\hat{X}$ and $\hat{Y}$ are only active in sample B

	Reads			Percentages		
Gene	A	B	interpretation	A	B	interpretation
X	20	20	no change	20	10	**50% down**
Y	80	80	no change	80	40	**50% down**
$\hat{X}$	0	20	switched on	0	10	switched on
$\hat{Y}$	0	80	switched on	0	40	switched on
	100	200				

Quantile normalization: Conclusion

From the examples, we can draw the conclusion that...

> ...the number of reads expected to map to a gene is not only dependent on the **expression level** and **length of the gene**, but also the **composition of the RNA population** that is being sampled. Thus, if a large number of genes are unique to, or highly expressed in, one experimental condition, the sequencing 'real estate' available for the remaining genes in that sample is decreased.
>
> *Robinson & Oshlack, Genome Biology 2010, 11:R25*

Quantile normalization as proposed solution

The quantile normalization procedure by Bullard *et al.* works as follows:

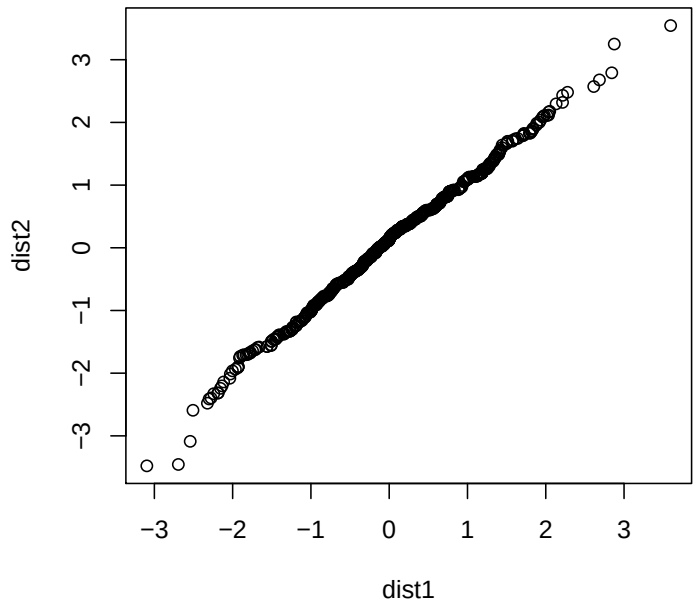
1. Count the reads for each gene
2. Ignore all genes with 0 reads
3. Compute the upper quartile, q_{75}
   q_{75} is the number of reads needed such that
   75% of the (nonzero) genes have less than q_{75} reads.
4. Divide all read counts by q_{75}.

Bullard et al. Bioinformatics 2010, 11:94

Visualizing quantile normalized data

After quantile normalisation one can visualise the distribution of the expression values in each sample using either

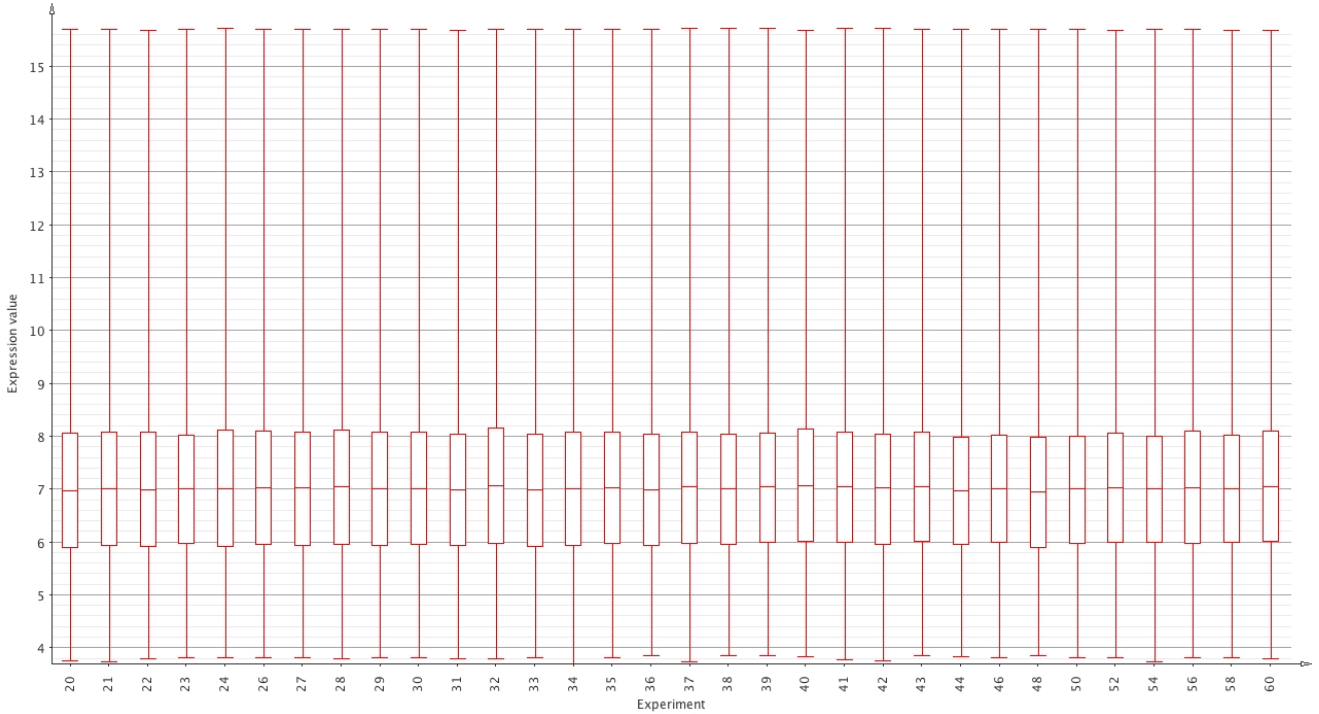
- A qq-plot: it plots the quantiles of two distributions as a scatter plot against each other. Two very similar distributions have almost a scatter along the diagonal:



Visualizing quantile normalized data

After quantile normalisation one can visualise the distribution of the expression values in each sample using either

- A boxplot: shows for each experiment the box plot of the expression values. Quantile normalised data should have very similar looking box plots.



Practical Session

Questions?

Practical Session

Questions?

Now off to the fifth practical session

Learning Objectives of Practical Session

- Run HTSeq-count
- Load .bam files after RNA-seq mapping with STAR into Mayday SeaSight
- Learn how to normalise count data in Mayday