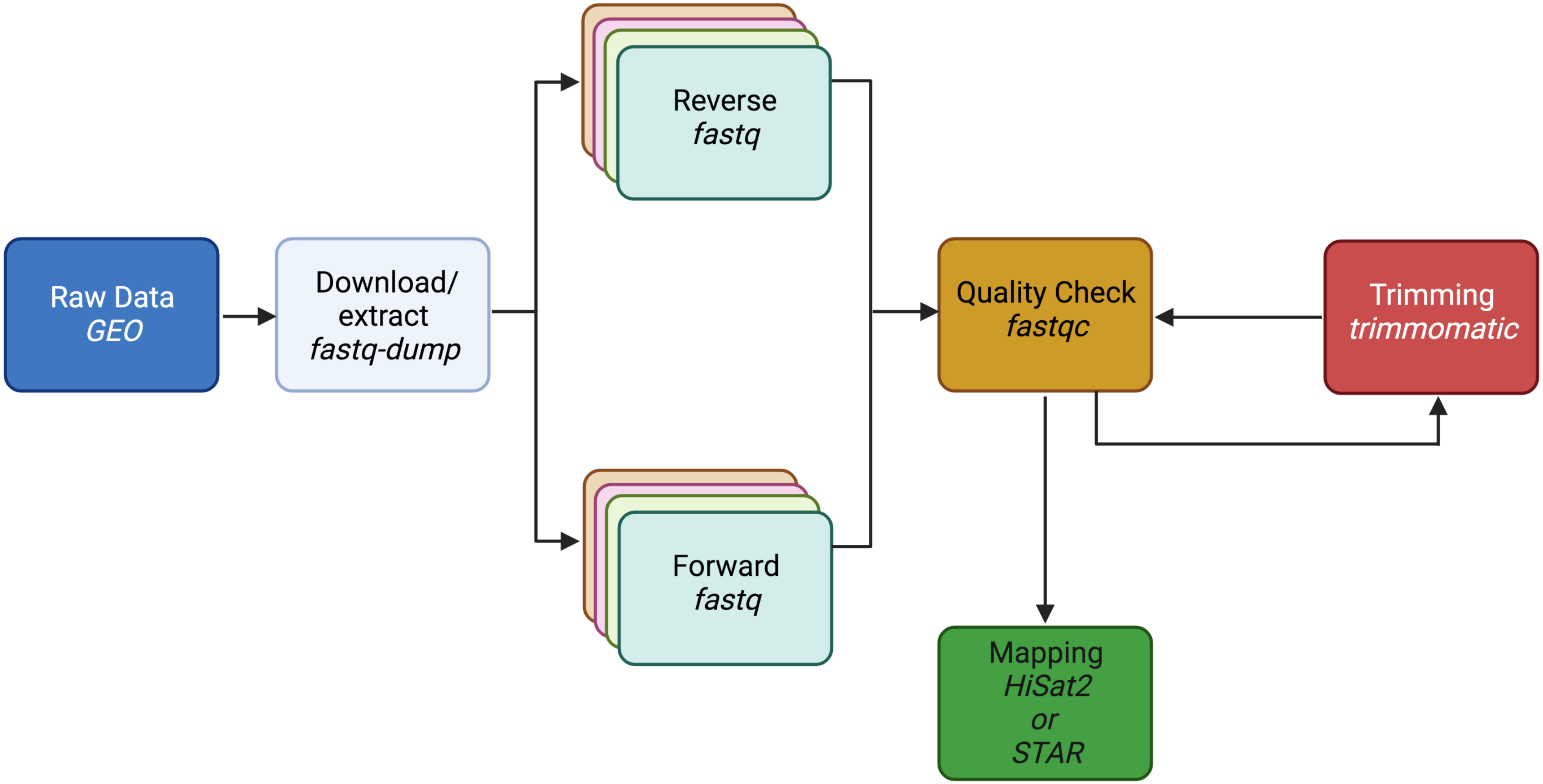
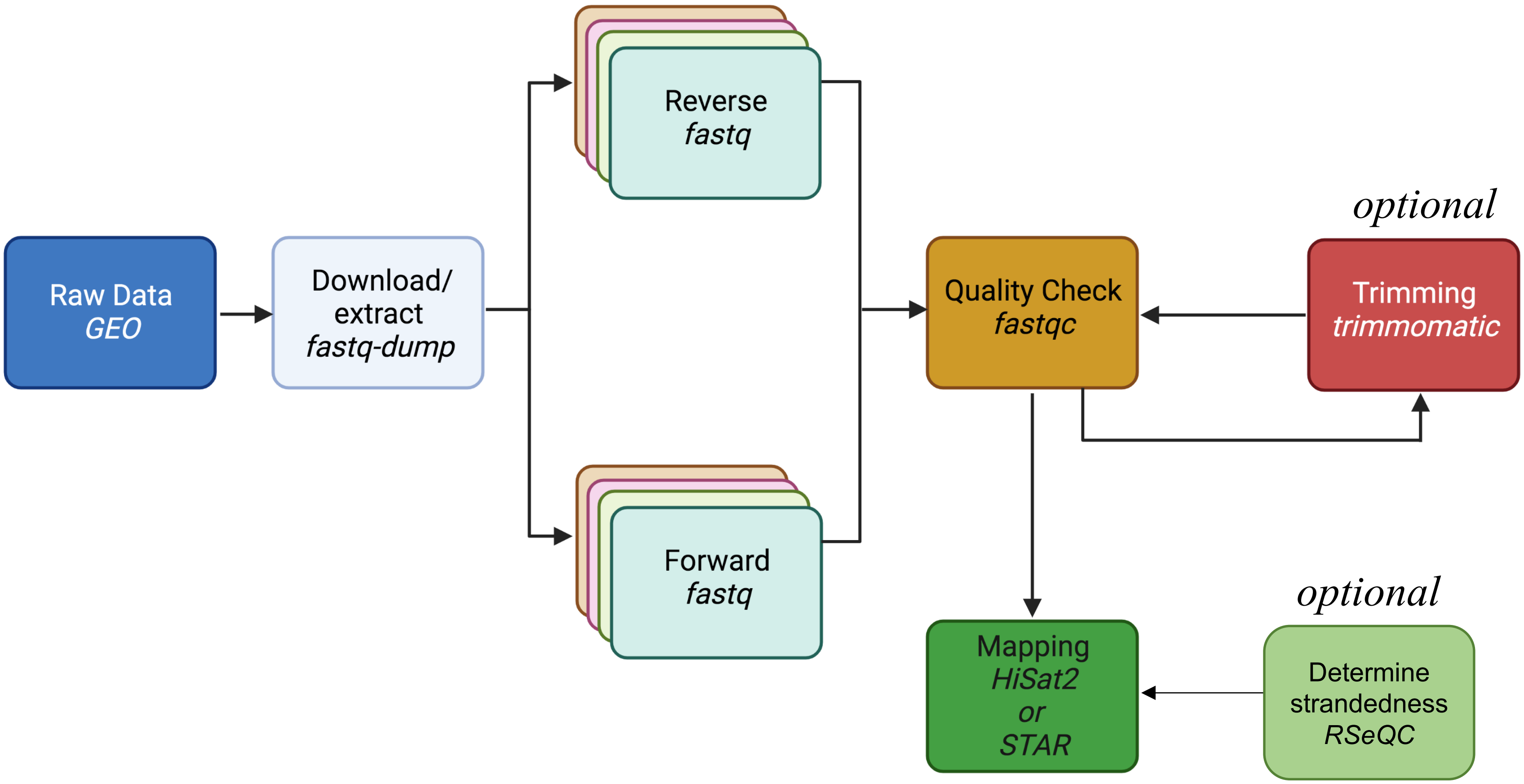
RSeQC & HTSeq

March 9th, 2023

MARCH 2023

SUN	MON	TUE	WED	THU	FRI	SAT
			1	2	3	4
5	6	7	8	9 **today**	10	11
12	13 **HW#8**	14 **SPRING BREAK**	15	16	17 St. Patrick's Day	18
19	20	21 **R intro** *Meant for beginners*	22	23	24	25
26	27	28	29	30	31 **HW#9** *Sooner the better*	


Raw Data
GEO
Download/
extract
fastq-dump
Reverse
fastq
Forward
fastq
Quality Check
fastqc
Trimming
trimmomatic
Mapping
HiSat2
or
STAR


Raw Data
GEO
Download/
extract
fastq-dump
Reverse
fastq
Forward
fastq
Quality Check
fastqc
optional
Trimming
trimmomatic
Mapping
HiSat2
or
STAR
optional
Determine
strandedness
RSeQC

Pre & post QC

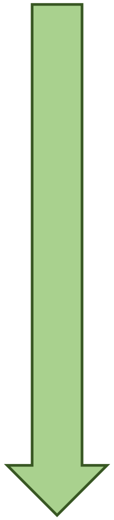
- Before mapping:
  - *How to identify and remove reads with low base calls?*
  - *How to identify and remove reads with linkers/adaptors ?*
  - *How to screen for potential species/vector/ribosomal contamination?*
  - *How is your library complexity?*
- After Mapping:
  - *What is percentage of reads aligned?*
  - *Is your sequencing library stranded or unstranded?*
  - *How could I know if the high expression levels are due to real biological signal or to PCR artefacts?*

QC Programs

- **raw reads** QC
  - adapter/primer/other contaminating and over-represented sequences
  - sequencing quality
  - GC distributions
  - duplication levels

- **aligned reads** QC
  - % (uniquely) aligned reads
  - % exonic vs. intronic/intergenic
  - gene diversity
  - gene body coverage
  - strandedness

Pre-alignment: FastQC, fastp



Post-alignment: RSeQC, QoRTs

2 popular post-alignment QC packages

RSeQC

- commands and outputs are not standardized
- most results can be integrated with the help of MultiQC

http://rseqc.sourceforge.net/

QoRTs

- less clunky than RSeQC
- offers many checks that are already part of FastQC
- stratifies genes by expression strength for many checks
- output is not easily integrated with MultiQC

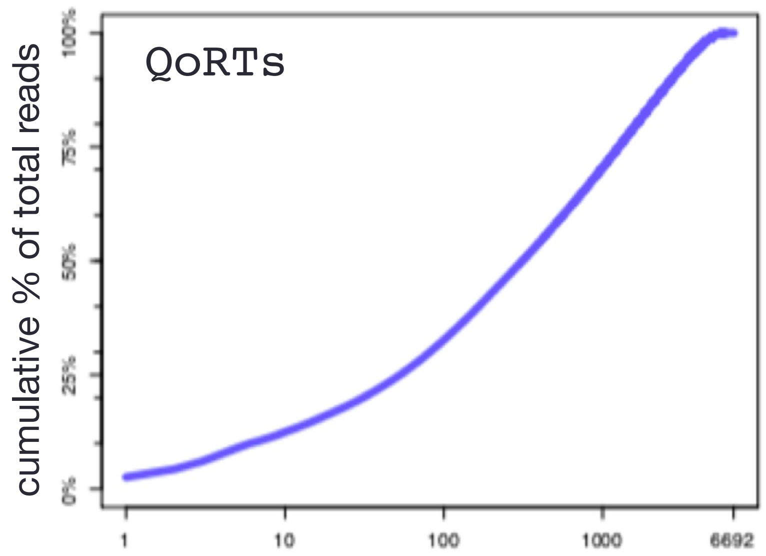
https://hartleys.github.io/QoRTs/

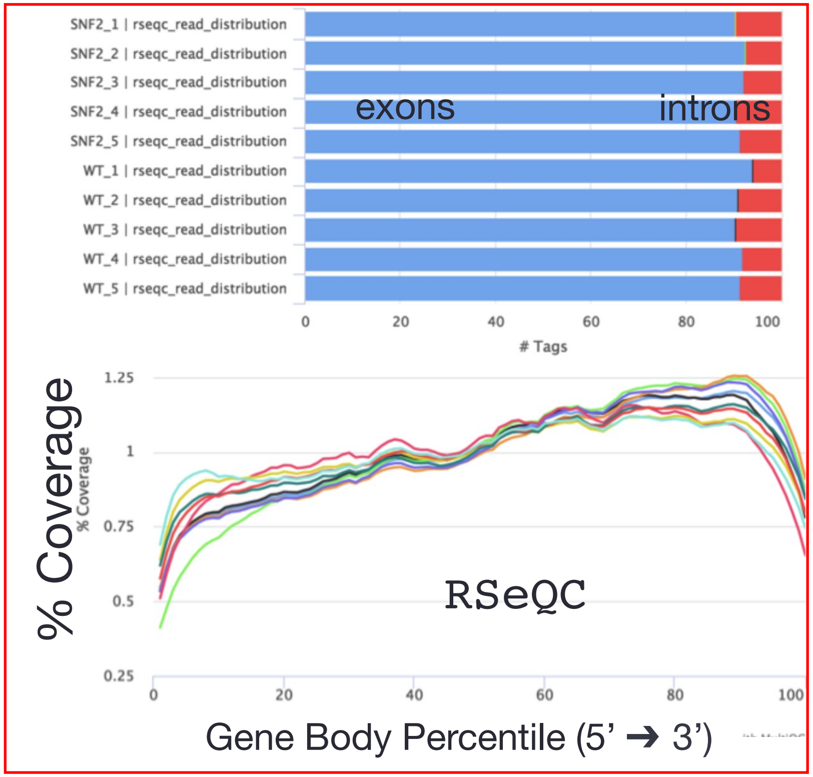
Typical biases of RNA-seq

- lack of **gene diversity**:
  - dominance of rRNAs, tRNAs or other highly abundant transcripts

- **read distribution**
  - high intron coverage: incomplete poly(A) enrichment
  - many intergenic reads: gDNA contamination

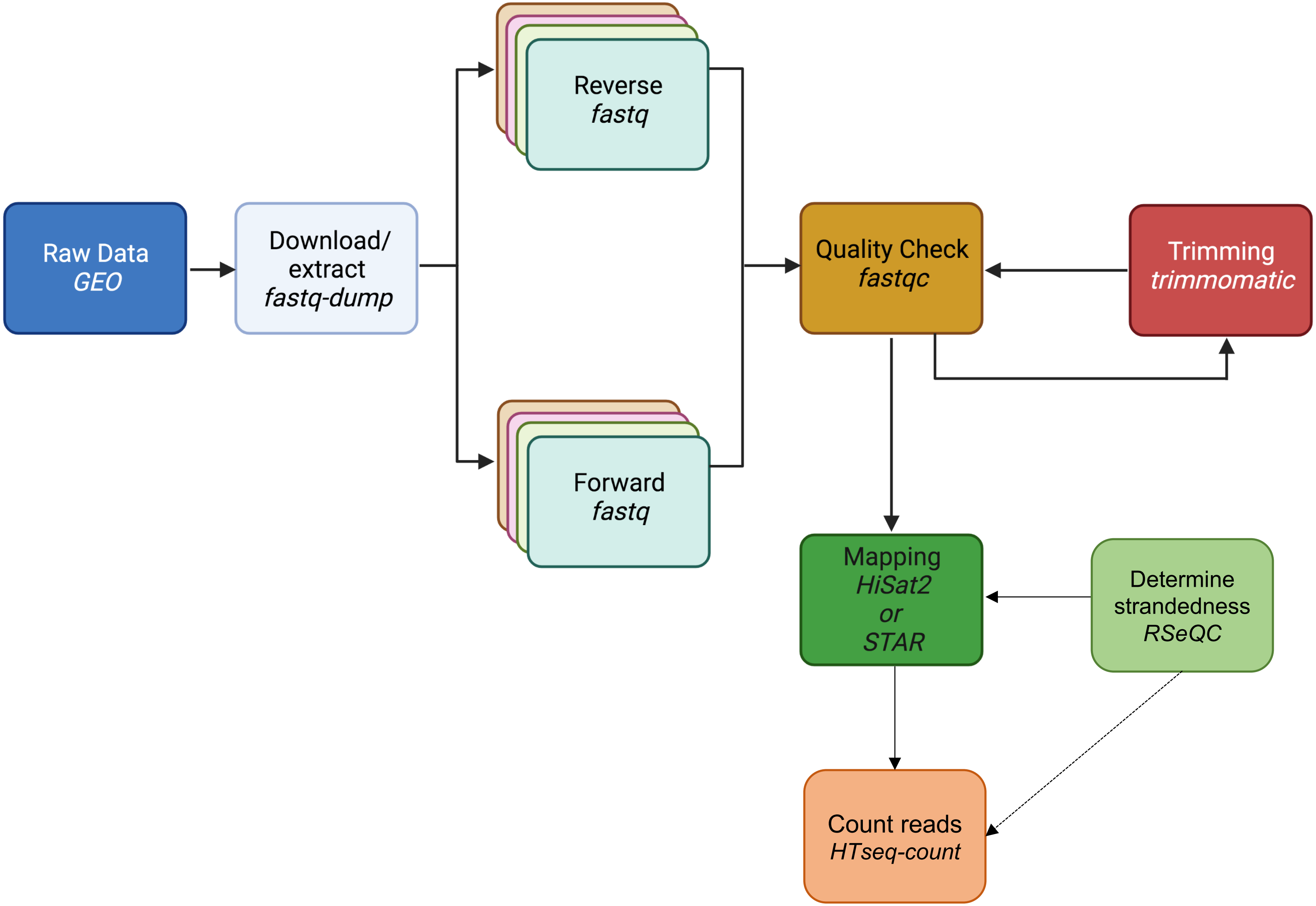
- **gene body coverage**
  - 3’ bias: RNA degradation + poly(A) enrichment





Installing RSeQC

- We will install RSeQC using conda
- Conda is an open-source management system
- Conda quickly installs, runs, and updates packages and their dependencies
- For this installation we will be creating a ‘conda environment’ called rseqc
- To use rseqc program in the future, you will need to perform ‘conda activate rseqc’



Raw Data
GEO

Download/
extract
fastq-dump

Reverse
fastq

Forward
fastq

Quality Check
fastqc

Trimming
trimmomatic

Mapping
HiSat2
or
STAR

Determine
strandedness
RSeQC

Count reads
HTseq-count

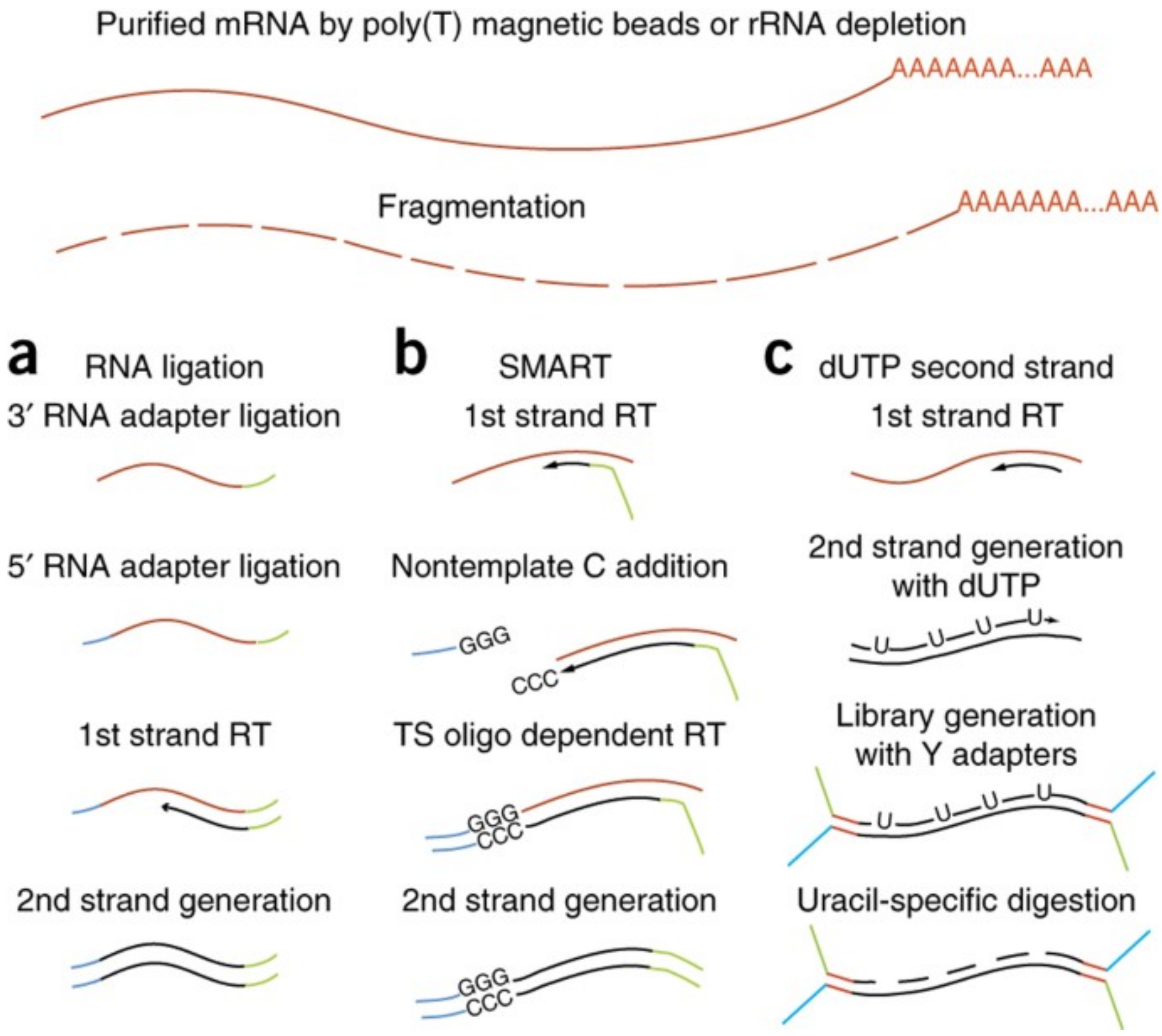
Take a break to install

Stranded libraries

- A major decision to be made during the library preparation step is whether to preserve RNA strand information.
- Unlike DNA molecules, RNA molecules exist as single-stranded threads that could result from the sense or antisense strand.
- The creation of stranded libraries are now standard with Illumina TruSeq ‘stranded’ RNA-Seq kits
- This means that with a great amount of certainty you can identify which strand of DNA the RNA was transcribed from

Three widely used protocols for strand-specific RNA-Seq library prep

Adds two different adapters to ends



Purified mRNA by poly(T) magnetic beads or rRNA depletion

Fragmentation

a RNA ligation

3′ RNA adapter ligation

5′ RNA adapter ligation

1st strand RT

2nd strand generation

b SMART

1st strand RT

Nontemplate C addition

TS oligo dependent RT

2nd strand generation

c dUTP second strand

1st strand RT

2nd strand generation with dUTP

Library generation with Y adapters

Uracil-specific digestion

Adds dUTP

Why retain stranded information?

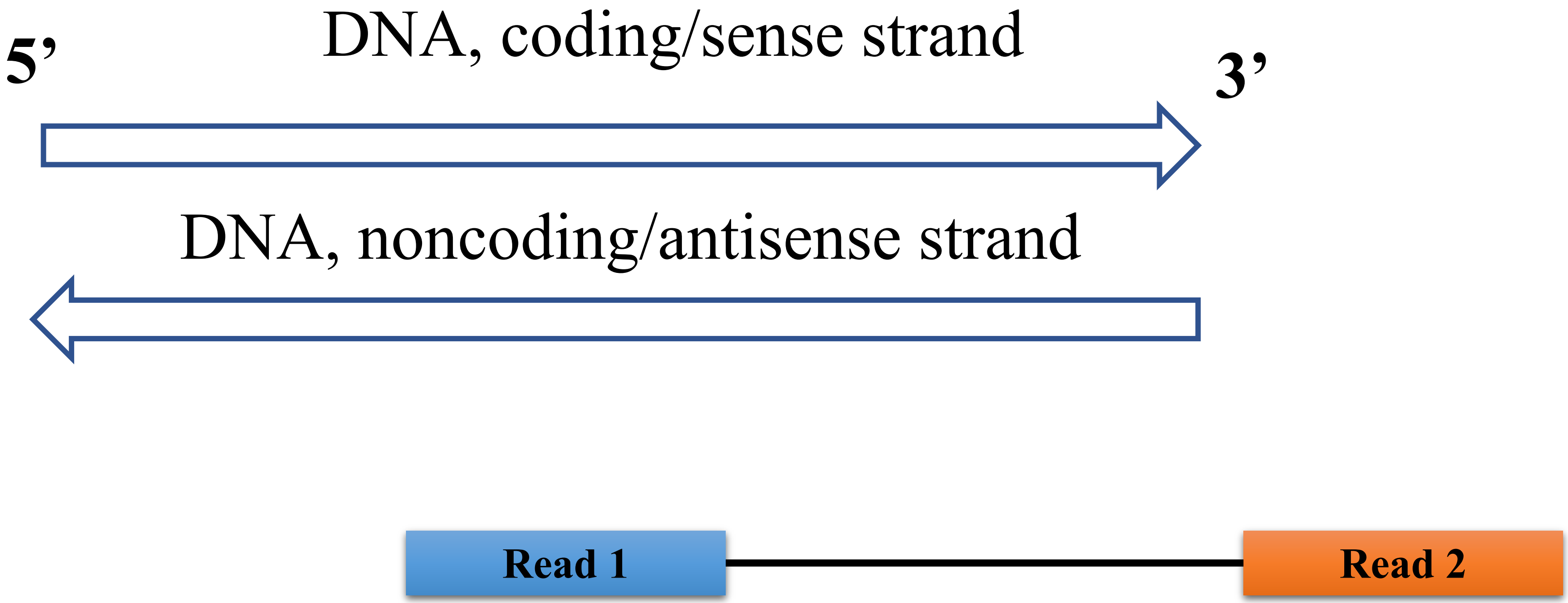
- It makes sense to begin with the most information possible – even if immediately that is not of interest
- Useful for identifying antisense transcripts, mapping splicing events, and detecting overlapping transcripts.
- They are commonly used in studies of transcriptomics, gene expression analysis, and RNA editing, and *de novo* assembly.

Why is this important to determine prior to counting?

- If you use wrong directionality parameter in the read counting step with HTSeq, the reads are considered to be from the wrong strand.
- This means you won't get any counts, and if there is a gene in the same location on the other strand, your reads are counted for the ***wrong gene***.
- So its important to check, if you are unsure, using tools!

Three scenarios when it comes to stranded libraries

- Forward (secondstrand) – reads resemble the gene sequence
- Reverse (firststrand) – reads resemble the complementary sequence
- Unstranded



5’

DNA, coding/sense strand

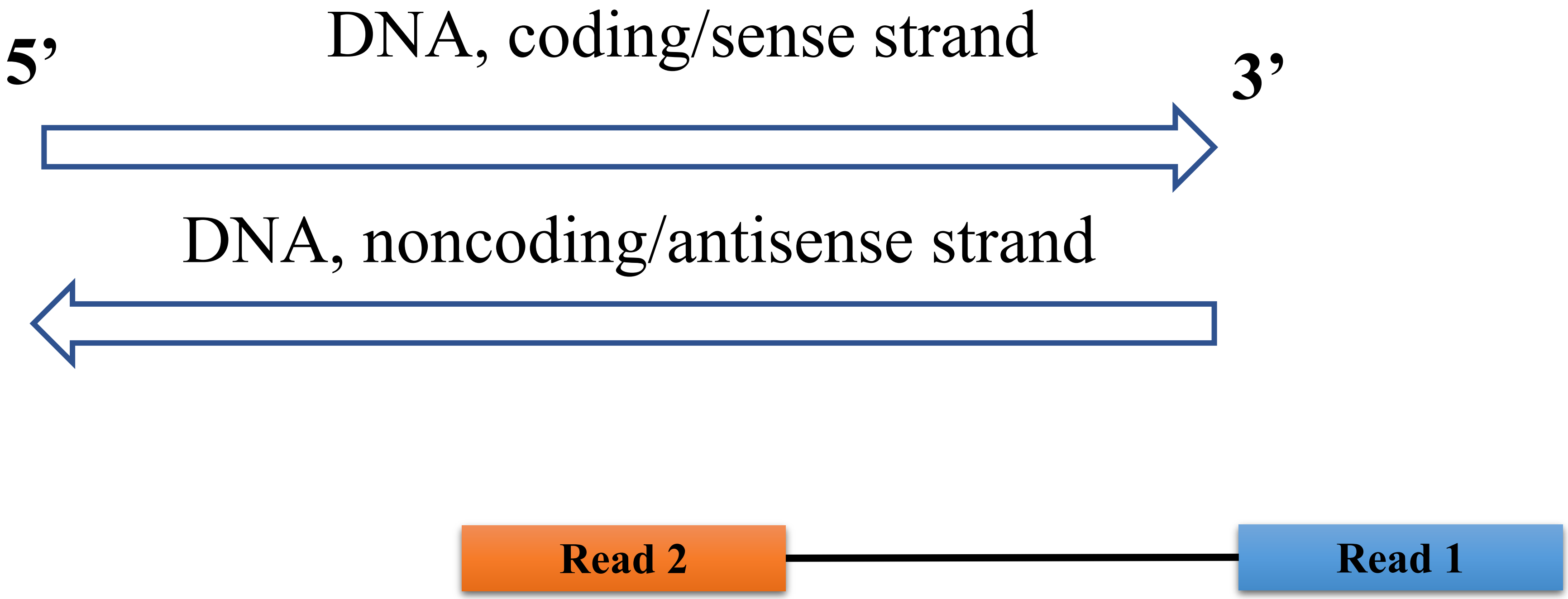
3’

DNA, noncoding/antisense strand

Read 1

Read 2

If sequences of Read 1 align to the coding, sense strand – the library is “stranded”



5’

DNA, coding/sense strand

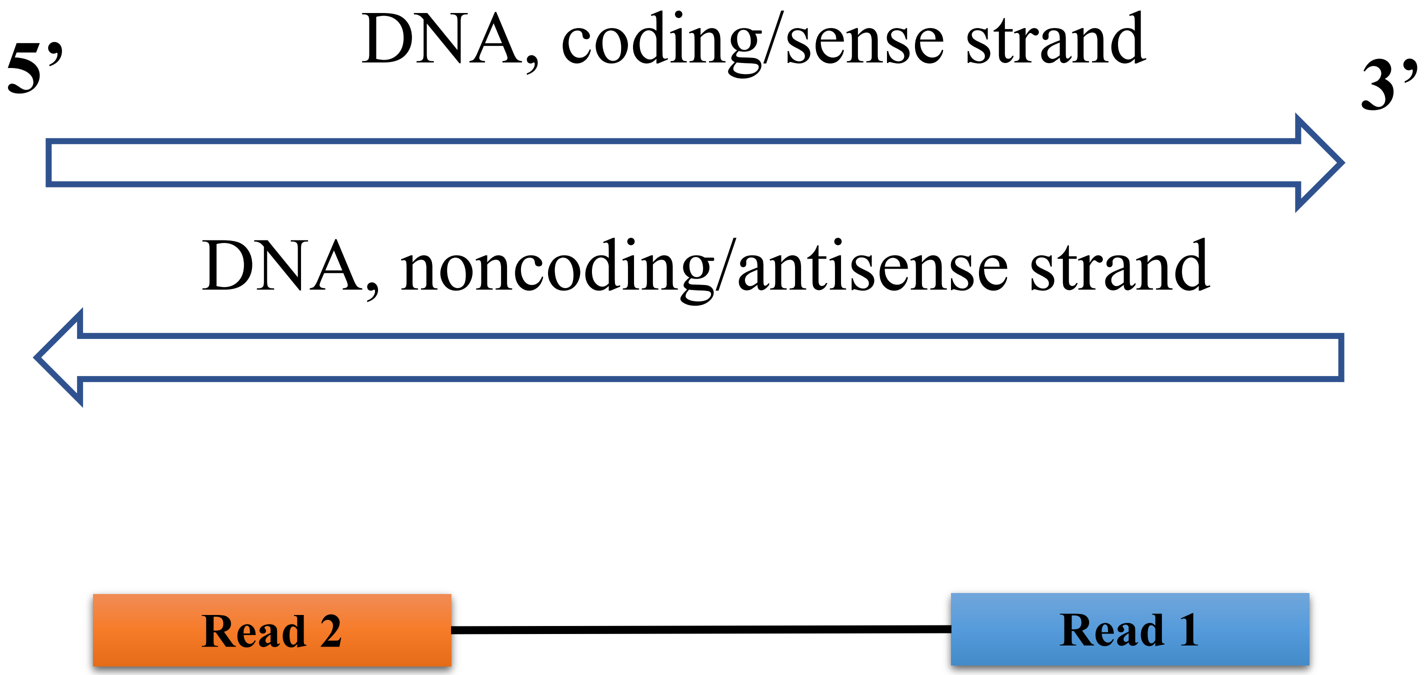
3’

DNA, noncoding/antisense strand

Read 2

Read 1

If sequences of Read 2 align to the coding, sense strand – the library is “ reverse stranded”



5’

DNA, coding/sense strand

3’

DNA, noncoding/antisense strand

Read 2

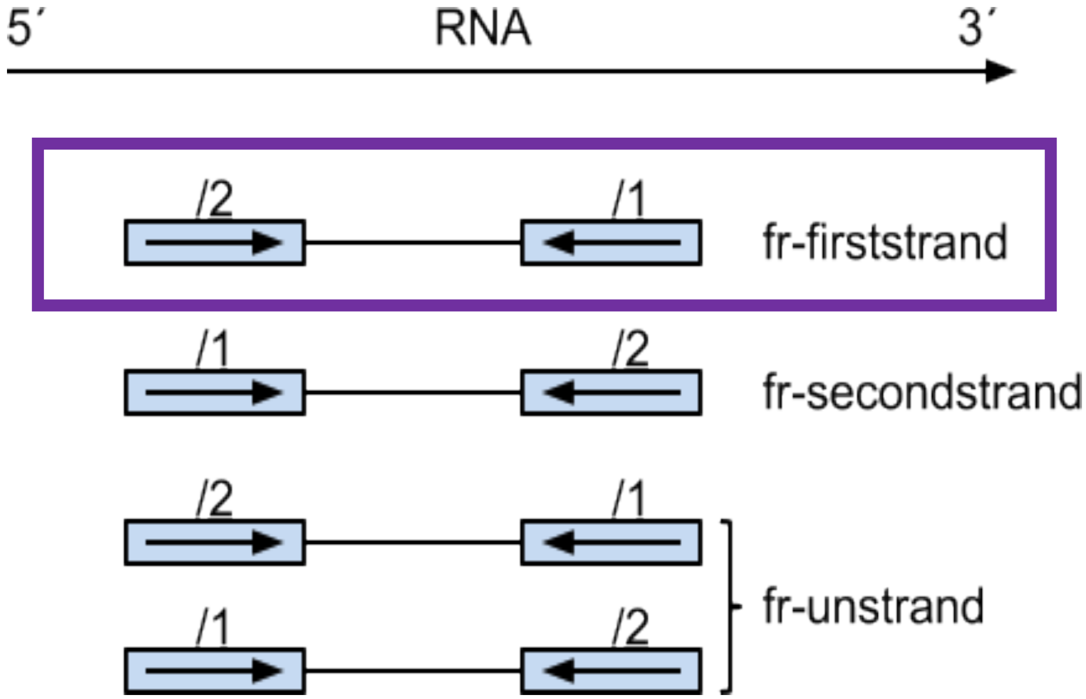
Read 1

If sequences both Read 1 and Read 2 align to the coding, sense strand – the library is “unstranded”

Different tools have different names for stranded settings:

	Option 1 **RF/fr-firststrand**	**Option 2** **FR/fr-secondstrand**	**Option 3** **Unstranded**
HISAT2	R/RF (for PE) --rna-strandedness R (for SE)	F/FR	Default
STAR	n/a	n/a	n/a
SALMON	-I ISR	-I ISR	-I IU
HTSeq	stranded=reverse	stranded=yes	stranded=no
Methods or Kits	dUTP Illumina TruSeq NEBNext Ultra II	Ligation Standard SOLID, NuGEN, 10X	Standard Illumina NuGEN, SMARTer

Infer_experiment.py pair-end RNA-seq



The second read (read 2) is from the original RNA strand/template, first read (read 1) is from the opposite strand.

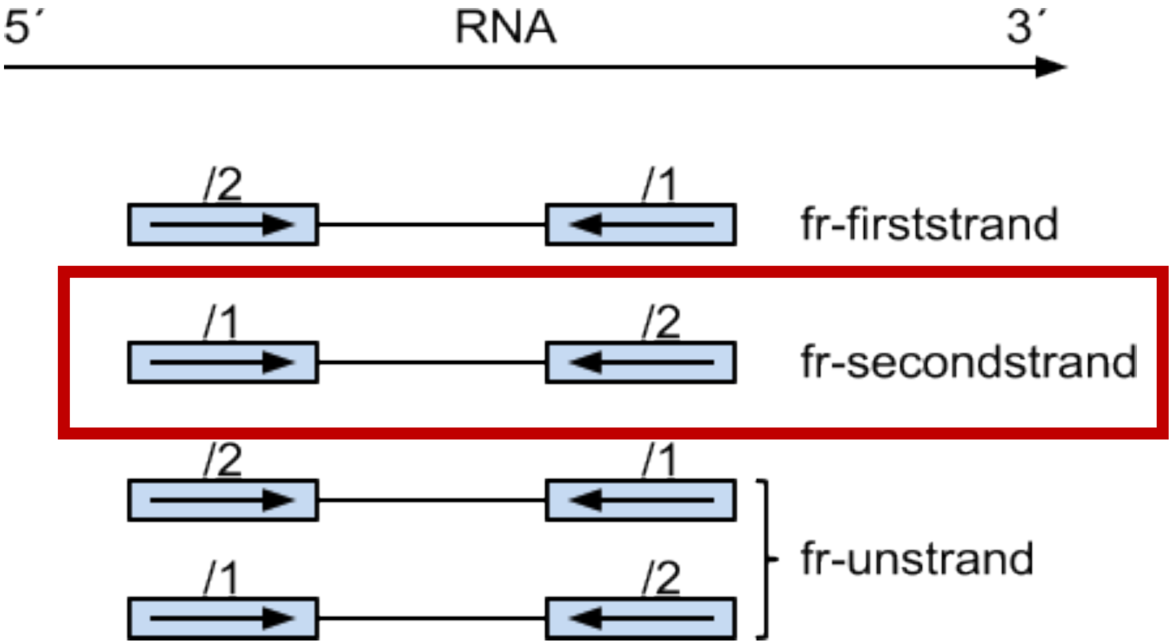
```
Fraction of reads explained by "1++,1--,2+-,2-+": 0.0169
Fraction of reads explained by "1+-,1-+,2++,2-- ": 0.8827
```

Strand-specific pair-end RNA-seq data using dUTP protocol

Option 1
RF/fr-firststrand

	Option 1 RF/fr-firststrand	Option 2 FR/fr-secondstrand	Option 3 Unstranded
HISAT2	R/RF (for PE) --rna-strandedness R (for SE)	F/FR	Default
STAR	n/a	n/a	n/a
SALMON	-I ISR	-I ISR	-I IU
HTSeq	**stranded=reverse**	stranded=yes	stranded=no
Methods or Kits	dUTP Illumina TruSeq NEBNext Ultra II	Ligation Standard SOLID, NuGEN, 10X	Standard Illumina NuGEN, SMARTer

Infer_experiment.py pair-end RNA-seq



The first read (read 1) is from the original RNA strand/template, second read (read 2) is from the opposite strand.

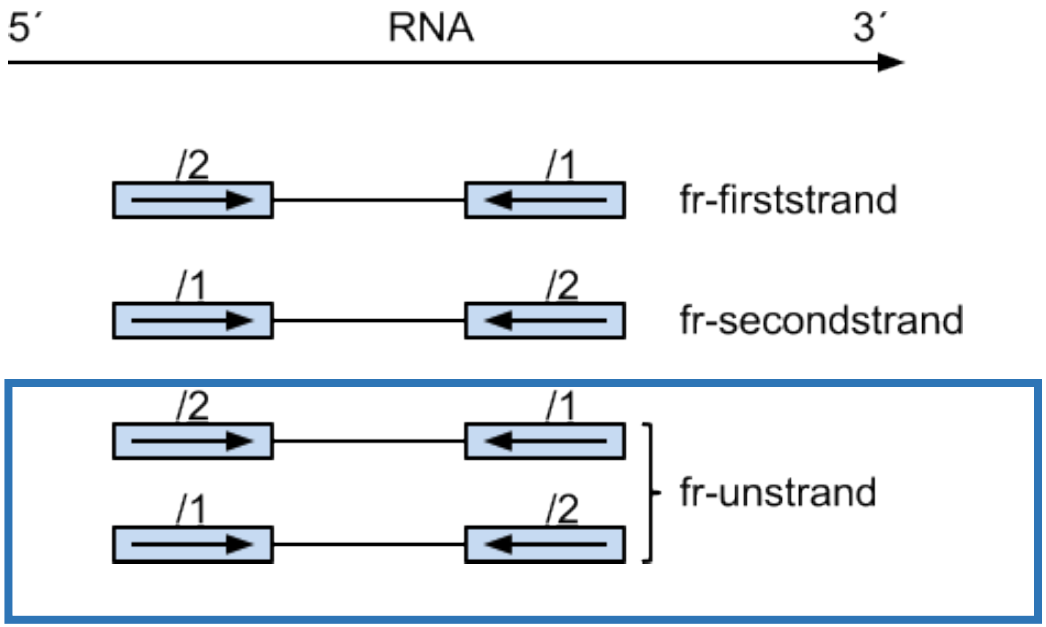
Fraction of reads explained by "1++,1--,2+-,2-+": 0.9807
Fraction of reads explained by "1+-,1-+,2++,2-- ": 0.0193

Strand-specific pair-end RNA-seq data using Ligation protocol

Option 2
FR/fr-secondstrand

	Option 1 **RF/fr-firststrand**	**Option 2** **FR/fr-secondstrand**	**Option 3** **Unstranded**
HISAT2	R/RF (for PE) --rna-strandedness R (for SE)	F/FR (for PE) --rna-strandedness F (for SE)	Default
STAR	n/a	n/a	n/a
SALMON	-I ISR	-I ISR	-I IU
HTSeq	stranded=reverse	**stranded=yes**	stranded=no
Methods or Kits	dUTP Illumina TruSeq NEBNext Ultra II	Ligation Standard SOLID, NuGEN, 10X	Standard Illumina NuGEN, SMARTer

Infer_experiment.py pair-end RNA-seq

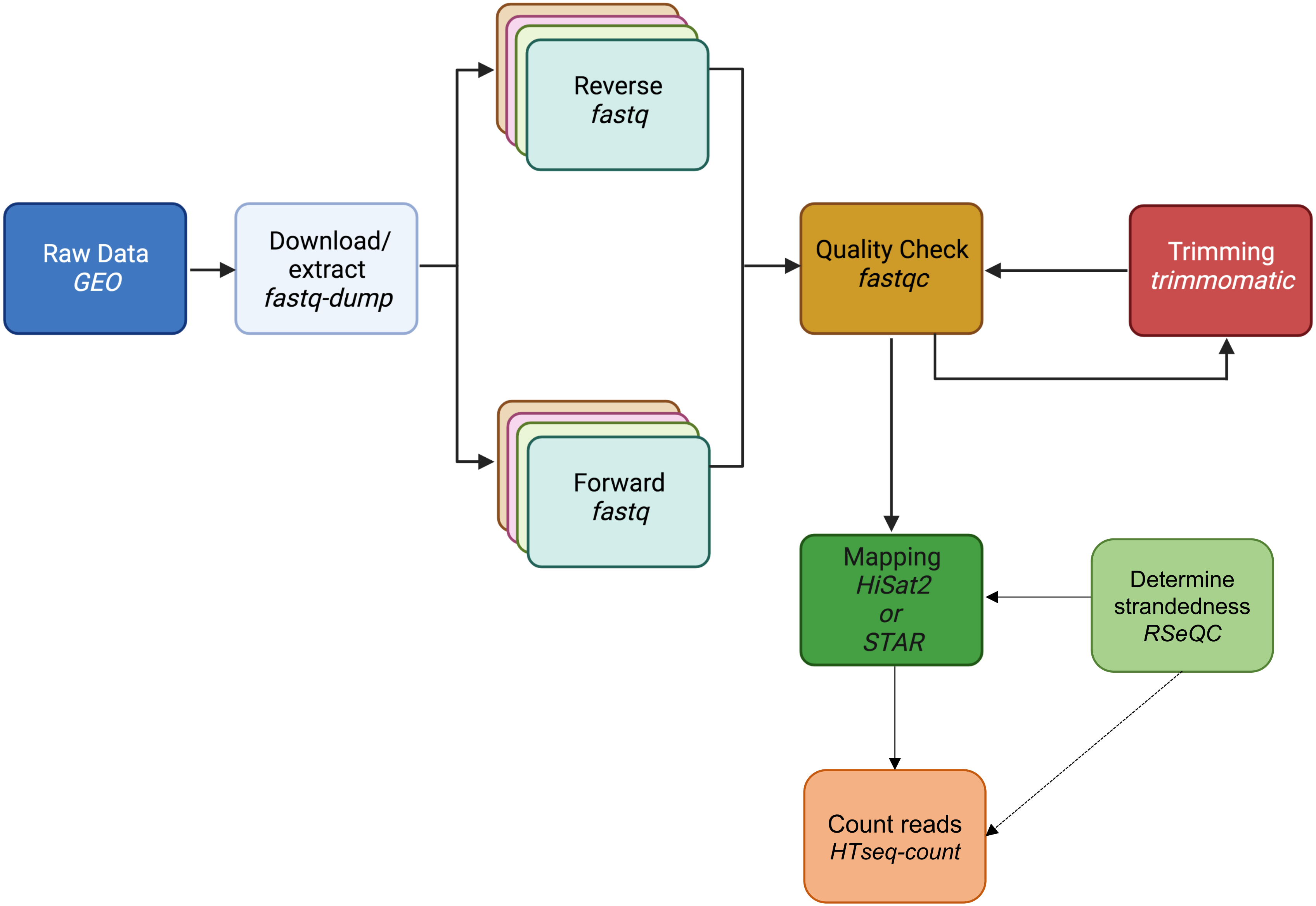


```
Fraction of reads failed to determine: 0.0648
Fraction of reads explained by "1++,1--,2+-,2-+": 0.4590
Fraction of reads explained by "1+-,1-+,2++,2--": 0.4763
```

Information regarding the strand is not conserved (it is lost during the amplification of the mRNA fragments).

Option 3
Unstranded

	Option 1 RF/fr-firststrand	Option 2 FR/fr-secondstrand	Option 3 Unstranded
HISAT2	R/RF (for PE) --rna-strandedness R (for SE)	F/FR (for PE) --rna-strandedness F (for SE)	Default
STAR	n/a	n/a	n/a
SALMON	-I ISR	-I ISR	-I IU
HTSeq	stranded=reverse	stranded=yes	**stranded=no**
Methods or Kits	dUTP Illumina TruSeq NEBNext Ultra II	Ligation Standard SOLID, NuGEN, 10X	Standard Illumina NuGEN, SMARTer



Take a break to run RSeQC to infer strandedness

Is your library stranded or not stranded?

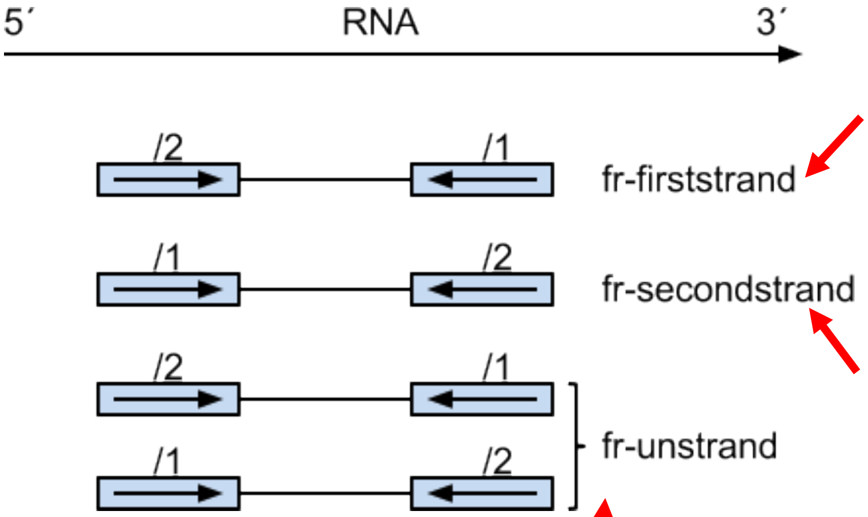
- RSeQC (http://rseqc.sourceforge.net/)

- `infer_experiment.py -i sample.bam -r gene_model.bed`

What would you choose for the unknown?

	Option 1 RF/fr-firststrand	Option 2 FR/fr-secondstrand	Option 3 Unstranded
HISAT2	R/RF (for PE) --rna-strandedness R (for SE)	F/FR (for PE) --rna-strandedness F (for SE)	Default
STAR	n/a	n/a	n/a
SALMON	-I ISR	-I ISR	-I IU
HTSeq	stranded=reverse	stranded=yes	stranded=no
Methods or Kits	dUTP Illumina TruSeq NEBNext Ultra II	Ligation Standard SOLID, NuGEN, 10X	Standard Illumina NuGEN, SMARTer

Summary

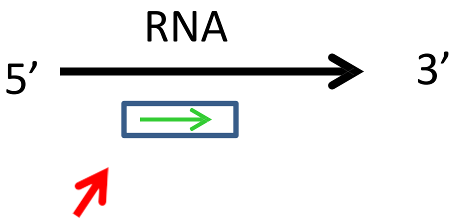


Fraction of reads explained by "1++,1--,2+-,2-+": 0.0193
Fraction of reads explained by "1+-,1-+,2++,2-- ": 0.8827

Fraction of reads explained by "1++,1--,2+-,2-+": 0.9807
Fraction of reads explained by "1+-,1-+,2++,2-- ": 0.0193

Fraction of reads failed to determine: 0.0648 Fraction of reads explained by "1++,1--,2+-,2-+":0.4590
Fraction of reads explained by "1+-,1-+,2++,2--": 0.4763

Infer_experiment.py
single-end RNA-seq



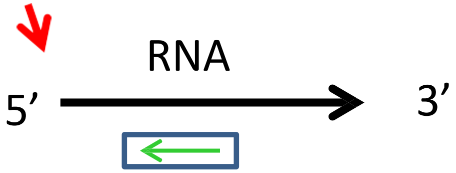
Two different ways to strand reads:

i) ++,--

read mapped to '+' strand indicates parental gene on '+' strand
read mapped to '-' strand indicates parental gene on '-' strand

ii) +-,-+

read mapped to '+' strand indicates parental gene on '-' strand
read mapped to '-' strand indicates parental gene on '+' strand



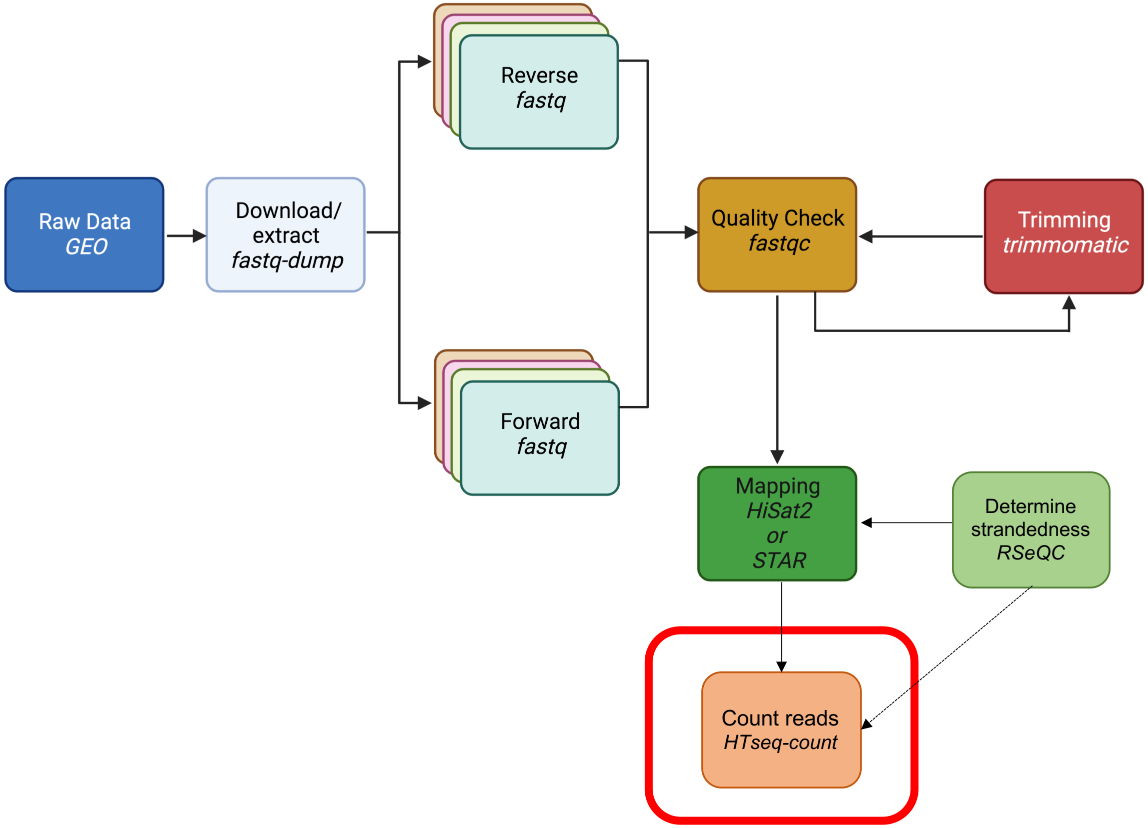
Strand-specific example:

```
Fraction of reads failed to determine: 0.0170
Fraction of reads explained by "++,--": 0.9669
Fraction of reads explained by "+-,-+": 0.0161
```

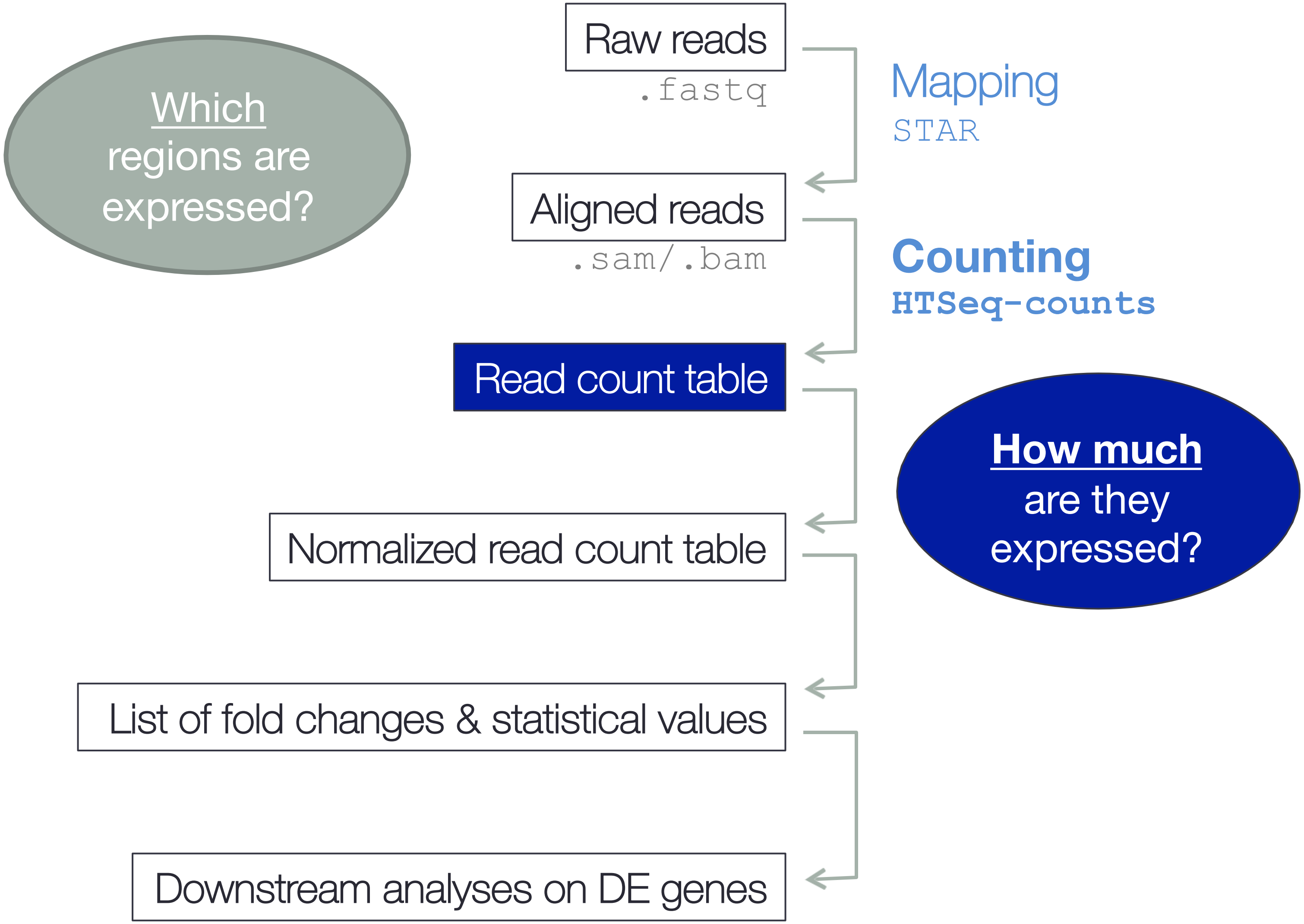
FR/fr-secondstrand

stranded=yes

COUNTING READS



Bioinformatics workflow of RNA-seq analysis



Which regions are expressed?

Raw reads

.fastq

Mapping

STAR

Aligned reads

.sam/.bam

Counting

HTSeq-counts

Read count table

How much are they expressed?

Normalized read count table

List of fold changes & statistical values

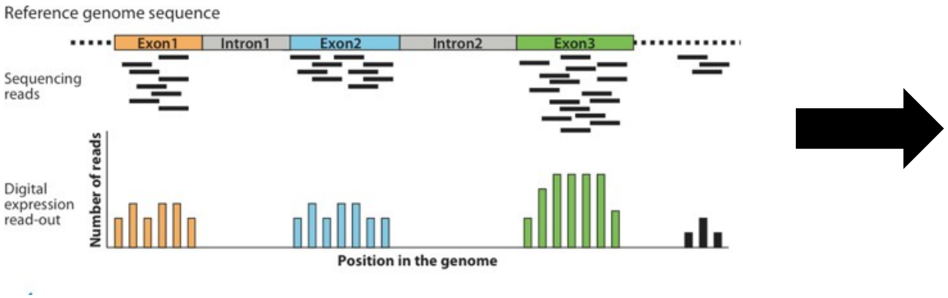
Downstream analyses on DE genes

Gene counting programs

- HTSeq (Anders et al.2015, Bioinformatics 31:2)
- Cufflinks (Trapnell et al, 2010, Nat Biotech 28:5)
- StringTie (Pertea et al. 2015, Nat Biotech 33:3)
- featureCounts

We are using HTSeq as this approach will obtain gene-level quantification by directly overlapping with gene loci

Counting per-gene alignments



	sample1	sample2	sample3	sample4	...
gene1	999	701	616	595	
gene2	532	520	41	26	
gene3	14	36	305	322	
...					

- **HTSeq** package
    - Anders, Pyl & Huber, 2015, *Bioinformatics 31*:2
    - Hompage at https://htseq.readthedocs.io/
    - Allows *per-exon* counts
    - Designed for *differential gene expression testing*
    - Includes the `htseq-count` command

Counting features with htseq-count

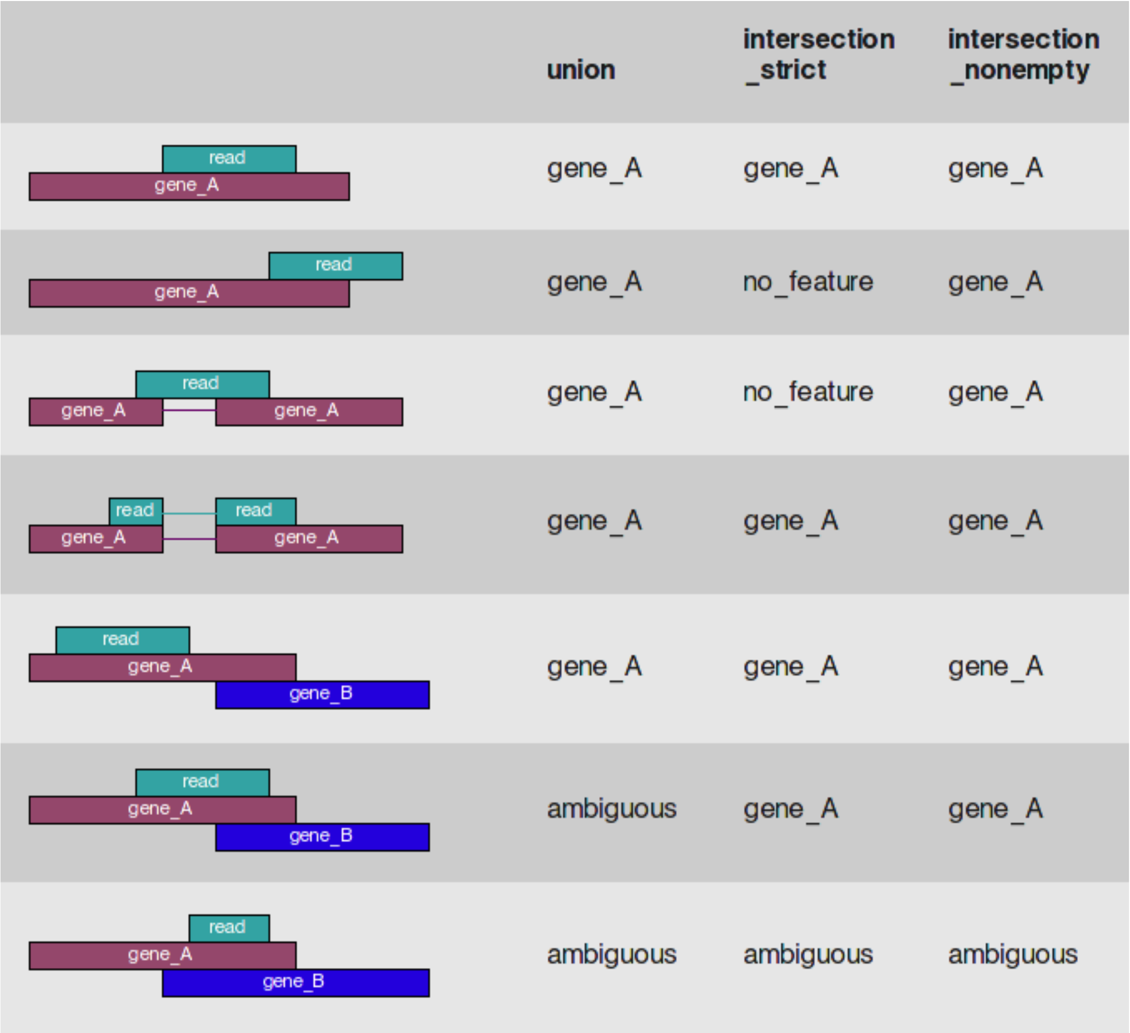
- What features are of interest? Gene, transcript, and/or exon counts?

  type=exon

- What happens if a read overlaps with multiple features?

  mode=union

- Is the RNA stranded, reversed strand, or unstranded?



	union	intersection_strict	intersection_nonempty
read gene_A	gene_A	gene_A	gene_A
read gene_A	gene_A	no_feature	gene_A
read gene_A gene_A	gene_A	no_feature	gene_A
read read gene_A gene_A	gene_A	gene_A	gene_A
read gene_A gene_B	gene_A	gene_A	gene_A
read gene_A gene_B	ambiguous	gene_A	gene_A
read gene_A gene_B	ambiguous	ambiguous	ambiguous

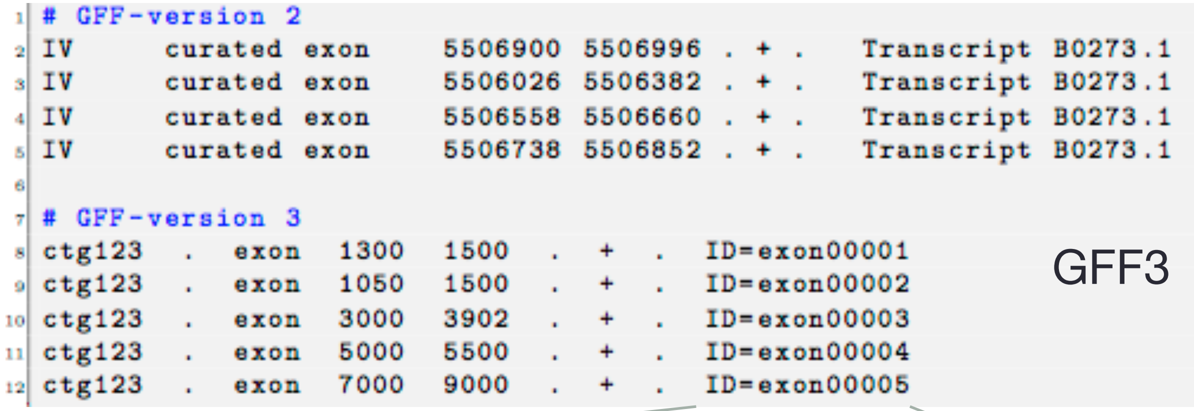
Storing annotation information

- representing genome coordinates + description/name
  - intron–exon structures, start and stop codons, UTRs, alternative transcripts
- various formats (all are plain text files): GFF2, GFF3, GTF, BED, SAF…

GTF ("GFF2.5")

1. reference coordinate
2. source
3. annotation type
4. start position
5. end position
6. score
7. strand
8. frame/phase
9. attributes: <TYPE VALUE>; <TYPE VALUE>; <TYPE VALUE>

GFF2

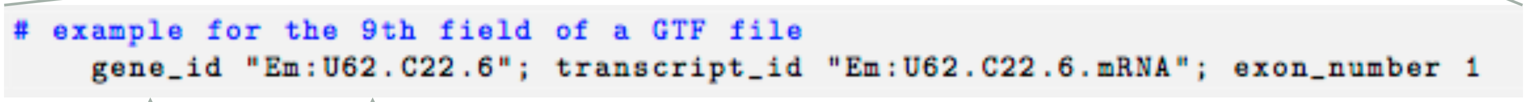


```
# GFF-version 2
IV      curated exon    5506900 5506996 . + .   Transcript B0273.1
IV      curated exon    5506026 5506382 . + .   Transcript B0273.1
IV      curated exon    5506558 5506660 . + .   Transcript B0273.1
IV      curated exon    5506738 5506852 . + .   Transcript B0273.1

# GFF-version 3
ctg123  .  exon  1300  1500  .  +  .  ID=exon00001
ctg123  .  exon  1050  1500  .  +  .  ID=exon00002
ctg123  .  exon  3000  3902  .  +  .  ID=exon00003
ctg123  .  exon  5000  5500  .  +  .  ID=exon00004
ctg123  .  exon  7000  9000  .  +  .  ID=exon00005
```

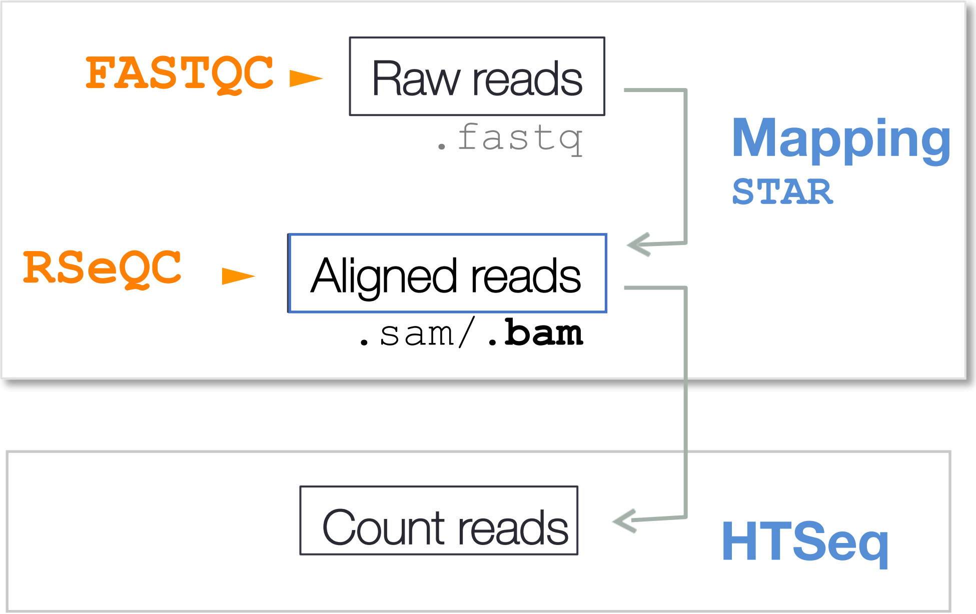
GFF3

GTF



```
# example for the 9th field of a GTF file
    gene_id "Em:U62.C22.6"; transcript_id "Em:U62.C22.6.mRNA"; exon_number 1
```

Summary



- We **downloaded fastq.gz** files from the SRA via SRAtool-kit (fastq-dump)
- We did **QC** of the raw reads using **FastQC** (1x per sample) and summarized the results for the numerous fastq files per sample it using **MultiQC**
- We **aligned** the raw reads using **STAR and HISAT2**
- We performed **additional QC** on those BAM files using **RSeQC**
- We then counted read-gene overlaps with **HTSeq**