SALMON – *Don't count…quantify!*

- free, quick, little memory
- takes into account experimental attributes and biases typical of RNA-seq data, including positional biases in coverage, sequence-specific biases at the 5′ and 3′ end of sequenced fragments, fragment-level GC bias, strand-specific protocols, and the fragment length distribution
- input: set of target transcripts

SALMON

mapping-based mode

alignment-based mode

MAPPING-BASED MODE

- Mapping instead of alignment
- Salmon index
- Decoy-aware transcriptome
- Outputs .fs file

Building index:

> ./bin/salmon index –t transcripts.fa –i transcripts_index –decoys decoys.txt –k 31

Paired-end reads:

> ./bin/salmon quant -i transcripts_index -l <LIBTYPE> -1 reads1.fq -2 reads2.fq --validateMappings -o transcripts_quant

Single-end reads:

> ./bin/salmon quant -i transcripts_index -l <LIBTYPE> -r reads.fq --validateMappings -o transcripts_quant

MAPPING-BASED MODE

Multiple samples:

> ./bin/salmon quant -i index -l <LIBTYPE> -1 lib_1_1.fq lib_2_1.fq -2 lib_1_2.fq lib_2_2.fq --validateMappings -o out

> ./bin/salmon quant -i index -l <LIBTYPE> -1 <(cat lib_1_1.fq lib_2_1.fq) -2 <(cat lib_1_2.fq lib_2_2.fq) --validateMappings -o out

Zipped files:

> ./bin/salmon quant -i index -l <LIBTYPE> -1 lib_1_1.fq.gz lib_2_1.fq.gz -2 lib_1_2.fq.gz lib_2_2.fq.gz --validateMappings -o out

> ./bin/salmon quant -i index -l <LIBTYPE> -1 <(gunzip -c lib_1_1.fq.gz lib_2_1.fq.gz) -2 <(gunzip -c lib_1_2.fq.gz lib_2_2.fq.gz) --validateMappings -o out

ALIGMENT-BASED MODE

- Pre-aligned data
- .bam files
- Outputs .fs file

> ./bin/salmon quant -t transcripts.fa -l <LIBTYPE> -a aln.bam -o salmon_quant

LIBTYPE

- automatic library type detection in alignment-based mode
- argument -l A

1. the relative orientation of the reads (only if the library is paired-end)

```
I = inward
O = outward
M = matching
```

LIBTYPE

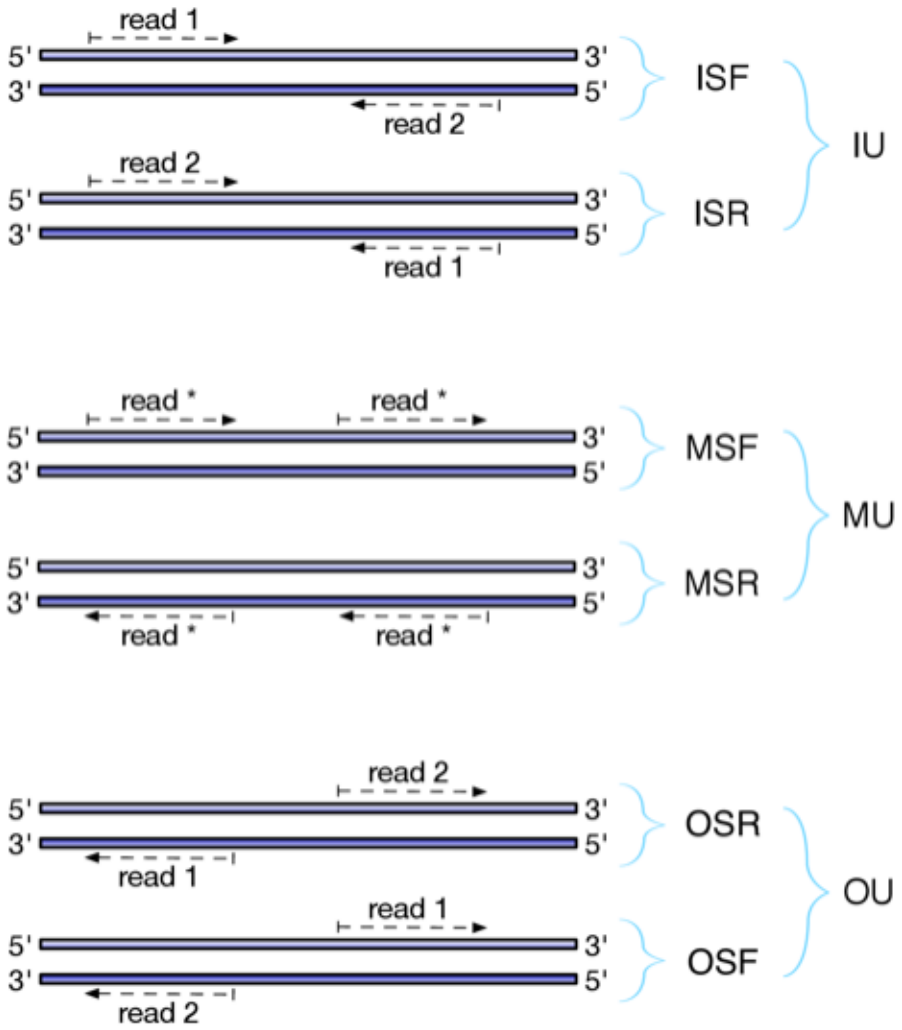
2. the strandedness of the library

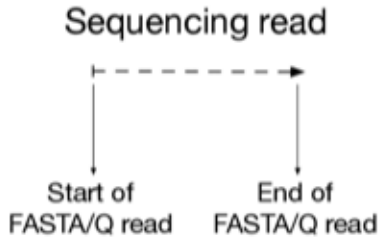
```
S = stranded
U = unstranded
```

3. the directionality of the reads (only if the library is stranded)

```
F = read 1 (or single-end read) comes from the forward strand
R = read 1 (or single-end read) comes from the reverse strand
```

LIBTYPE


read 1
5'
3'
3'
5'
read 2
ISF
read 2
5'
3'
3'
5'
read 1
ISR
IU
read *
read *
5'
3'
3'
5'
MSF
5'
3'
3'
5'
read *
read *
MSR
MU
read 2
5'
3'
3'
5'
read 1
OSR
read 1
5'
3'
3'
5'
read 2
OSF
OU


Sequencing read
Start of
FASTA/Q read
End of
FASTA/Q read

EXAMPLES

```
IU (an unstranded paired-end library where the reads face each other)
```

```
SF (a stranded single-end protocol where the reads come from the forward strand)
```

```
OSR (a stranded paired-end protocol where the reads face away from each other,
     read1 comes from reverse strand and read2 comes from the forward strand)
```

THREADS

- alignment-based mode: 8-12 (4 threads for BAM decompression, rest for quantification) for maximum speed
- mapping-based mode: more threads = faster quantification
- -p argument (default = maximum number of available threads)

OUTPUT

Name	Length	EffectiveLength	TPM	NumReads
ENST00000456328	1657	1785.304	0.054490	3.722479
ENST00000450305	632	250.000	0.000000	0.000000
ENST00000488147	1351	1530.937	3.793490	222.229533
ENST00000619216	68	3.000	34.844416	4.000000
ENST00000473358	712	519.262	0.000000	0.000000
ENST00000469289	535	250.000	0.000000	0.000000
ENST00000607096	138	5.000	0.000000	0.000000
ENST00000417324	1187	250.000	0.000000	0.000000
ENST00000461467	590	250.000	0.000000	0.000000
ENST00000606857	840	250.000	0.000000	0.000000
ENST00000642116	1414	250.000	0.000000	0.000000
ENST00000492842	939	250.000	0.000000	0.000000

TPM

- transcripts per kilobase million
- how to calculate:
  1. divide the read counts by the length of each gene in kilobases. This gives you reads per kilobase (RPK)
  2. then count how many RPK values you have in a sample and divide this number by million to get your “per million” scaling factor
  3. and then divide each RPK value by the scaling factor and the result is TPM

SOURCES

- https://combine-lab.github.io/salmon/
- https://salmon.readthedocs.io/en/latest/salmon.html
- https://nf-co.re/modules/salmon_quant#input
- https://sailfish.readthedocs.io/en/develop/salmon.html
- https://www.rna-seqblog.com/rpkm-fpkm-and-tpm-clearly-explained/