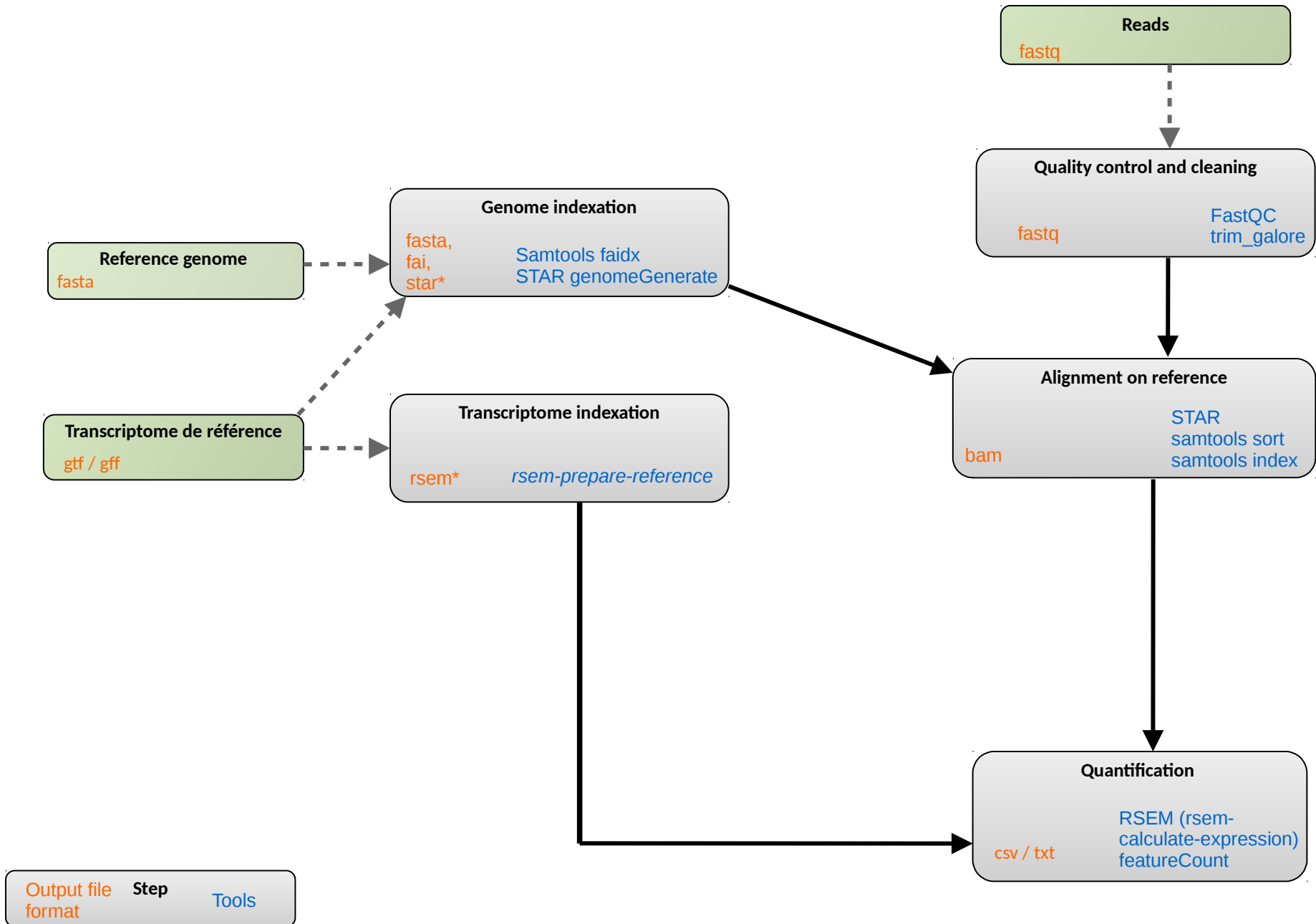
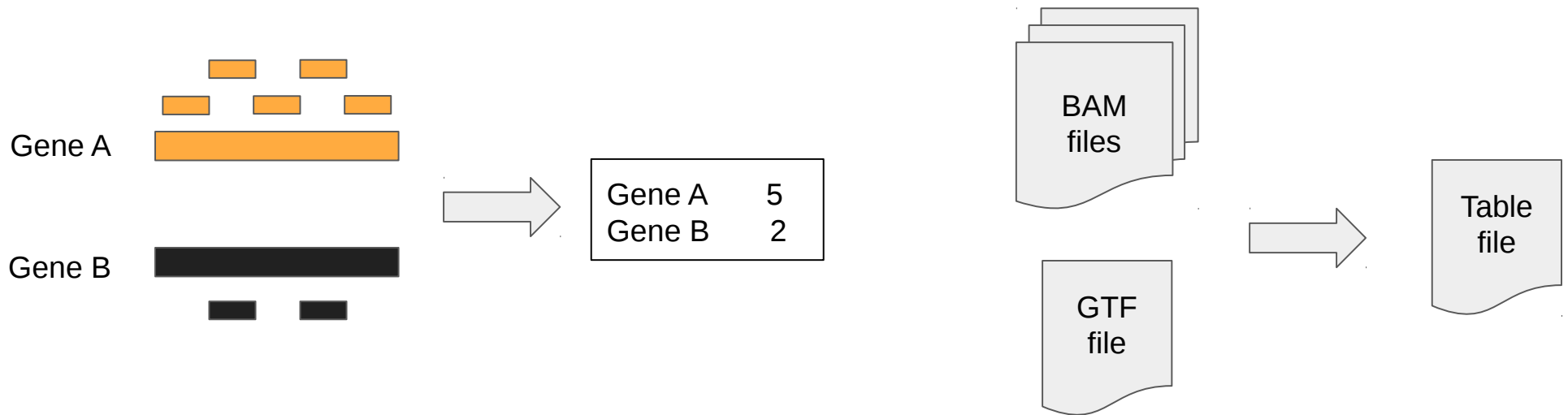


Analysis workflow



Quantification

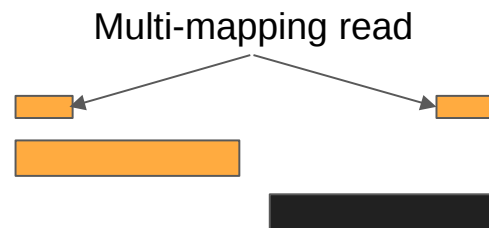
Quantification: estimation of expression based on a read count.



Estimation of:

- gene expression
- transcript expression
- exon expression

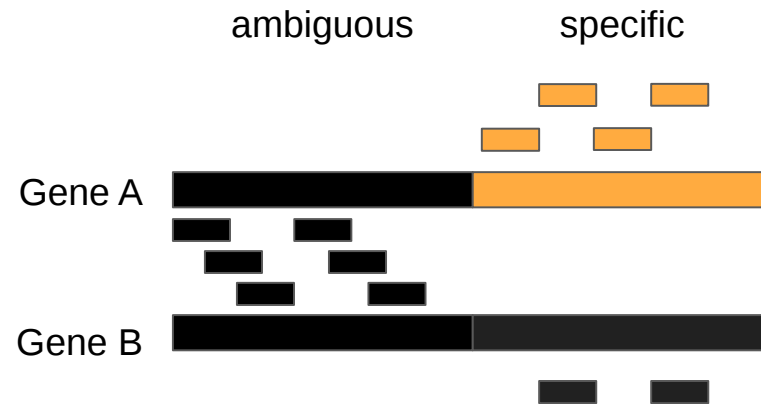
Difficult cases



Every quantification tools uses its own rules!

Raw counts vs estimation

Raw count vs estimation: what to do with ambiguous reads?



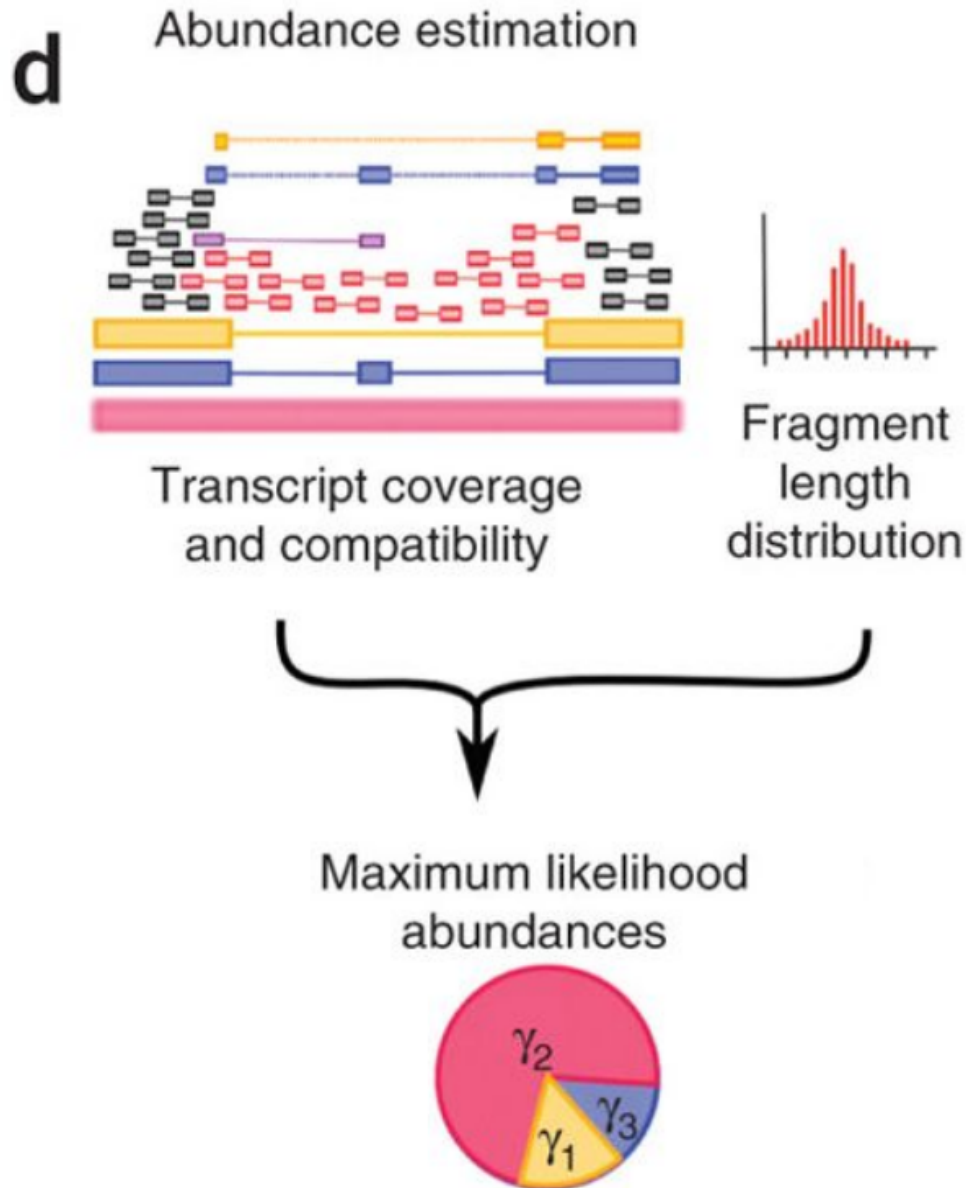
Pros estimation:

- Use more reads.
- More accurate?

Cons estimation:

- Underlying model inaccurate.
- Raw counts for differential expression does not matter much.

Transcript expression



Trapnell C *et al.* Nature Biotechnology 2010; 28:511-515

Raw counts tool: featureCounts

featureCounts: an efficient general purpose program for assigning sequence reads to genomic features

Yang Liao^{1,2}, Gordon K. Smyth^{1,3} and Wei Shi^{1,2,*}

¹Bioinformatics Division, The Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Parkville, VIC 3052,

²Department of Computing and Information Systems and ³Department of Mathematics and Statistics, The University of Melbourne, Parkville, VIC 3010, Australia

Associate Editor: Martin Bishop

- Levels : exon, transcript, gene
- Multiple option for :
 - Paired reads
 - Assignment of reads
 - Oriented library
- Also exists: HTseq-Count



Raw counts tool: featureCounts

```
module load bioinfo/subread-1.6.0
```

Command line:

```
featureCounts [options] -a <annotation_file> -o  
<output_file> input_file1 [input_file2]
```

Inputs :

- Gtf : annotation file (- a)
- Bams: input files

Some options :

- t [exon] Specify the feature type. Only rows which have the matched feature type in the provided GTF annotation file will be included for read counting.
- g [gene_id] Specify the attribute type used to group features (eg. Exons) into meta-features (eg. genes), when GTF annotation is provided.




Raw counts tool: featureCounts

- **Q** The minimum mapping quality score a read must satisfy in order to be counted. For paired-end reads, at least one end should satisfy this criteria. 0 by default.
- **-primary** If specified, only primary alignments will be counted.
- **-minOverlap** Specify the minimum number of overlapped bases required to assign a read to a feature. 1 by default.
- **p** If specified, fragments (or templates) will be counted instead of reads.
- **P** If specified, paired-end distance will be checked when assigning
- **d** Minimum fragment/template length, 50 by default.
- **D** Maximum fragment/template length, 600 by default.
- **B** If specified, only fragments that have both ends successfully aligned will be considered for summarization.
- **T [1]** Number of the threads.

Estimation tool: RSEM

RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome

Bo Li and Colin N Dewey 

BMC Bioinformatics 2011 12:323 | DOI: 10.1186/1471-2105-12-323 | © Li and Dewey; licensee BioMed Central Ltd. 2011

Received: 10 May 2011 | Accepted: 4 August 2011 | Published: 4 August 2011

- Exhaustive tool
- Levels : transcript, gene
- May be used without reference genome (RNA-Seq *de novo*)

- Also exists: cufflinks



RSEM : Prepare reference

Command line:

```
module load bioinfo/RSEM-XXX
```

```
rsem-prepare-reference --gtf annot.gtf  
genome.fasta rsem_lib
```

Output files:

- *rsem_lib.grp*, *rsem_lib.ti*, *rsem_lib.seq*, and *rsem_lib.chrlist* are for internal use.
- *rsem_lib.idx.fa*: the transcript sequences
- *rsem_lib.n2g.idx.fa*: same, with N → G



RSEM: calculate expression

Command line:

```
rsem-calculate-expression --alignments  
alignment.bam rsem_lib quant
```

Outputs:

- `quant.isoforms.results`: isoform level expression estimates
- `quant.genes.results`: same for genes
- `quant.stat`: directory with stats on various aspects of this step



RSEM: calculate expression

Other parameters:

- -paired-end: specify paired-end reads
- p N: use N CPUs
- -seed N: seed for random number generators
- -calc-ci: calculate 95% credibility intervals and posterior mean estimates.
- ci-memory 30000: size in MB of the buffer used for computing CIs
- -estimate-rspd: estimate the read start position distribution
- no-bam-output: do not output any BAM file (produced by internal mapper)



Output file format

- `effective_length`: # positions that can generate a fragment
- `expected_count`: read count, with mapping prob. and read qual
- TPM: Transcripts Per Million, relative transcript abundance, see *infra*
- FPKM: Fragments Per Kilobase of transcript per Million mapped reads, see *infra*
- IsoPct: isoform percentage
- `posterior_mean_count`,
`posterior_standard_deviation_of_count`,
`pme_TPM`, `pme_FPKM`: estimates calculated Gibbs sampler



Output file format

- IsoPct_from_pme_TPM: isoform percentage calculated from pme TPM values
- TPM_ci_lower_bound, TPM_ci_upper_bound, FPKM_ci_lower_bound, FPKM_ci_upper_bound: bounds of 95% credibility intervals
- TPM_coefficient_of_quartile_variation, RPKM_coefficient_of_quartile_variation: coefficients of quartile variation, a robust way of measuring the ratio between the standard deviation and the mean

RPKM vs FPKM vs TPM

RPKM: Reads Per Kb of transcript per Million mapped

- $r = \#$ reads on a gene
- $k =$ size of the gene (in kb)
- $m = \#$ reads in the sample (in millions)

$$\text{RPKM} = r / (k m)$$

FPKM: Fragments Per Kilobase...

- Same with $f = \#$ fragments (2 reads in PE) on a gene

Meaning:

If you sequence at depth 10^6 , you will have $x = \text{FPKM}$ fragments of a 1kb-gene.

RPKM vs FPKM vs TPM

TMP:

- $r_i = \#$ reads on a gene i
- $s_i = \text{size of the gene } i$
- $cpbi = r_i / s_i$
- $cpb = \sum cpbi$
- $TMP_i = cpbi / cpb \times 10^6$

Remark:

- $TMP_i = FPKM_i / (\sum FPKM_j) \times 10^6$

Meaning:

If you have 10^6 transcripts, $x = TMP_i$ will originate from gene i .

RPKM vs FPKM vs TPM

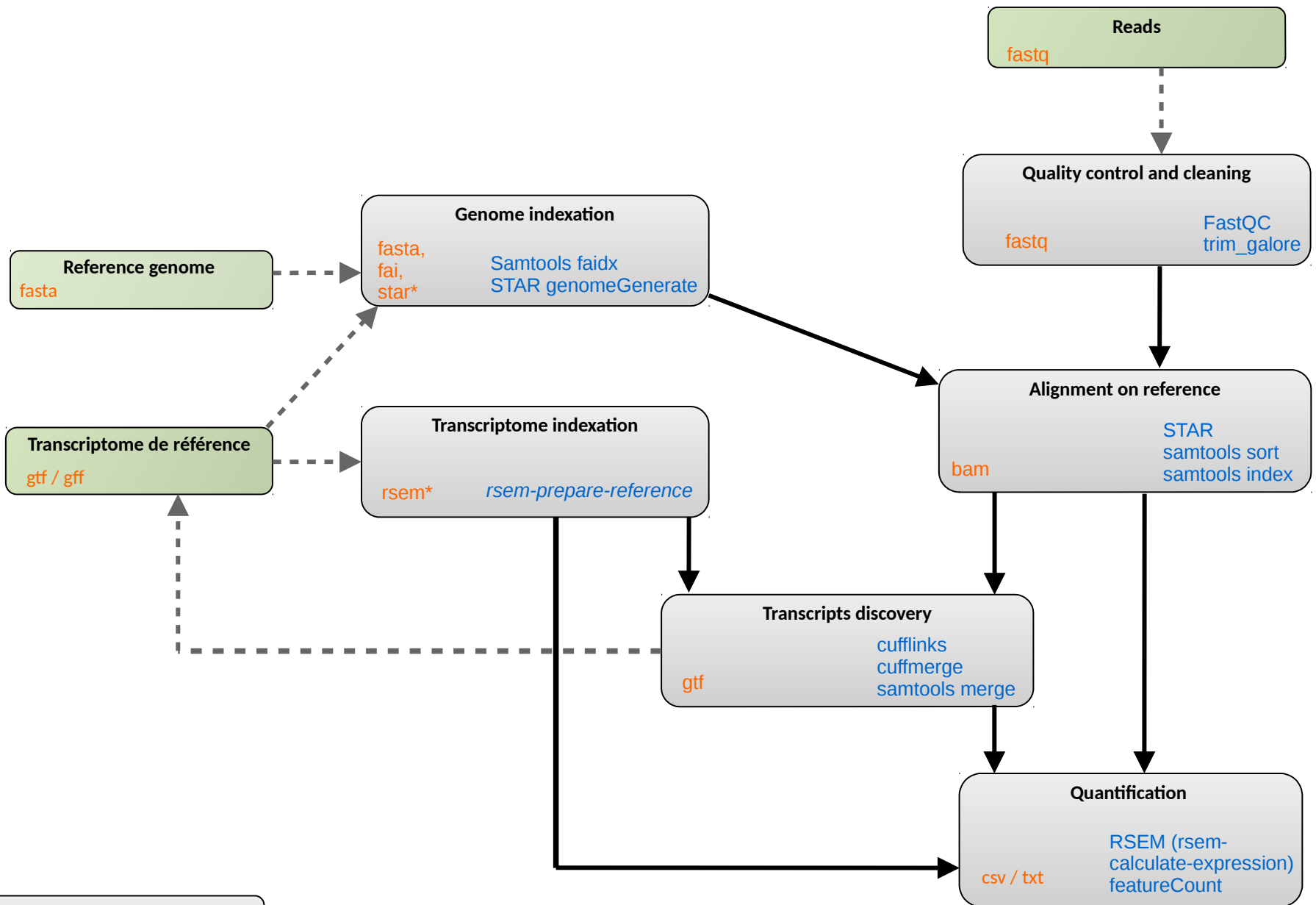
- These are refinement of library size normalization, with gene length effect.
- RPKM should not be used for PE reads.
- TPM tend to be favored now w.r.t. R/FPKM.
- None of them should be used for differential expression: only raw counts.

Ask your questions to the stats guys.

Quantification

Exercise 6

RNAseq pipeline : all steps



Output file format Step Tools

How to choose count matrix ?

- Quality of the annotation :
 - do not forget to check the genes structure with IGV
 - presence of genes of interest
 - too many transcripts
 - quality metrics with gffcompare
 - number of covered gene
- Number of mapped reads
- Number of assigned reads

Next step

From count matrix to DEG :

- Normalization
 - Differential expression analysis
 - End more ... GO enrichment
- ... an overview

Satisfaction form

<https://enquetes.inra.fr/index.php/84236>