Tutorial

Bioinformatics analysis of RNA-Seq data

Toulouse, 22-23 septembre 2015

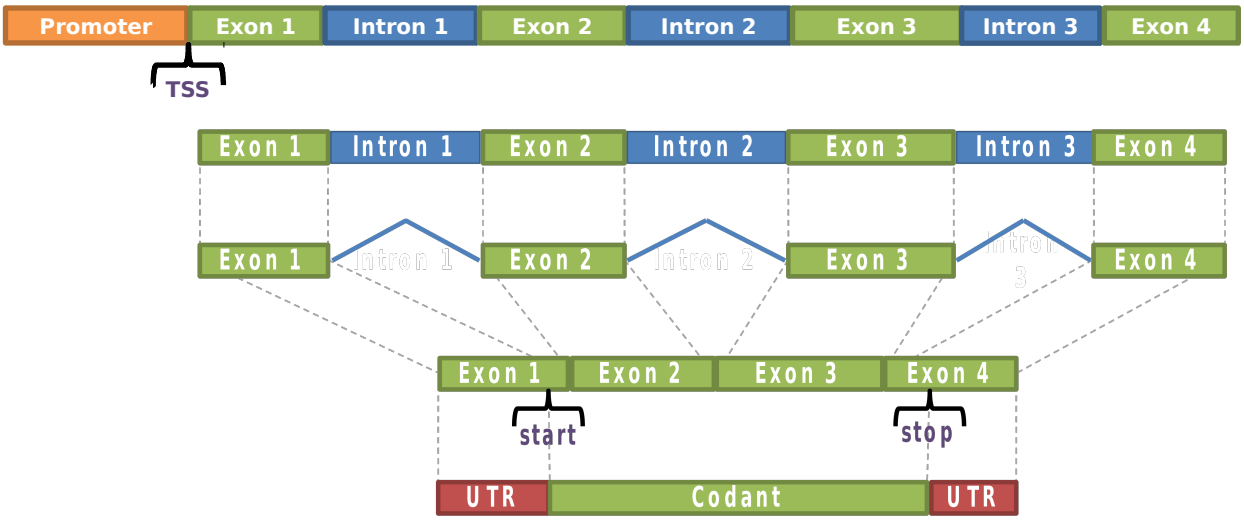
Céline Noirot
Plateforme Bioinformatique - INRA Toulouse

Content

1. RNAseq

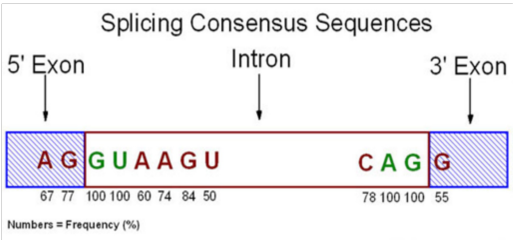
1. Reminders

Gene : functional unit of DNA that contains the instructions for creating a functional product



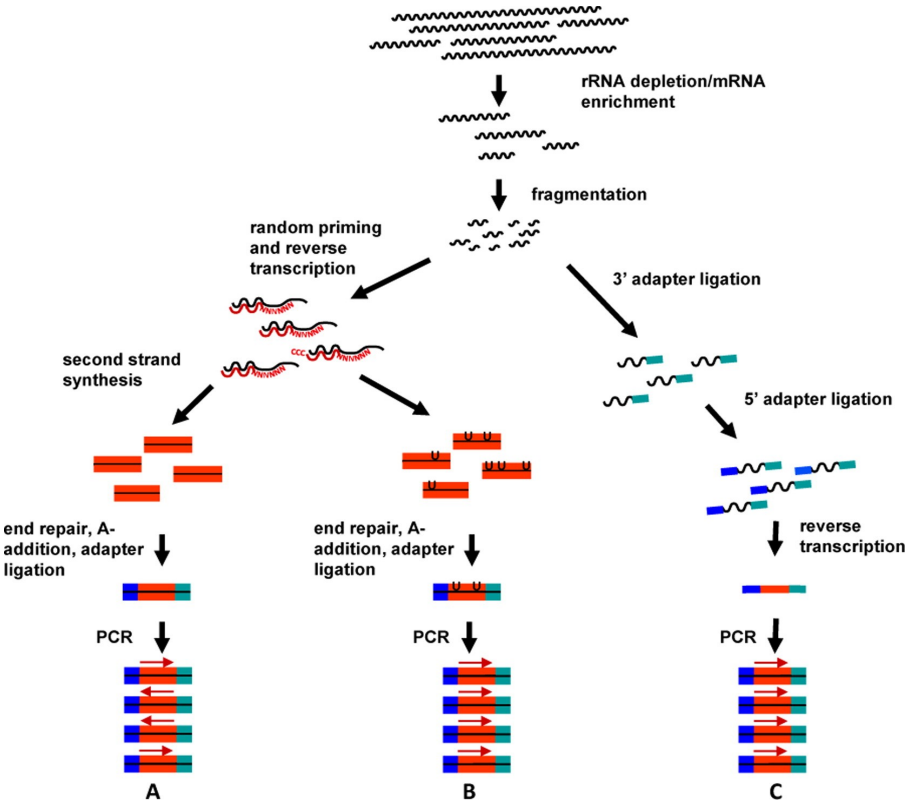
Promoter : ribosomal binding zone
TSS : transcription start site
Exon : region of the mRNA included in the transcript
Intron : non transcribed region

Splicing : introns excision before translation
Transcript : portion of DNA transcribed into RNA molecule
UTR : Untranslated region

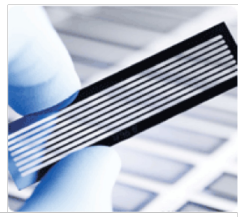


2. RNAseq library preparation

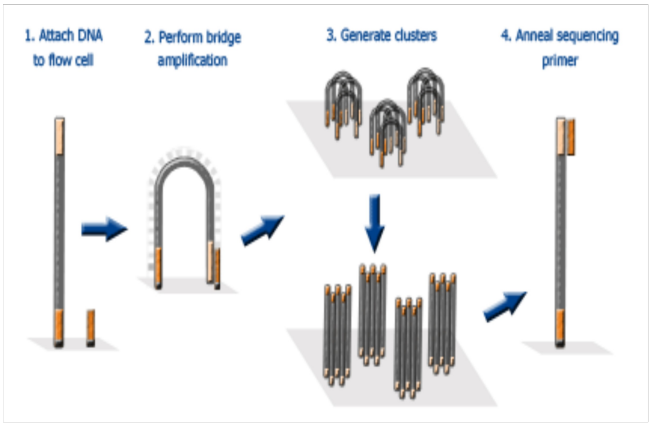
The most common RNA-seq protocols fall in three main classes. (A) Classical Illumina protocol. Random-primed double-stranded cDNA synthesis is followed by adapter ligation and PCR. (B) One class of strand-specific methods relies on marking one strand by chemical modification. The dUTP second strand marking method follows basically the same procedure as the classical protocol except that dUTP is incorporated during second strand cDNA synthesis, preventing this strand from being amplified by PCR. Most current transcriptome library preparation kits follow the dUTP method. (C) The second class of strand-specific methods relies on attaching different adapters in a known orientation relative to the 5′ and 3′ends of the RNA transcript. The Illumina ligation method is a well-know example of this class and is based on sequential ligation of two different adapters. Most current small RNA library preparation kits follow the RNA ligation method.

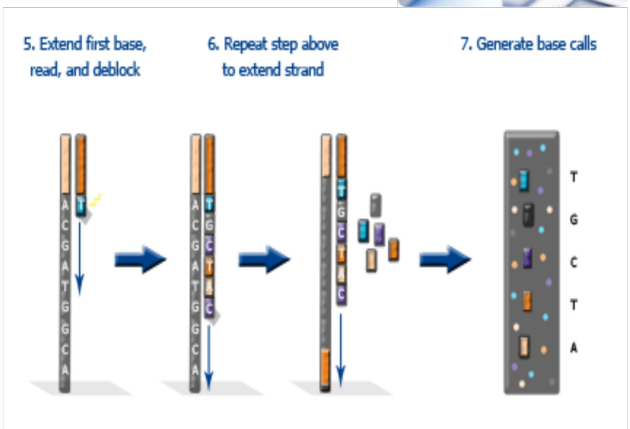


More info about Illumina sequencing :



1 Flowcell : 8 Lane
1 flowcell Hiseq 2500 : 2 Billion of reads single or 4 Billion of paired reads.





3. Different choices

Depletion or enrichment?

- rRNA depletion (eucaryote or procaryote)
- enrichment by poly-A selection (eucaryote)

More info: *Comparison of RNA-Seq by poly (A) capture, ribosomal RNA depletion, and DNA microarray for expression profiling, BMC Genomics , 2014*

Single-end or Paired-end?

- Specific adapter
- Mapping more accurate

Use a strand specific methods?

- Useful for studying anti-sense expression

Multiplexing?

- Add tag sequence to group multiple samples to be sequenced on a single sequencing run

Number of replicates?

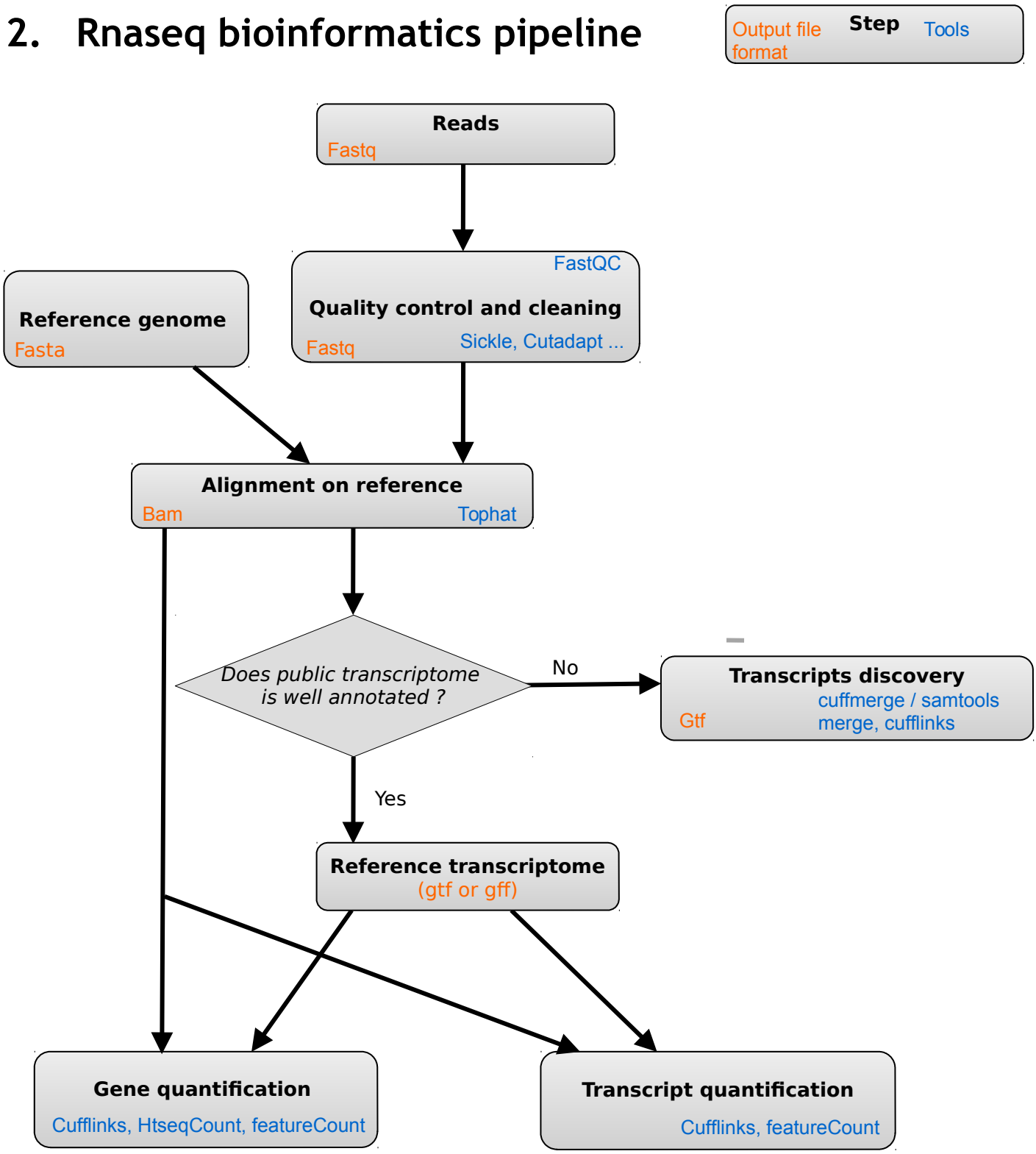
- Balance depth / number of repetitions:
    - at least 3 biological replicates
    - Pearson correlation between 2 samples must be > 0.92
    - If correlation < 0.9 , this should be repeated or redone

- Number of read : Between 10M and 100M reading samples according to the study.

More info : *A Comparative Study of Techniques for Differential Expression Analysis on RNA-Seq Data Zhang et al. 2014*

---

2. Rnaseq bioinformatics pipeline



3. Files format

1. FASTA

File type	Sequence
Name meaning	Format used by the tool named 'FastA' (fast alignment)
Who generates it?	Almost all
Who reads it?	Almost all, you

Example

```
>sequence1
CGATGTACGCTAGAT
```

Explanations

Each sequence begins with a '>', followed by the name of the sequence. Although this is not mandatory, it is recommended that the name of the sequence is unique within the file . The sequence itself follows .

2. FASTQ

File type	Reads
Name meaning	As FASTA, with the quality (Q)
Who generates it?	Sequencers
Who reads it?	Mapping tools, visualization tools, you

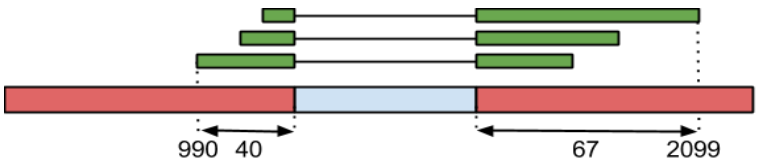
Example

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Explanations

A FASTQ file normally uses four lines per sequence.

- Line 1 begins with a '@' character and is followed by a sequence identifier and an *optional* description (like a FASTA title line).
- Line 2 is the raw sequence letters.
- Line 3 begins with a '+' character and is *optionally* followed by the same sequence identifier (and any description) again.
- Line 4 encodes the quality values for the sequence in Line 2.



The quality is encoded, each character correspond to a number . In general, the association is as follows:

!	"	#	$	%	&	'	(	)	*	+	,	-	.	/	0	1	2	3	4
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

5	6	7	8	9	:	;	<	=	>	?	@	A	B	C	D	E	F	G	H	I
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40

Each number represent the probability p that the base call is incorrect. The standard Sanger variant to assess reliability of a base call is $Q=-10\log_{10}(p)$. For Example , the character C code number 34. It therefore represents a probability of error of about 4.10^{-4}. The rightmost codes therefore represent the best qualities.
Caution: for relatively old data , there are other quality encodings (i.e. other associations between characters and numbers) .

More info

- Wikipedia Page : http://en.wikipedia.org/wiki/FASTQ_format
- NAR article: http://nar.oxfordjournals.org/content/38/6/1767.full

---

3. BED

File type	Annotation
Name meaning	Browser Extensible Format
Who generates it?	Annotation tools, TopHat.
Who reads it ?	Viewer, you

Example

```
<chrom> <chromStart> <chromEnd> <name> <score> <strand> <thickStart> <thickEnd> <itemRgb> <blockCount> <blockSizes> <blockStarts>
chr1   990   2099   JUNC00001560   3   +   990   2099   255,0,0   2   40,67   0,1042
```

Explanations

Each line is an annotation. The information is tabulated, ie each row contains a fixed number of columns (here, 12), separated by tabs. **Only the 3 first fields are required.**
The BED format is used for many types of annotations. We describe here for the annotation of junctions:

1. (CHR 1) the number of chromosome (or scaffold)
2. (990) The starting position of the junction
3. (2099) The ending position of the feature
4. (JUNC00001560) Systematic junction name
5. (3) number of reads covering the junction
6. (+) strand
7. (990) same as column 2
8. (2099) same as column 3
9. (255,0,0) not important
10. (2) not important
11. (40.67) maximum size readings covering exon left and right of the intron.
12. (0.1042) not important

More info

- Documentation : http://genome.ucsc.edu/FAQ/FAQformat.html#format1

---

4. GTF

File type	Annotation
Name meaning	Gene Transfer Format
Who generates it?	Annotation tools
Who reads it?	Visualization tools, TopHat, you

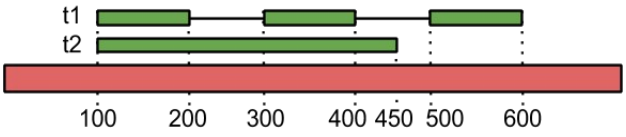
Example

```
<seqname> <source> <feature> <start> <end> <score> <strand> <frame> [attributes]
chr20   example   exon   100   200   .   +   .   gene_id "g1"; transcript_id "t1";
chr20   example   exon   300   400   .   +   .   gene_id "g1"; transcript_id "t1";
chr20   example   exon   500   600   .   +   .   gene_id "g1"; transcript_id "t1";
chr20   example   exon   100   450   .   +   .   gene_id "g1"; transcript_id "t2";
```

Explanations

This is another format to 9 tabbed fields. Each line contains a feature description:

1. (chr20) chromosome
2. (example) the source of the annotation, usually the tool that generated the annotation
3. (exon) the type of annotation; here we have exons, but it could CDS, start, intron ...
4. (100) the beginning of the annotation
5. (200) the ending of the annotation
6. (.) The score field indicates a degree of confidence in the feature's existence and coordinates
7. (+) The strand
8. (.) The frame
9. (gene_id "g1"; transcript_id "t1";) Attributes. This is a catch-all. One can find the common name of the gene.



Mandatory attributes :

- *gene_id value;* A globally unique identifier for the genomic locus of the transcript. If empty, no gene is associated with this feature.
- *transcript_id value;*A globally unique identifier for the predicted transcript. If empty, no transcript is associated with this feature.

More info

- Documentation : http://mblab.wustl.edu/GTF22.html
- The GTF is specifically adapted GFF format less constrained. Documentation GFF size: http://www.sequenceontology.org/gff3.shtml

---

5. SAM

File type	Mapping
Name meaning	Sequence Alignment/Map
Who generates it?	Mapping tools (BWA, Bowtie, STAR ...)
Who reads it?	Samtools, you

Example

```
@SQ SN:chr1 LN:45
r001  99   chr1  7  30  8M2I4M1D3M =  37 39  TTAGATAAAGGATACTG *
r002  0    chr1  9  30  3S6M1P1I4M *  0  0   AAAAGATAAGGATA    *
r003  0    chr1  9  30  5S6M       *  0  0   GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004  0    chr1  16 30  6M14N5M    *  0  0   ATAGCTTCAGC       *
r003  2064 chr1  29 17  6H5M       *  0  0   TAGGC            * SA:Z:ref,9,+,5S6M,30,1;
r001  147  chr1  37 30  9M         =  7  -39 CAGCGGCAT         * NM:i:1
```

Explanations

It is a TAB-delimited text format consisting of a header section, which is optional, and an alignment section. Header lines start with '@', while alignment lines do not.

The header provides information on the genome or the mapping . The header lines all start with an @ , followed by two letters. The line @SQ SN : CHR1 LN : 45 reads :

- @ : We are in a header
- SQ: with respect to a reference sequence (chromosomes)
- SN : CHR1 : The name of a sequence is CHR1
- LN : 45: its size is 45 bp

There are many different types of header that we will not strudy here.

The body is tabulated. Each line has 11 mandatory fields. These fields always appear in the same order and must be present, but their values can be `0' or `*' (depending on the field) if the corresponding information is unavailable.

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	`[!-?A-~]{1,255}`	Query template NAME
2	FLAG	Int	$[0,2^{16}-1]$	bitwise FLAG
3	RNAME	String	`\*|[!-()+-<>-~][!-~]*`	Reference sequence NAME
4	POS	Int	$[0,2^{31}-1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0,2^{8}-1]$	MAPping Quality
6	CIGAR	String	`\*|([0-9]+[MIDNSHPX=])+`	CIGAR string
7	RNEXT	String	`\*|=|[!-()+-<>-~][!-~]*`	Ref. name of the mate/next read
8	PNEXT	Int	$[0,2^{31}-1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31}+1,2^{31}-1]$	observed Template LENgth
10	SEQ	String	`\*|[A-Za-z=.]+`	segment SEQuence
11	QUAL	String	`[!-~]+`	ASCII of Phred-scaled base QUALity+33

Here is the alignment corresponding to the previous SAM example.

```
Coor              11111  11111222222222233333333333444444
         12345678901234  56789012345678901234567890123345
chr1     AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

+r001/1       TTAGATAAAGGATA*CTG
+r002        aaaAGATAA*GGATA
+r003      gcctaAGCTAA
+r004                      ATAGCT..............TCAGC
-r003                          ttagctTAGGC
-r001/2                                      CAGCGGCAT
```

The CIGAR format (Compact Idiosyncratic Gapped Alignment Report).
It details the alignment of a read on a reference sequence. Alignment is read from left to right, and composed by a sequence of pairs (number, letter). For Example , the cigar 5M1I5M consist of:

- 5M: 5 matches between the fragment and sequence
- 1I: insertion into the fragment
- 5M: 5 more matches.

```
read: ACGTAGATCGA
chr1: ACGTA-ATCGA
```

The possible letters are :

- M: a match (careful, it may be a SNP , but not one indel)
- I: insertion relative to the reference
- D: deletion
- N: intron

There are other letters, we will not detail them here.

Other information about field 12 :
This is a catch-all field where each mapping tool of discretion that information to add. The information has special training, such as: TA: l: value, where:

- TA: is a pair of letter describing the field
- l: another letter (not important)
- value: the field value.

Here are some fields that may be of interest:

- NM: number of mismatches (counting indels) in alignment
- AS: Alignment score generated by aligner
- XN: number of ambiguous bases
- XM: number of mismatches (excluding indels)
- XO: number of openings of gaps
- X0: optimal number of matches
- X1: number of sub-optimal matches
- MD: String for mismatching positions.
- RG: Read group. Value matches the header RG-ID tag if @RG is present in the header.
- YT: Mapping description
  - UU: “single-end”
  - CP: the pair is correctly aligned
  - DP: the pair is not correctly aligned (ex : inversion or gene fusion)
  - UP: one fragment mapped
- SA: another mapping possible.

There are many other possible tags that vary with mapping tools.

More info

- Official documentation: http://samtools.github.io/hts-specs/SAMv1.pdf
- Bioinformatics article introducing the format : http://bioinformatics.oxfordjournals.org/content/25/16/2078.long
- SAM format is not intended to be read by programs . Format BAM (below) is made to it.

---

6. BAM

File type	Mapping
Name meaning	Binary sAM
Who generates it?	Samtools, Mapping tools
Who reads it?	Visualization tools, alignment processing tools

Explanations

It is simply the SAM file, binary (easily readable for a machine) and compressed.

---

7. BAI

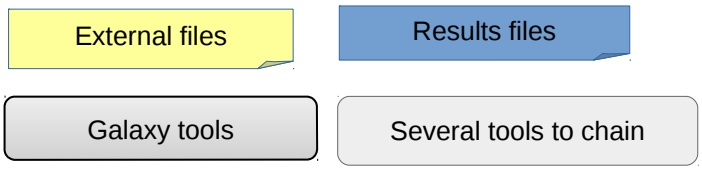
File type	Index
Name meaning	BAm Index
Who generates it?	Samtools
Who read it?	Visualization tools

Explanations

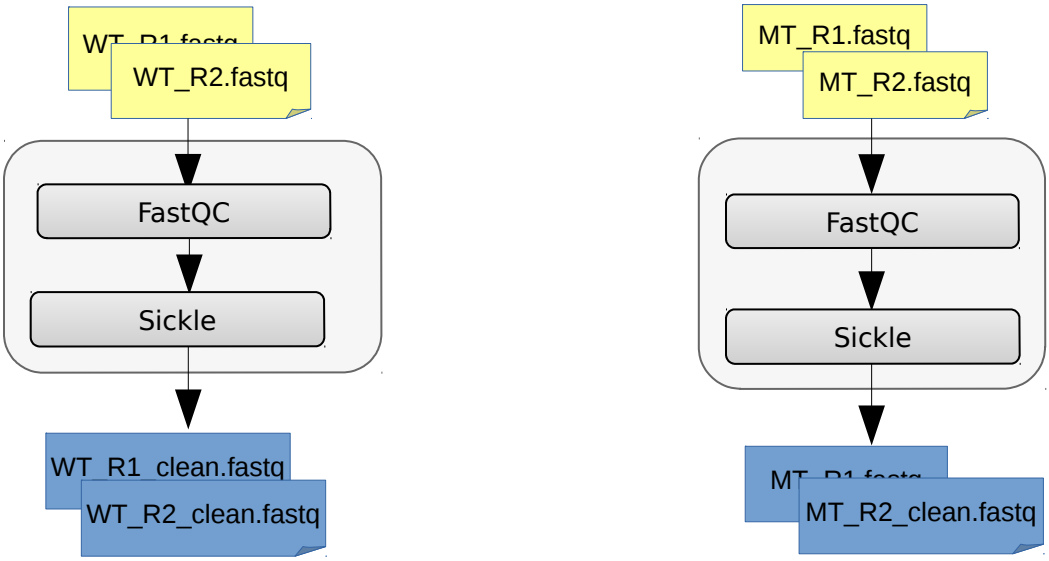
This is a binary file that indexes a BAM file. Can be seen as a "table of contents", which would serve as visualization tools to speed up the display of data in a BAM file (usually very large).

4. Detailed pipeline.

Here we detail a pipeline with two sample.



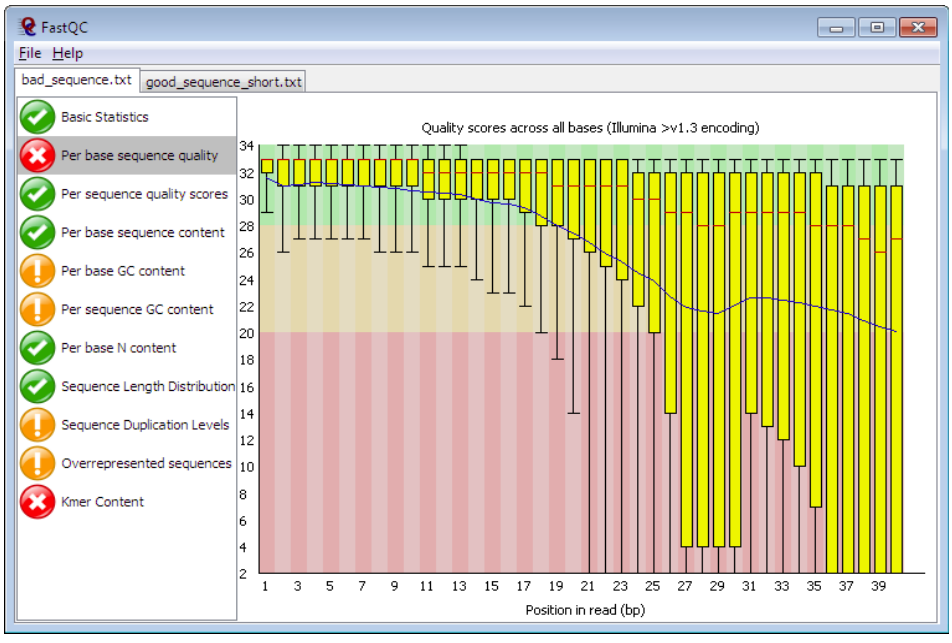
1. Quality control and cleaning



Modern high throughput sequencers can generate hundreds of millions of sequences in a single run. Before analysing this sequence to draw biological conclusions you should always perform some simple quality control checks to ensure that the raw data looks good and there are no problems or biases in your data which may affect how you can usefully use it.
FastQC is a DNA-specific software, so several test will failed.

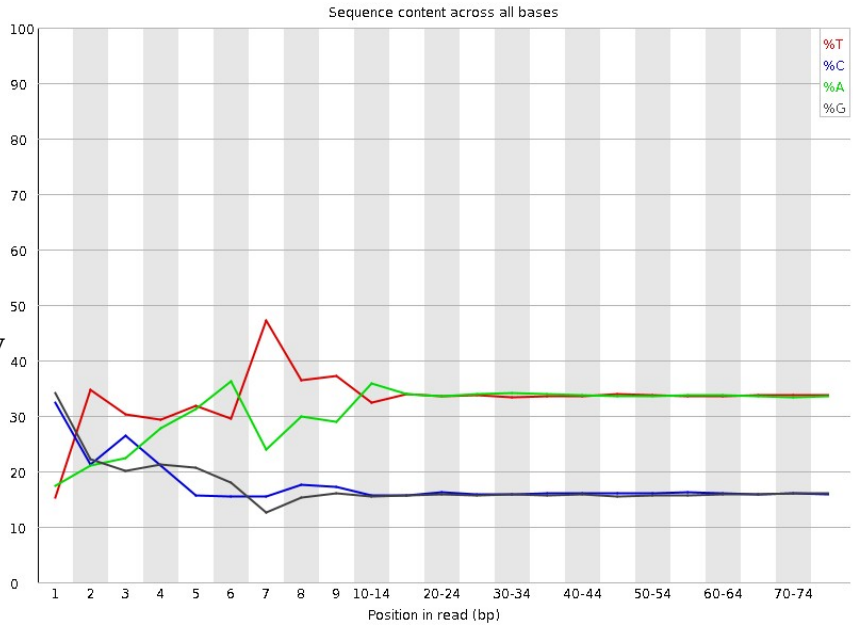
Quality along the reads

Base of reads with quality lower than 30 should be remove.



Random hexamer priming

Generation of cDNA using random hexamer priming induces biases in the nucleotide composition at the beginning of transcriptome sequencing reads from the Illumina Genome Analyzer. The bias is independent of organism and laboratory and impacts the uniformity of the reads along the transcriptome. We provide a read count reweighting scheme, based on the nucleotide frequencies of the reads, that mitigates the impact of the bias.

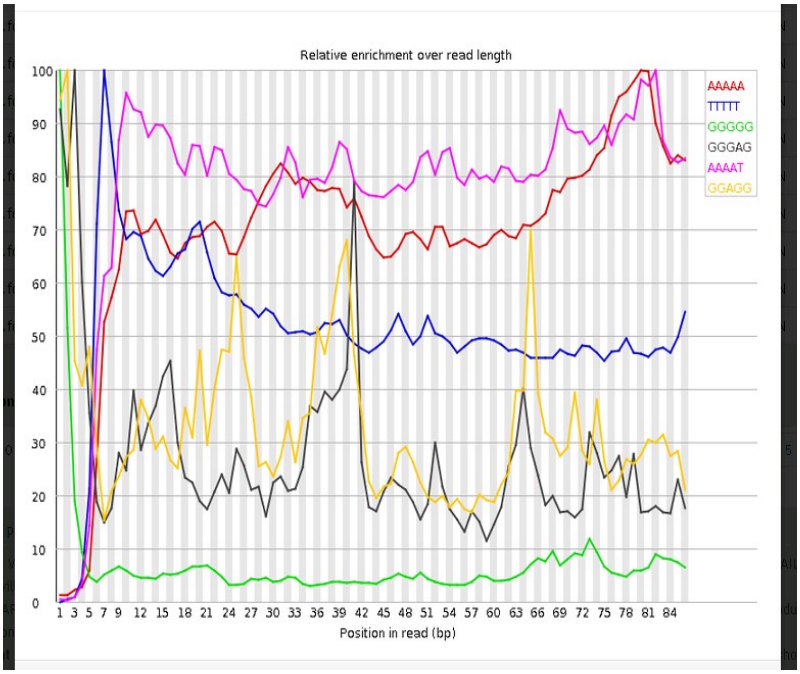


http://nar.oxfordjournals.org/content/38/12/e131.long

Kmer content (PolyA/PolyT)

The analysis of overrepresented sequences will spot an increase in any exactly duplicated sequences, but there are a different subset of problems where it will not work.

- If you have very long sequences with poor sequence quality then random sequencing errors will dramatically reduce the counts for exactly duplicated sequences.
- If you have a partial sequence which is appearing at a variety of places within your sequence then this won't be seen either by the per base content plot or the duplicate sequence analysis.



Overexpressed sequences : detect adapter

Overrepresented sequences

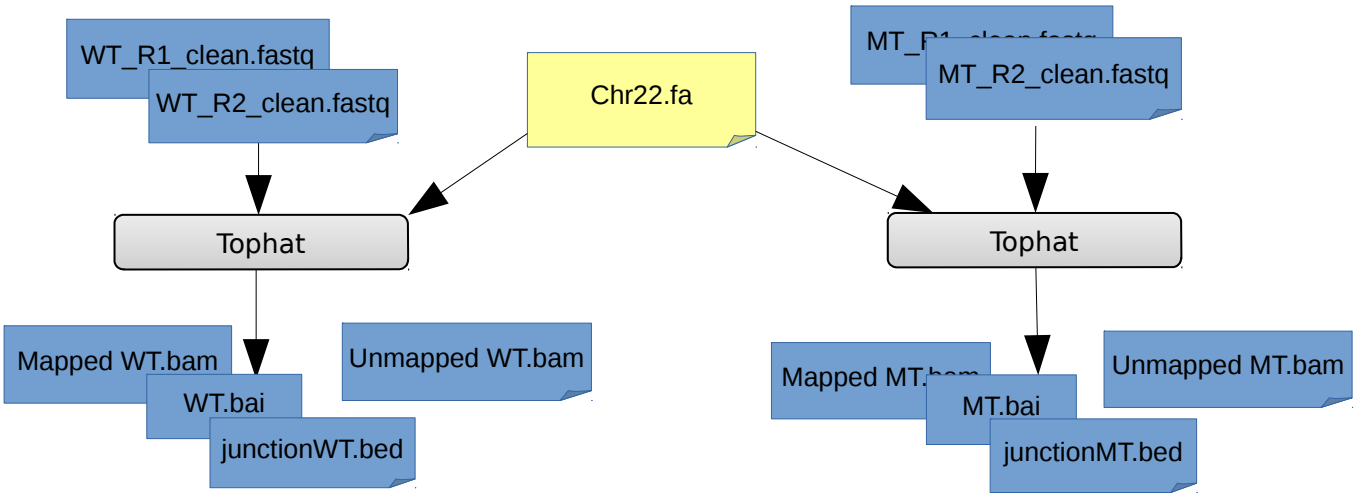
Sequence	Count	Percentage	Possible Source
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCGATCT	8122	8.122	Illumina Paired End PCR Primer 2 (100% over 40bp)
GATCGGAAGAGCGGTTCAGCAGGAATGCCGAGATCGGAAG	5086	5.086	Illumina Paired End PCR Primer 2 (97% over 36bp)

---

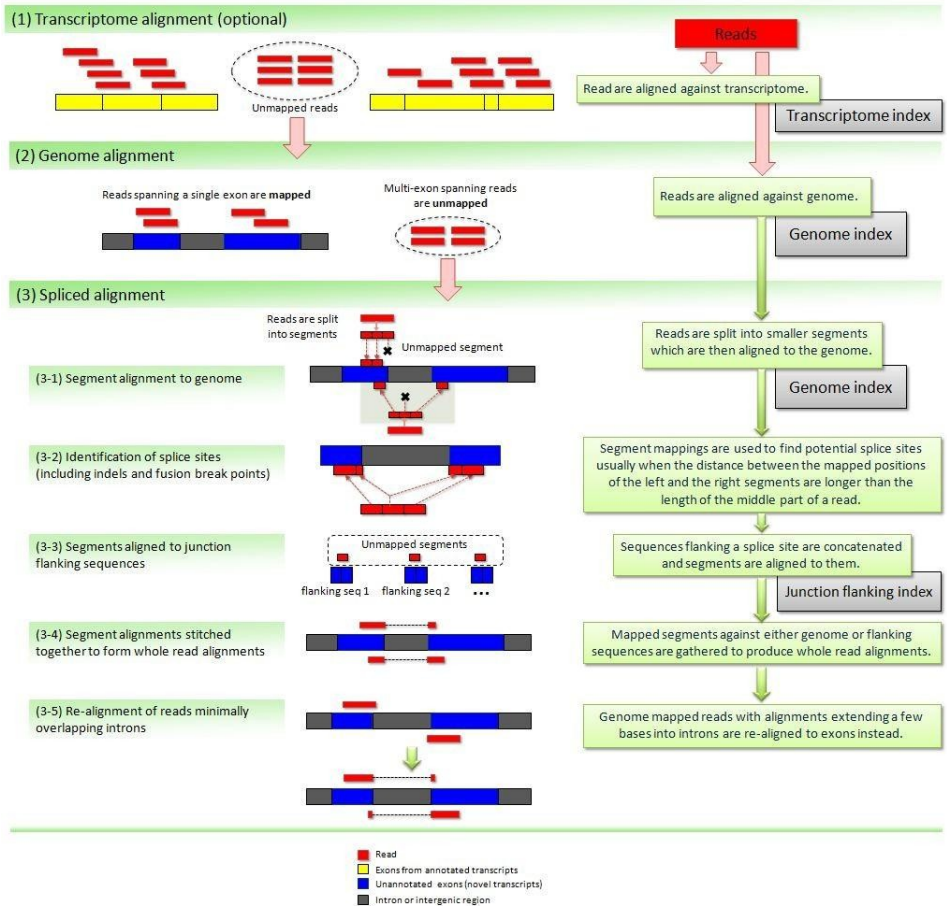
2. Alignment on reference genome

If the genome of interest does not exist in Galaxy Tophat tool, please make a request to support.

Splicing alignment can be performed with Tophat (which is based on Bowtie2). If you have a transcriptome file (GTF), provide it.



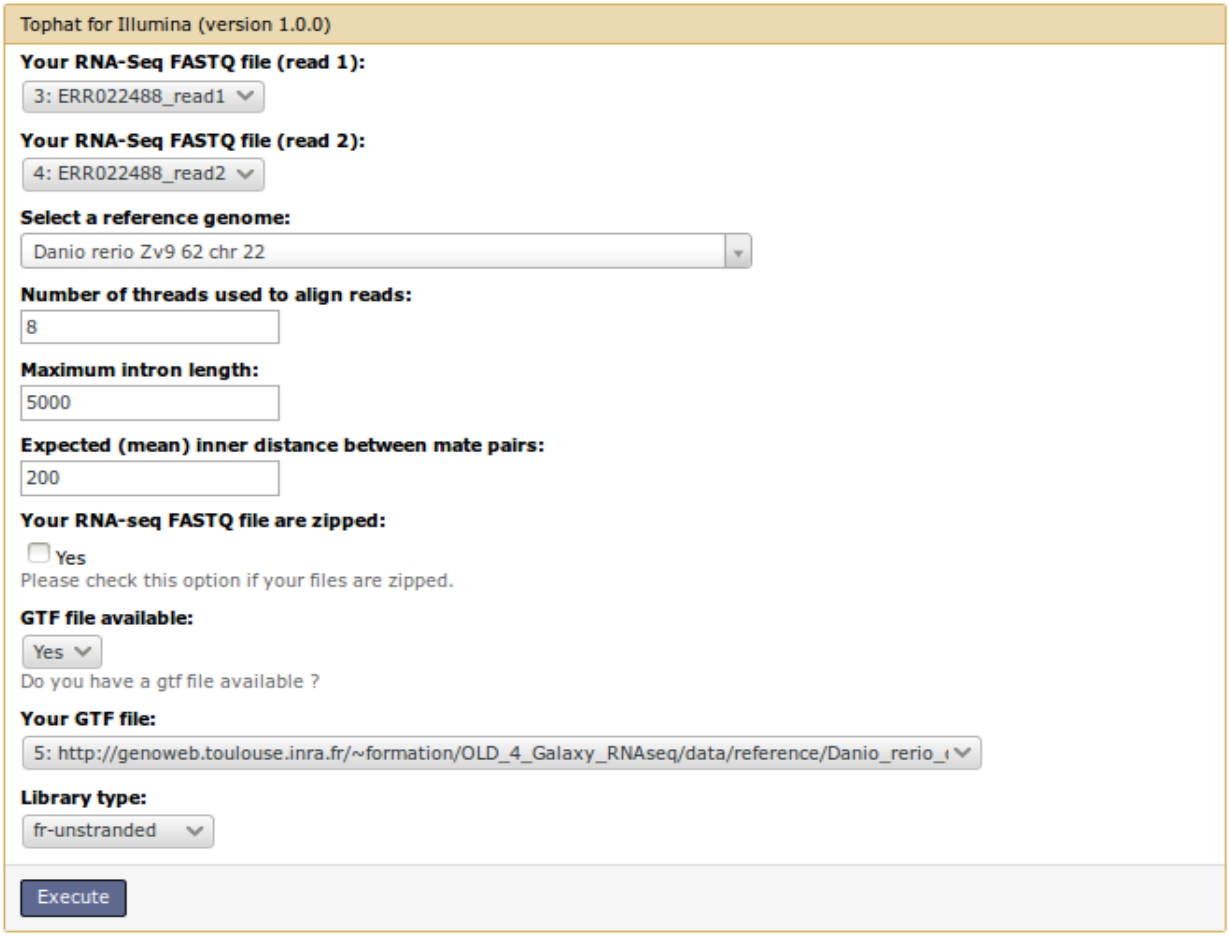
Tophat overview :



More information about spliced aligner:

- Tophat, A spliced read mapper for RNA-Seq - **http://ccb.jhu.edu/software/tophat/index.shtml**
- STAR is an ultrafast universal RNA-seq aligner - https://code.google.com/p/rna-star/
- Tools comparison : The RNA-seq Genome Annotation Assessment Project (Engström et al., Nature Methods, 2013) - http://www.nature.com/nmeth/journal/v10/n12/full/nmeth.2714.html

Tophat galaxy tool



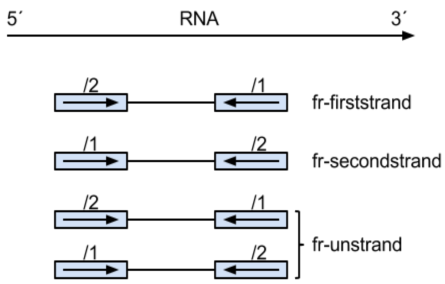
Number of threads : to parallelized job use 8 or 16

Maximum intron length : Depend on your species

TopHat documentation : « Usually, setting the maximum intron size to 4 or 5 Kb is sufficient to discover most junctions while keeping the number of false positives low. »

Library type (eg figure) :

Depend on your protocol



fr-unstranded	*Standard Illumina*	*Reads from the left-most end of the fragment (in transcript coordinates) map to the transcript strand, and the right-most end maps to the opposite strand.*
fr-firststrand	*dUTP, NSR, NNSR*	*Same as above except we enforce the rule that the right-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during first strand synthesis is sequenced.*
fr-secondstrand	*Ligation, Standard SOLiD*	*Same as above except we enforce the rule that the left-most end of the fragment (in transcript coordinates) is the first sequenced (or only sequenced for single-end reads). Equivalently, it is assumed that only the strand generated during second strand synthesis is sequenced.*

Expected mean inner distance between mate

Important parameter which depend on your experience. Usualy 500bp

---

3. Discovering new transcript

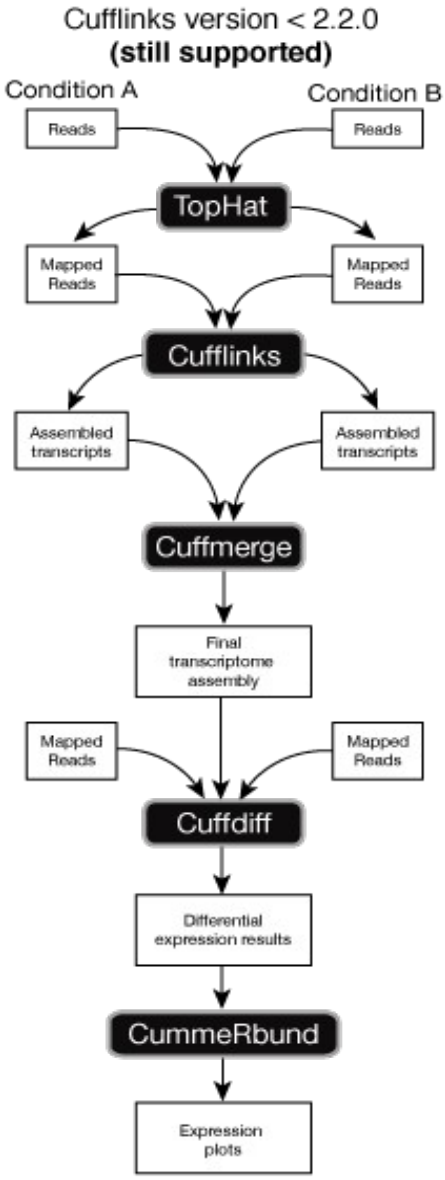
Cufflinks overview

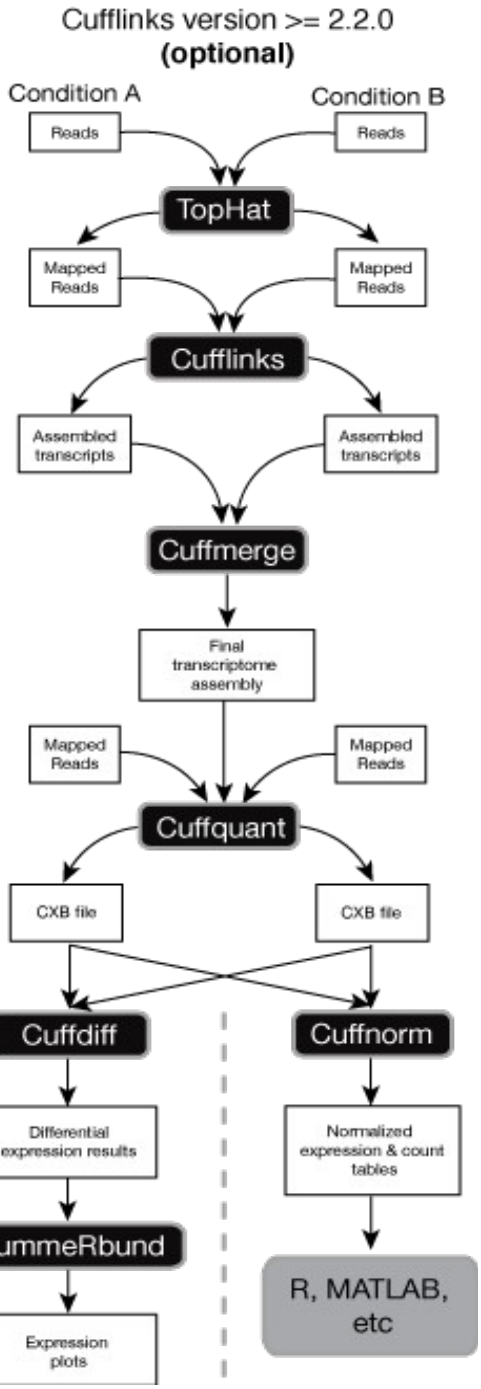
Cufflinks is both the name of a suite of tools and a program within that suite. The program “Cufflinks” assembles transcriptomes from RNA-Seq data and quantifies their expression.

Cufflinks:

- **discovers new transcript (assemble transcript)**
- quantifies the abundance (no raw count, so cannot be use with edgeR or DEseq)
- compares annotations (cuffcompare)
- performs differential analysis (cuffdiff)

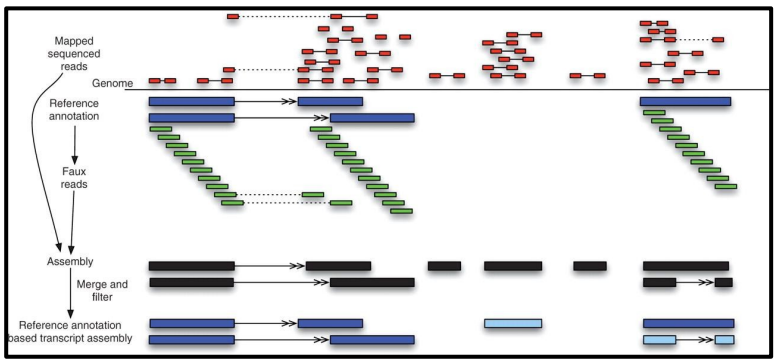
Cufflinks pipeline



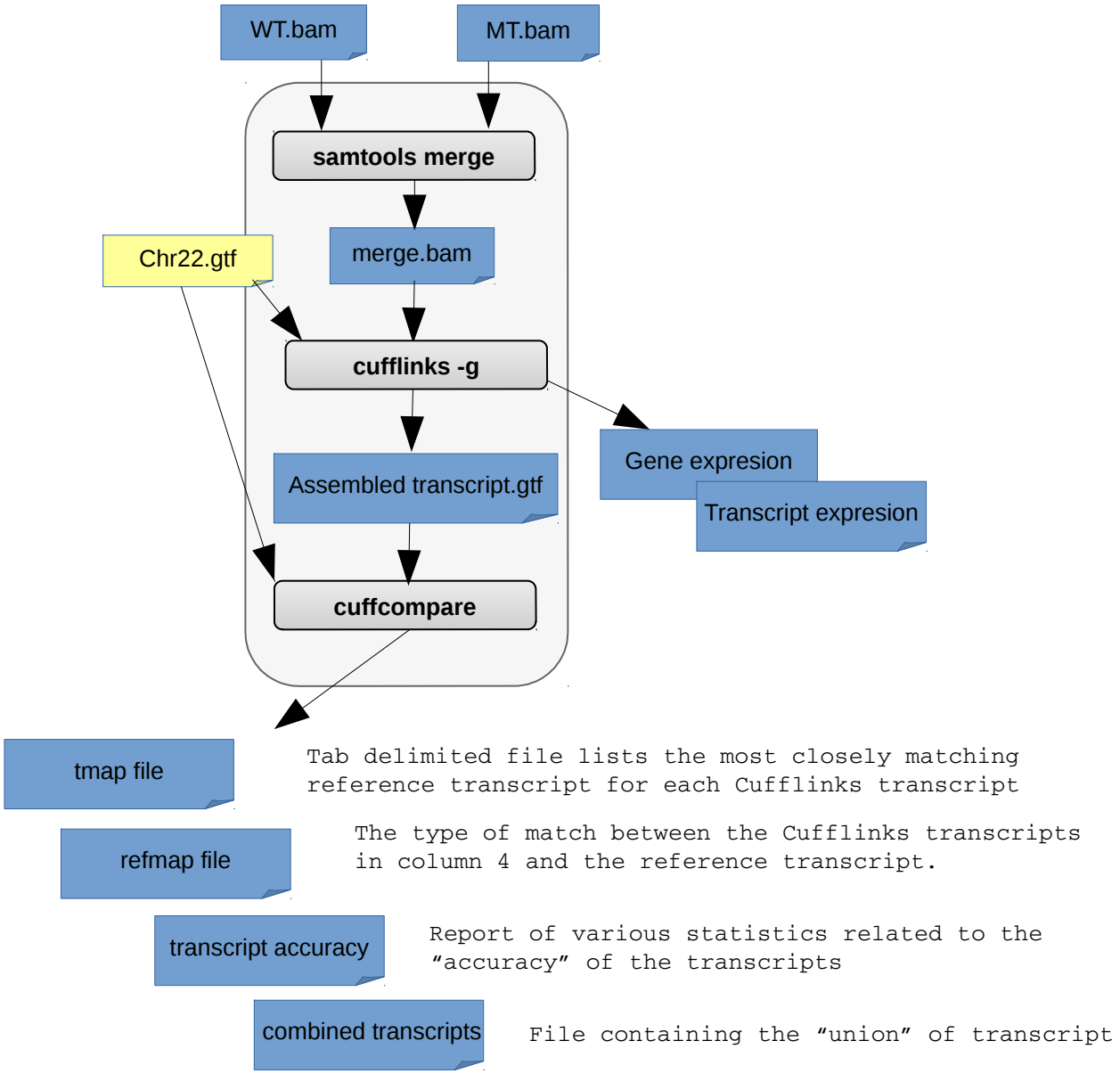


Cufflink transcript assembly

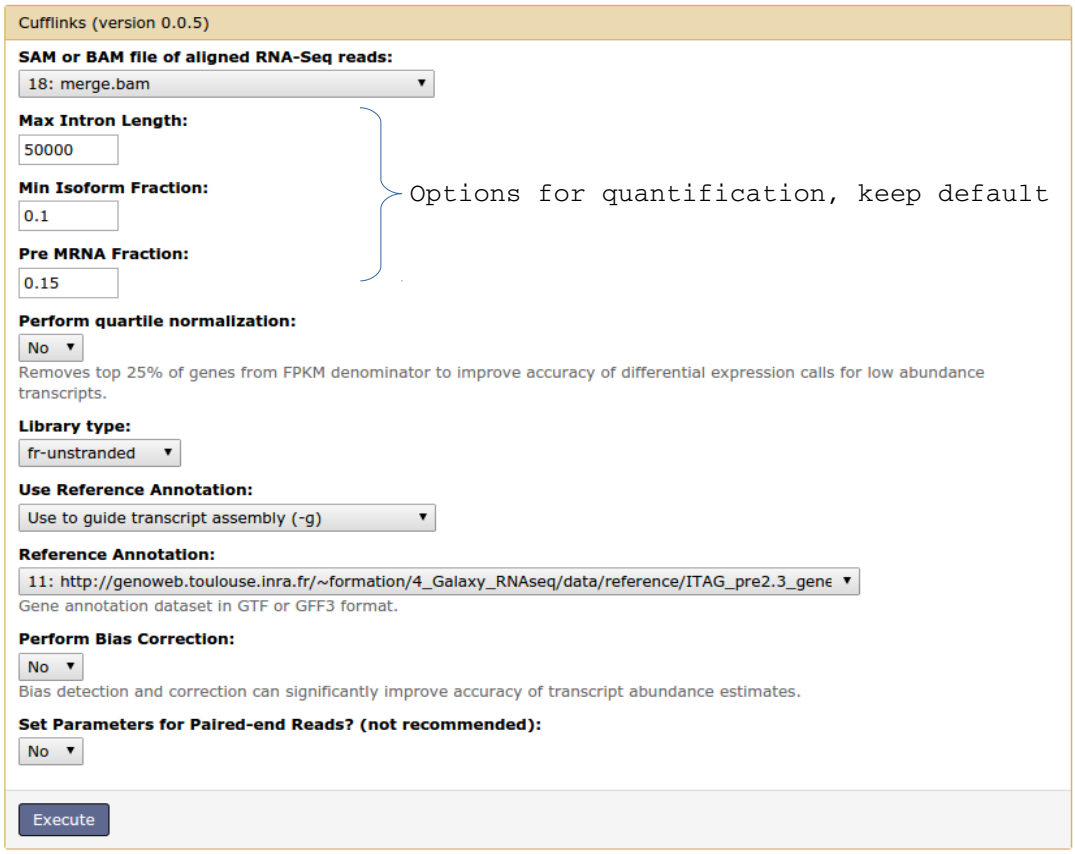
Illustration of RABT (Reference Annotation Based Transcripts Assembly) algorithm:



To discover the maximum of transcript and alternative form you should use all the condition, so we suggest you to merge all alignment, and then to discover new transcript using the option “Use guide transcript assembly.”



Cufflinks Galaxy tool view :



Max intron len :

The maximum intron length. Cufflinks will not report transcripts with introns longer than this, and will ignore SAM alignments with REF_SKIP CIGAR operations longer than this. The default is 300,000.

Library type :

Tells Cufflinks to use the supplied reference annotation a GFF file to guide RABT assembly.

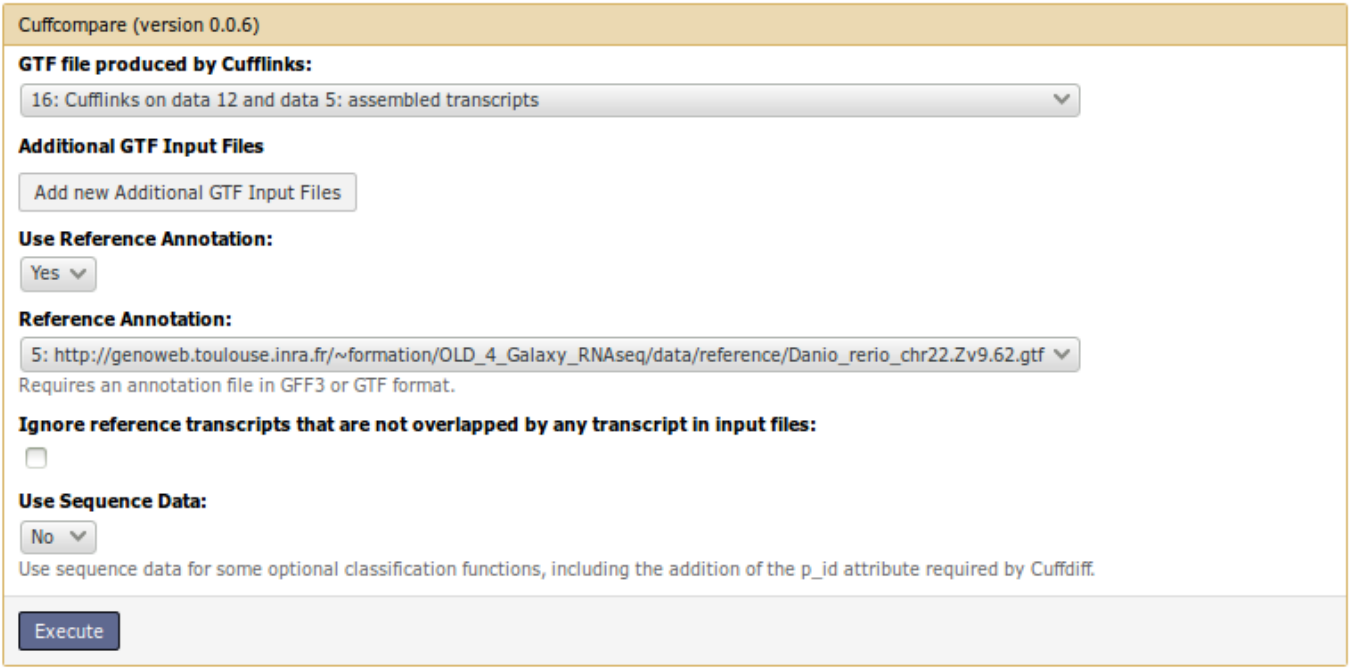
Use reference annotation :

Tells Cufflinks to use the supplied reference annotation a GFF file to guide RABT assembly. Reference transcripts will be tiled with faux-reads to provide additional information in assembly. Output will include all reference transcripts as well as any novel genes and isoforms that are assembled.

Cuffcompare Galaxy tool view :

The program cuffcompare:

- compares your assembled transcripts to a reference annotation
- tracks Cufflinks transcripts across multiple experiments (e.g. across a time course)



Output files :

1. transcript accuracy : Report of various statistics related to the « accuracy » of the transcripts
2. tmap file : Tab delimited file lists the most closely matching reference transcript for each Cufflinks transcript
3. refmap file : The type of match between the Cufflinks transcripts in column 4 and the reference transcript.
4. combined transcripts : File containing the « union » of transcript

Transfrag class codes which are in previous defined files 2,3,4 :

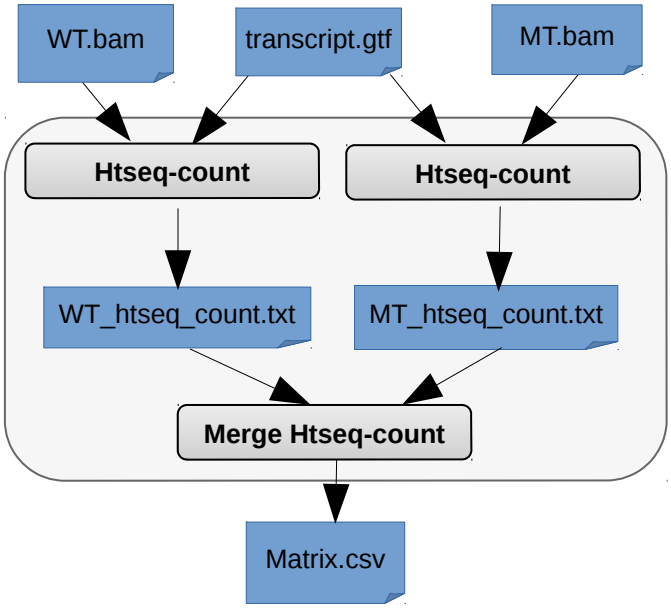
Priority	Code	Description
1	=	Complete match of intron chain
2	c	Contained
3	j	Potentially novel isoform (fragment): at least one splice junction is shared with a reference transcript
4	e	Single exon transfrag overlapping a reference exon and at least 10 bp of a reference intron, indicating a possible pre-mRNA fragment.
5	i	A transfrag falling entirely within a reference intron
6	o	Generic exonic overlap with a reference transcript
7	p	Possible polymerase run-on fragment (within 2Kbases of a reference transcript)
8	r	Repeat. Currently determined by looking at the soft-masked reference sequence and applied to transcripts where at least 50% of the bases are lower case

Priority	Code	Description
9	u	Unknown, intergenic transcript
10	x	Exonic overlap with reference on the opposite strand
11	s	An intron of the transfrag overlaps a reference intron on the opposite strand (likely due to read mapping errors)
12	.	(.tracking file only, indicates multiple classifications)

Once you have your reference transcriptome you can perform quantification.

4. Quantification with Htseq-count

Purpose : count how many reads map to each feature (gene).



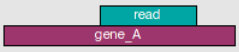
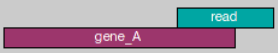
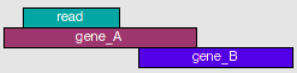
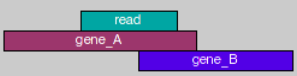
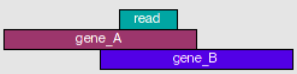
HtseqCount principe

A feature is here an interval (i.e., a range of positions) on a chromosome or a union of such intervals.

In the case of RNA-Seq, the features are typically genes, where each gene is considered here as the union of all its exons. One may also consider each exon as a feature, e.g., in order to check for alternative splicing. For comparative ChIP-Seq, the features might be binding region from a pre-determined list.

Special care must be taken to decide how to deal with reads that overlap more than one feature. The htseq-count script allows to choose between three modes.

The following figure illustrates the effect of these three modes.

	union	intersection_strict	intersection_nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous	gene_A	gene_A
	ambiguous	ambiguous	ambiguous

HtseqCount Galaxy Tool

Skip all reads with alignment quality lower than… :
Default 0 ; 30 is a good quality.
Feature type :
Use exon to count reads aligned to Exon.
GFF attribute :
Use gene_id to group reads count by gene_id.
If you set transcript_id, all reads which mapped to alternative form will be set as ambiguous.
BAM are sorted by:
by default by produced by tophat are sorted by Chr/pos.
Mode : view previously.

Output of HTseqCount
The script outputs a table with counts for each feature, followed by the special counters :

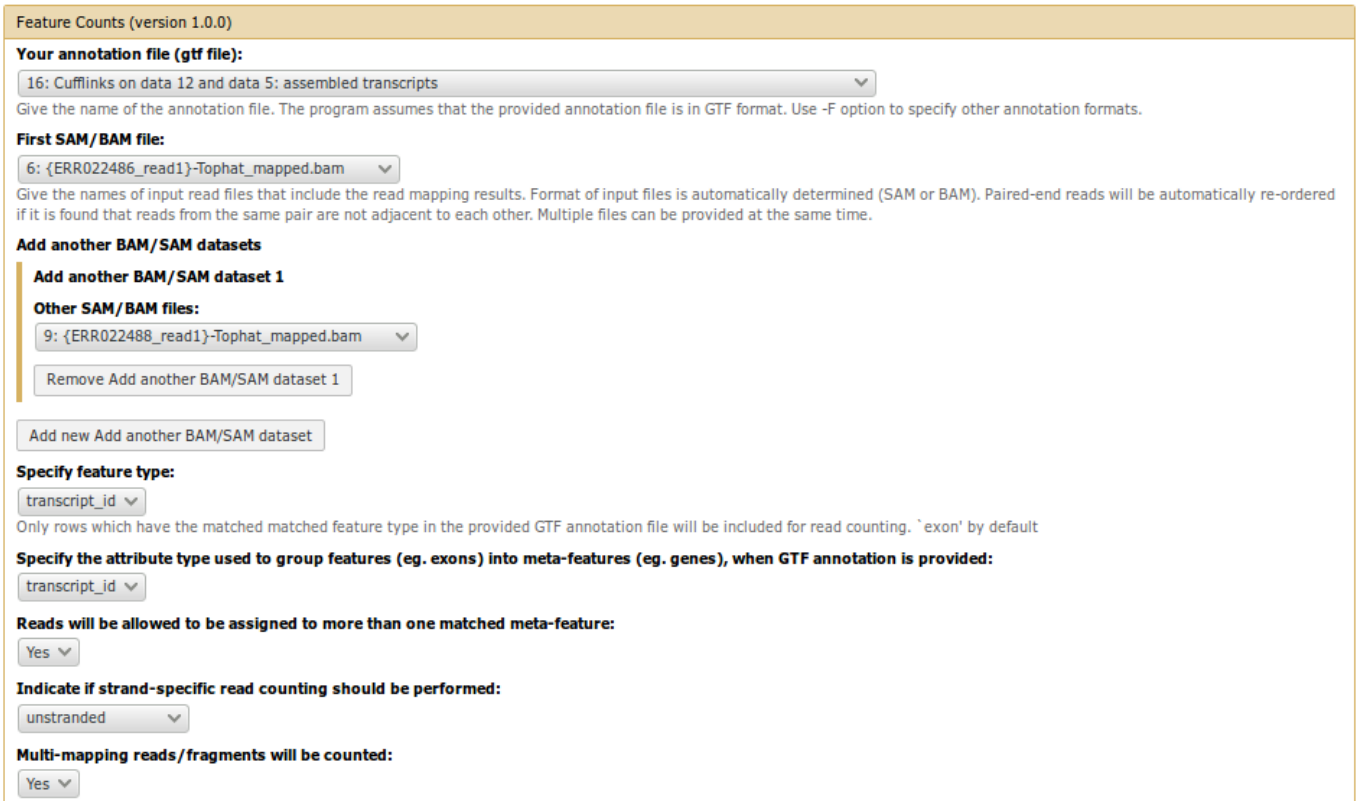
- __no_feature: reads (or read pairs) which could not be assigned to any feature (set *S* as described above was empty).
- __ambiguous: reads (or read pairs) which could have been assigned to more than one feature and hence were not counted for any of these (set *S* had mroe than one element).
- __too_low_aQual: reads (or read pairs) which were skipped due to the -a option, see below
- __not_aligned: reads (or read pairs) in the SAM file without alignment
- __alignment_not_unique: reads (or read pairs) with more than one reported alignment. These reads are recognized from the NH optional SAM field tag. (If the aligner does not set this field, multiply aligned reads will be counted multiple times, unless they getv filtered out by due to the -a option.)

5. Quantification with featureCounts

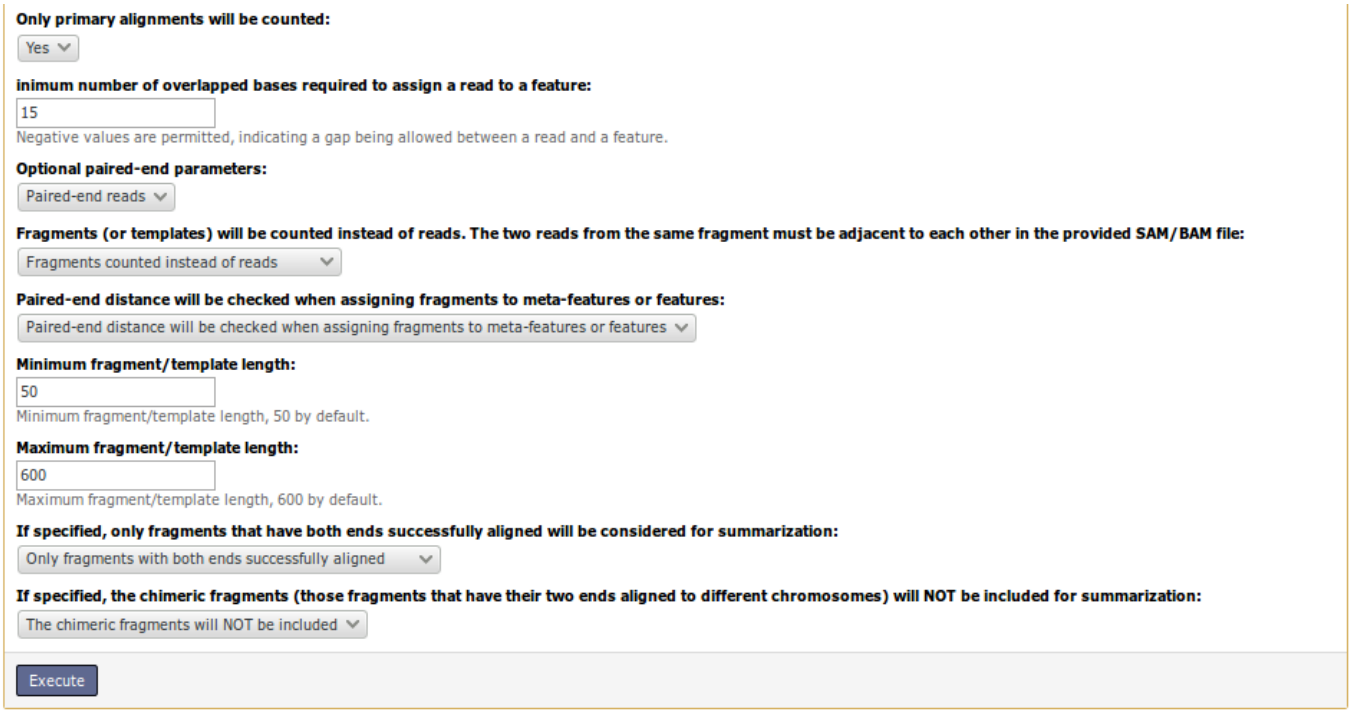
Purpose: transcript or gene quantification

- Quantification level: exon, gene, transcript,
- 1 read can be attributed to several feature,
- Reads with multiple alignment can be taken into account.
- Take several bam in input and directly generate matrix file.

FeatureCounts galaxy tool



Specify the feature type. Only rows which have the matched
matched feature type in the provided GTF annotation file
will be included for read counting. `exon' by default.
Specify the attribute type used to group features (eg. exons)
into meta-features (eg. genes), when GTF annotation is provided.
`gene_id' by default. This attribute type is usually the gene
identifier. This argument is useful for the meta-feature level
summarization.
Allow reads to be count several time : Yes/No
Multi-mapping reads/fragments (ie. a multi-mapping read will be counted up to N
times if it has N reported mapping locations). The program uses
the `NH' tag to find multi-mapping reads.
Indicate if **strand-specific** read counting should be performed.
It has three possible values: 0 (unstranded), 1 (stranded) and
2 (reversely stranded). 0 by default.



Only primary alignment: Primary and secondary alignments are identified using bit 0x100 in the Flag field of SAM/BAM files. All primary alignments in a dataset will be counted no matter they are from multi-mapping reads or not.

If specified, fragments (or templates) will be counted instead of reads. This option is only applicable for paired-end reads. The two reads from the same fragment must be adjacent to each other in the provided SAM/BAM file.

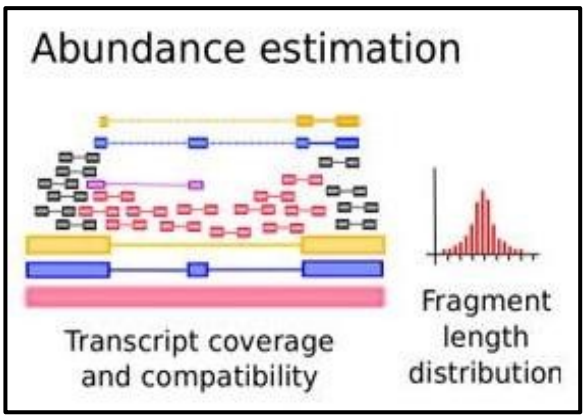
If specified, only fragments that have both ends successfully aligned will be considered for summarization. This option is only applicable for paired-end reads.

If specified, **the chimeric fragments** (those fragments that have their two ends aligned to different chromosomes) will NOT be included for summarization. This option is only applicable for paired-end read data.

6. Quantification with Cufflinks

Purpose : transcript estimation

If you want to use cufflinks to quantify we highly advise to use the whole package till differential expression like show in cufflinks presentation.
Here is an explanation about how cufflinks estimate the abundance and attribute reads to a feature.



RPKM :
Reads **P**er **K**ilobase of exon per **M**illion fragments mapped :

R = Number of mapped reads
N = Total number of reads in the library
L = Exon size in gene in bp

$$RPKM = \frac{10^9 \times R}{N \times L}$$

FPKM :
Fragments **P**er **K**ilobase of exon per **M**illion fragments mapped
1 pair of reads = 1 fragment

More general information about RNAseq/NGS

Seqanswer : *http://seqanswers.com/*
Biostar : *https://www.biostars.org/*
RNA-Seq blog : *http://rna-seqblog.com/*

NOTES