RNA-Seq data analysis

17-18 octobre 2019

Céline Noirot et Matthias Zytnicki

Genotoul Bioinfo

MIA TOULOUSE

Material

- **Slides:**
  - pdf : one per page
    http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/doc/Rnaseq_training_012019.pdf
  - pdf : three per page with comment lines

    http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/doc/Rnaseq_training_012019_3p.pdf

- **Hands on:**
  - Exercises:
    http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/doc/Tps/RNAseq_TP_ligne_cmd_enonce-Octobre2019.pdf
  - Data files: http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/data
  - Results files: http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/doc/Tps/Correction.txt

Session organisation

Day 1

Morning (9h00 -12h30) :

- Biological reminds
- Sequence quality
  Theory & exercises
- Spliced read mapping
  Theory & Exercises & Visualisation

Afternoon (14h-17h) :

- Expression quantification
  Theory + exercises

- mRNA calling
  Theory & exercises & Visualisation

Day 2

Morning (9h00 -12h30) :

- Models comparison
  Theory & exercises

- Hovering differential gene expression analyse

Summary – Biological reminds

- Transcriptome specificity
- High throughput sequencers
- Illumina protocol, paired-end library, directional library
- Experimental protocol
- RNAseq specific bias
- How to retrieve public data

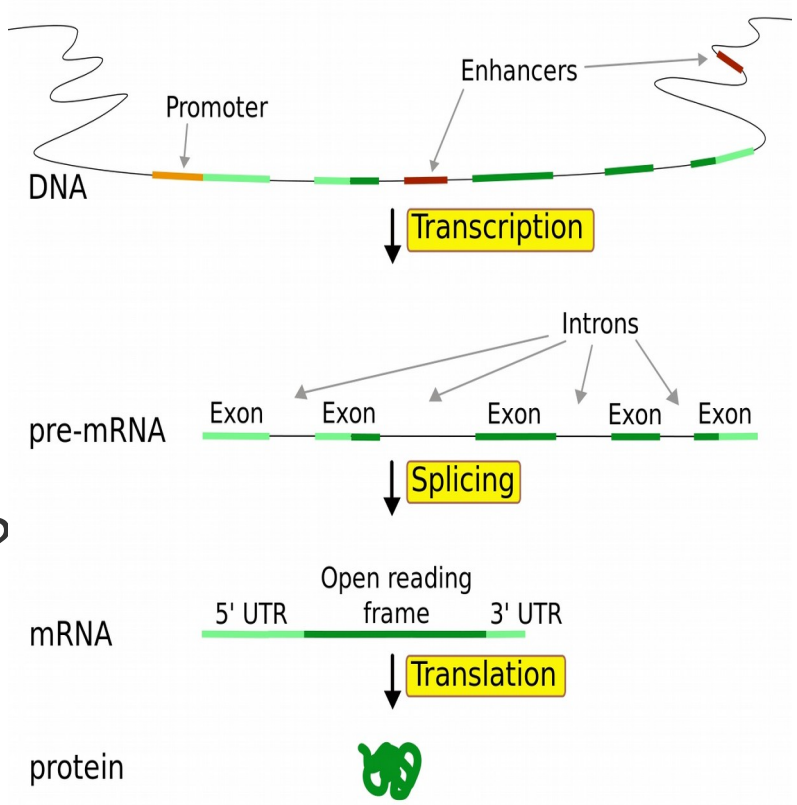
Context

Prerequis :

- Reference genome available
- RNAseq sequencing
  (sequence of transcript)

Try to answer to :

- How to map transcript to the genome ?
- How to discover new transcript ?
- What are the alternative transcript ?



Source : en.wikipedia.org/wiki/User:Forluvoft/sandbox

Transcriptome variability

- Many types of transcripts (mRNA, ncRNA, cis-natural antisense, fusion gene ...)
- Many isoform (non canonical splice sites, intron retention …)
- Number of transcripts
  - possible variation factor between transcripts: 10^6 or more,
  - expression variation between samples.
- Allele specific expression

Transcriptome variability *(ENCODE)*

GENCODE | Data | Stats

Statistics about the current Human GENCODE Release (version 28)

* The statistics derive from the **gtf file** that contains only the annotation of the main chromosomes.

For details about the calculation of these statistics please see the **README_stats.txt** file.

Compare with the previous release (GENCODE 27) »

Version 28 (November 2017 freeze, GRCh38) - Ensembl 92, 93

General stats

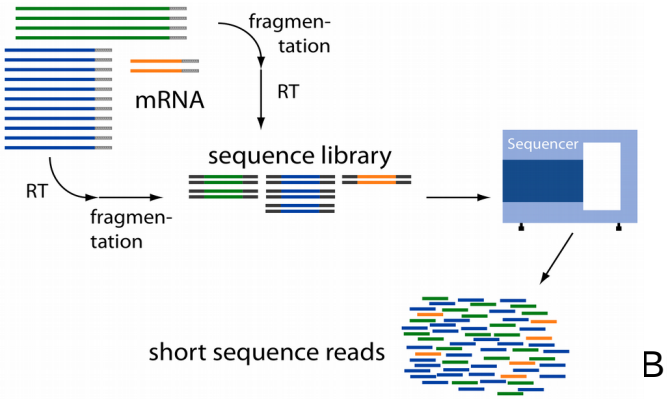
Total No of Genes	58381
Protein-coding genes	19901
Long non-coding RNA genes	15779
Small non-coding RNA genes	7569
Pseudogenes	14723
- processed pseudogenes:	10693
- unprocessed pseudogenes:	3519
- unitary pseudogenes:	218
- polymorphic pseudogenes:	38
- pseudogenes:	18

Total No of Transcripts	203835
Protein-coding transcripts	82335
- full length protein-coding:	56541
- partial length protein-coding:	25794
Nonsense mediated decay transcripts	14889
Long non-coding RNA loci transcripts	28468

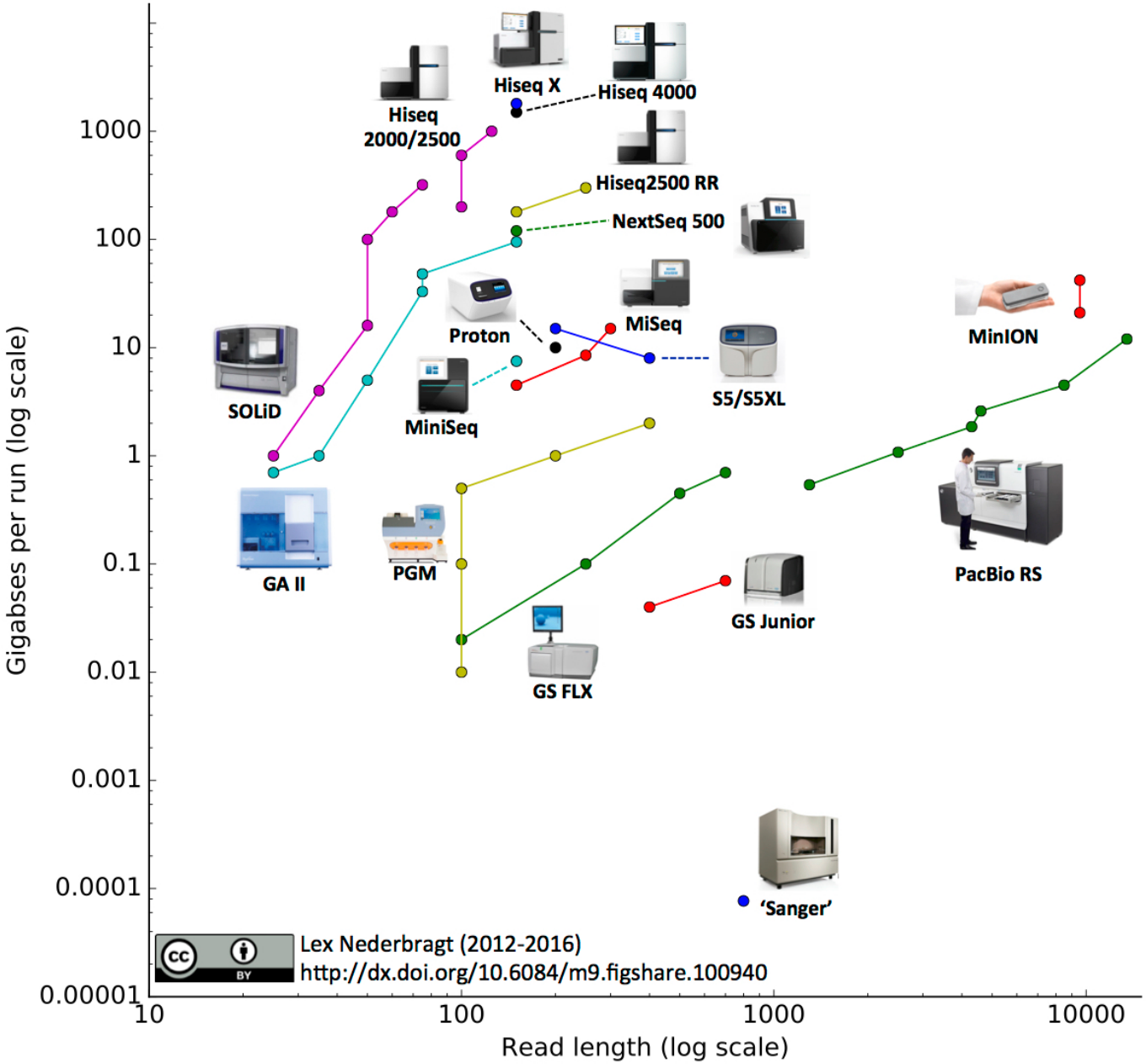
https://www.gencodegenes.org/stats/current.html

What is « new » with RNA-Seq ?

- No prior knowledge of sequence needed
- Specificity of what is measured
- Increased dynamic range of measure, more sensitive detection
- Direct quantification
- Good reproducibility
- Different levels : genes, transcripts, allele specificity, structure variations
- New feature discovery: transcripts, isoforms, ncRNA, structures (fusion...)
- Possible detection of SNPs, ...



Sequencing platforms



https://flxlexblog.wordpress.com/2016/07/08/developments-in-high-throughput-sequencing-july-2016-edition/#more-790

Illumina Sequencing platforms

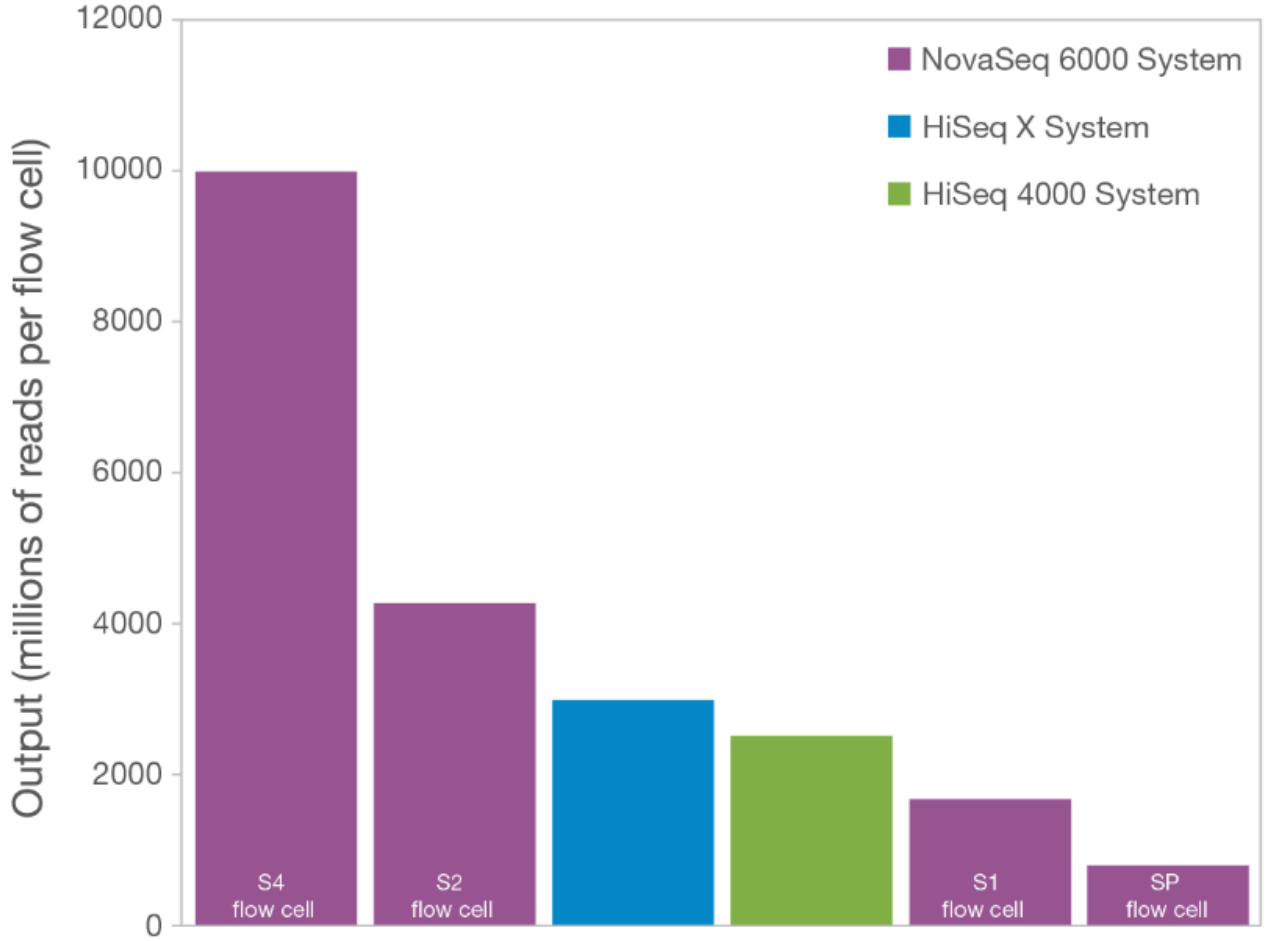


Figure 2: The NovaSeq 6000 System offers the broadest output range—The NovaSeq 6000 System generates from 80 Gb and 800 M reads to 3 Tb and 10 B reads of data in single flow cell mode. In dual flow cell mode, output can be up to 6 Tb and 20 B reads. The tunable output makes the NovaSeq 6000 System accessible for a wide range of applications.

https://www.illumina.com/content/dam/illumina-marketing/documents/products/datasheets/novaseq-6000-system-specification-sheet-770-2016-025.pdf

Illumina RNA-Seq protocol

1 Library Preparation

Fragment DNA
Repair ends
Add A overhang
Ligate adapters
Purify

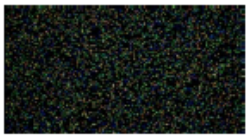
2 Cluster Generation

Hybridize to flow cell
Extend hybridized template
Perform bridge amplification
Prepare flow cell for sequencing

3 Sequencing

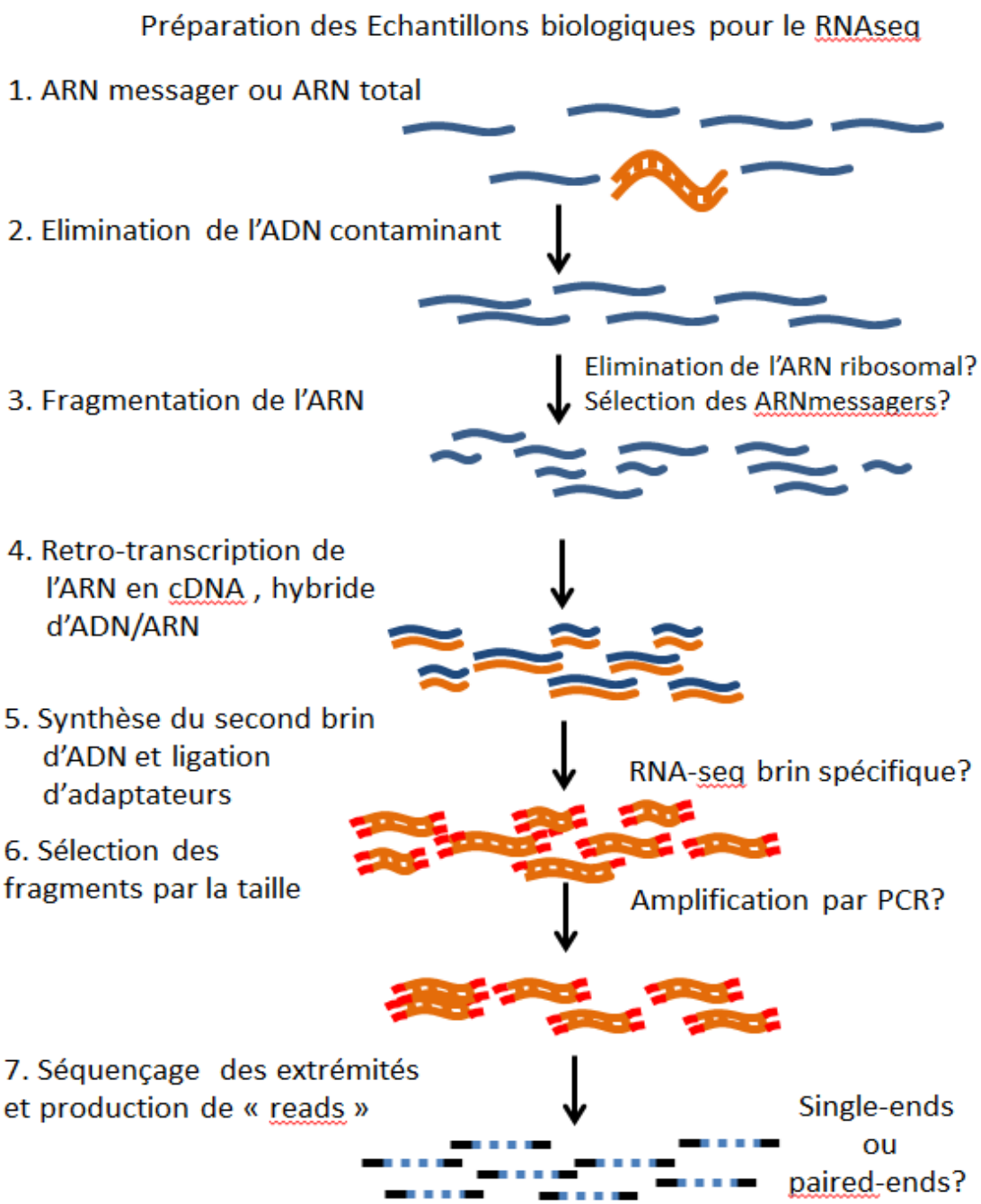
Perform sequencing
Generate base calls

4 Data Analysis

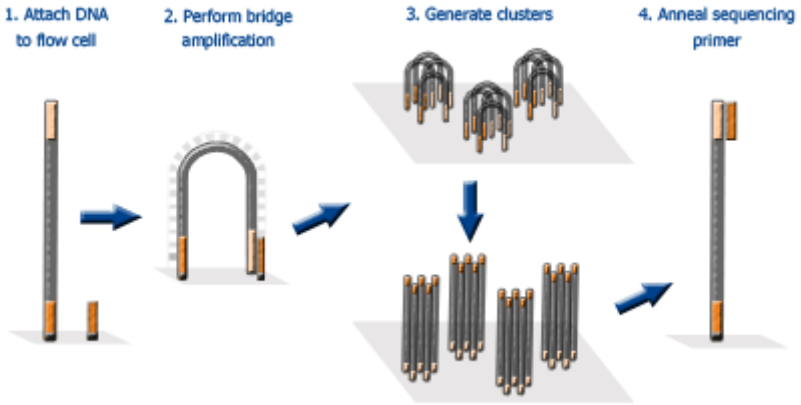


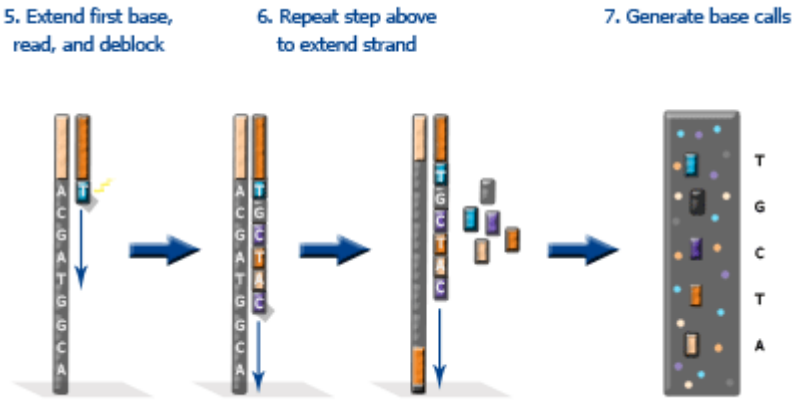
Images
Intensities
Reads
Alignments

RNA-Seq library preparation



Clusters generation / Sequencing





https://www.eurofinsgenomics.co.in/en/eurofins-genomics/product-faqs/next-generation-sequencing/general-technical-questions/what-is-the-principal-of-the-illumina-sequencing-technology.aspx

How to define experimental protocol ?

- Ribo-depletion or polyA-selection ?
- Single-end or paired-end ?
- How long should my reads be ?
- How many replicates ?
  - Technical or/and biological replicates ?
- How many reads for each sample?
- How many conditions for a full transcriptome ?

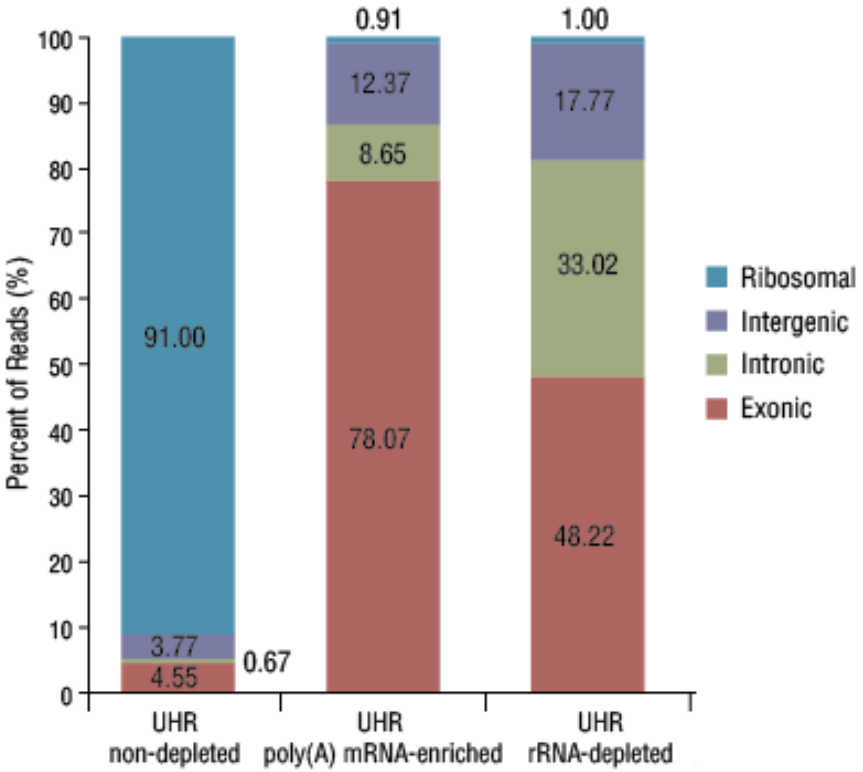
Déplétion / Enrichissement ?

- Similar results

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

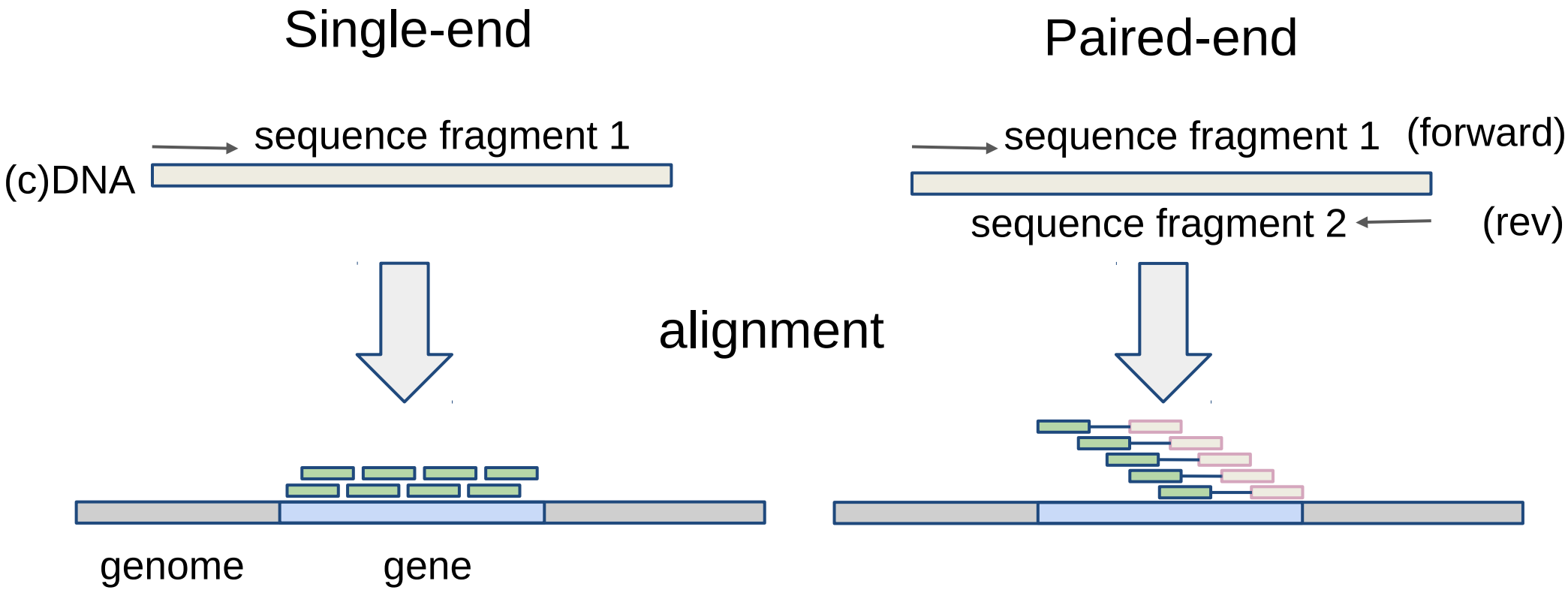
- RNA depletion:
  - For bacterial
  - ARN more varied
  - CircRNA
  - Some ncRNA

- polyA enrichment:
  - More reads into exons
  - Less biological material
  - No transcript without PolyA or partially degraded
  - No circRNA biais



https://content.neb.com/products/e6310-nebnext-rrna-depletion-kit-human-mouse-rat

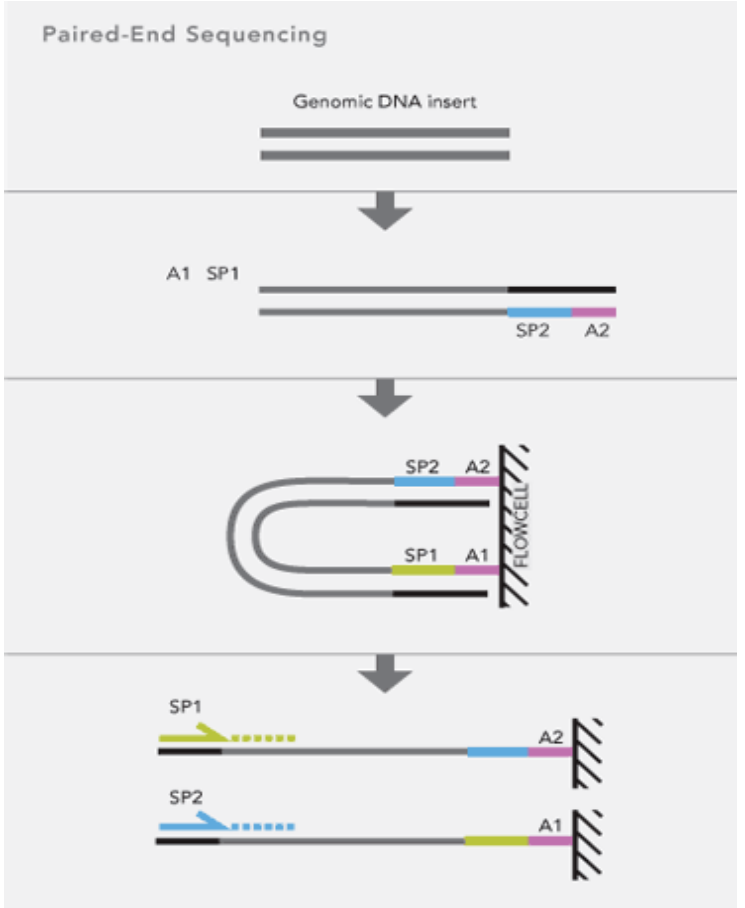
Paired-end VS single-end



- The cDNA size hive the insert size (ex. 200-500 pb).
- The fragment are usually forward-reverse.

Paired-end sequencing

- Modification of the standard single-read DNA library preparation facilitates reading both ends of each fragment
- Improvement of mapping
- Help to detect structural variations in the genome like insertions or deletions, copy number variations, and genome rearrangements



Adapter (A1 and A2) with sequencing primer sites (SP1 and SP2) are ligated onto DNA fragments. Template clusters are formed on the flow cell by bridge amplification and then sequenced by synthesis from the paired primers sequentially.

Strand specific RNA-Seq protocol

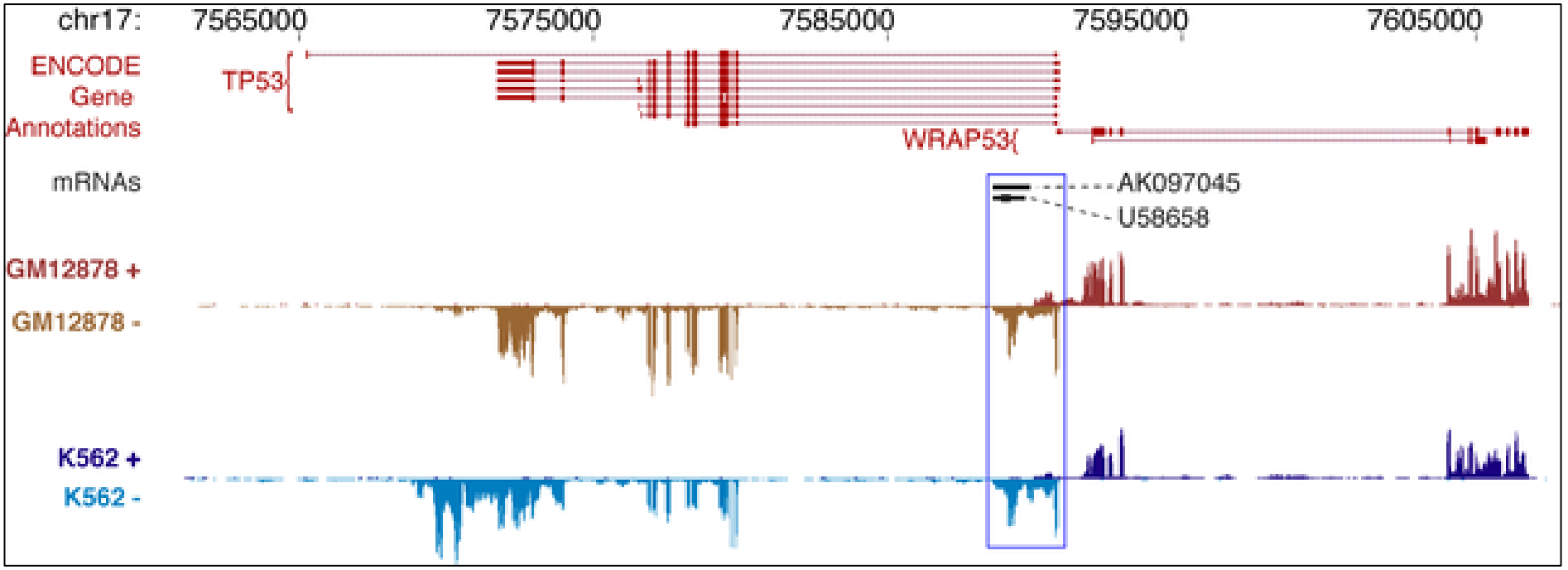
Nat Methods. 2010 Sep;7(9):709-15. Epub 2010 Aug 15.

Comprehensive comparative analysis of strand-specific RNA sequencing methods.

Levin JZ, Yassour M, Adiconis X, Nusbaum C, Thompson DA, Friedman N, Gnirke A, Regev A.

Broad Institute of Massachusetts Institute of Technology and Harvard University, Cambridge, Massachusetts, USA.
jlevin@broadinstitute.org

Abstract



Experimental protocol:
Depth VS Replicates

- Encode (2016):
  - Experiments should be performed with **two or more biological replicates**, unless there is a compelling reason why this is impractical or wasteful
  - Replicate concordance: the gene level quantification should have a Spearman correlation of >0.9 between isogenic (same donor) replicates and >0.8 between anisogenic (different donor) replicates.
- Between **30M and 100M reads** per sample depending on the study.

https://www.encodeproject.org/documents/cede0cbe-d324-4ce7-ace4-f0c3eddf5972/@@download/attachment/ENCODE%20Best%20Practices%20for%20RNA_v2.pdf

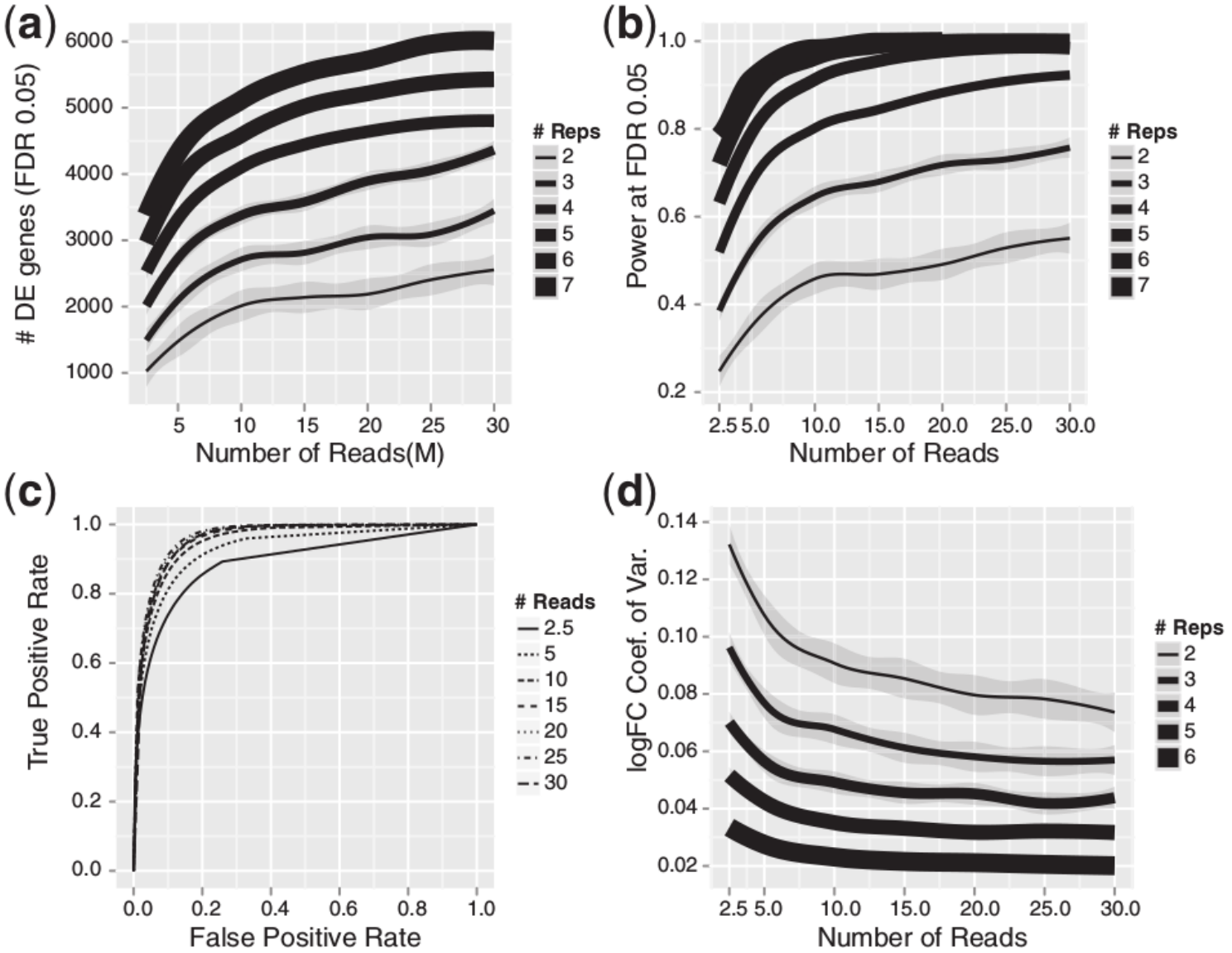
Experimental protocol:
Depth VS Replicates

Gene expression

Advance Access publication December 6, 2013

RNA-seq differential expression studies: more sequence or more replication?

Yuwen Liu[1,2], Jie Zhou[1,3] and Kevin P. White[1,2,3,*]



Retrieve public data

Why ?

- Because there is a lot of public data that would be sufficient for your analysis
- The authors often use only part of the data to answer their own problems
- Perhaps you don’t need to sequence your own data

Retrieve public data

ENA https://www.ebi.ac.uk/ena

ENA > Search & Browse > Download > Downloading read data

Downloading read data

Sequencing reads are available for download through FTP and Aspara protocols in their original format and in an archive generated fastq formats described here.

- Submitted data files
- Archive generated fastq files
- Downloading files using FTP
- Downloading files using Globus GridFTP
- Downloading files using ENA Browser
- Downloading files using Aspera

Submitted data files

Submitted data files are organised by submission accession number under vol1/ directory in ftp.sra.ebi.ac.uk:
ftp://ftp.sra.ebi.ac.uk/vol1/<submission accession prefix>/<submission accession>

where <submission accession prefix> contains the first 6 letters and numbers of the SRA Submission accession. For example, the files submitted in the SRA Submission ERA007448 are available at: ftp://ftp.sra.ebi.ac.uk/vol1/ERA007/ERA007448/.

Archive generated fastq files

Archive generated fastq files are organised by run accession number under vol1/fastq directory in ftp.sra.ebi.ac.uk:

ftp://ftp.sra.ebi.ac.uk/vol1/fastq/<dir1>[/<dir2>]/<run accession>

<dir1> is the first 6 letters and numbers of the run accession (e.g. ERR000 for ERR000916),

<dir2> does not exist if the run accession has six digits. For example, fastq files for run ERR000916 are in

Search & Browse

- Data formats
  - Genome assemblies
- Marker portal
- Taxon portal
- Programmatic access
  - Data retrieval
  - Taxon portal
  - Marker portal
  - Search
  - File reports
  - XREF service
- Genome assembly database
- Taxonomy Service
  - Translation tables
- Download
  - Sequences
    - Feature level products
  - Reads
  - Taxonomy
- Sequence search

Retrieve public data

SRA https://www.ncbi.nlm.nih.gov/sra

NCBI Site map | All databases | Search

Sequence Read Archive

Main | Browse | Search | Download | Submit | Documentation | Software | Trace Archive | Trace Assembly | Trace BLAST

Overview

The Sequence Read Archive (SRA) stores raw sequence data from "next-generation" sequencing technologies including Illumina, 454, IonTorrent, Complete Genomics, PacBio and OxfordNanopores. In addition to raw sequence data, SRA now stores alignment information in the form of read placements on a reference sequence.

SRA is NIH's primary archive of high-throughput sequencing data and is part of the international partnership of archives (INSDC) at the NCBI, the European Bioinformatics Institute and the DNA Database of Japan. Data submitted to any of the three organizations are shared among them.

Please check SRA Overview for more information.

Submitting to SRA

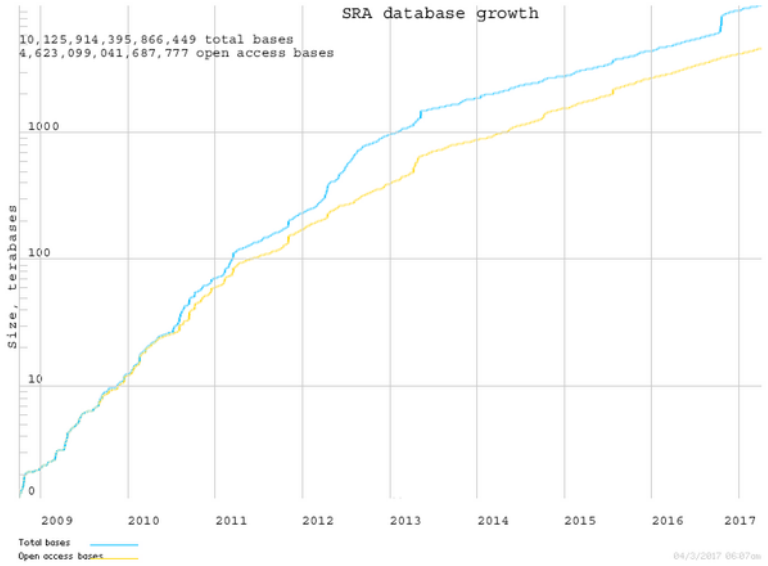
Making data available to the research community enhances reproducibility and allows for new discovery by comparing data sets.

- Submission Quick Start
- Frequently Asked Questions
- Submitter Login

Using SRA Data with SRA Toolkit

Use SRA data to validate experimental results, increase sample sizes, determine variance and open up new avenues of research.

- Documentation
- Usage Guide
- Download
- Get sources code on GitHub (for developers using SRA)



Save in CSV format

Retrieve public data

SRX4792876: GSM3415475: HS2191_control_S7_R1_001; Homo sapiens; RNA-Seq
1 ILLUMINA (NextSeq 500) run: 26.6M spots, 2G bases, 782.3Mb downloads

Submitted by: NCBI (GEO)

Study: Glucocorticoid induced gene signature in human skin
PRJNA494527 • SRP163234 • All experiments • All runs
show Abstract

Sample: HS2191_control_S7_R1_001
SAMN10171026 • SRS3872085 • All experiments • All runs
Organism: Homo sapiens

Library:
Instrument: NextSeq 500
Strategy: RNA-Seq
Source: TRANSCRIPTOMIC
Selection: cDNA
Layout: SINGLE
Construction protocol: Total RNA from whole human skin, and HaCaT keratinocyte cell cultures were isolated with RiboPure kit (Ambion, Life Technologies, Grand Island, NY, USA). The RNA samples were treated with TURBOTM DNase (Ambion), checked for quality and integrity with the Agilent 2100 bioanalyzer and used for RNASeq. Due to the conical shape of punch skin biopsies, the RNA was mostly extracted from keratinocytes with minimal contribution of dermal cells RNA libraries were prepared for sequencing using standard Illumina protocols

Experiment attributes:
GEO Accession: GSM3415475

Links:

Runs: 1 run, 26.6M spots, 2G bases, 782.3Mb

Run	# of Spots	# of Bases	Size	Published
SRR7959222	26,580,098	2G	782.3Mb	2018-10-04

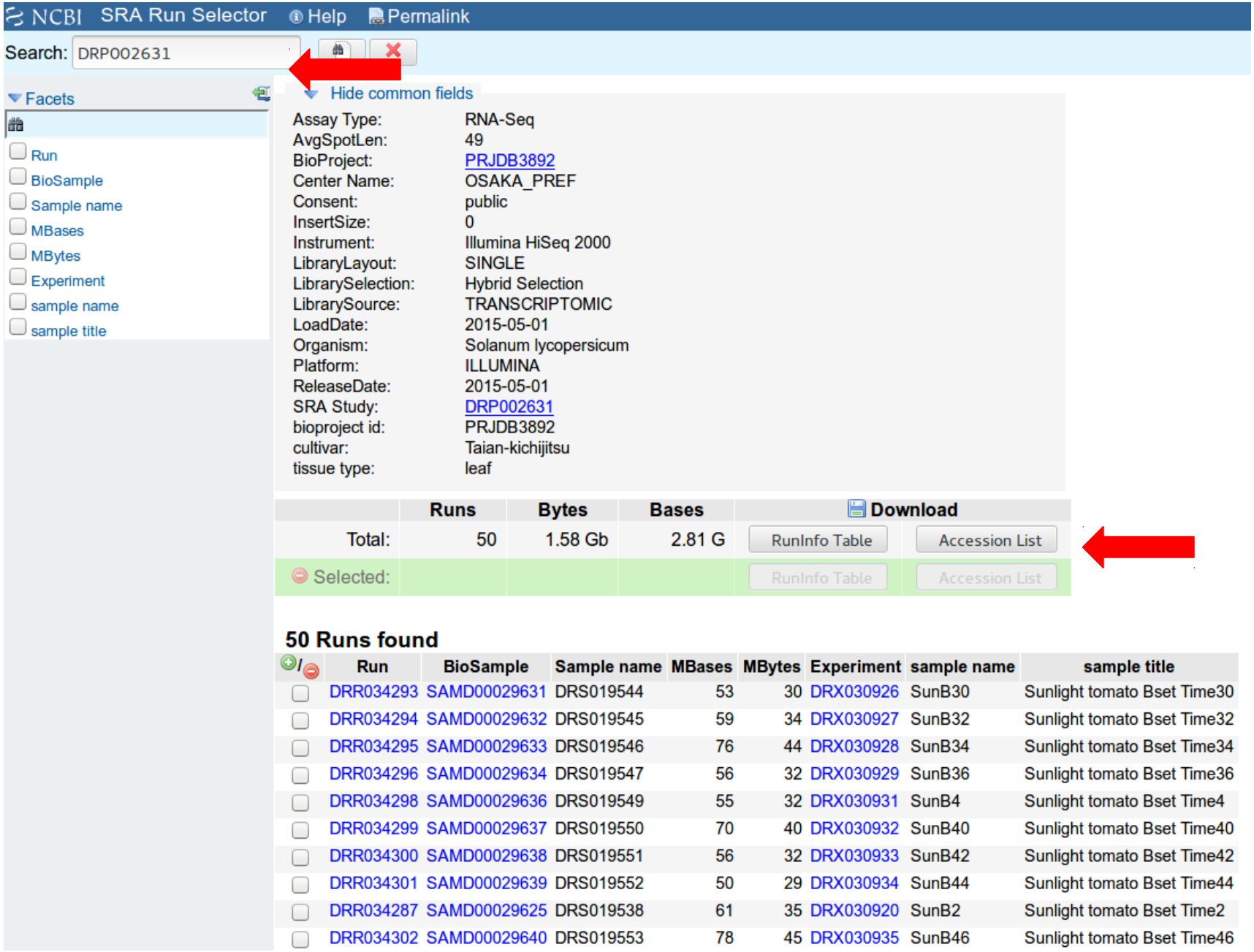
Accession : SRX/ERX/DRX

SRPxxxxxx : Project
SRXxxxxxx : Experiement
SRRxxxxxx : Run

GSMxxxxxx : GEO id

http://bioinfo.genotoul.fr/index.php/faq/bioinfo_tips_faq/

Retrieve public data



Retrieve public data

NCBI SRA Run Selector | Help | Permalink

Search: DRP002631

Facets

- Run
- BioSample
- Sample name
- MBases
- MBytes
- Experiment
- sample name
- sample title

Hide common fields

Assay Type:	RNA-Seq
AvgSpotLen:	49
BioProject:	PRJDB3892
Center Name:	OSAKA_PREF
Consent:	public
InsertSize:	0
Instrument:	Illumina HiSeq 2000
LibraryLayout:	SINGLE
LibrarySelection:	Hybrid Selection
LibrarySource:	TRANSCRIPTOMIC
LoadDate:	2015-05-01
Organism:	Solanum lycopersicum
Platform:	ILLUMINA
ReleaseDate:	2015-05-01
SRA Study:	DRP002631
bioproject id:	PRJDB3892
cultivar:	Taian-kichijitsu
tissue type:	leaf

	Runs	Bytes	Bas
Total:	50	1.58 Gb	2.
Selected:			

SRR_Acc_List.txt (/tmp/mozilla_choede0) – gedit

Fichier Édition Affichage Rechercher Outils Documents

Ouvrir Enregistrer Annuler

SRR_Acc_List.txt

```
DRR034293
DRR034294
DRR034295
DRR034296
DRR034298
DRR034299
DRR034300
DRR034301
DRR034287
DRR034302
DRR034291
DRR034305
DRR034290
DRR034292
DRR034303
```

Texte brut Largeur des tabulations : 8 Lig 1, Col 1 INS

50 Runs found

Run	BioSample	Sample name	MBases	MBytes	Experiment	sample name	sample title
DRR034293	SAMD00029631	DRS019544	53	30	DRX030926	SunB30	Sunlight tomato Bset Time30
DRR034294	SAMD00029632	DRS019545	59	34	DRX030927	SunB32	Sunlight tomato Bset Time32
DRR034295	SAMD00029633	DRS019546	76	44	DRX030928	SunB34	Sunlight tomato Bset Time34
DRR034296	SAMD00029634	DRS019547	56	32	DRX030929	SunB36	Sunlight tomato Bset Time36
DRR034298	SAMD00029636	DRS019549	55	32	DRX030931	SunB4	Sunlight tomato Bset Time4
DRR034299	SAMD00029637	DRS019550	70	40	DRX030932	SunB40	Sunlight tomato Bset Time40
DRR034300	SAMD00029638	DRS019551	56	32	DRX030933	SunB42	Sunlight tomato Bset Time42
DRR034301	SAMD00029639	DRS019552	50	29	DRX030934	SunB44	Sunlight tomato Bset Time44
DRR034287	SAMD00029625	DRS019538	61	35	DRX030920	SunB2	Sunlight tomato Bset Time2
DRR034302	SAMD00029640	DRS019553	78	45	DRX030935	SunB46	Sunlight tomato Bset Time46

Retrieve public data

- On genologin, use sratoolkit to :
  - download raw file
  - and convert format.

```
mkdir ~/work/ncbi
ln -s ~/work/ncbi ~/ncbi
module load bioinfo/sratoolkit.2.8.2-1
prefetch <sra_accession> --max-size
(20G by default)
```

Files are created into:

```
~/work/ncbi/public/sra/
```

Convertion

```
fastq-dump --gzip  sra_file.sra
```

Summary - Sequence quality

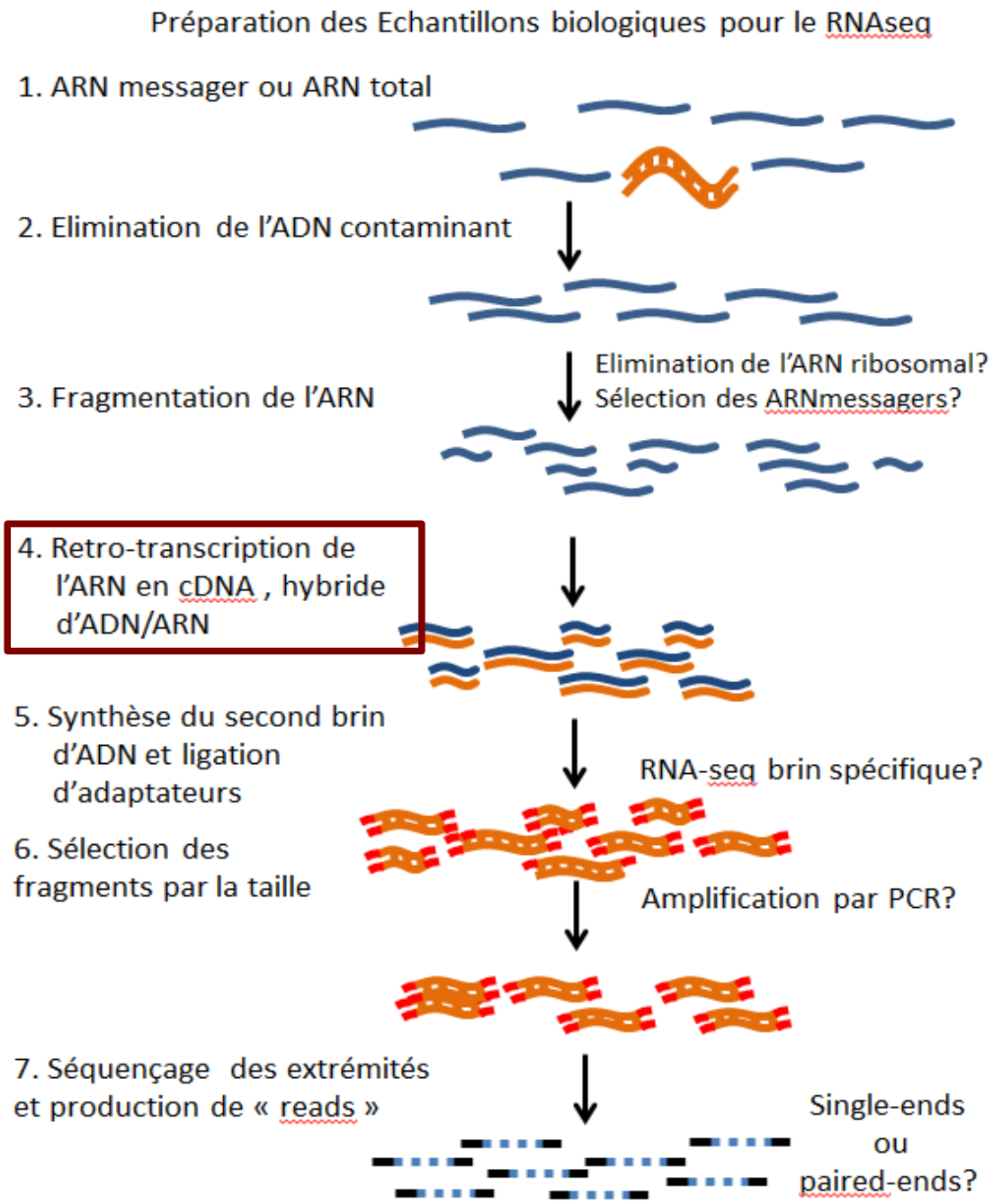
- Known RNAseq biais
- How to check the quality ?
- How to clean the data ?

RNAseq specific bias

- Influence of the library preparation
- Random hexamer priming
- Positional bias and sequence specificity bias.
  Robert et al. Genome Biology, 2011,12:R22
- Transcript length bias
- « Mappability »

Hexamer random priming bias

Random priming
→ not so random



Hexamer random priming bias

Published online 14 April 2010 *Nucleic Acids Research, 2010, Vol. 38, No. 12 e131*
doi:10.1093/nar/gkq224

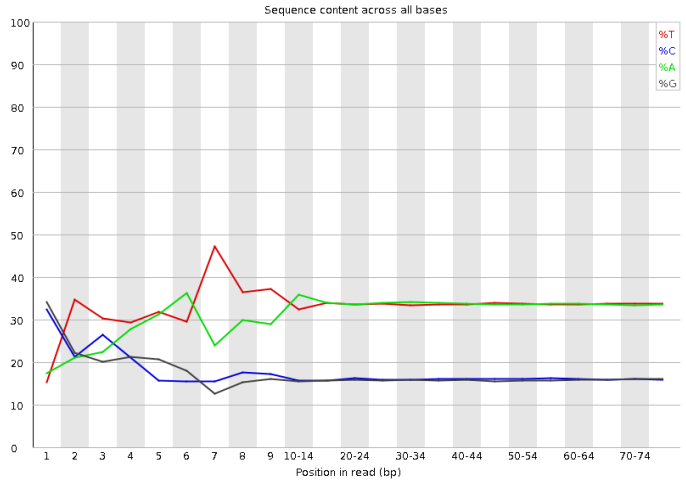
Biases in Illumina transcriptome sequencing caused by random hexamer priming

Kasper D. Hansen[1,*], Steven E. Brenner[2] and Sandrine Dudoit[1,3]

ABSTRACT

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.

- A strong distinctive pattern in the nucleotide frequencies of the first 13 positions at the 5'-end :
  - sequence specificity of the polymerase
  - due to the end repair performed



- Reads beginning with a hexamer over-represented in the hexamer distribution at the beginning relative to the end are down-weighted

Transcript length bias

Biol Direct. 2009 Apr 16;4:14.

Transcript length bias in RNA-seq data confounds systems biology.

Oshlack A, Wakefield MJ.

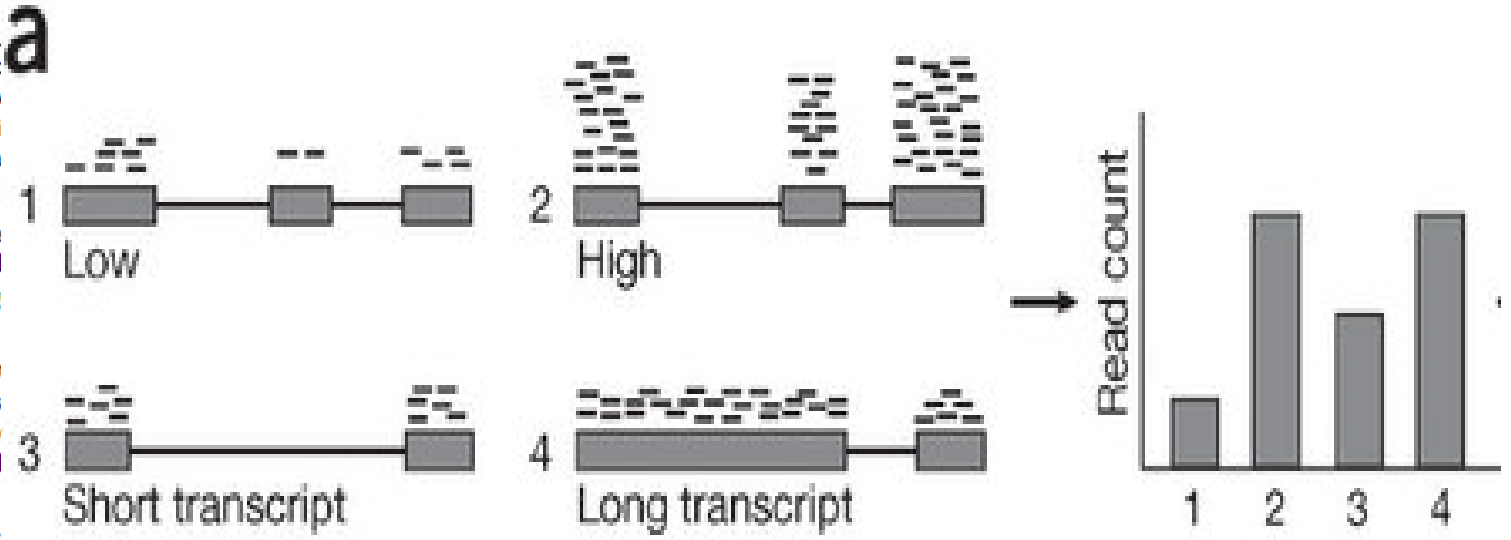
Abstract

Background: Several recent transcriptome analysis (RNA genome transcriptional profil genomic sequences. As yet, a still in the stages of exploring

Results: We investigated the published data sets. For stand call differentially expressed g transcript.

Conclusion: Transcript leng current protocols for RNA-s expressed genes, and in parti other multi-gene systems biol

Reviewers: This article was Cloonan (nominated by Mark



– *the differential expression of longer transcripts is more likely to be identified than that of shorter transcripts*

BIOINFORMATICS ORIGINAL PAPER Vol. 27 no. 5 2011, pages 662–669 doi:10.1093/bioinformatics/btr005

Gene expression Advance Access publication January 19, 2011

Length bias correction for RNA-seq data in gene set analyses

Liyan Gao[1,†], Zhide Fang[2,†], Kui Zhang[1], Degui Zhi[1] and Xiangqin Cui[1,*]

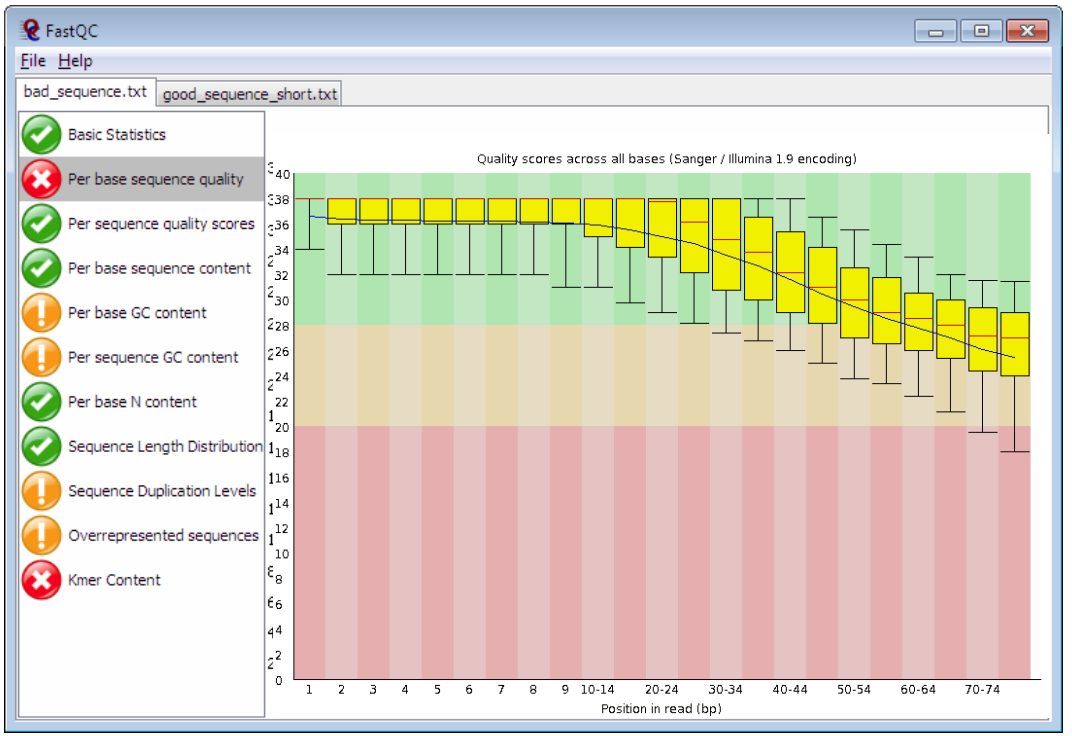
Bias “mappability”

- Quality of the reference genome influence results
  - assembly
  - finishing

- Sequence composition
- Repeated sequences
- Annotation quality

Verifying RNA-Seq quality

FastQC :
http://www.bioinformatics.bbsrc.ac.uk/projects/fastqc/



Has been developed for genomic data

fastq format

- Standard for storing outputs of HTS
- A text-based format for storing a read and its corresponding quality scores
- 1 read <-> 4 lines

```
@HWI-ST218:596:C90JYANXX:8:1101:1293:2188 1:N:0:ATTCAGAATAATCTTA
NCTAAGTGTTAGGGGGTTTCCGCCCCTTAGTGCTGCAGCTAACGCATTAAGCACTCCGCCTGGGGAGTACGGTCGCAAGACTGAAAA
+
#<3?BFGGGGGGEGGGGGGGEGGGGGG@F1FGGGGGGDDGG1FB</9FE=EGGGGGGGG>GGGGBGGGGG<<C/BDGGGGGGC=GGG
```

1. Begins with '@' character and is followed by a sequence identifier
2. The raw sequence
3. Begins with a '+' character and is optionally followed by the same sequence identifier
4. Encodes the quality values for th read , contains the same number of symbols as letters in the read

fastq format

- Sequence identifier

@EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG

1. Begins with '@' character and is followed by a sequence identifier

EAS139	the unique instrument name
136	the run id
FC706VJ	the flowcell id
2	flowcell lane
2104	tile number within the flowcell lane
15343	'x'-coordinate of the cluster within the tile
197393	'y'-coordinate of the cluster within the tile
1	the member of a pair, 1 or 2 *(paired-end or mate-pair reads only)*
Y	Y if the read is filtered, N otherwise
18	0 when none of the control bits are on, otherwise it is an even number
ATCACG	index sequence

fastq format

- Base quality (Sanger standard)

```
@HWI-ST218:596:C90JYANXX:8:1101:1293:2188 1:N:0:ATTCAGAATAATCTTA
NCTAAGTGTTAGGGGGTTTCCGCCCCTTAGTGCTGCAGCTAACGCATTAAGCACTCCGCCTGGGGAGTACGGTCGCAAGACTGAAAA
+
#<3?BFGGGGGGEGGGGGGGEGGGGGG@F1FGGGGGGDDGG1FB</9FE=EGGGGGGGG>GGGGBGGGGG<<C/BDGGGGGGC=GGG
```

ASCII-encoded version of the PHRED quality given by $Q_{\mathrm{PHRED}} = -10 \times \log_{10}(P_e)$

SANGER=PHRED+33 : H=ASCII(40+33)

$$Q = -10 \log_{10} P \Leftrightarrow P = 10^{\frac{-Q}{10}}$$

Score de qualité phred	Probabilité d'une identification incorrecte	Précision de l'identification d'une base
10	1 pour 10	90 %
20	1 pour 100	99 %
30	1 pour 1000	99.9 %
40	1 pour 10000	99.99 %
50	1 pour 100000	99.999 %

Published online 16 December 2009 *Nucleic Acids Research, 2010, Vol. 38, No. 6 1767–1771*
doi:10.1093/nar/gkp1137

SURVEY AND SUMMARY

The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants

Peter J. A. Cock[1,*], Christopher J. Fields[2], Naohisa Goto[3], Michael L. Heuer[4] and Peter M. Rice[5]

fastq format

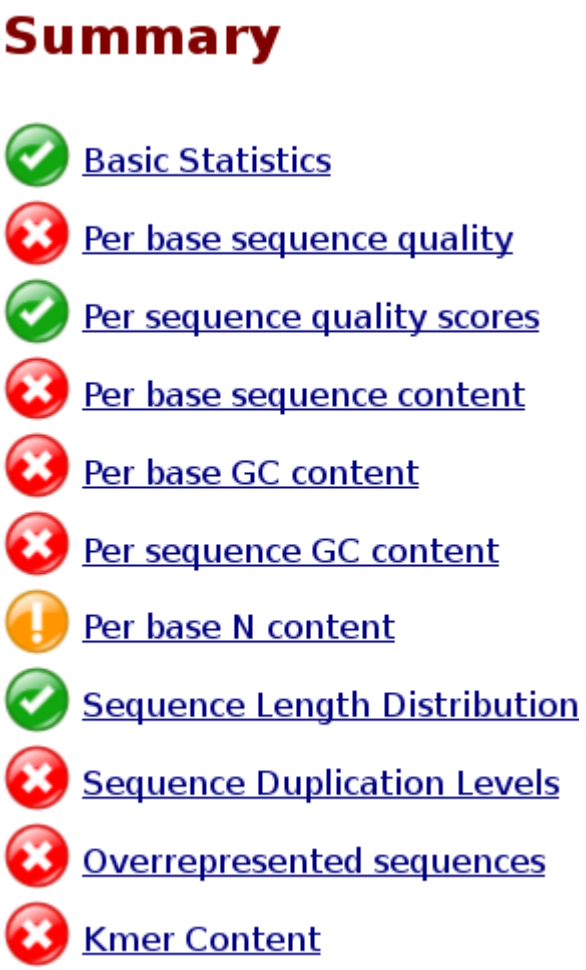
```
@EAS54_6_R1_2_1_413_324
CCCTTCTTGTCTTCAGCGTTTCTCC
+
;;3;;;;;;;;;;;;7;;;;;;;88
```

la proba d'une erreur : $Q_{\mathrm{PHRED}} = -10 \times \log_{10}(P_e)$

```
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
................................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
.................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL.....................................................
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
|                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
                          -5....0........9.............................40
                               0........9.............................40
                                  3.....9.............................40
 0.2......................26...31........41

S - Sanger        Phred+33,  raw reads typically (0, 40)
X - Solexa        Solexa+64, raw reads typically (-5, 40)
I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
J - Illumina 1.5+ Phred+64,  raw reads typically (3, 40)
    with 0=unused, 1=unused, 2=Read Segment Quality Control Indicator (bold)
    (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

fastqQC Report



The analysis in FastQC is performed by a series of analysis modules.

Quick evaluation of whether the results of the module seem :

- entirely normal (green tick),
- slightly abnormal (orange triangle)
- or very unusual (red cross).

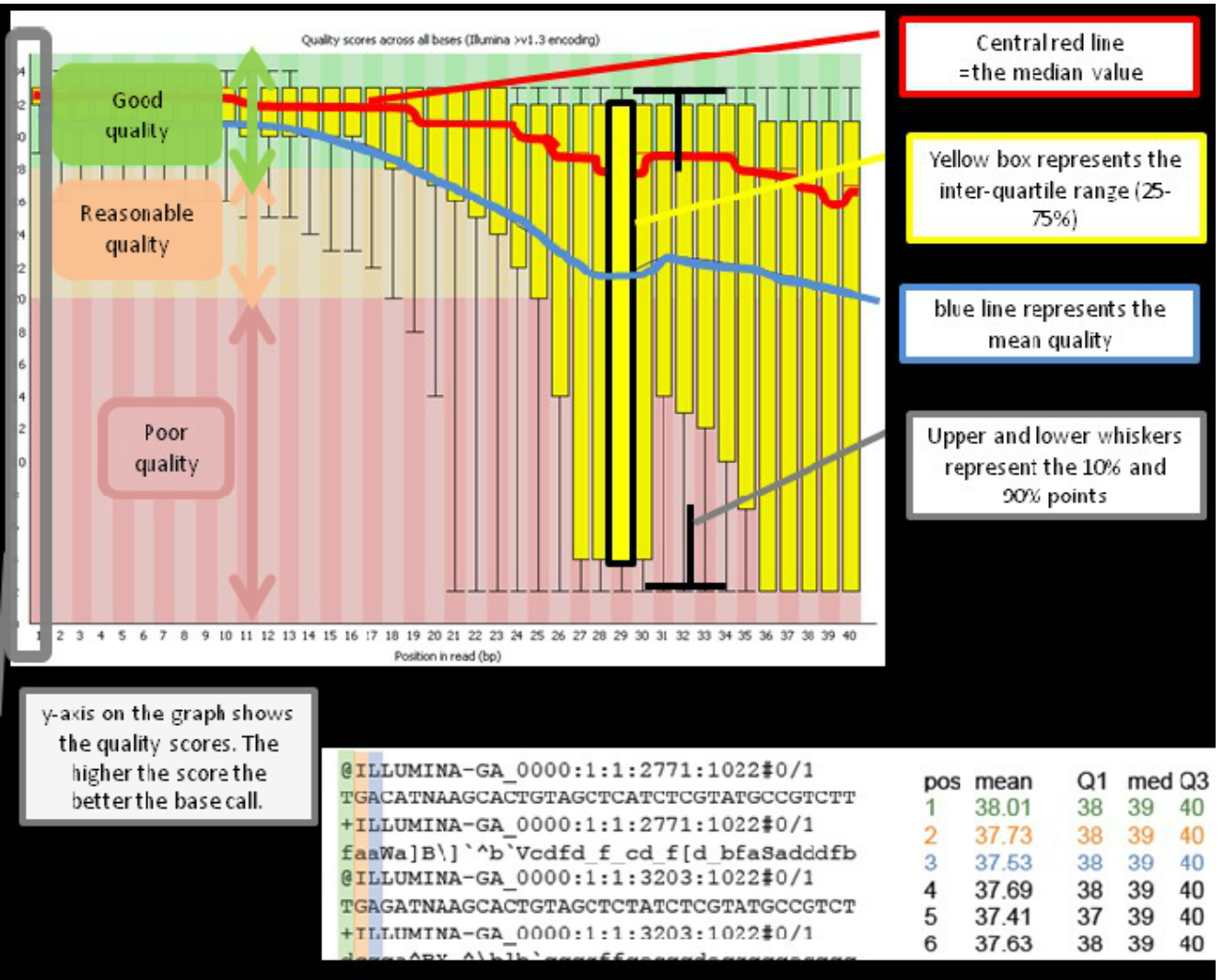
These evaluations must be taken in the context of what you expect from your library. A 'normal' sample as far as FastQC is concerned is random and diverse.

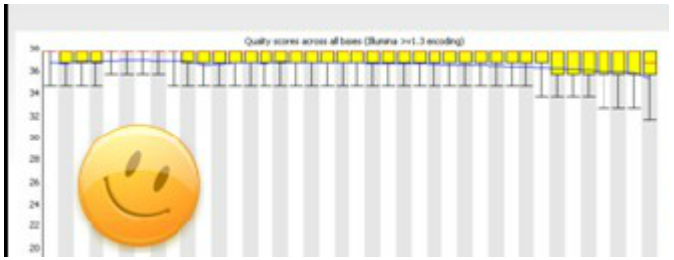
fastqQC Report

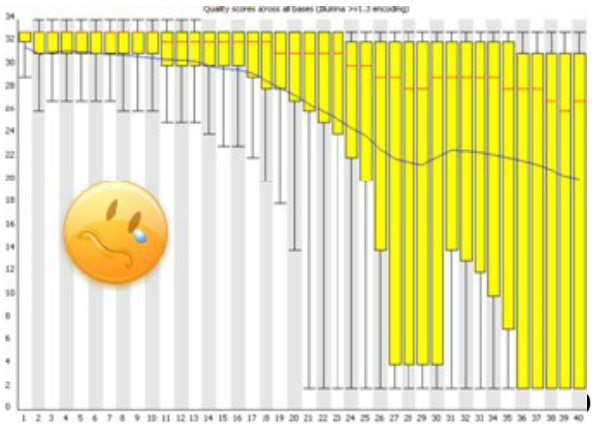
Statistics per Base Sequence Quality

This view shows an overview of the range of quality values across all bases at each position in the FastQ file.

Common to see base calls falling into the orange area towards the end of a read.





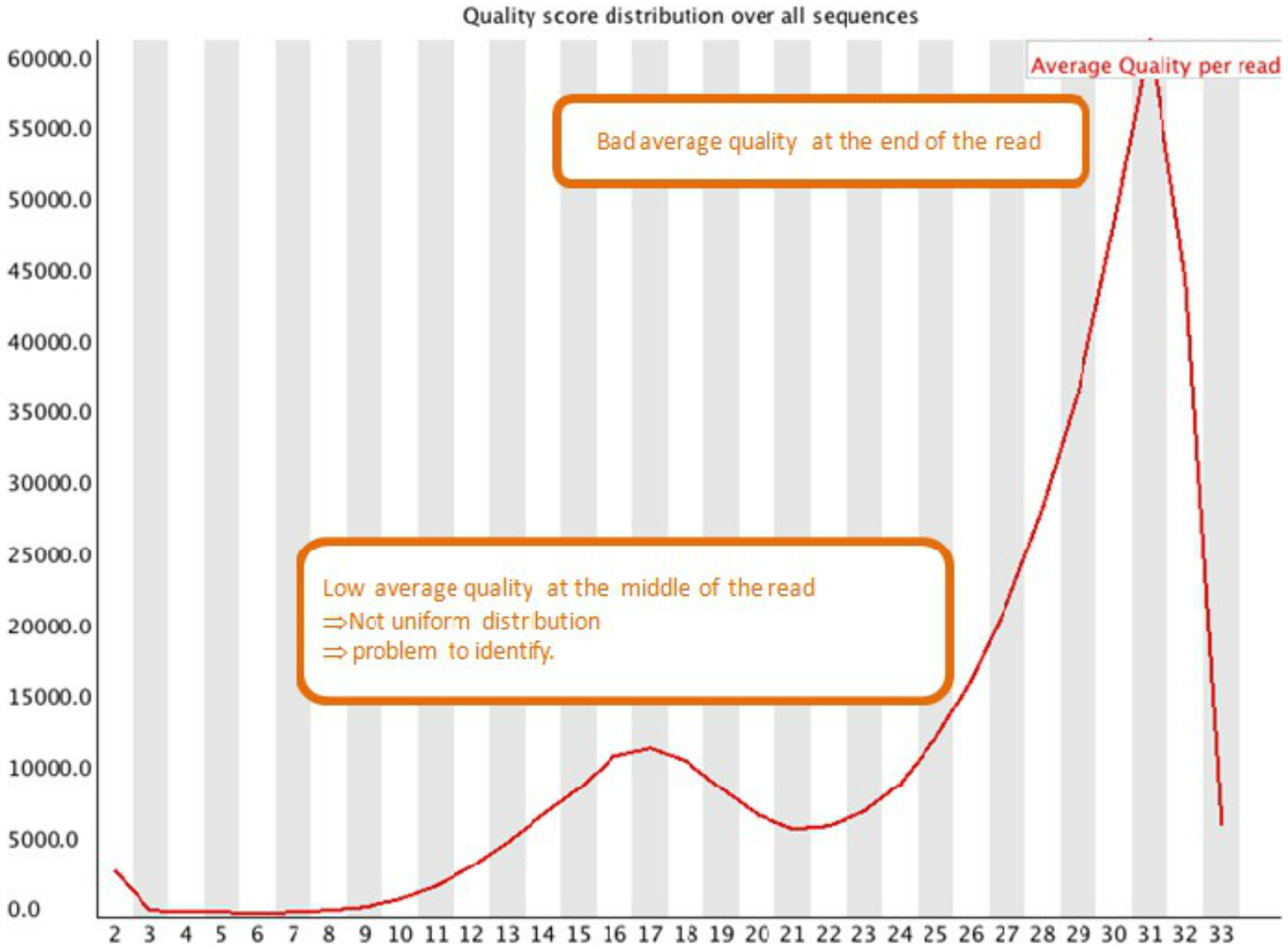


& Quality

fastqQC Report

Statistics per Sequence Quality Score

See if a subset of your sequences have universally low quality values.

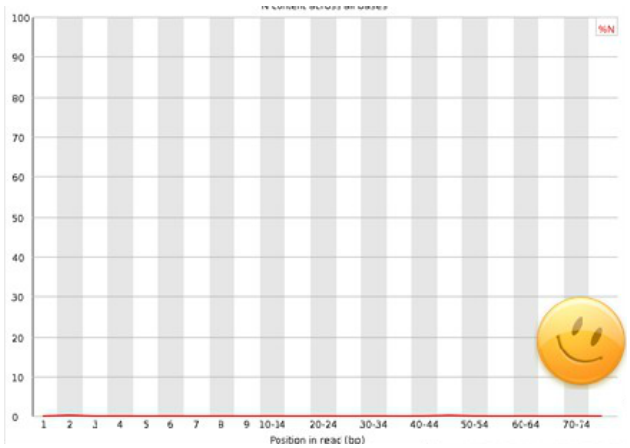


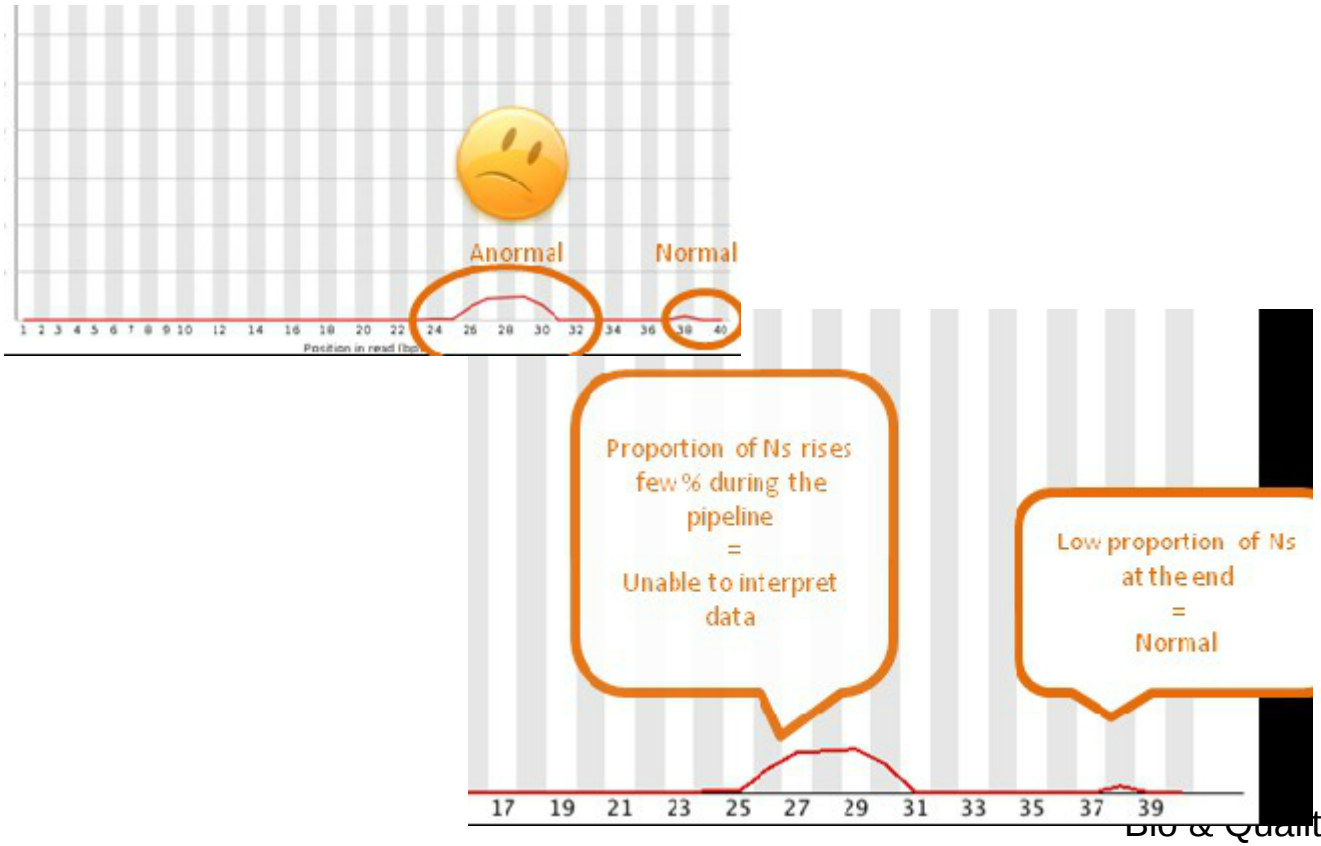
fastqQC Report

Statistics per Base N Content

This module plots out the percentage of base calls at each position for which an N was called.

Usual to see a very low proportion of Ns appearing nearer the end of a sequence.





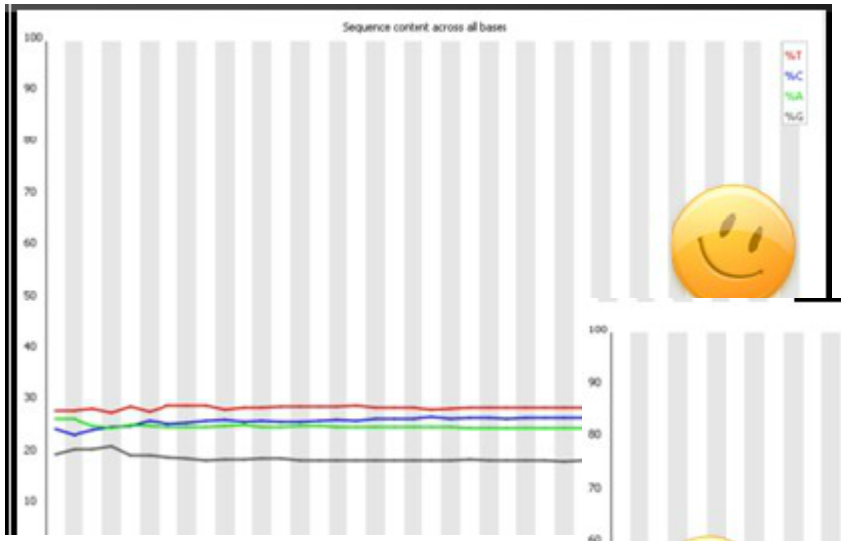
fastqQC Report

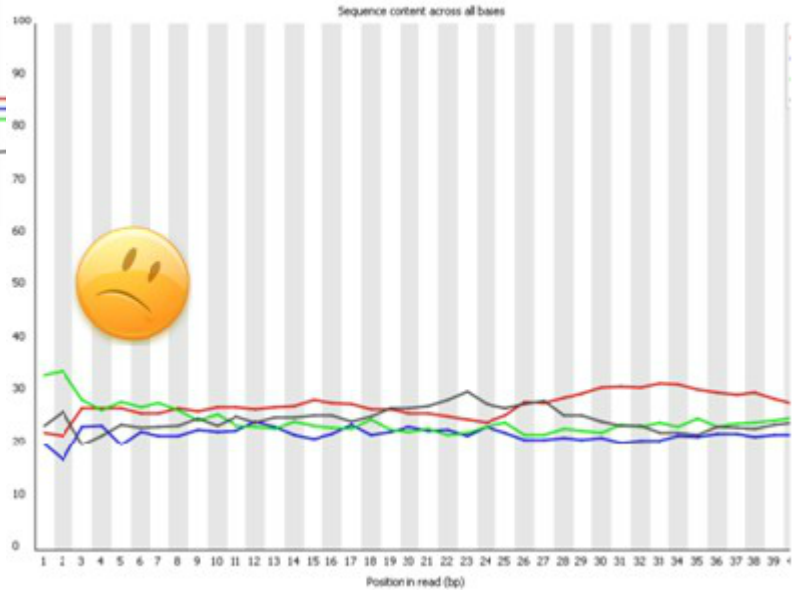
Statistics Per Base Sequence Content

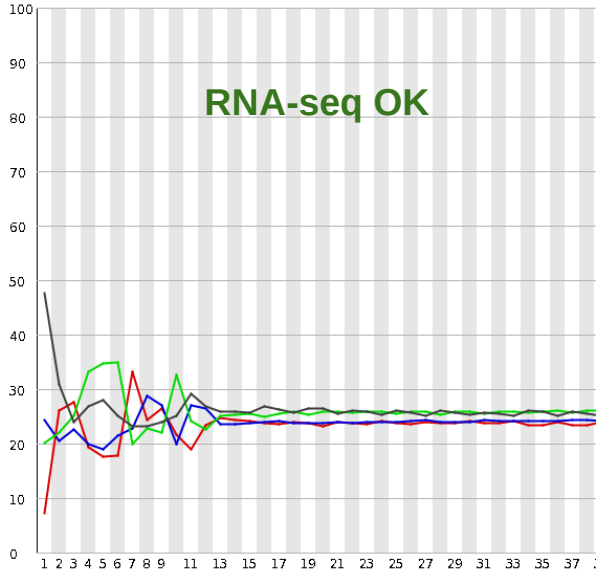
Per Base Sequence Content plots out the proportion of each base position in a file for which each of the four normal DNA bases has been called.

In a random library : little/no difference between the different bases of a sequence run, so the lines in this plot should run parallel with each other.

If strong biases which change : overrepresented sequence contaminating your library.







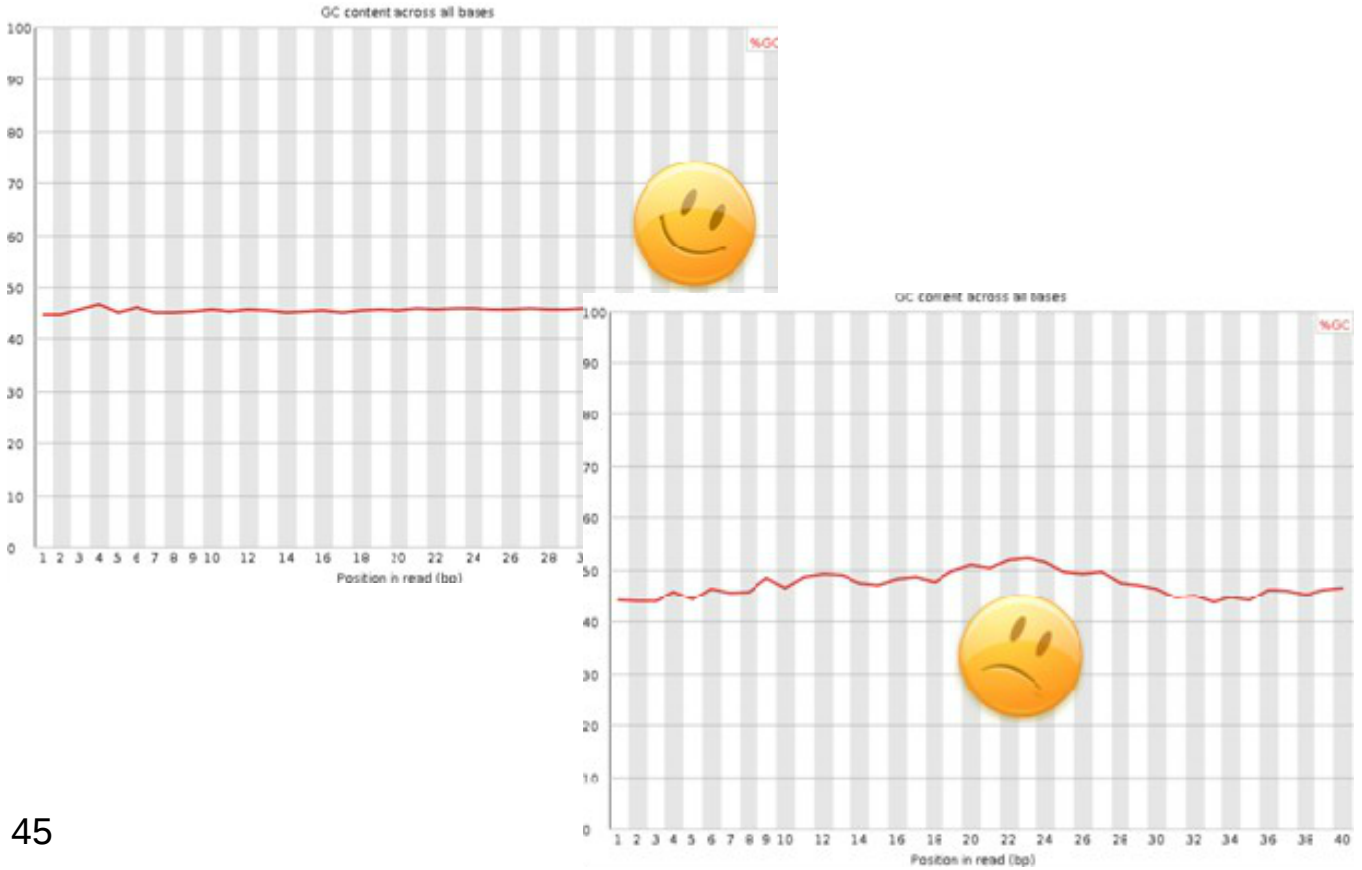
fastqQC Report

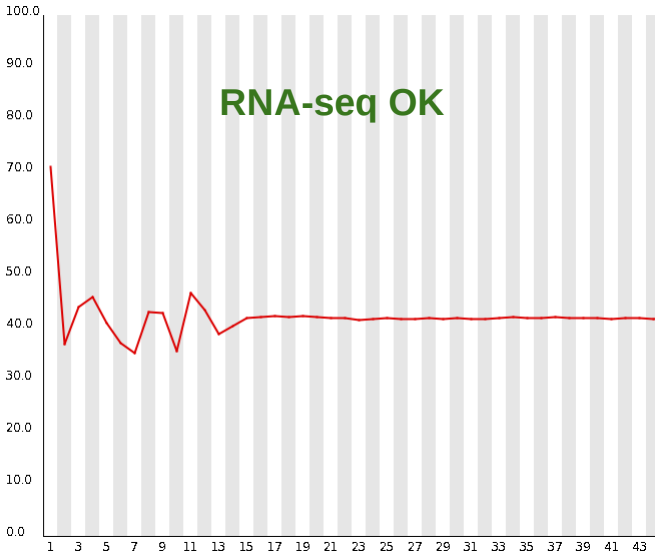
Statistics per Base GC Distribution

Per Base GC Content plots out the GC content of each base position in a file.

Random library : little/no difference between the different bases of a sequence run
=> plot horizontally.
The overall GC content should reflect the GC content of the underlying genome.

GC bias: changes in different bases, overrepresented sequence contaminating your library.
=> plot not horizontally.

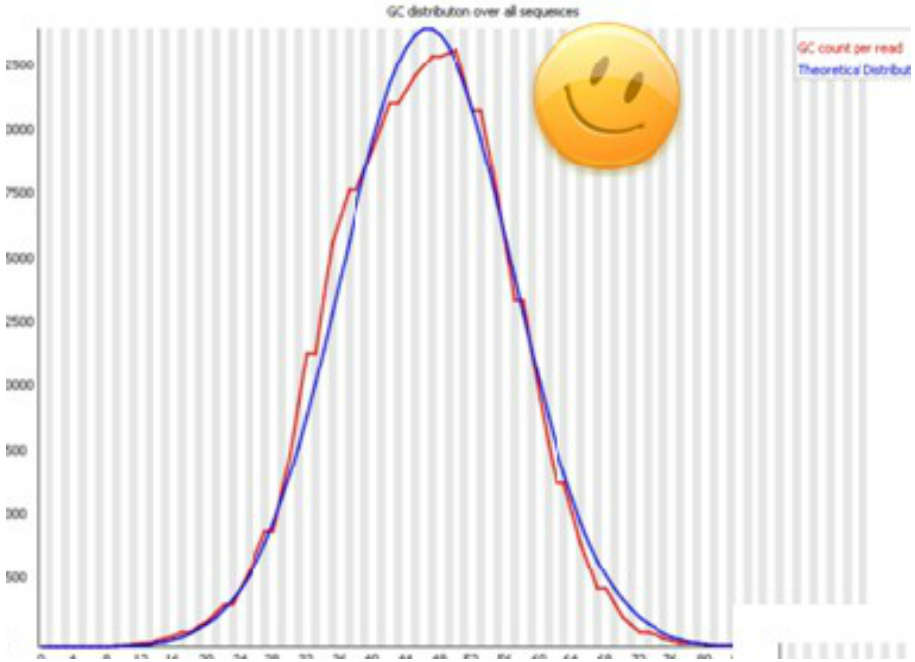


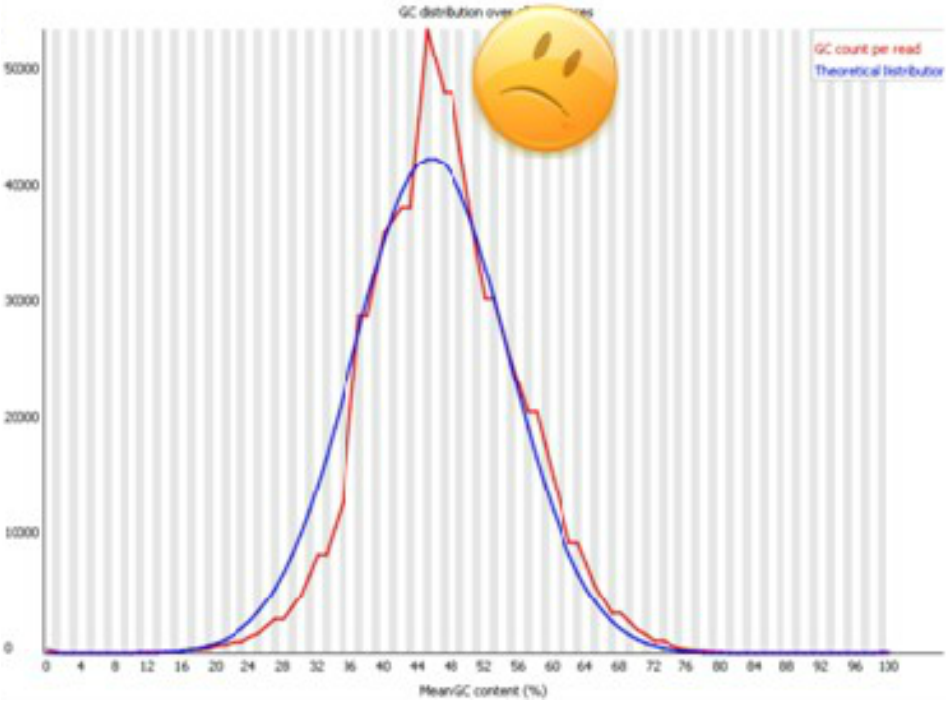


fastqQC Report

Statistics per Sequence GC Content

This module measures the GC content across the whole length of each sequence in a file and compares it to a modeled normal distribution of GC content.



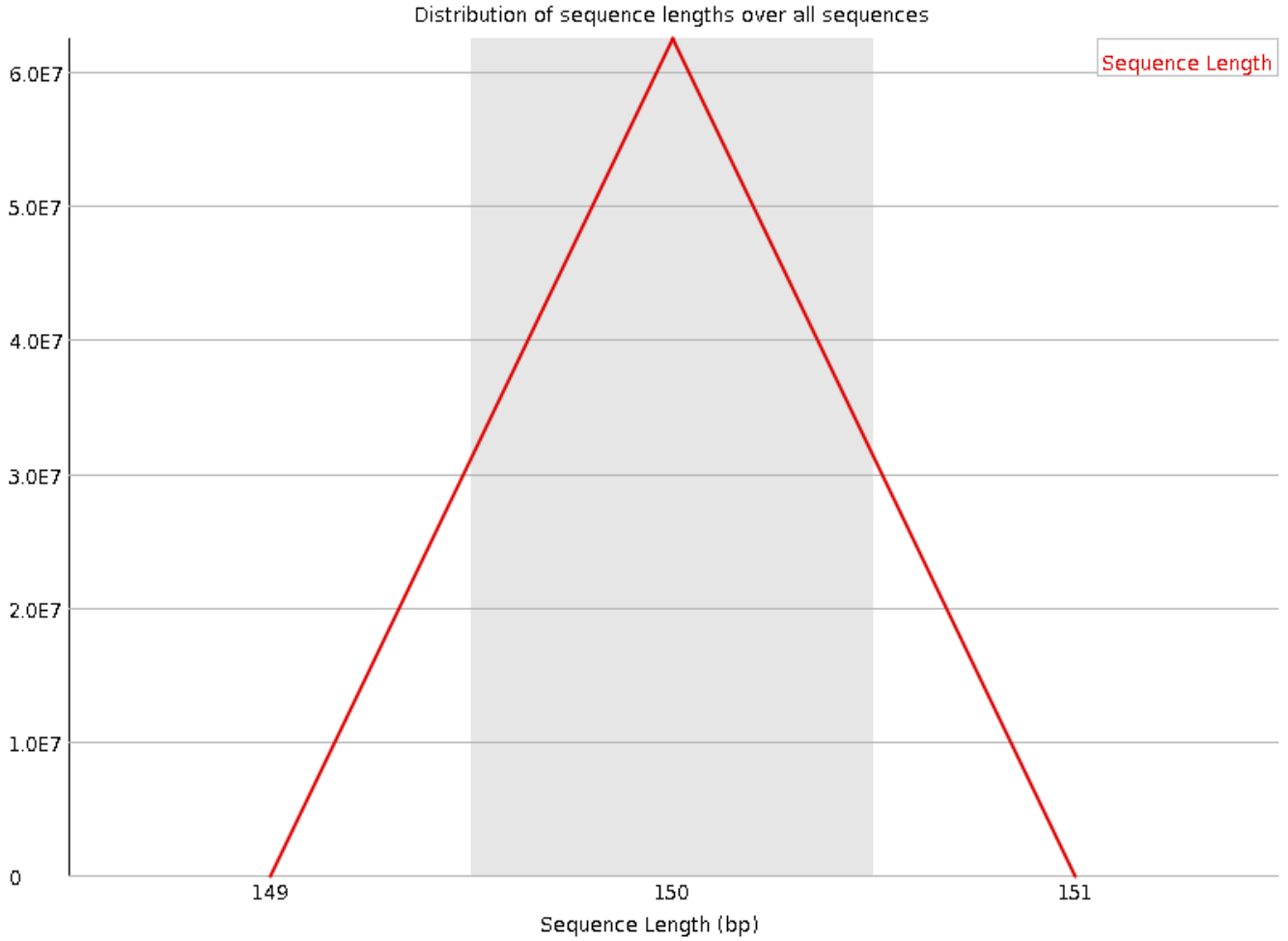


fastqQC Report

Statistics per Sequence Length Distribution

Some sequence fragments contain reads of wildly varying lengths.

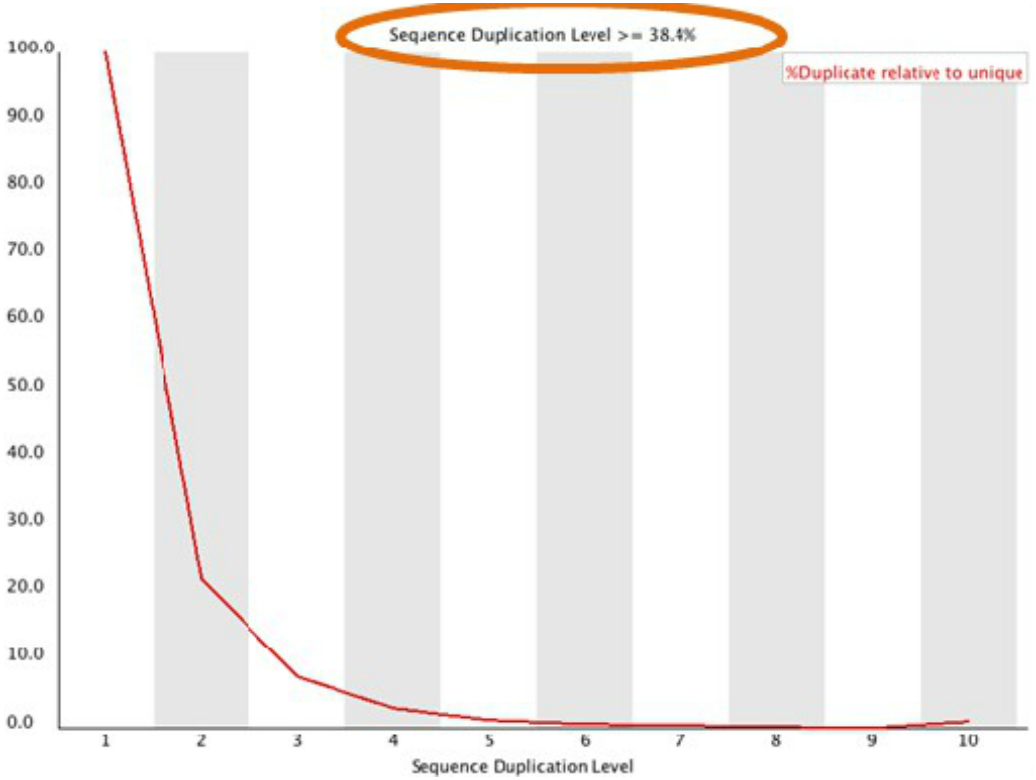
Even within uniform length libraries some pipelines will trim sequences to remove poor quality base calls from the end.

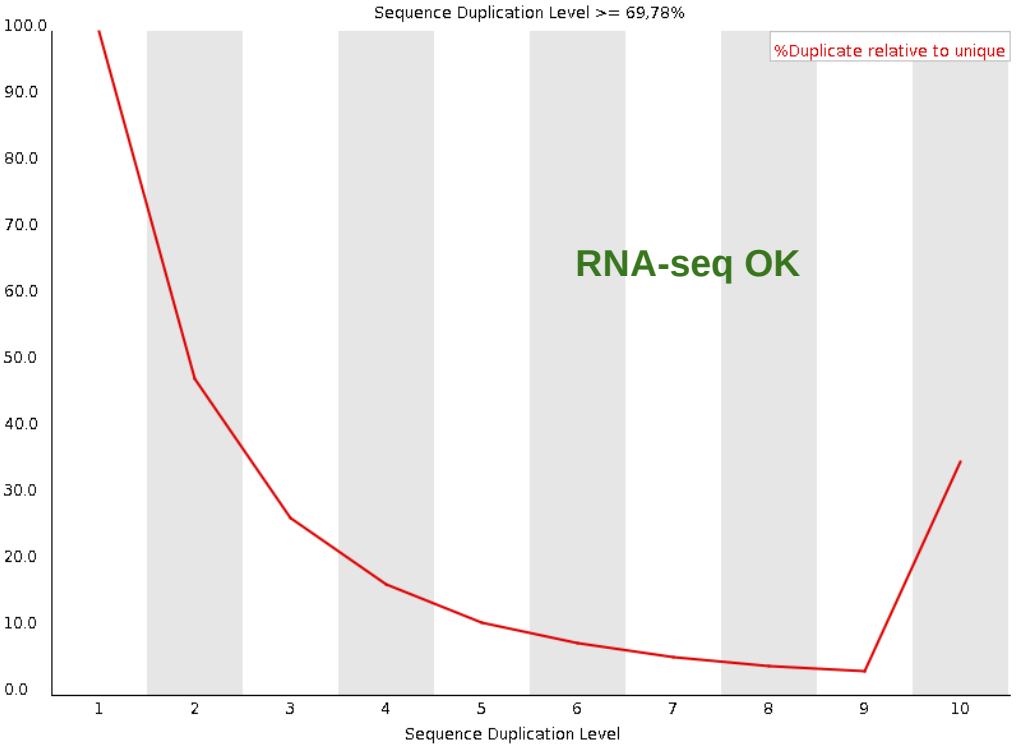


fastqQC Report

Statistics per Duplicate Sequences

High level of duplication indicate an enrichment biais.

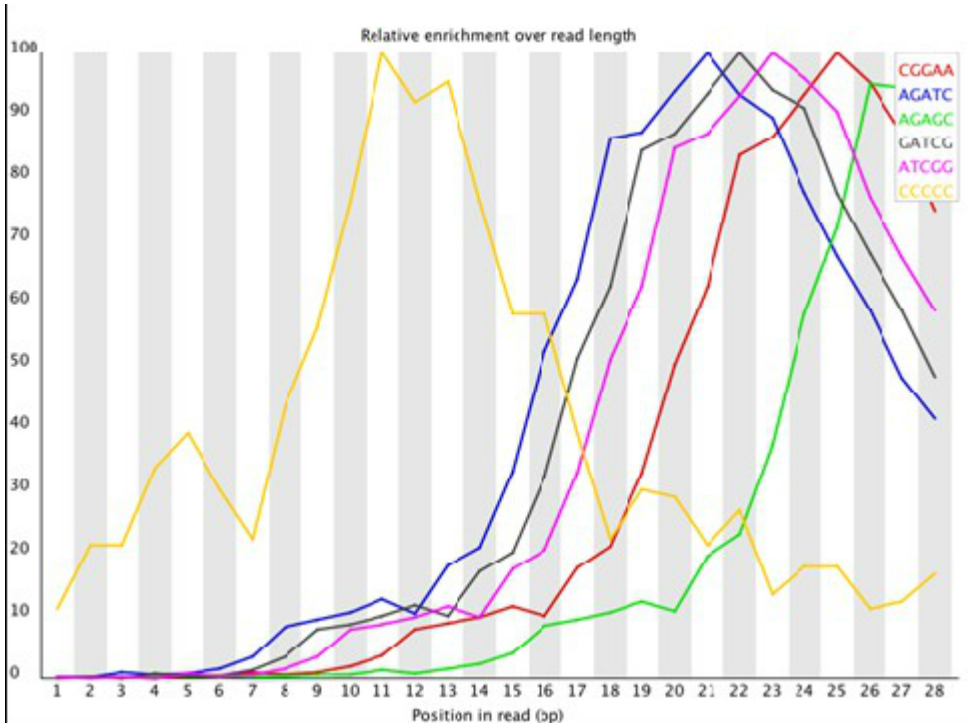


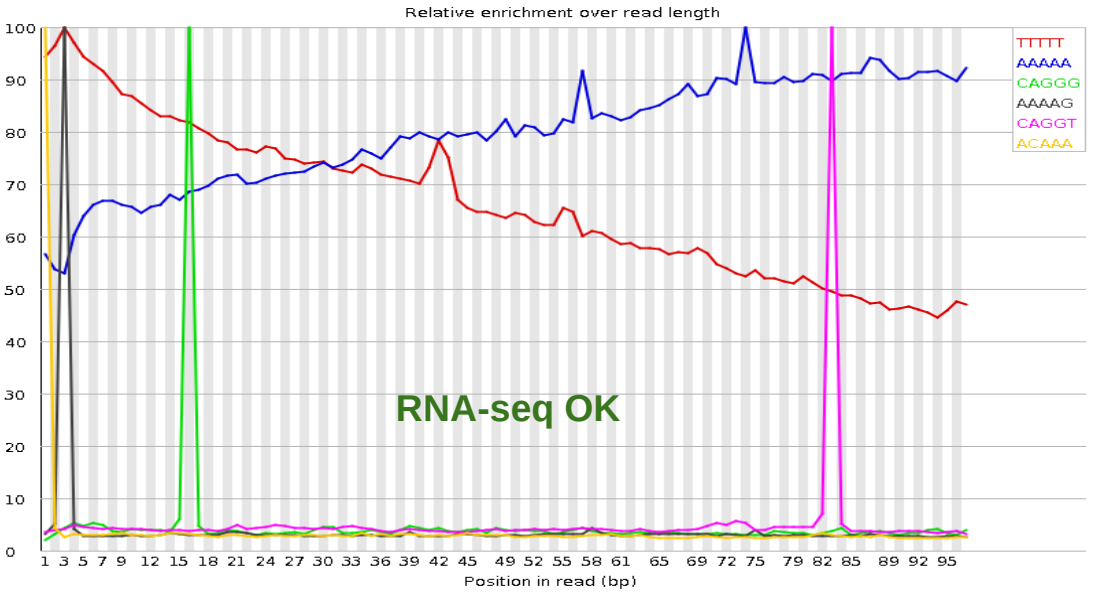


fastqQC Report

Overrepresented Kmers

- A kmer is a subsequence of length k
- Should spot overrepresented sequences, give a good impression of any contamination.
- Kmers showing a rise towards the end of the library indicate progressive contamination with adapters.
- Check for adaptor sequence or poly-A sequence





Sequence	Count	Obs/Exp Overall	Obs/Exp Max	Max Obs/Exp Position
TTTTT	47499960	4.84021	7.2762637	3
AAAAA	18101385	4.2297845	5.3006034	74
CAGGG	12486915	2.3769662	49.03375	16
AAAAG	10728075	2.3667703	56.233307	3

Take home message on quality analysis

Elements to be checked :

- Random priming effect
- K-mer (polyA, polyT)
- Adaptor presence

Alignment on reference for the second quality check and filtering.

A good run?:

- Expected number of reads produced,
- Length of the reads expected (150pb),
- Random selection of the nucleotides and the GC%,
- Good alignment: very few unmapped reads, pairs mapped on opposite strands.

Cleaning analysis

- Cleaning :
  - Low quality bases
  - Adaptors

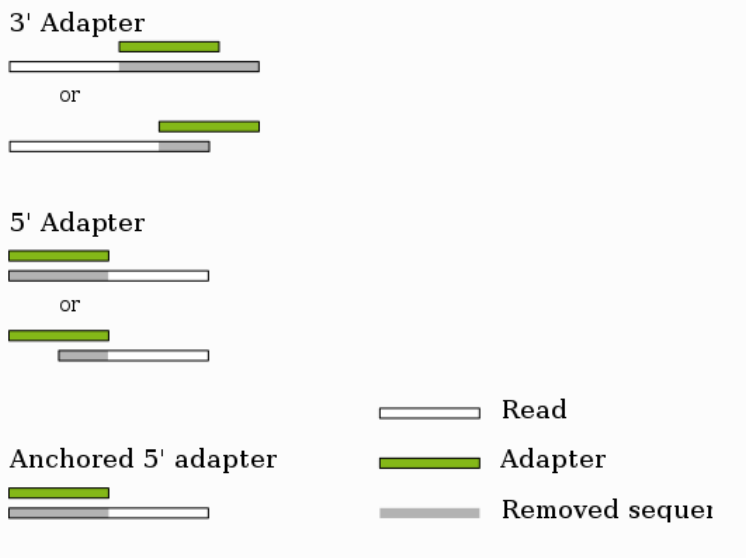
- Software :
  - Trim_galore
  - Cutadapt
  - Trimmomatic
  - Sickle
  - PRINSEQ
  - ...

Cutadapt

- Searches & removes adapter & tag in all reads.
- Trim quality
- Filter too short or untrimmed reads (in a separate output file).

```
module load bioinfo/cutadapt-1.8.3-python-2.7.2
cutadapt -a ADAPTER_FWD -A ADAPTER_REV -o out1.fatsq -p
out2.fastq  reads1.fastq reads2.fastq
```

Ex.: cutadapt -a AACCGGTT -o output.fastq input.fastq
(3’ adapter, single read)
Input file : fasta, fastq or compressed (gz, bz2, xz).



Source : http://cutadapt.readthedocs.io/en/stable/guide.html

trim_galore

- Detect automatically adaptor
- Trim adaptor
- Trim low quality bases
- Trim N bases
- Remove read with length lower than 20b

```
module load bioinfo/cutadapt-1.14-python-2.7.2
module load bioinfo/FastQC_v0.11.7
module load bioinfo/TrimGalore-0.4.5
mkdir DIR
trim_galore    --fastqc
               --stringency 3
               --length 25
               --trim-n
               -o DIR
               --paired <read1> <read2>
```

Hands-on: quality control

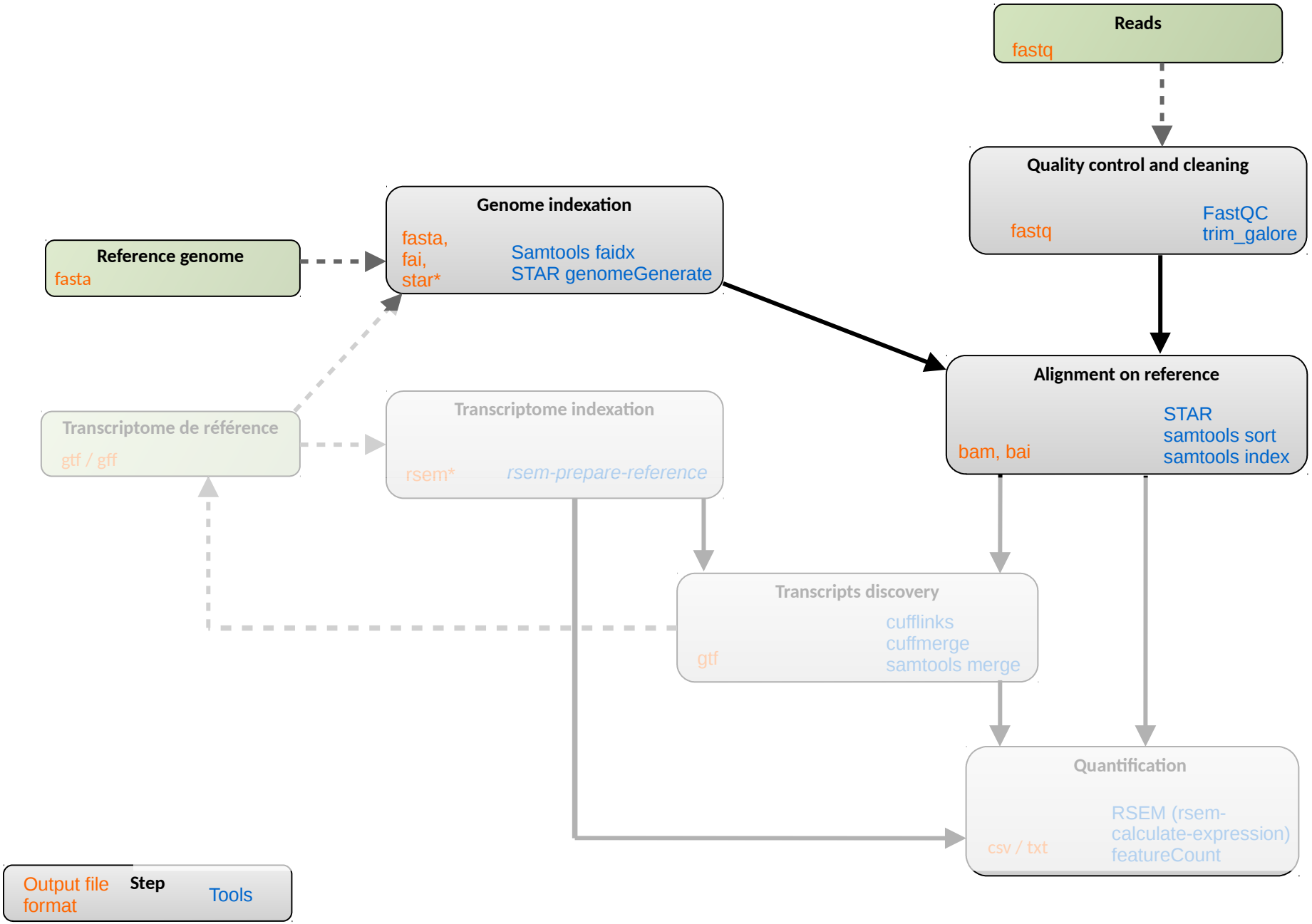
Data for the exercises:

- from Mohammed Zouine (ENSAT)
- tomato wild type and mutant type (without seeds) with the transcription factor Sl-ARF8 (auxine response factor 8) overexpressed
- clonal lineage
- paired, 100 pb non stranded
- triplicated
- in the publication process
- subsampled on chromosome 6 for faster analysis

Use FastQC and trim_galore

Exercise 1 : quality control of used datasets
cleaning used datasets

Analysis workflow



Summary -

Spliced read mapping & Visualisation

1. What is a spliced aligner?
2. Reference genome & transcriptome files formats
3. STAR principle and usage
4. BAM & Bed files formats
5. Visualisation with IGV

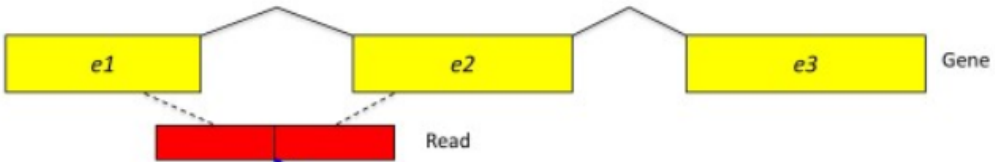
Aim -

Spliced read mapping & Visualisation

Aim: Discover the true location (origin) of each read on the reference.

Problems:

- Some features (repetitive regions, assembly errors, missing information) make it impossible for some reads.
- Reads may be split by potentially thousands of bases of intronic sequence.

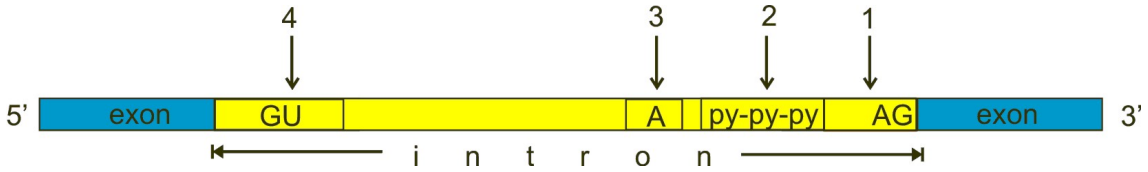


And:

Do it in/with reasonable time/resources.

Splice sites

- Canonical splice site:
- which accounts for more than 99% of splicing
- GT and AG for donor and acceptor sites



http://en.wikipedia.org/wiki/RNA_splicing

- Non-canonical site:
- GC-AG splice site pairs, AT-AC pairs

- Trans-splicing: splicing that joins two exons that are not within the same RNA transcript

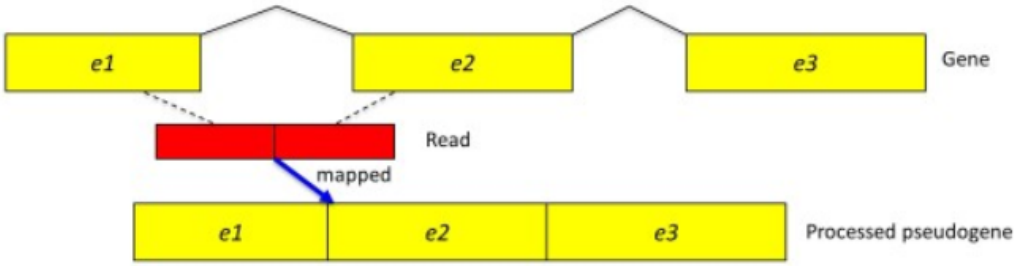
Nucleic Acids Res. 2000 Nov 1;28(21):4364-75.

Analysis of canonical and non-canonical splice sites in mammalian genomes.

Burset M, Seledtsov IA, Solovyev VV.

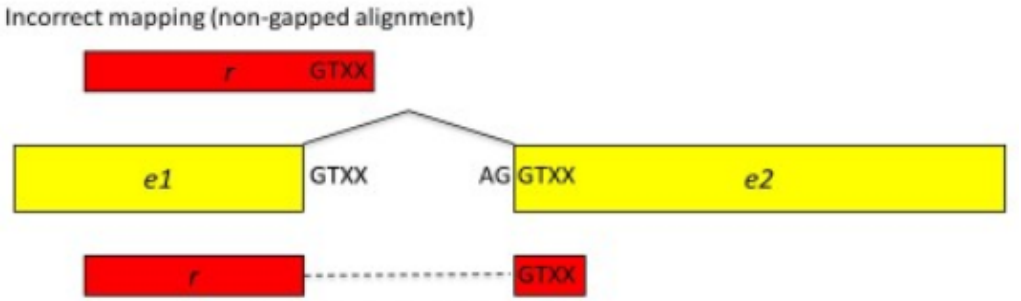
Hard case

- Lot of variations (sequencing errors, mutations)
- Repeats
- Reads spanning 3+ exons
- Gene or pseudogene



Kim et al, Genome Biology, 2013

- Small end “anchor”

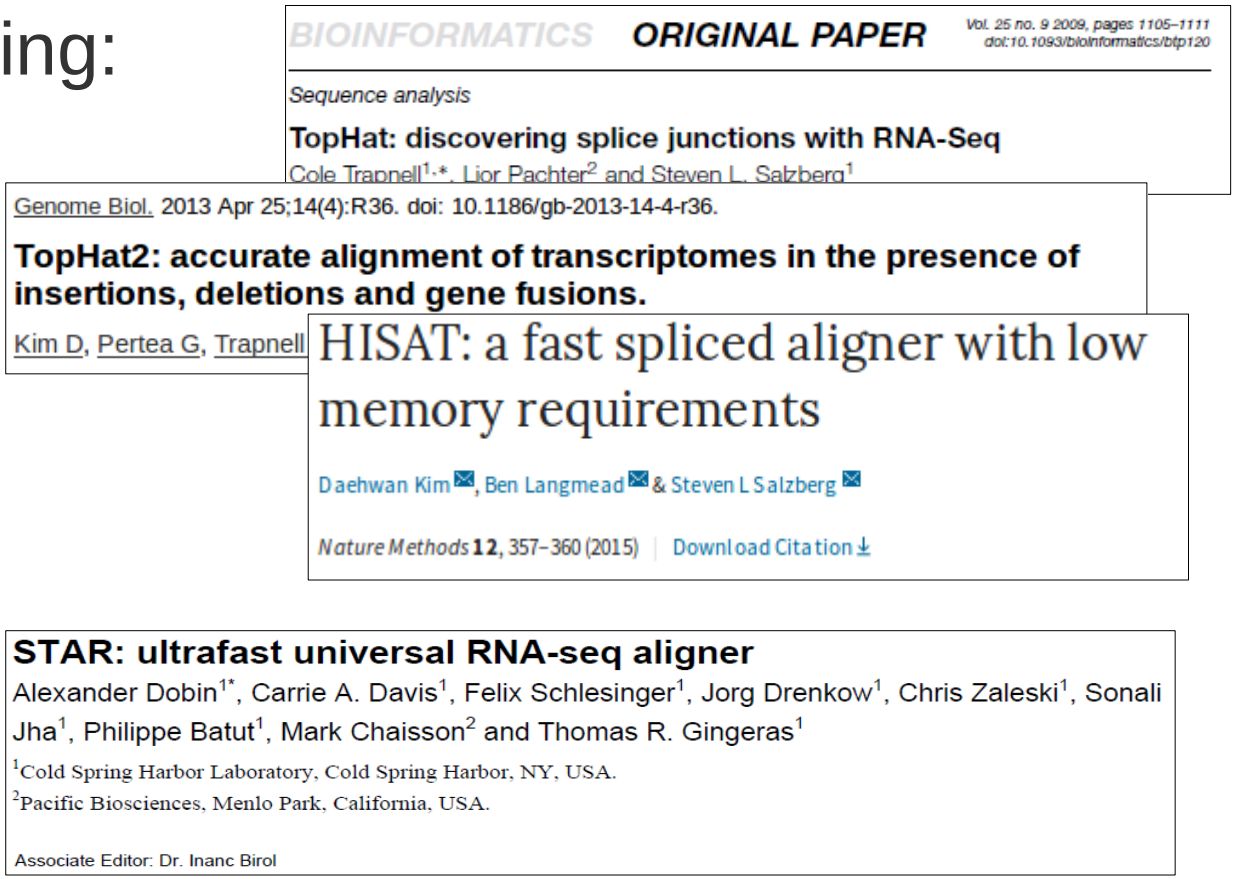


- Unknown junction inside poorly rarely expressed gene

Most used tools

Tools for splice-mapping:

- ~~Tophat:~~
- HISAT
- STAR:



BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 9 2009, pages 1105–1111 doi:10.1093/bioinformatics/btp120

Sequence analysis

TopHat: discovering splice junctions with RNA-Seq

Cole Trapnell[1,*], Lior Pachter[2] and Steven L. Salzberg[1]

Genome Biol. 2013 Apr 25;14(4):R36. doi: 10.1186/gb-2013-14-4-r36.

TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions.

Kim D, Pertea G, Trapnell

HISAT: a fast spliced aligner with low memory requirements

Daehwan Kim, Ben Langmead & Steven L Salzberg

Nature Methods **12**, 357–360 (2015) | Download Citation

STAR: ultrafast universal RNA-seq aligner

Alexander Dobin[1*], Carrie A. Davis[1], Felix Schlesinger[1], Jorg Drenkow[1], Chris Zaleski[1], Sonali Jha[1], Philippe Batut[1], Mark Chaisson[2] and Thomas R. Gingeras[1]

[1]Cold Spring Harbor Laboratory, Cold Spring Harbor, NY, USA.

[2]Pacific Biosciences, Menlo Park, California, USA.

Associate Editor: Dr. Inanc Birol

Benchmarks

NATURE METHODS | ANALYSIS

Simulation-based comprehensive benchmarking of RNA-seq aligners

Giacomo Baruzzo, Katharina E Hayer, Eun Ji Kim, Barbara Di Camillo, Garret A FitzGerald & Gregory R Grant

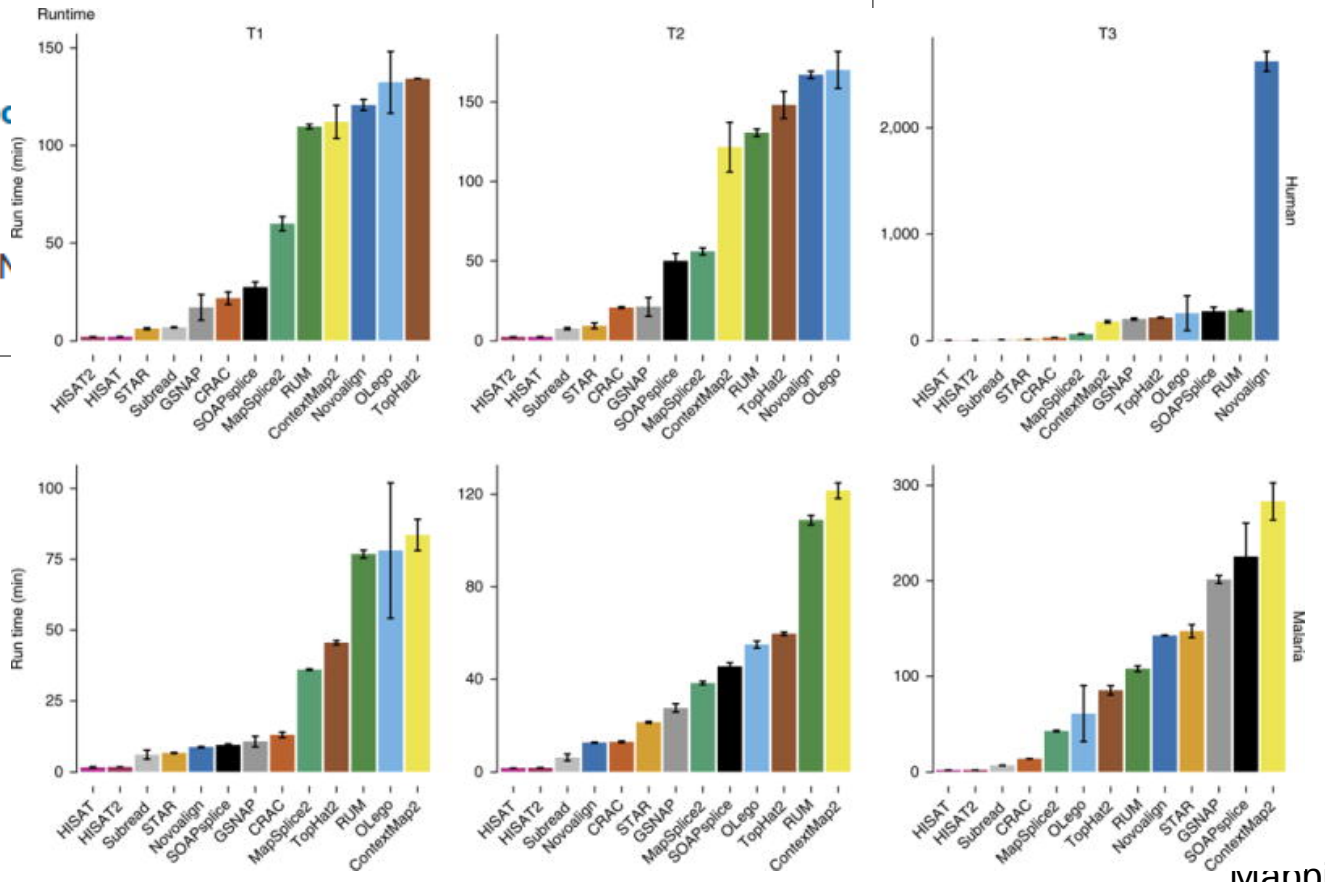
Affiliations | Contributions | Correspo

Nature Methods **14**, 135–139 (2017) |

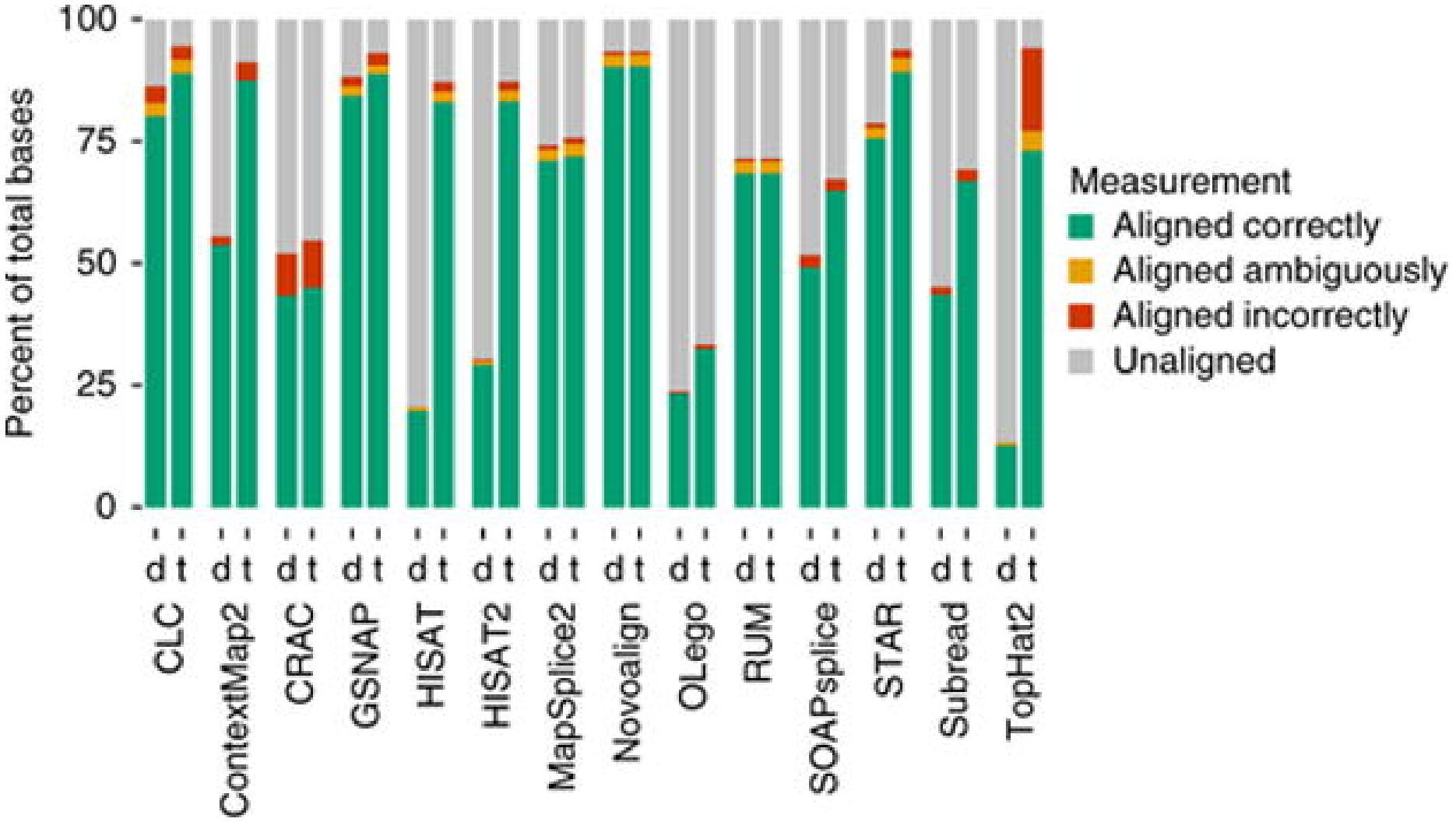
Received 18 April 2016 | Accepted 15 M

| Corrected online **22 December 2016**

Run time:



Tuning parameters



on the human-T3-data base-level statistics.

« Therefore, an algorithm that is robust to parameter settings and exhibits good performance using defaults is desirable »

« most reliable general-purpose aligners appear to be CLC, Novoalign, GSNAP, and STAR. »

rnaSTAR

OXFORD JOURNALS Bioinformatics (Oxford, England)

Bioinformatics. 2013 Jan; 29(1): 15–21. PMCID: PMC3530905

Published online 2012 Oct 25. doi: 10.1093/bioinformatics/bts635

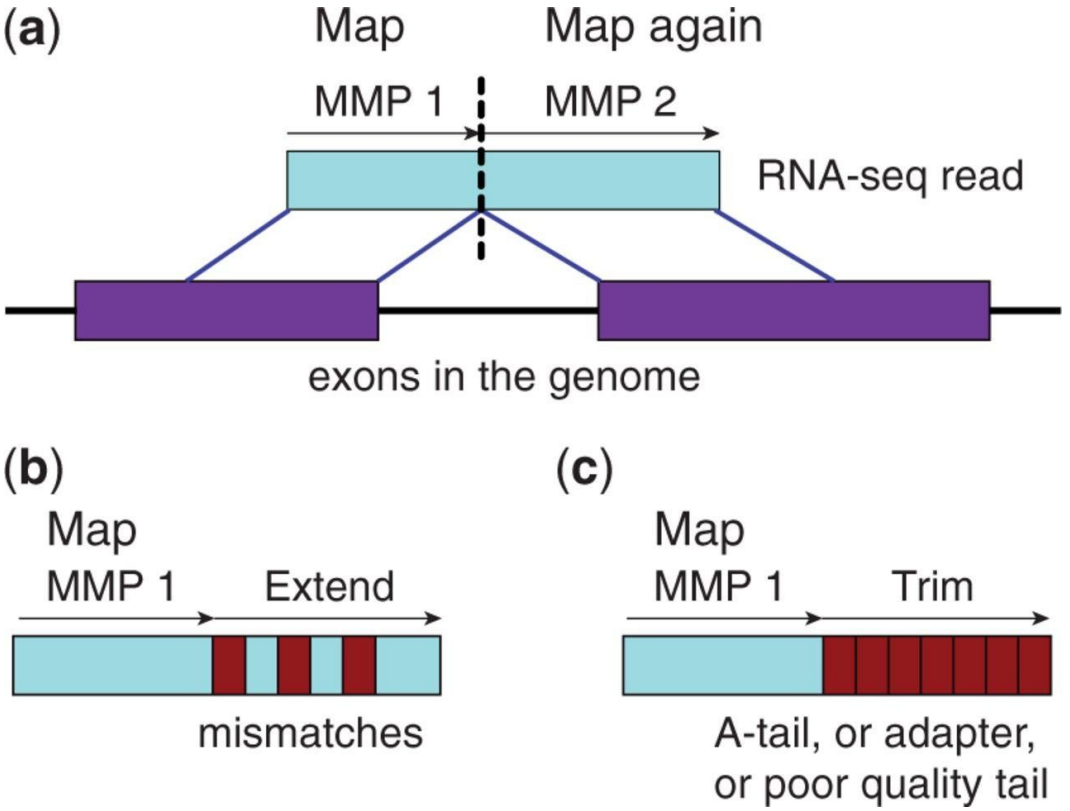
STAR: ultrafast universal RNA-seq aligner

Alexander Dobin,[1,*] Carrie A. Davis,[1] Felix Schlesinger,[1] Jorg Drenkow,[1] Chris Zaleski,[1] Sonali Jha,[1] Philippe Batut,[1] Mark Chaisson,[2] and Thomas R. Gingeras[1]

- Spliced Transcripts Alignment to a Reference
- Outperforms other aligners by more than a factor of 50 in mapping speed

rnaSTAR strategy

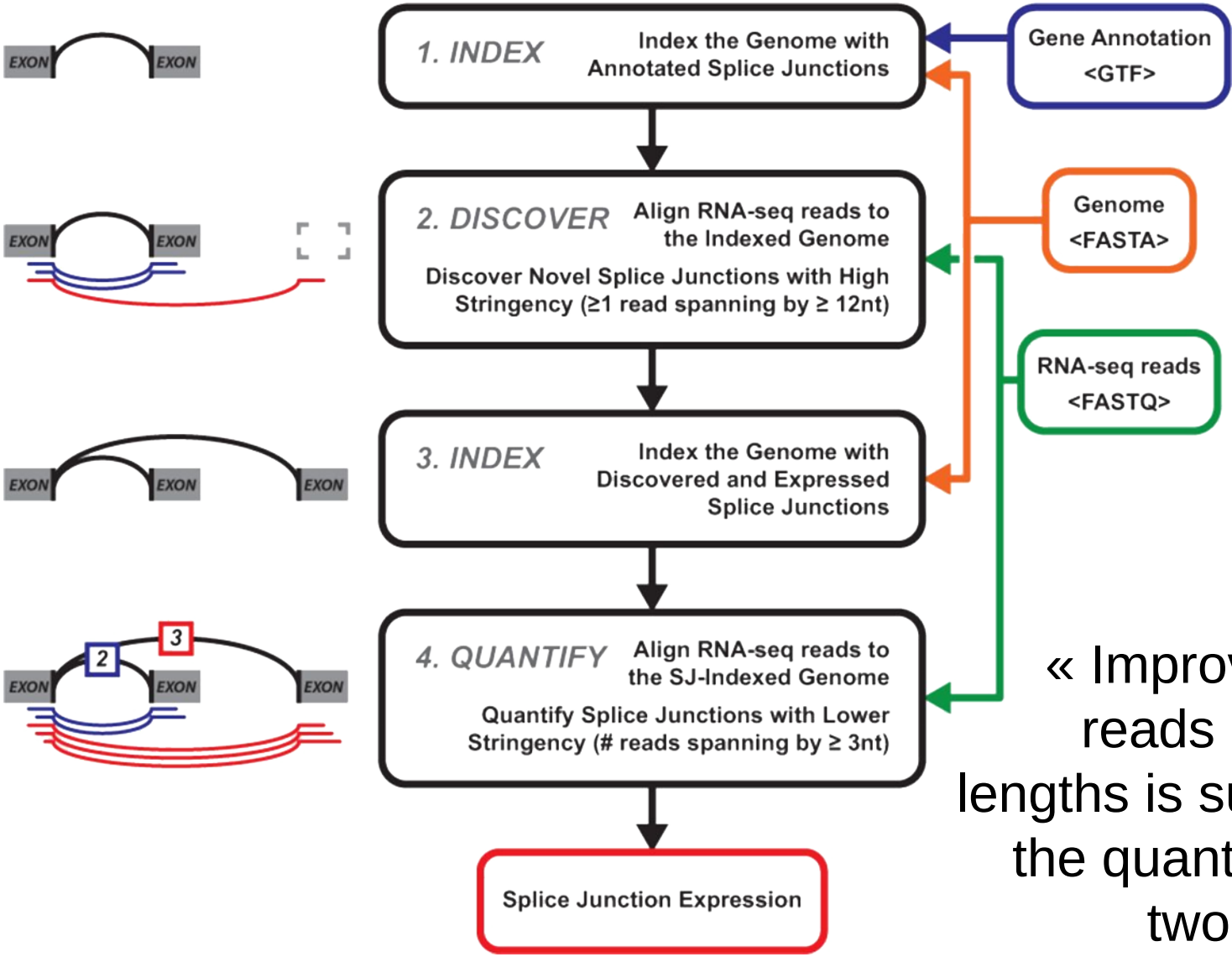
- search for a MMP from the 1st base
- MMP search repeated for the unmapped portion next to the junction
- do it in both fwd and rev directions
- cluster seeds from the mates of paired-end RNA-seq reads



Dobin *et al*, Bioinformatics, 2011

Soft-clipping is the main difference between Tophat and STAR

STAR : two passes strategy



« Improved ability to align reads by short spanning lengths is sufficient to explain the quantification benefit of two-pass alignment »

Veeneman et al, Bioinformatics, 2016

STAR indexing

```
module load bioinfo/starXXX
STAR --runMode genomeGenerate --genomeDir genome_dir --genomeFastaFiles genome.fasta
```

To use *N* CPUs, add: `--runThreadN N`
With an annotation: `--sjdbGTFfile annot.gtf`

Some precomputed indices are already available:
http://labshare.cshl.edu/shares/gingeraslab/www-data/dobin/STAR/STARgenomes
or on your preferred platform: /bank/STARdb

Where to find a reference genome?

Retrieving the genome file (fasta):

- The Genome Reference Consortium

http://www.ncbi.nlm.nih.gov/projects/genome/assembly/grc/

- NCBI chromosome naming with « | » not well supported by mapping software

- Prefer EMBL

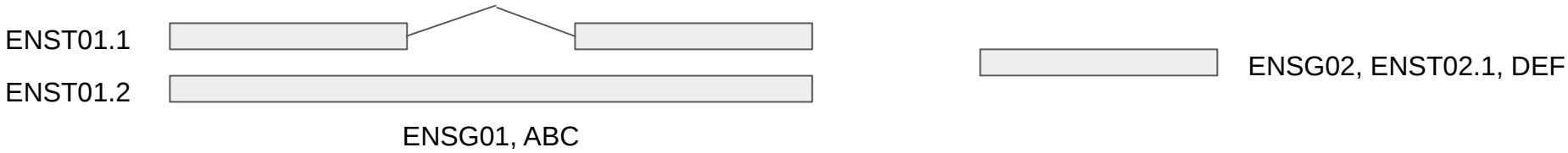
http://www.ensembl.org/info/data/ftp/index.html

Reference transcriptome file

What is a **GTF** file ?

- An annotation file: loci of coding genes (transcripts, CDS, UTRs), non-coding genes, etc.
- Gene Transfer Format (derived from GFF)

```
chr source  feature start end  score strand frame [attributes]
1   ENSEMBL exon    1000  2000 .     +      .     gene_id "ENSG01"; transcript_id “ENST01.1”; gene_name "ABC";
1   ENSEMBL exon    3000  4000 .     +      .     gene_id "ENSG01"; transcript_id “ENST01.1”; gene_name "ABC";
1   ENSEMBL exon    1000  4000 .     +      .     gene_id "ENSG01"; transcript_id “ENST01.2”; gene_name "ABC";
1   ENSEMBL exon    5000  6000 .     +      .     gene_id "ENSG02"; transcript_id “ENST02.1”; gene_name "DEF";
```



- `gene_id` *value* : unique identifier for the gene.
- `transcript_id` *value* : unique identifier for the transcript.

The chromosome names MUST be the same in the gtf file and fasta files (e.g. `chr1` vs `Chr1` vs `1`).

http://genome.ucsc.edu/FAQ/FAQformat.html#format4

Hands-on : STAR

Exercise n°3

Create a directory for the genome and annotation files.

Get the FASTA and GTF files from:
http://genoweb.toulouse.inra.fr/~formation/19_Rnaseq_Cli/data/reference/

Create the STAR index.
Tip: you can allocate *N* CPUs with the `sbatch -c 8`

STAR alignment

```
module load bioinfo/starXXX
STAR --genomeDir genome_dir
--readFilesIn read1.fastq.gz read2.fastq.gz
--readFilesCommand zcat
--sjdbGTFfile transcriptome.gtf
--alignIntronMin 20 --alignIntronMax 500000
--outSAMtype BAM SortedByCoordinate → sort
--outSAMstrandField intronMotif → for cufflinks
--alignSoftClipAtReferenceEnds No → for cufflinks
--outSAMattrIHstart 0 → for cufflinks or StringTie
--outFilterType BySJout → filter by splice site
--outFilterIntronMotifs RemoveNoncanonical → filter
--quantMode TranscriptomeSAM GeneCounts → for RSEM
 --outSAMattributes All → more information
--outFileNamePrefix sampleName
--runThreadN 4
```

STAR options

Intron size

```
--alignIntronMin  20
--alignIntronMax 500000
```

Allow soft-clipping past the end of chr (for cufflinks No)

`--alignSoftClipAtReferenceEnds No` [default Yes]

Output format:

`--outSAMtype BAM SortedByCoordinate [SAM]`

Output SAM/BAM alignments to transcriptome into a separate file (for RSEM)

`--quantMode TranscriptomeSAM`

→ need `:--sjdbGTFfile annot.gtf`

Output read unmapped

`--outReadsUnmapped Fastx`

STAR options

Add more tags:
`--outSAMattributes All`

Default output file name: `Aligned.bam` Modify prefix:
`--outFileNamePrefix` *`prefix`*

Infer strand using intron motifs (for Cufflinks)
`--outSAMstrandField intronMotif [None]`

Start IH at `--outSAMattrIHstart 0 [1]` (for Cufflinks)

STAR options

Remove reads that did not pass the junction filter:

`--outFilterType BySJOut [Normal]`

Filter out alignments with non-canonical intron motifs

`--outFilterIntronMotifs RemoveNoncanonical`

Mismatches :

`--outFilterMismatchNmax [default: 10]`

Limit multimap outputed:

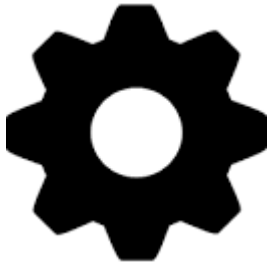
`--outFilterMultimapNmax [Default: 10]`

> Flag of secondary alignment 0x100

Too short alignemnt

`--outFilterMatchNminOverLread 0.66`

`--outFilterScoreMinOverLread 0.66`



STAR - two passes mode

- First pass: discover new junctions.

- Second pass: run again with knowing the new junctions. (most useful for poorly annotated genomes.)

`--twopassMode [`**`None`**`|Basic]`

Defines the number of reads to be mapped in the 1st pass :
`--twopass1readsN [-1]`

STAR Output files

Outputs (w/o specific options except `BAM SortedByCoordinate`):

- `Aligned.sortedByCoord.out.bam`: list of read alignments in SAM format compressed
- `Log.out`: main log file with a lot of detailed information about the run (for troubleshooting)
- `Log.progress.out`: reports job progress statistics
- `Log.final.out`: summary mapping statistics after mapping job is complete, very useful for quality control.
- `SJ.out.tab`: contains high confidence collapsed splice junctions in tab-delimited format

(chr, intron start, end, strand, intron motif, in database, # uniquely mapping reads, # multi, max. overhang)

STAR technical issues

- Temporary disk space:
  - Indexing the mouse genome requires 128GB and 1 hour on 6 slots.
  - Mapping a 16M paired-end reads requires 110GB and 4 mins on 6 slots.

- Available cluster:
  - New : 48 nodes with 32 cores and 256 GB of ram per node
  - Old : 68 nodes with 20 cores and 256 GB of ram per node

Hands-on : STAR

Exercise n°3

Map the 2 FASTQ files.

Do not forget to provide a different output file name for each set.

Index the output BAM files with:

```
samtools index file.bam
```

→ Then BAM format presentation.

SAM / BAM formats

Sequence Alignment/Map format:

- Each line stores an alignment/map

```
Coor    12345678901234  5678901234567890123456789012345
ref     AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

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

name flag chr start mapQ  cigar   nNext sNext tlen     seq          qual    tags
r001   99 ref    7    30 8M2I4M1D3M =     37    39 TTAGATAAAGGATACTG *
r002    0 ref    9    30 3S6M1P1I4M *      0     0 AAAAGATAAGGATA    *
r003    0 ref    9    30       5S6M *      0     0 GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004    0 ref   16    30    6M14N5M *      0     0 ATAGCTTCAGC       *
r003 2064 ref   29    17       6H5M *      0     0 TAGGC             * SA:Z:ref,9,+,5S6M,30,1;
r001  147 ref   37    30         9M =      7   -39 CAGCGGCAT         * NM:i:1
```

- Header stores genome information

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:45
```

Fields

```
Coor    12345678901234  5678901234567890123456789012345
ref     AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

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

name	flag	chr	start	mapQ	cigar	nNext	sNext	tlen	seq	qual	tags
r001	**99**	ref	7	**30**	8M2I4M1D3M	=	37	39	TTAGATAAAGGATACTG	*	
r002	**0**	ref	9	**30**	3S6M1P1I4M	*	0	0	AAAAGATAAGGATA	*	
r003	**0**	ref	9	**30**	5S6M	*	0	0	GCCTAAGCTAA	*	SA:Z:ref,29,-,6H5M,17,0;
r004	**0**	ref	16	**30**	6M14N5M	*	0	0	ATAGCTTCAGC	*	
r003	**2064**	ref	29	**17**	6H5M	*	0	0	TAGGC	*	SA:Z:ref,9,+,5S6M,30,1;
r001	**147**	ref	37	**30**	9M	=	7	-39	CAGCGGCAT	*	NM:i:1

- Flags: https://broadinstitute.github.io/picard/explain-flags.html
- MapQ: similar to a phred score
- nNext: = means same chr
- In general, * means NA

CIGAR

```
Coor    12345678901234  5678901234567890123456789012345
ref     AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

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

name	flag	chr	start	mapQ	**cigar**	nNext	sNext	tlen	seq	qual	tags
r001	99	ref	7	30	**8M2I4M1D3M**	=	37	39	TTAGATAAAGGATACTG	*	
r002	0	ref	9	30	**3S6M1P1I4M**	*	0	0	AAAAGATAAGGATA	*	
r003	0	ref	9	30	**5S6M**	*	0	0	GCCTAAGCTAA	*	SA:Z:ref,29,-,6H5M,17,0;
r004	0	ref	16	30	**6M14N5M**	*	0	0	ATAGCTTCAGC	*	
r003	2064	ref	29	17	**6H5M**	*	0	0	TAGGC	*	SA:Z:ref,9,+,5S6M,30,1;
r001	147	ref	37	30	**9M**	=	7	-39	CAGCGGCAT	*	NM:i:1

- `30M` means 30 matches or mismatches
- `I` and `D`: insertion/deletion
- `S` and `H`: soft/hard clipping

Tags

```
Coor    12345678901234  5678901234567890123456789012345
ref     AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

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

name flag chr start mapQ  cigar   nNext sNext tlen     seq          qual    tags
r001   99 ref    7    30 8M2I4M1D3M =     37    39 TTAGATAAAGGATACTG *
r002    0 ref    9    30 3S6M1P1I4M *      0     0 AAAAGATAAGGATA    *
r003    0 ref    9    30       5S6M *      0     0 GCCTAAGCTAA       * SA:Z:ref,29,-,6H5M,17,0;
r004    0 ref   16    30    6M14N5M *      0     0 ATAGCTTCAGC       *
r003 2064 ref   29    17       6H5M *      0     0 TAGGC             * SA:Z:ref,9,+,5S6M,30,1;
r001  147 ref   37    30         9M =      7   -39 CAGCGGCAT         * NM:i:1
```

- Format: *2-letter name*:*format*:*value* (many different)
- NM: # mismatches
- SA: chimeric reads
- NH, HI: # hits for this sequence, hit index
- AS: alignment score
- nM: # mismatches per fragment

SAM / BAM

BAM (Binary Alignment/Map) format:

- Compressed binary representation of SAM
- Greatly reduces storage space requirements to about 27% of original SAM
- samtools: reading, writing, and manipulating BAM files
- Most tools require a sorted and indexed BAM file.

- To be viewed a bam file must be indexed :
  `samtools index`

samtools

```
Program: samtools (Tools for alignments in the SAM format)
Version: 1.8 (using htslib 1.8)

Usage:   samtools <command> [options]

Commands:
  -- Indexing
     dict           create a sequence dictionary file
     faidx          index/extract FASTA
     index          index alignment

  -- Editing
     calmd          recalculate MD/NM tags and '=' bases
     fixmate        fix mate information
     reheader       replace BAM header
     targetcut      cut fosmid regions (for fosmid pool only)
     addreplacerg   adds or replaces RG tags
     markdup        mark duplicates

  -- File operations
     collate        shuffle and group alignments by name
     cat            concatenate BAMs
     merge          merge sorted alignments
```

```
module load bioinfo/samtools-1.8
```

Bam → sam

```
samtools view in.bam
```

Sam → bam

```
samtools view in.sam > out.bam
```

Sort

```
samtools sort -o out.bam in.bam
```

Index

```
samtools sort in.bam
```

Global options nb threads:

```
-@ 4
```

Visualizing alignments on IGV

http://www.broadinstitute.org/igv/home

NATURE BIOTECHNOLOGY | OPINION AND COMMENT | CORRESPONDENCE

Integrative genomics viewer

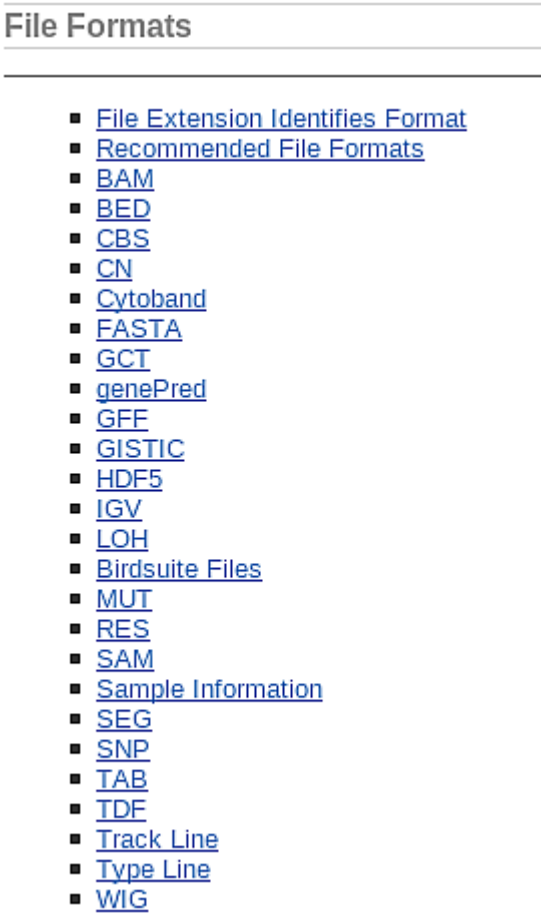
James T Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S Lander, Gad Getz & Jill P Mesirov

Affiliations | **Corresponding authors**

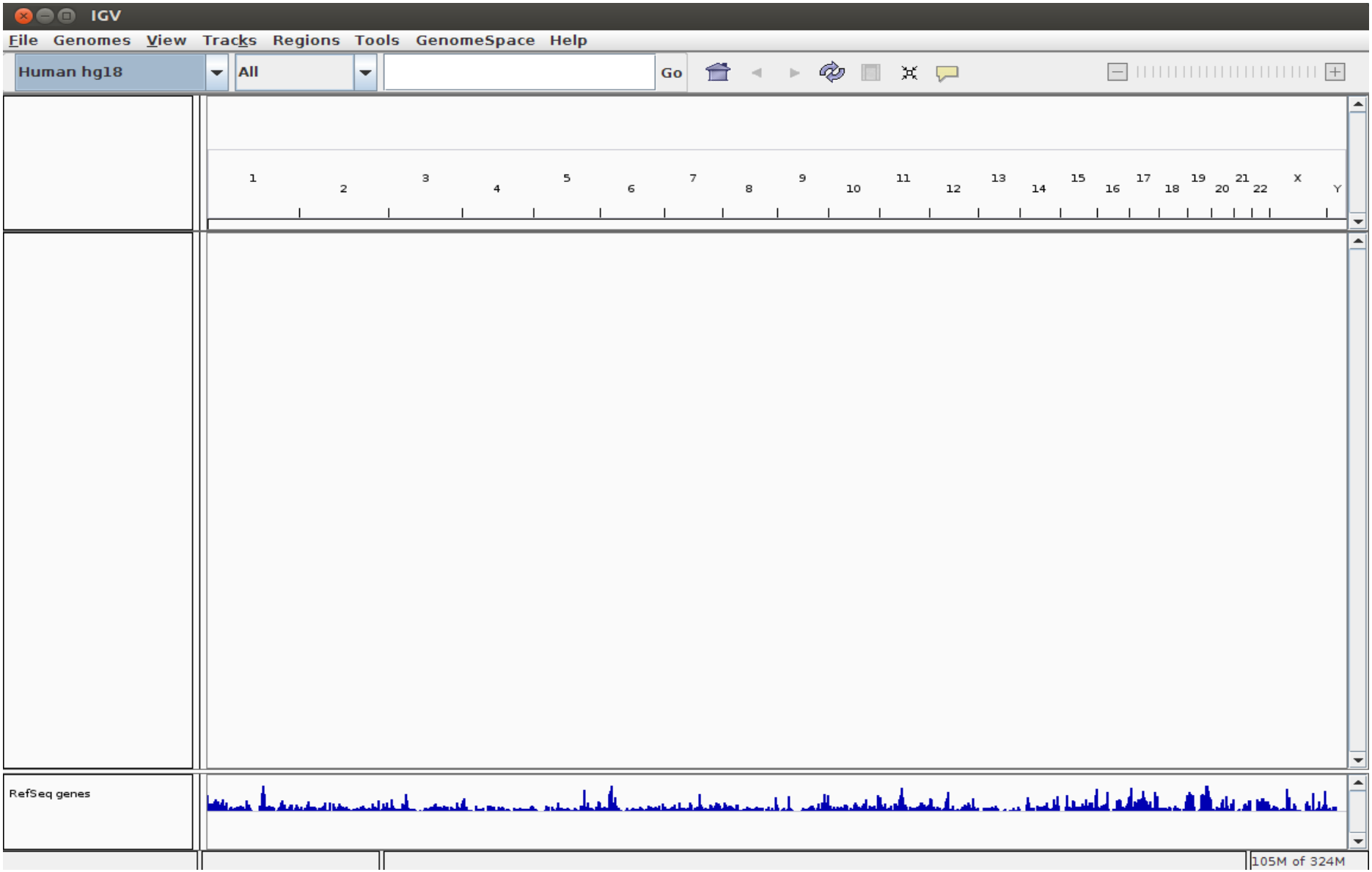
Nature Biotechnology **29**, 24–26 (2011) | doi:10.1038/nbt.1754
Published online 10 January 2011

Visualizing alignments on IGV

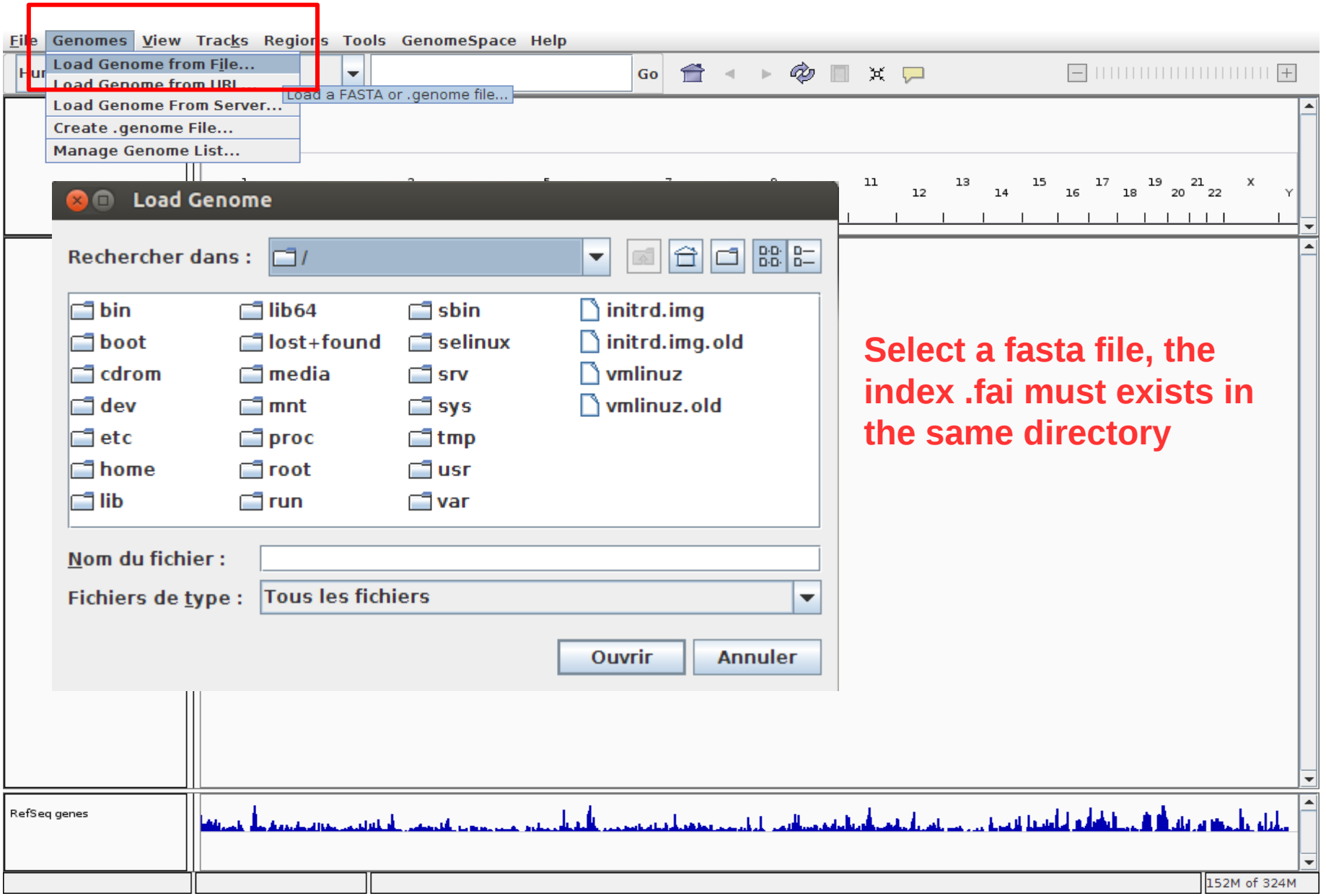
- High-performance visualization tool
- Interactive exploration of large, integrated datasets
- Supports a wide variety of data types
- Documentations
- Developed at the Broad Institute of MIT and Harvard



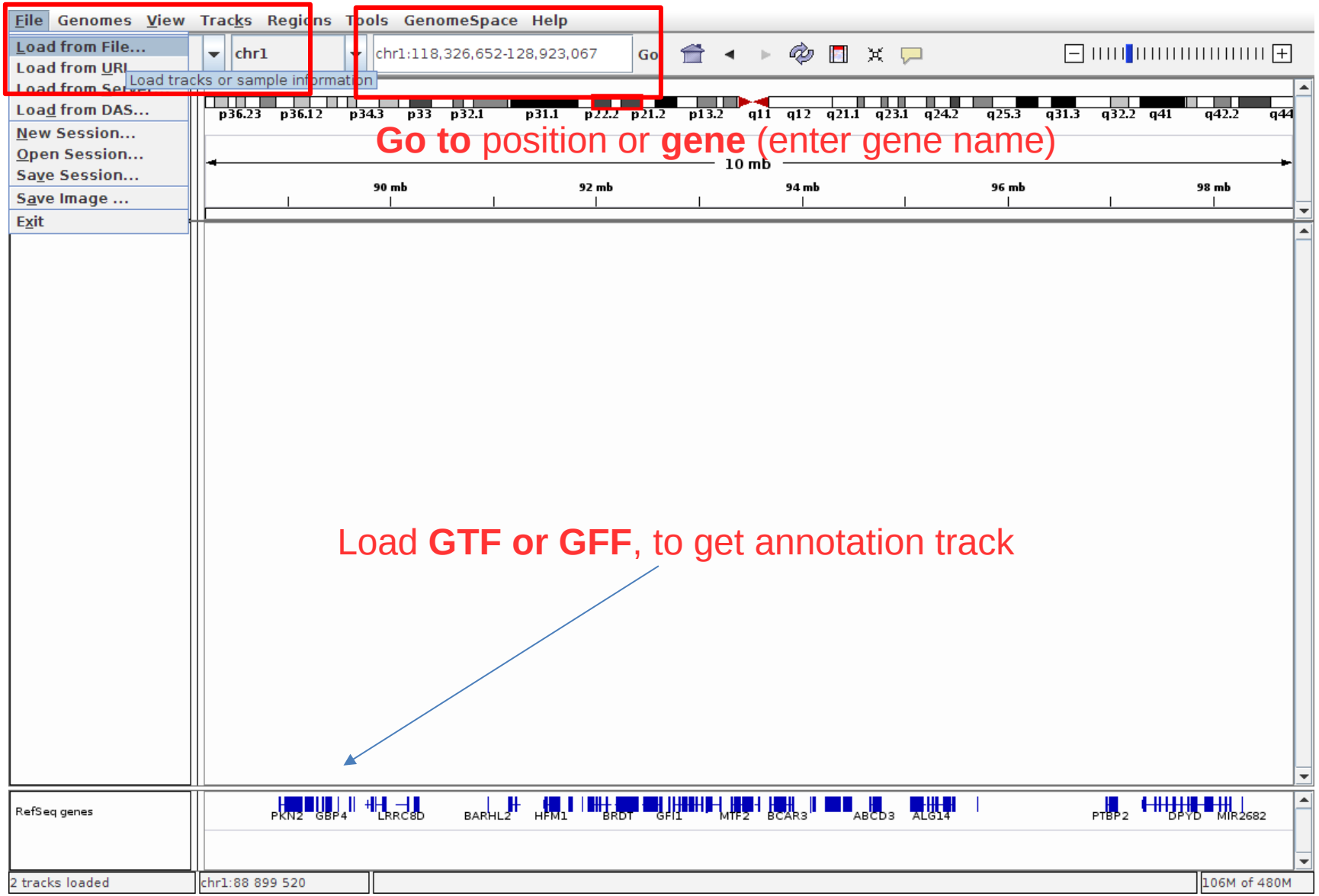
Visualizing alignments on IGV



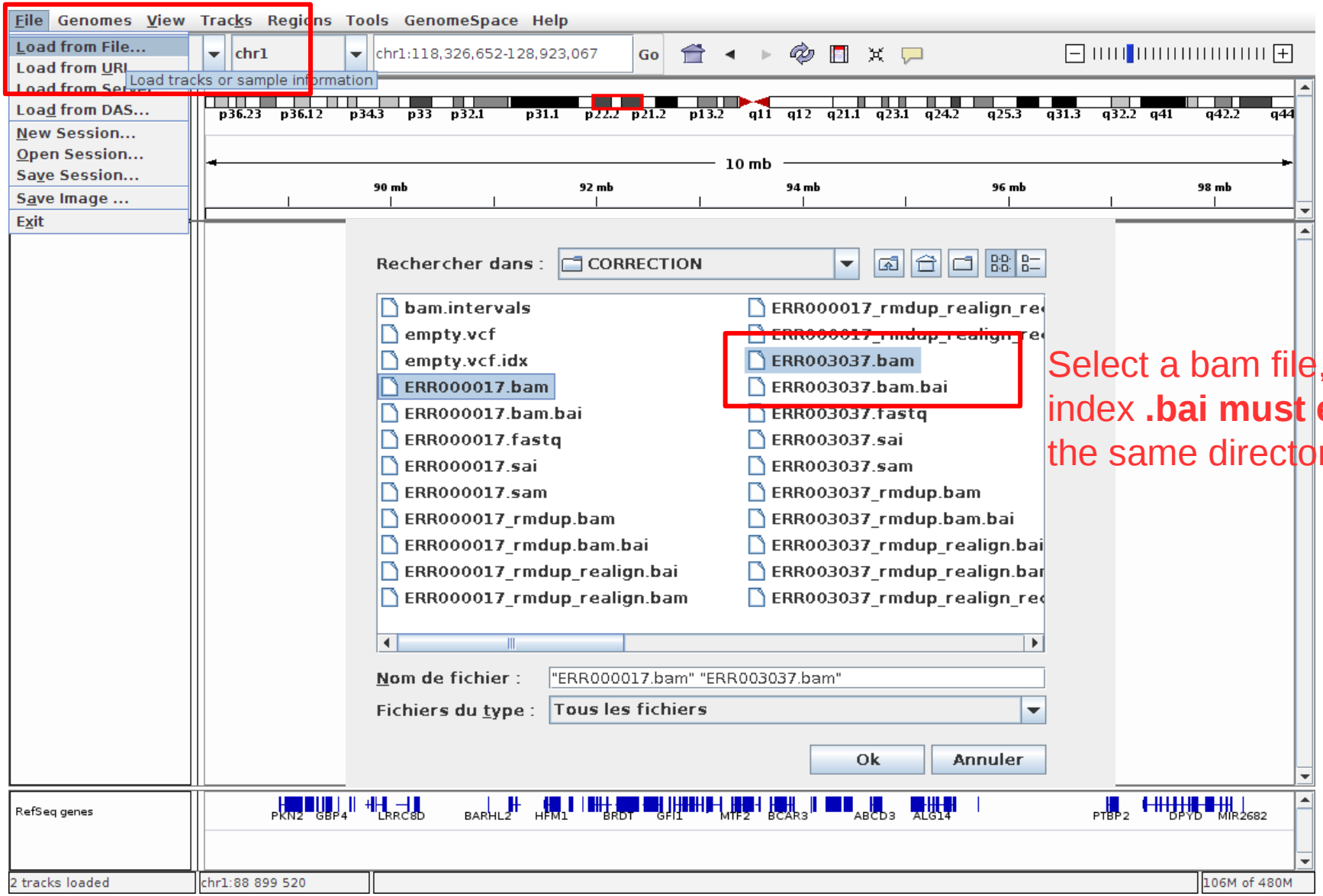
IGV : Load reference genome



IGV : Load annotation

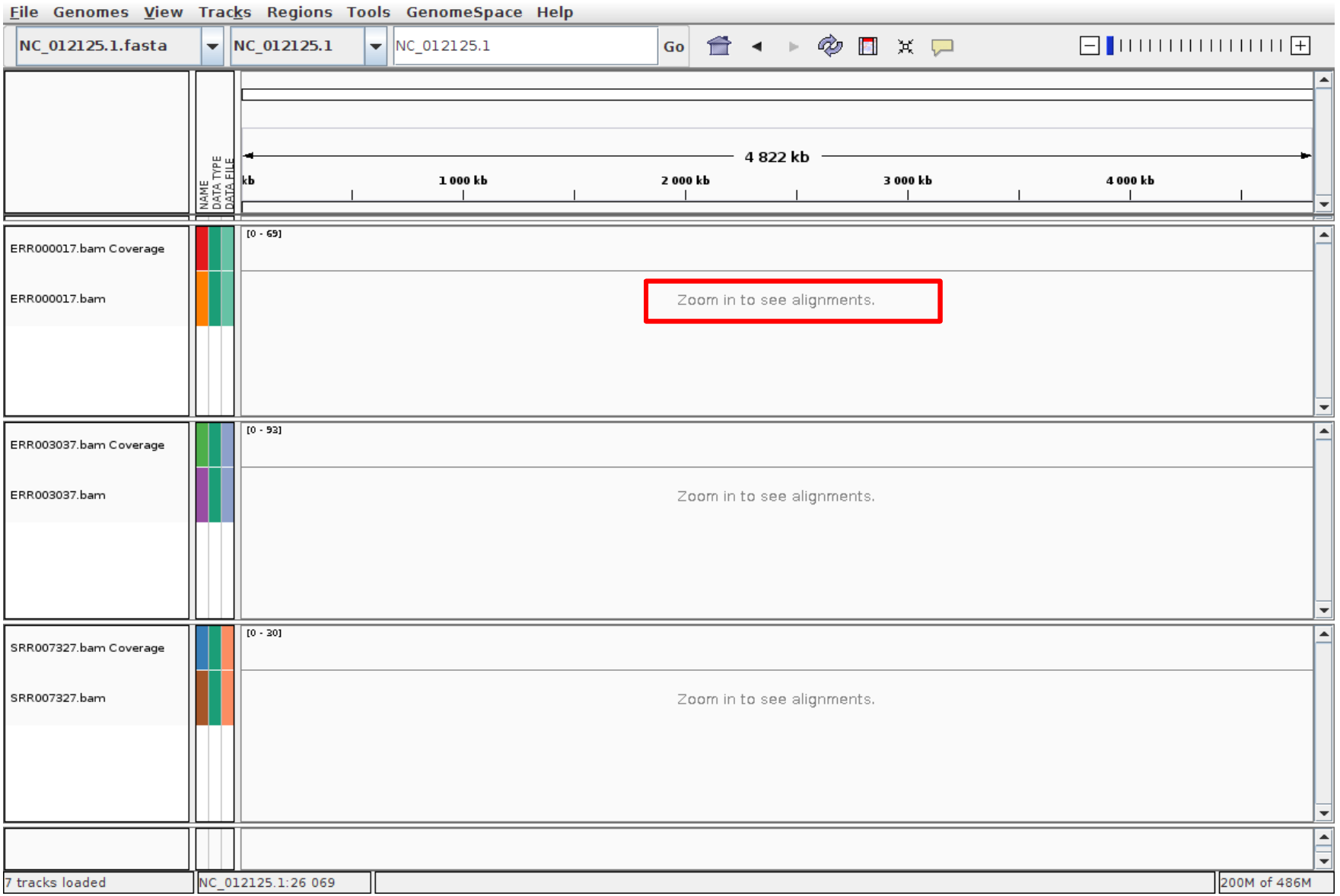


IGV : Load alignment

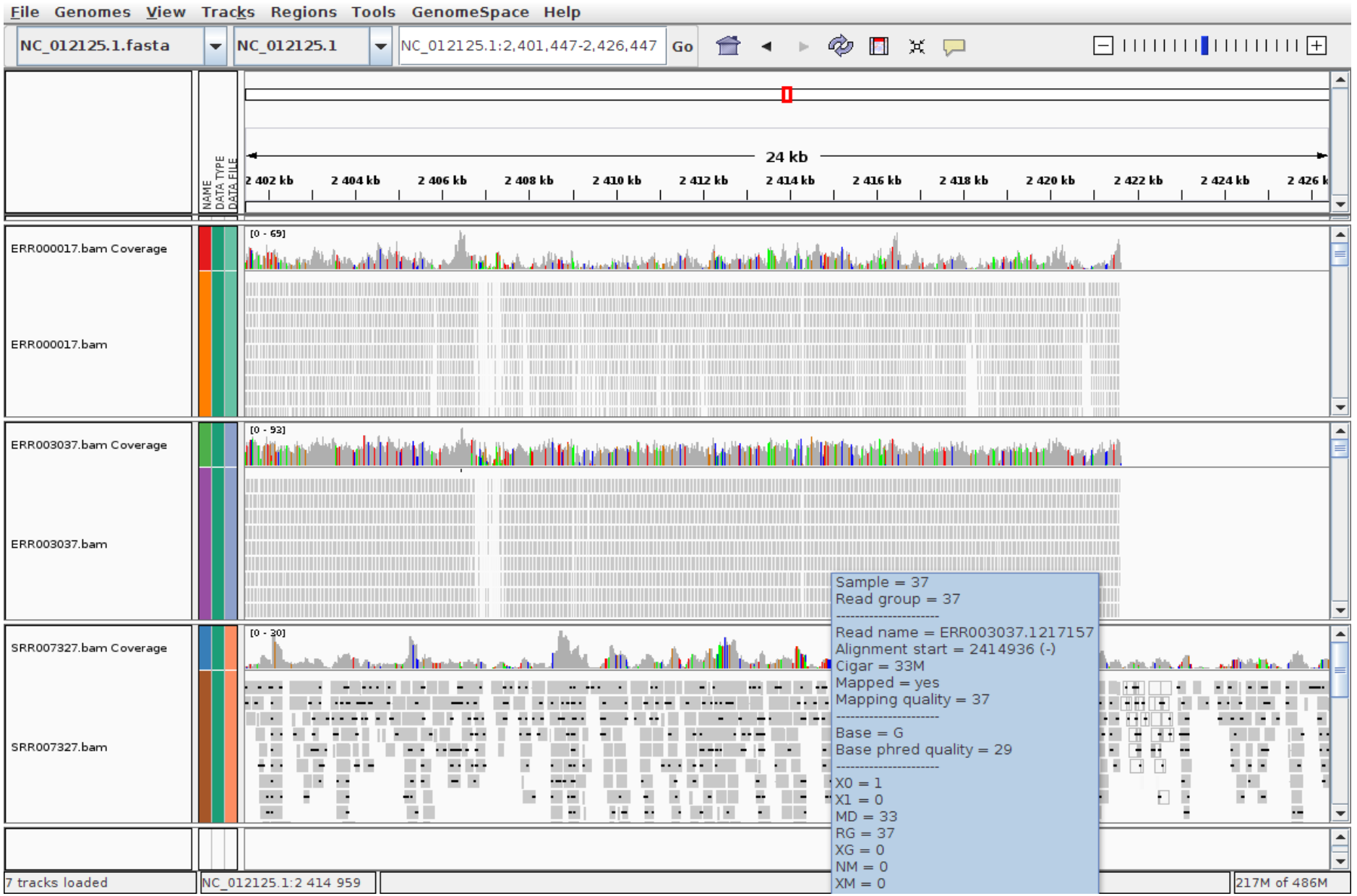


Select a bam file, the index **.bai must exists** in the same directory

IGV : Load alignment

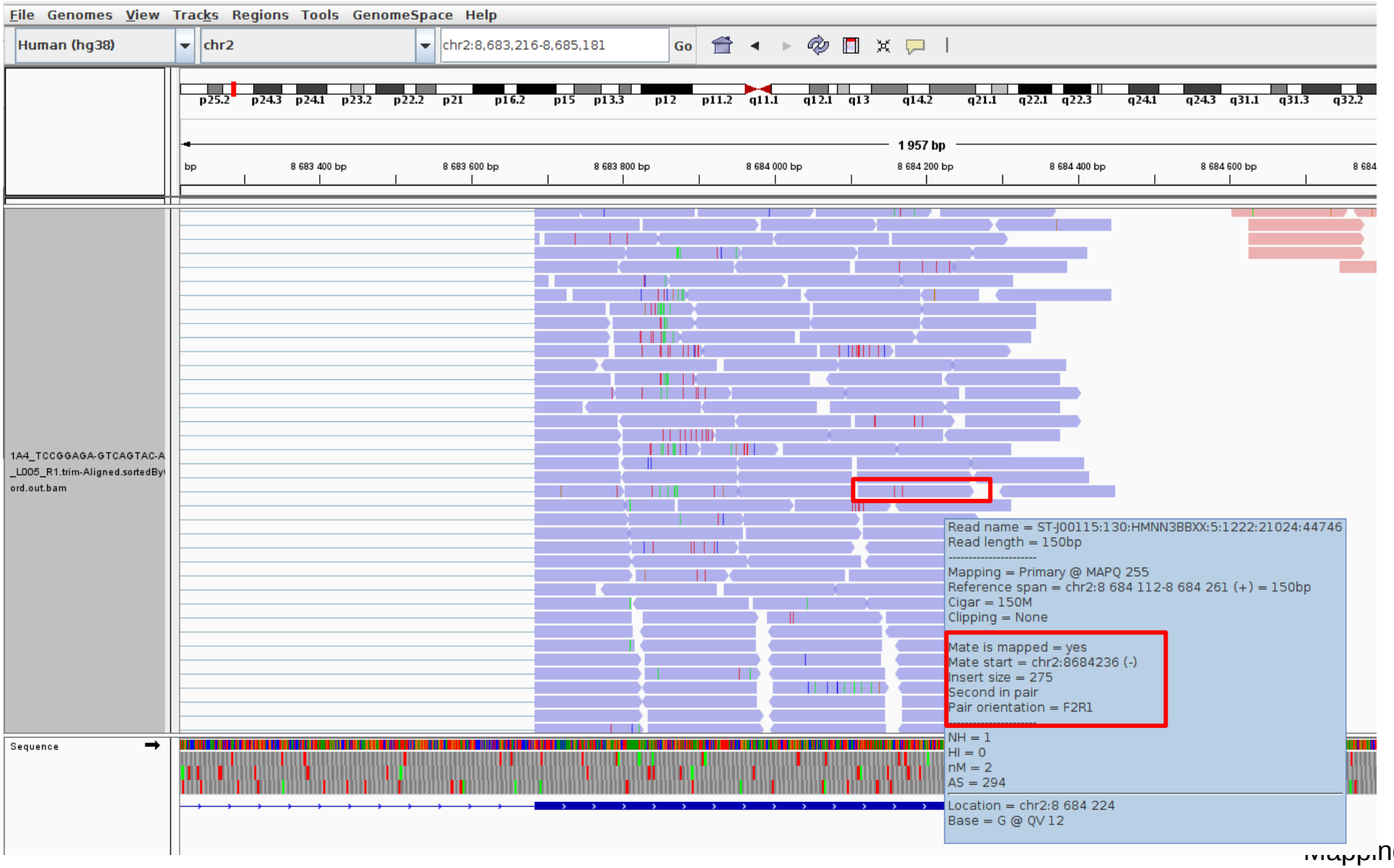


IGV : Load alignment



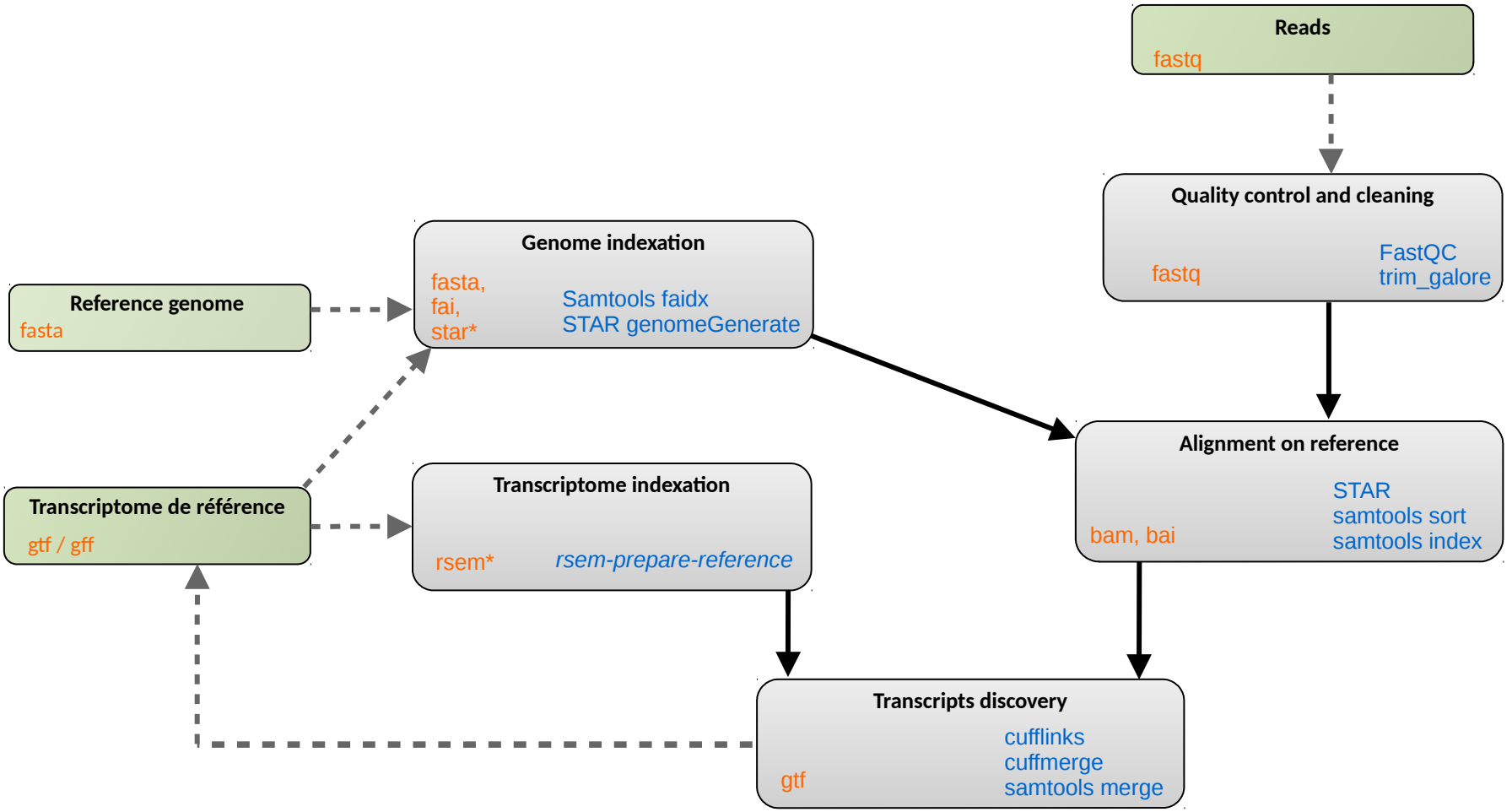
Find library orientation

Color alignment by > first-of-pair strand



Exercices 5

Analysis workflow

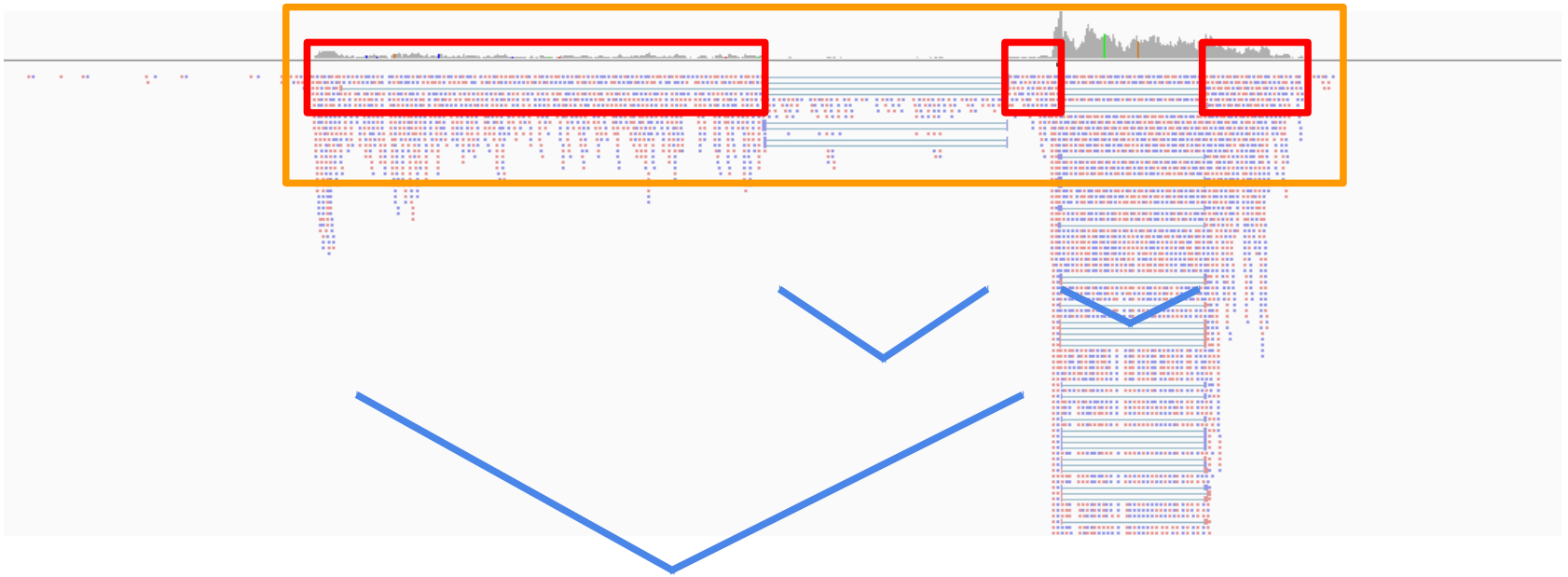


Summary -

mRNA calling & model comparison

- How to reconstruct transcript ?
- Cufflinks
- Compare models (cuffcompare)
- Merge annotation (cuffmerge)
- Which strategy ?

Transcript reconstruction



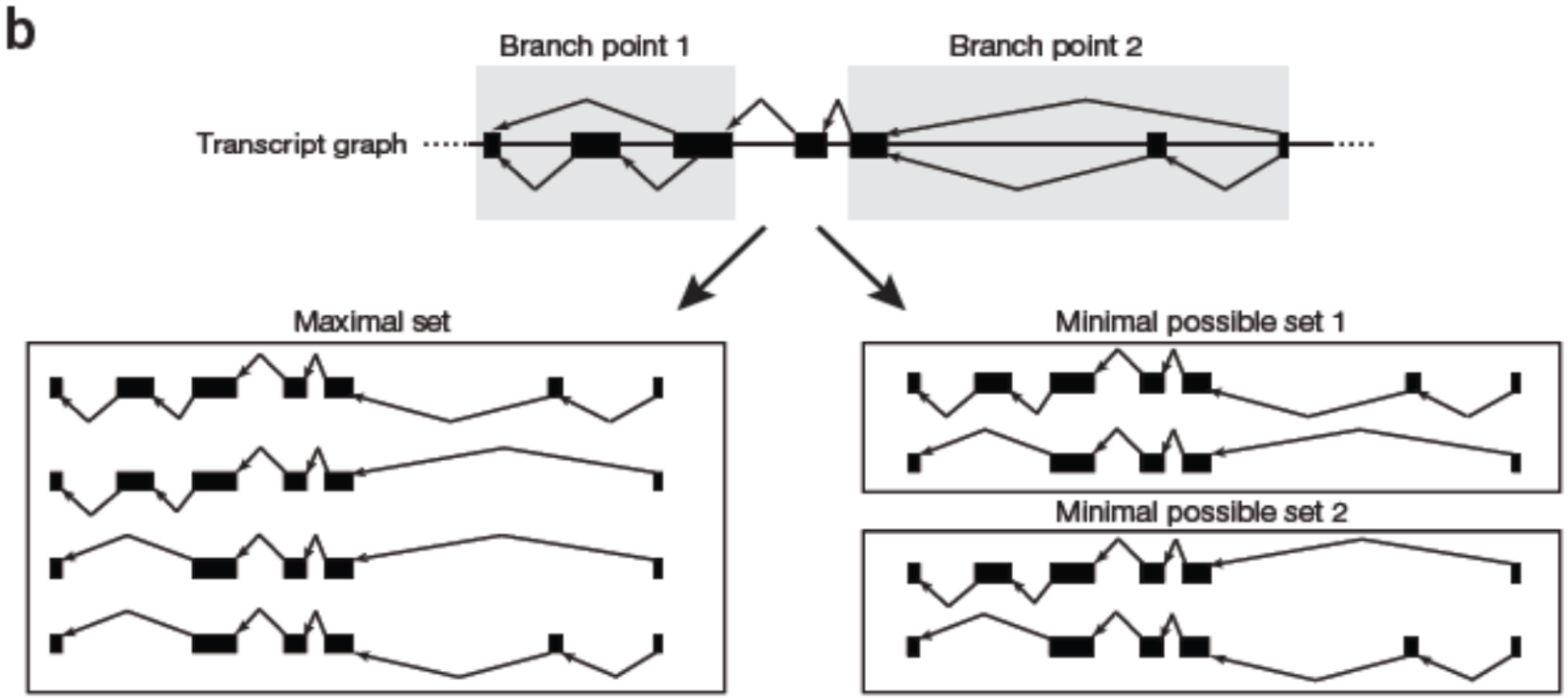
Gene location

Exon location

Junctions :

- between read pair junction
- within read junction

Model building strategies

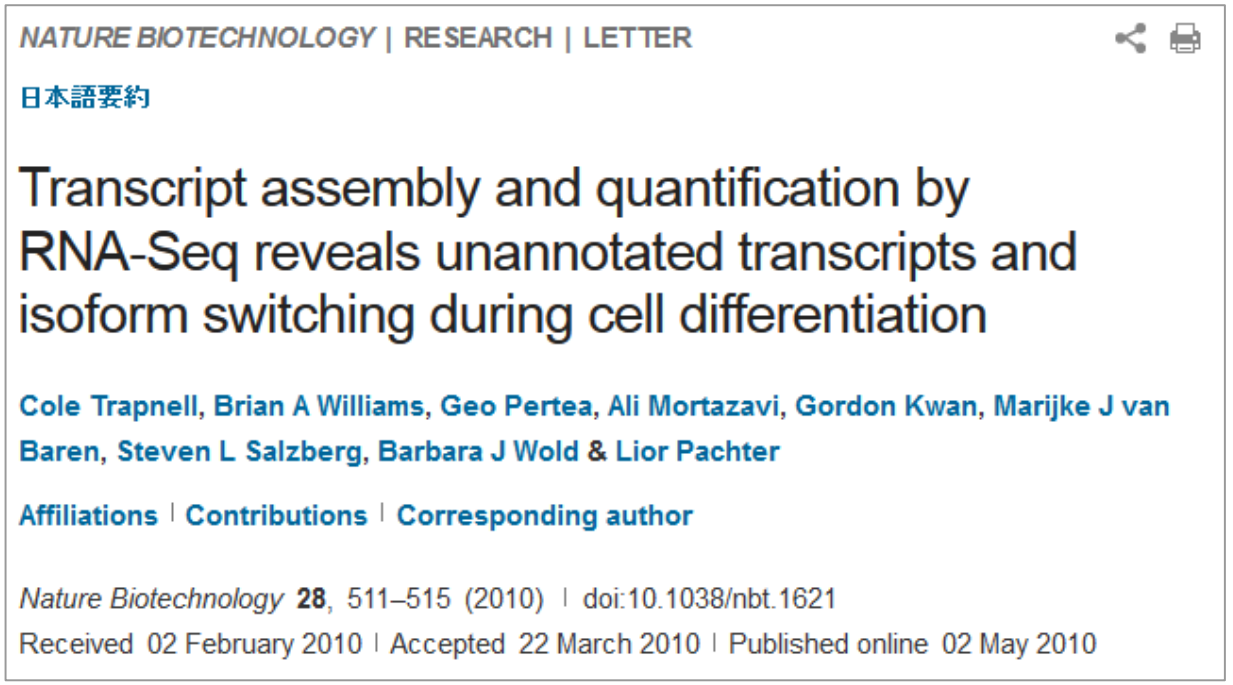


REVIEW

Computational methods for transcriptome annotation and quantification using RNA-seq

Manuel Garber[1], Manfred G Grabherr[1], Mitchell Guttman[1,2] & Cole Trapnell[1,3]

Cufflinks



NATURE BIOTECHNOLOGY | RESEARCH | LETTER

日本語要約

Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation

Cole Trapnell, Brian A Williams, Geo Pertea, Ali Mortazavi, Gordon Kwan, Marijke J van Baren, Steven L Salzberg, Barbara J Wold & Lior Pachter

Affiliations | Contributions | Corresponding author

Nature Biotechnology **28**, 511–515 (2010) | doi:10.1038/nbt.1621

Received 02 February 2010 | Accepted 22 March 2010 | Published online 02 May 2010

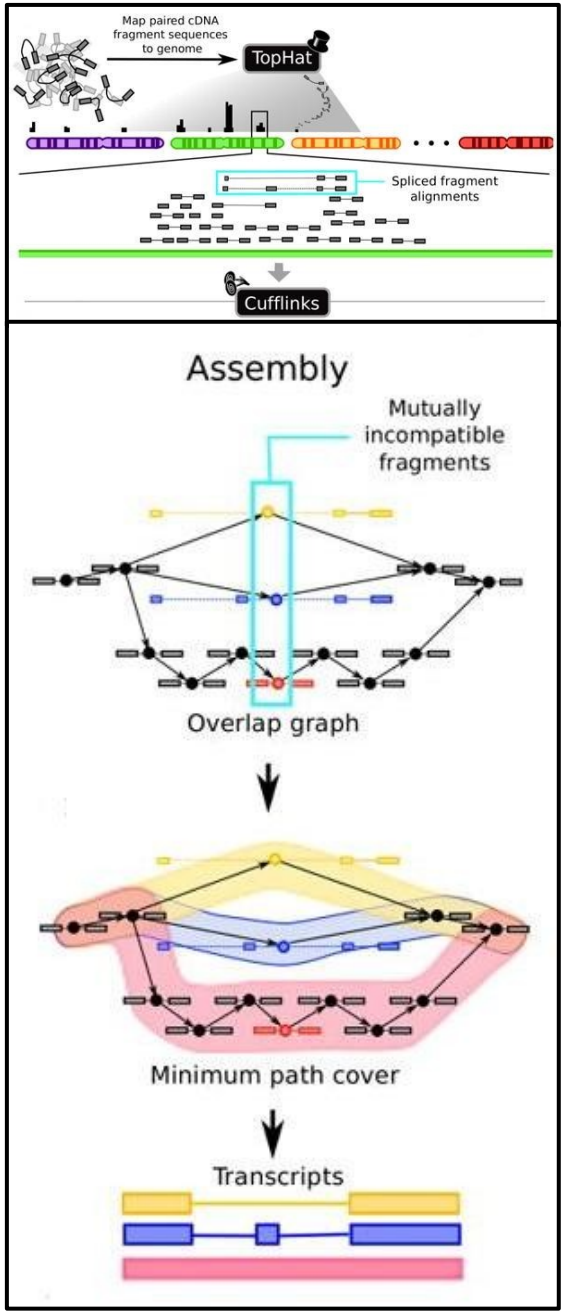
http://cole-trapnell-lab.github.io/cufflinks/

- **assembles transcripts**
- estimates their abundances: based on how many reads support each one
- Suite of software : cufflinks, cuffmerge, cuffcompare

Cufflinks transcript assembly

- Transcripts assembly:
  - fragments are divided into non-overlapping loci
  - each locus is assembled independently

- Cufflinks assembler
  - find the mini nb of transcripts that explain the reads
  - find a minimum path cover (Dilworth's theorem):
    - nb incompatible read = mini nb of transcripts needed
    - each path = set of mutually compatible fragments overlapping each other



Trapnell C et al. Nature Biotechnology 2010

Cufflinks inputs and options

```
module load bioinfo/cufflinks-2.2.1
```

- Command line:

```
cufflinks [options] <aligned_reads.(sam/bam)>
```

- Some options:

  `-h/--help`

  `-o/--output-dir`

  `-p/--num-threads`

  `-G/--GTF <reference_annotation.(gtf/gff)>`
  estimate isoform expression, no novel transcripts

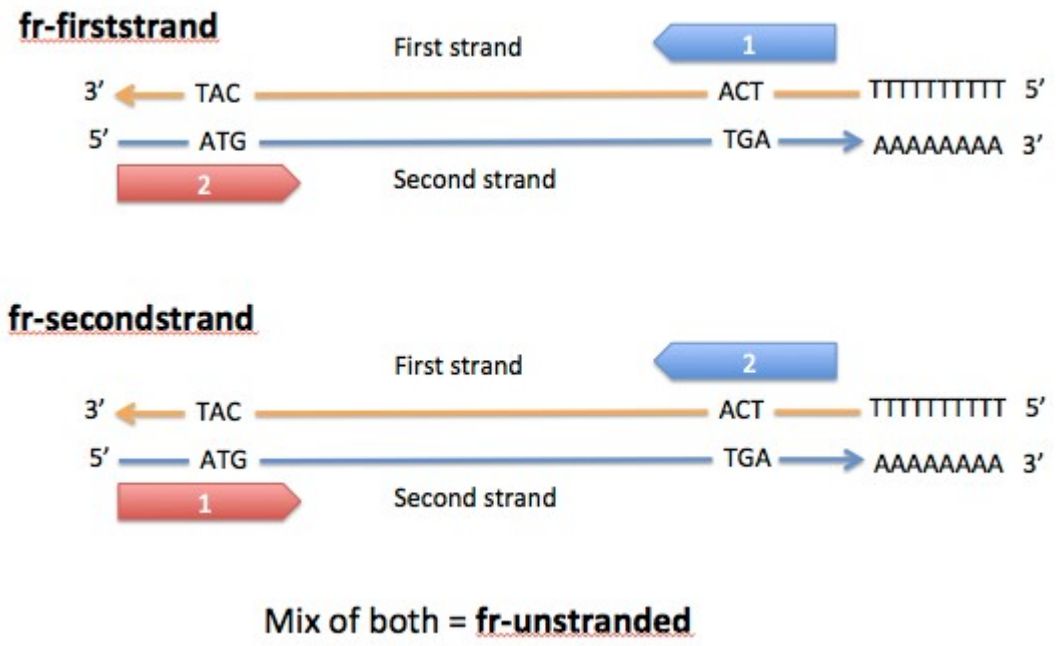
  `-g/--GTF-guide <reference_annotation.(gtf/gff)>`
  use reference transcript annotation to guide assembly

  `--max-bundle-length [3,500,000]`

  `--max-bundle-frags [500,000]`

  `--library-type`
  library prep used for input reads

Cufflinks library types



Mix of both = **fr-unstranded**

https://www.biostars.org/p/344264/

Cufflinks outputs

- `transcripts.gtf`
  contains assembled isoforms (coordinates and abundances)

- `genes.fpkm_tracking`
  contains the genes FPKM

- `isoforms.fpkm_tracking`
  contains the isoforms FPKM

- `skipped.gtf`
  contains skipped loci (too many fragments)

Cufflinks GTF description

`transcripts.gtf` (coordinates and abundances):

- contains assembled isoforms
- can be visualized with a genome viewer
- attributes: ids, FPKM, confidence interval, read coverage & support
  - score: most abundant isoform = 1000
    minor isoforms = minor FPKM/major FPKM
  - cov: estimate for depth across the transcript

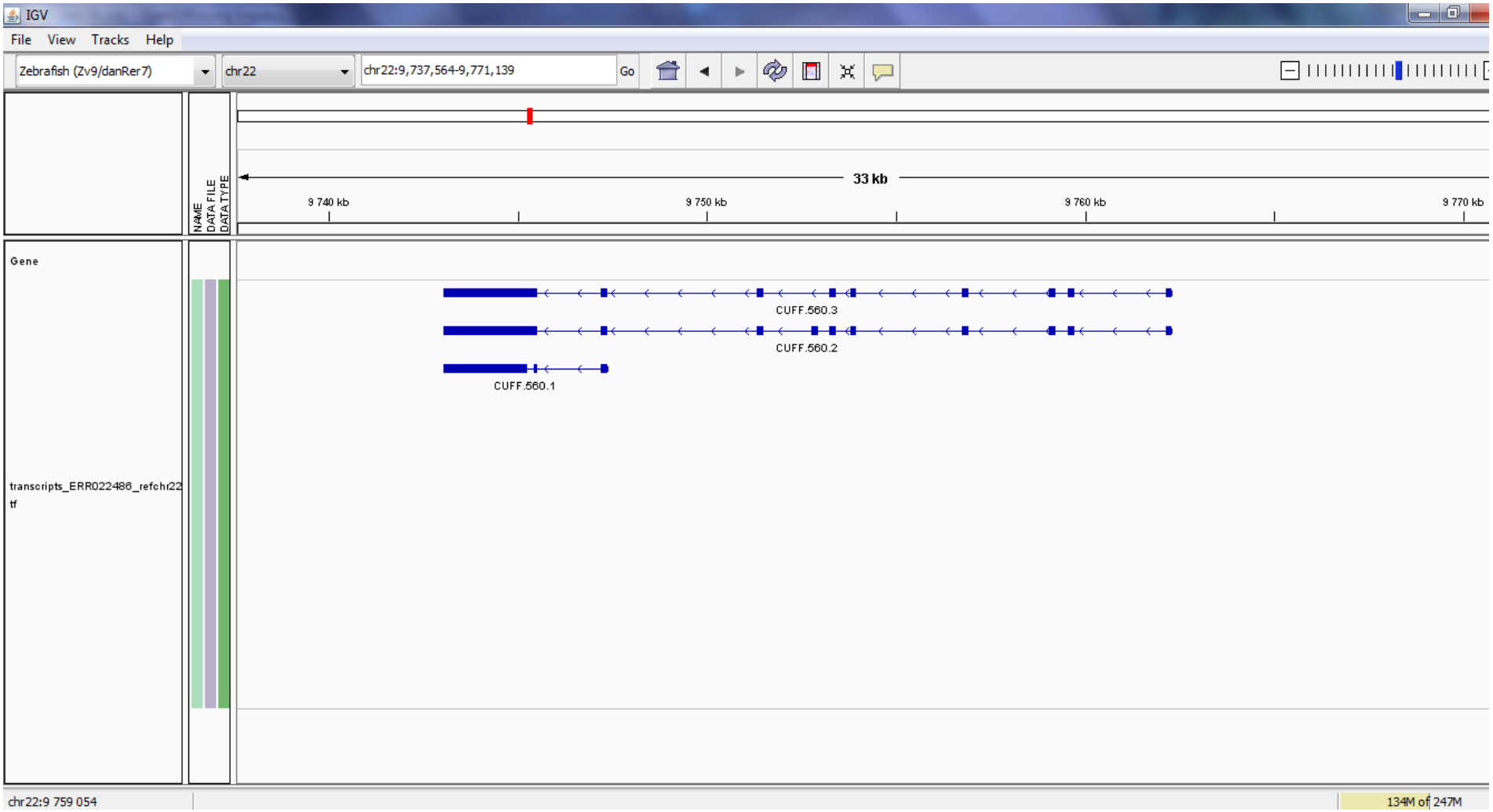
```
1   Cufflinks   transcript  459812  460830  1       -   .
1   Cufflinks   exon        459812  460830  1       -   .
1   Cufflinks   transcript  463572  478996  1000    -   .
1   Cufflinks   exon        463572  463746  1000    -   .
1   Cufflinks   exon        466228  466405  1000    -   .
```

```
gene_id "ENSBTAG00000013841"; transcript_id "ENSBTAT00000018387"; FPKM "0.0000000000"; frac "0.000000";
gene_id "ENSBTAG00000013841"; transcript_id "ENSBTAT00000018387"; exon_number "1"; FPKM "0.0000000000"; frac "0.000000";
gene_id "CUFF.2"; transcript_id "ENSBTAT00000015319"; FPKM "25.4745974237"; frac "1.000000";
gene_id "CUFF.2"; transcript_id "ENSBTAT00000015319"; exon_number "1"; FPKM "25.4745974237"; frac "1.000000";
gene_id "CUFF.2"; transcript_id "ENSBTAT00000015319"; exon_number "2"; FPKM "25.4745974237"; frac "1.000000";
```

```
conf_lo "0.000000"; conf_hi "0.000000"; cov "0.000000"; full_read_support "no";
conf_lo "0.000000"; conf_hi "0.000000"; cov "0.000000";
conf_lo "21.387219"; conf_hi "29.561976"; cov "422.904985"; full_read_support "yes";
conf_lo "21.387219"; conf_hi "29.561976"; cov "422.904985";
conf_lo "21.387219"; conf_hi "29.561976"; cov "422.904985";
```

Cufflinks GTF description

`transcripts.gtf` (coordinates and abundances): visualization in IGV



Cufflinks / Cuffcompare

Compare assemblies between conditions:

- compare your assembled transcripts to a reference annotation
- track Cufflinks transcripts across multiple experiments

Command:

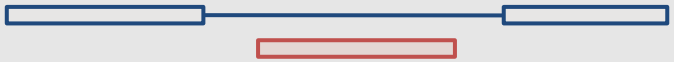
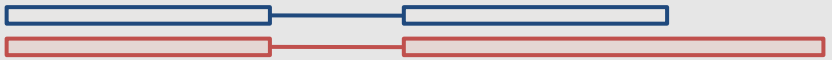
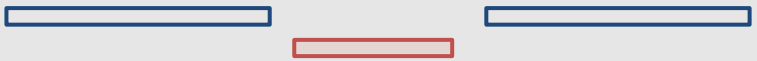
```
cuffcompare [-r <reference.gtf>] [-o <outprefix>] <input1.gtf>
…
```

Outputs:

- `<outprefix>.stats` - overall summary statistics
- `<outprefix>.combined.gtf` - “union” of all transfrags
- `<cuff_in>.refmap` - transfrags matching to reference transcript
- **`<cuff_in>.tmap`** - best reference transcript for each transfrag
- `<outprefix>.tracking` - tracking transfrags across samples

Cuffcompare

Class code de cuffcompare

=	complete match	
c	contained	
j	novel isoform	
e	single exon	
i	within intron	
o	exonic overlap	
p	polymerase run-on	
r	repeat	 > 50% lower case bases
u	unknown, intergenic	
x	exonic overlap on the opposite strand	
s	intronic overlap on the opposite strand	

http://cole-trapnell-lab.github.io/cufflinks/cuffcompare/index.html#transfrag-class-codes

Cufflinks / Cuffmerge

Merge together several assemblies:

- merge novel isoforms and known isoforms
- filters a number of transfrags that are probably artifacts
- build a new gene model describing all conditions

Command:

```
cuffmerge [options] -o <assembly_GTF_list>
```

Options:

- `-o/--output-dir`
- `-g/--ref-gtf`
- `-s/--ref-sequence`
- `--min-isoform-fraction`
  discard isoforms with abundance below this [0.05]
- `-p/--num-threads`

Cufflinks / Cuffmerge

`merged.gtf` (coordinates and legacy):

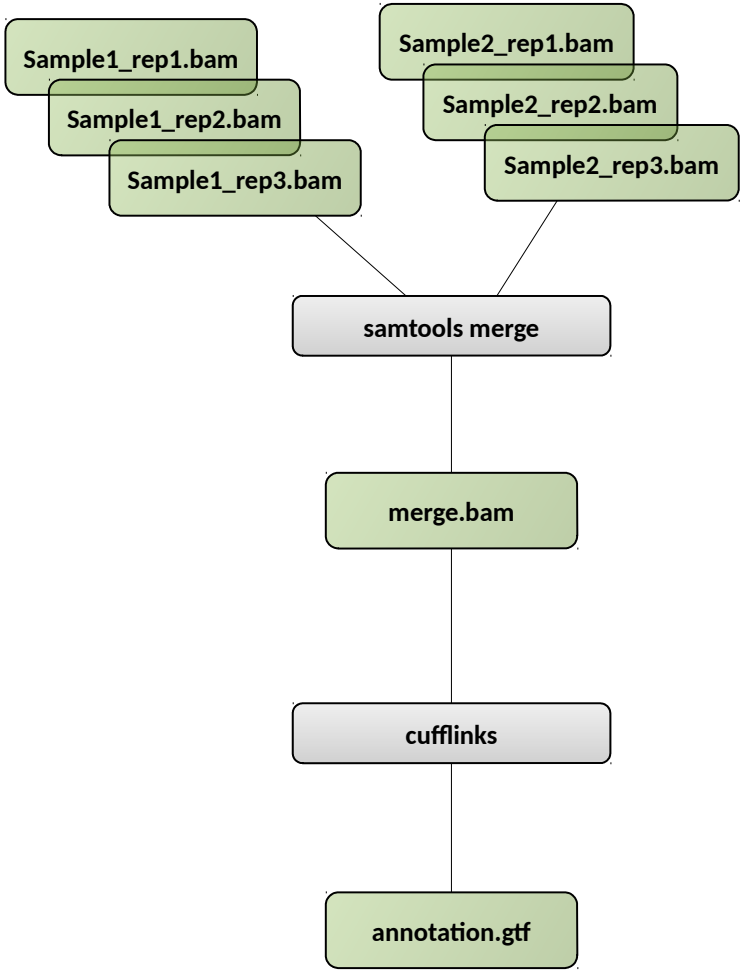
- contains merged input assemblies
- can be visualized with a genome viewer
- attributes: ids, name, oId, nearest_ref, class_code, tss_id, p_id

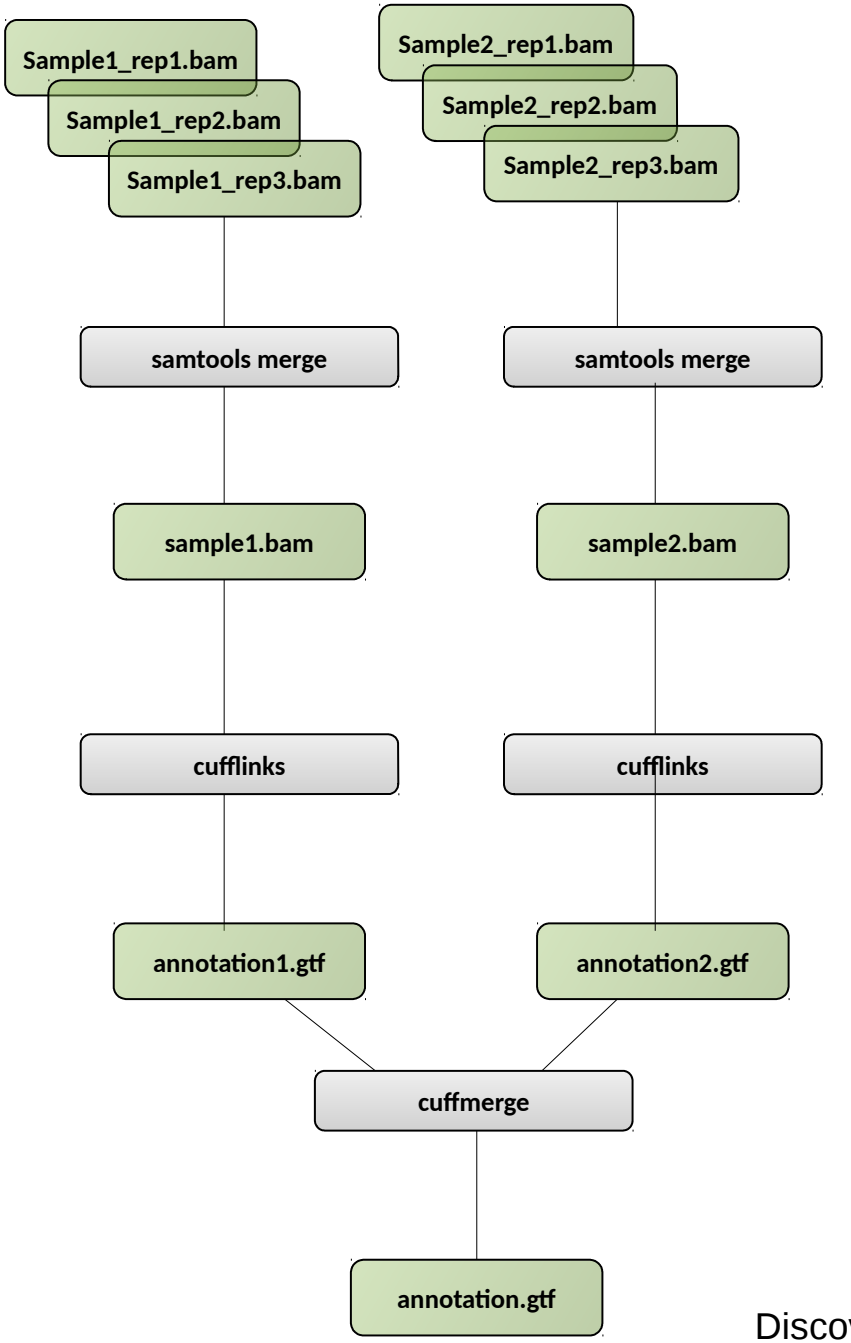
```
1   Cufflinks   exon    34627   35558   .   +   .
1   Cufflinks   exon    242394  242646  .   +   .
1   Cufflinks   exon    275623  275681  .   +   .
1   Cufflinks   exon    242402  242646  .   +   .
1   Cufflinks   exon    254559  254693  .   +   .
1   Cufflinks   exon    247340  249673  .   +   .
1   Cufflinks   exon    351546  351874  .   +   .
1   Cufflinks   exon    355064  355237  .   +   .
1   Cufflinks   exon    357793  357952  .   +   .
1   Cufflinks   exon    361144  362915  .   +   .
```

```
gene_id "XLOC_000001"; transcript_id "TCONS_00000001"; exon_number "1"; gene_name "ENSBTAG00000006858";
gene_id "XLOC_000002"; transcript_id "TCONS_00000002"; exon_number "1"; gene_name "CBX3";
gene_id "XLOC_000002"; transcript_id "TCONS_00000002"; exon_number "2"; gene_name "CBX3";
gene_id "XLOC_000002"; transcript_id "TCONS_00000003"; exon_number "1";
gene_id "XLOC_000002"; transcript_id "TCONS_00000003"; exon_number "2";
gene_id "XLOC_000003"; transcript_id "TCONS_00000004"; exon_number "1";
gene_id "XLOC_000004"; transcript_id "TCONS_00000005"; exon_number "1"; gene_name "RCAN1";
gene_id "XLOC_000004"; transcript_id "TCONS_00000005"; exon_number "2"; gene_name "RCAN1";
gene_id "XLOC_000004"; transcript_id "TCONS_00000005"; exon_number "3"; gene_name "RCAN1";
gene_id "XLOC_000004"; transcript_id "TCONS_00000005"; exon_number "4"; gene_name "RCAN1";
```

```
oId "ENSBTAT00000009004"; nearest_ref "ENSBTAT00000009004"; class_code "="; tss_id "TSS1";
oId "CUFF.1.1"; nearest_ref "ENSBTAT00000007283"; class_code "x"; tss_id "TSS2";
oId "CUFF.1.1"; nearest_ref "ENSBTAT00000007283"; class_code "x"; tss_id "TSS2";
oId "CUFF.1.2"; class_code "u"; tss_id "TSS2";
oId "CUFF.1.2"; class_code "u"; tss_id "TSS2";
oId "CUFF.2.1"; class_code "u"; tss_id "TSS3";
oId "CUFF.3.1"; nearest_ref "ENSBTAT00000037243"; class_code "j"; tss_id "TSS4";
oId "CUFF.3.1"; nearest_ref "ENSBTAT00000037243"; class_code "j"; tss_id "TSS4";
oId "CUFF.3.1"; nearest_ref "ENSBTAT00000037243"; class_code "j"; tss_id "TSS4";
oId "CUFF.3.1"; nearest_ref "ENSBTAT00000037243"; class_code "j"; tss_id "TSS4";
```

Which strategy ?





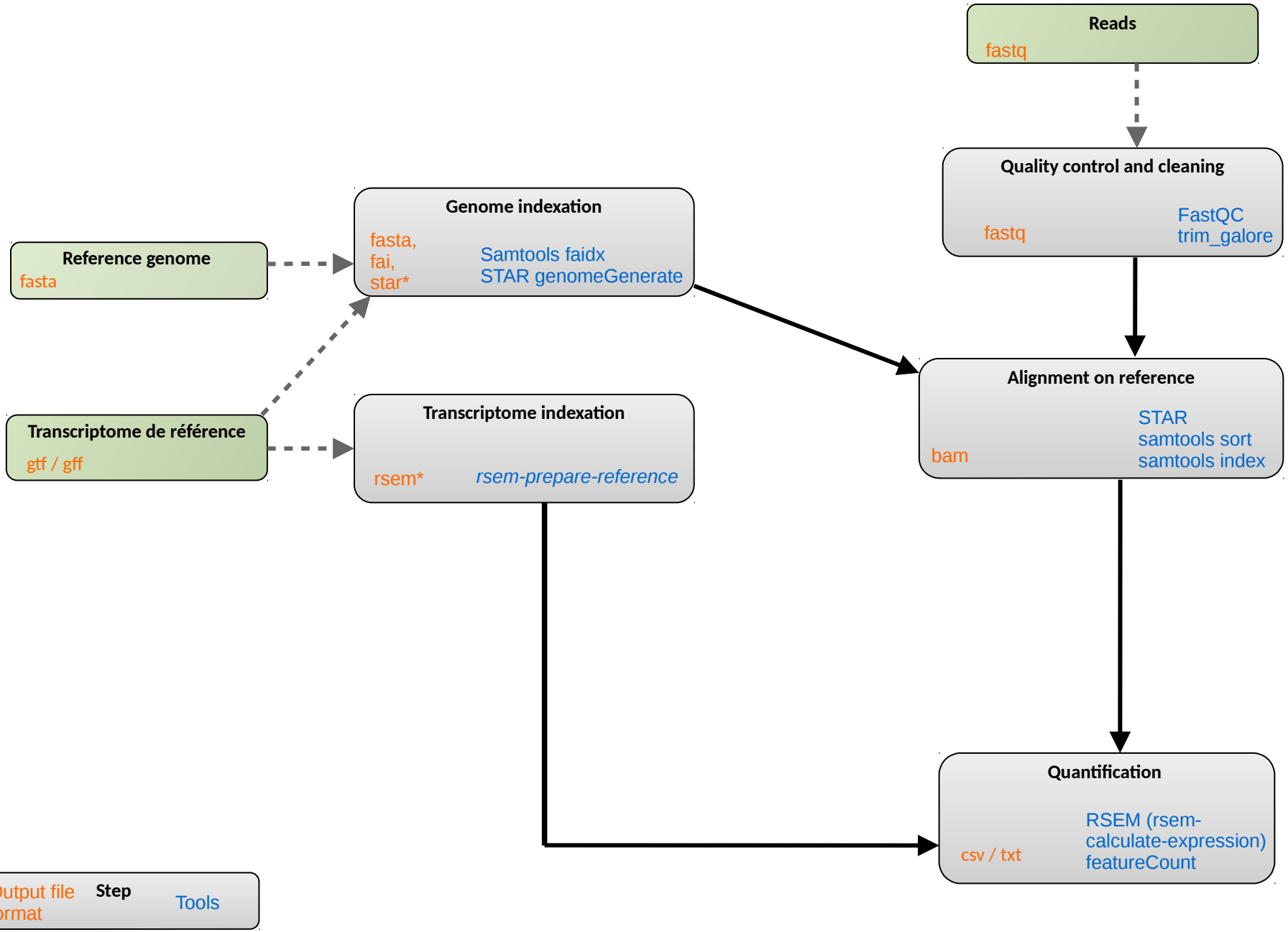
Hands-on: transcripts assembly

Using cufflinks:

Exercise 6:

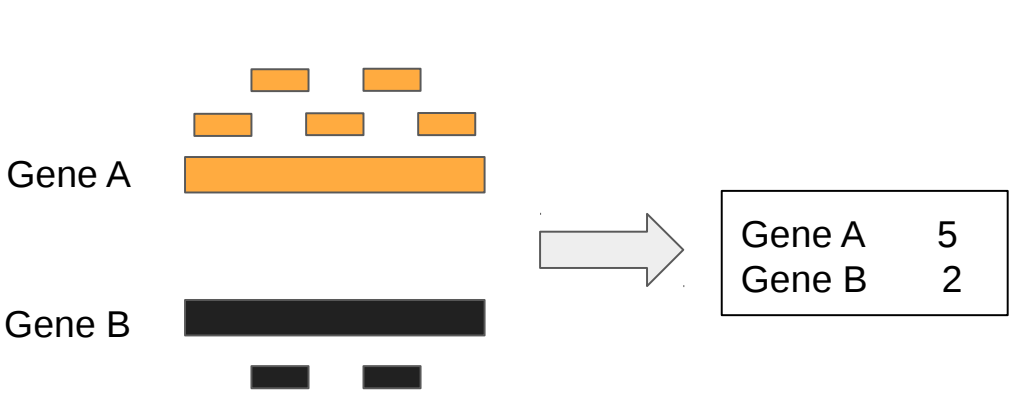
- reconstruct known and novel transcripts
- compare annotations

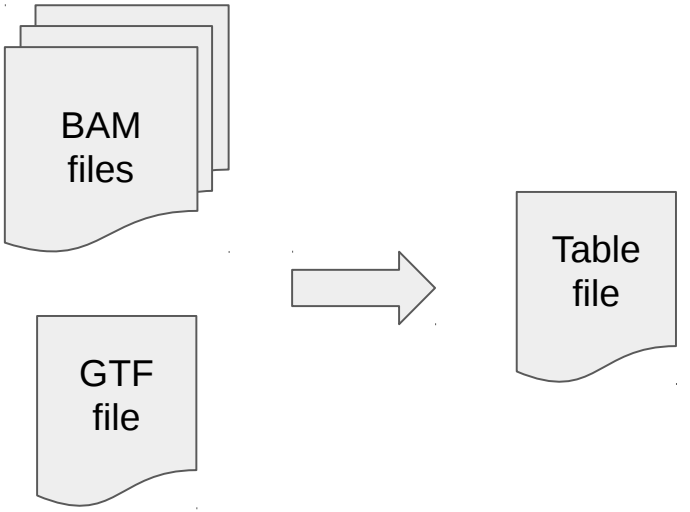
Analysis workflow



Quantification

Quantification: estimation of expression based on a read count.

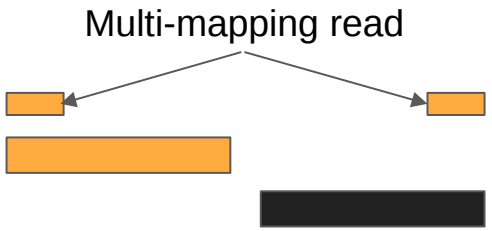


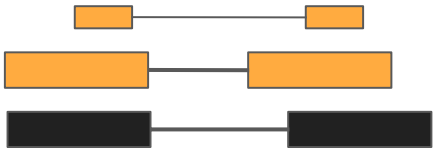


Estimation of:

- gene expression
- transcript expression
- exon expression

Difficult cases

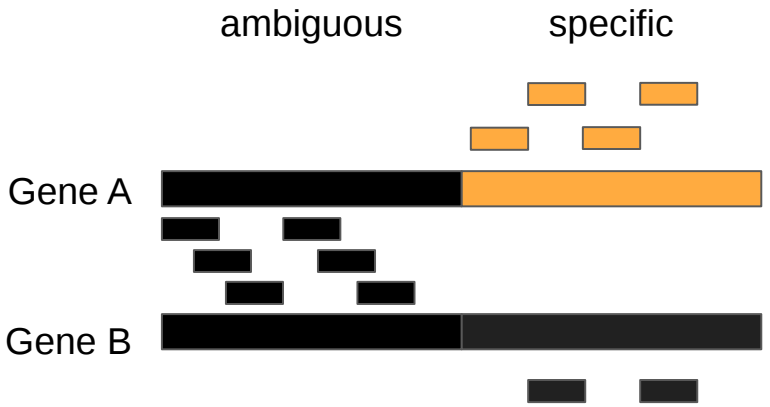




Every quantification tools uses its own rules!

Raw counts *vs* estimation

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



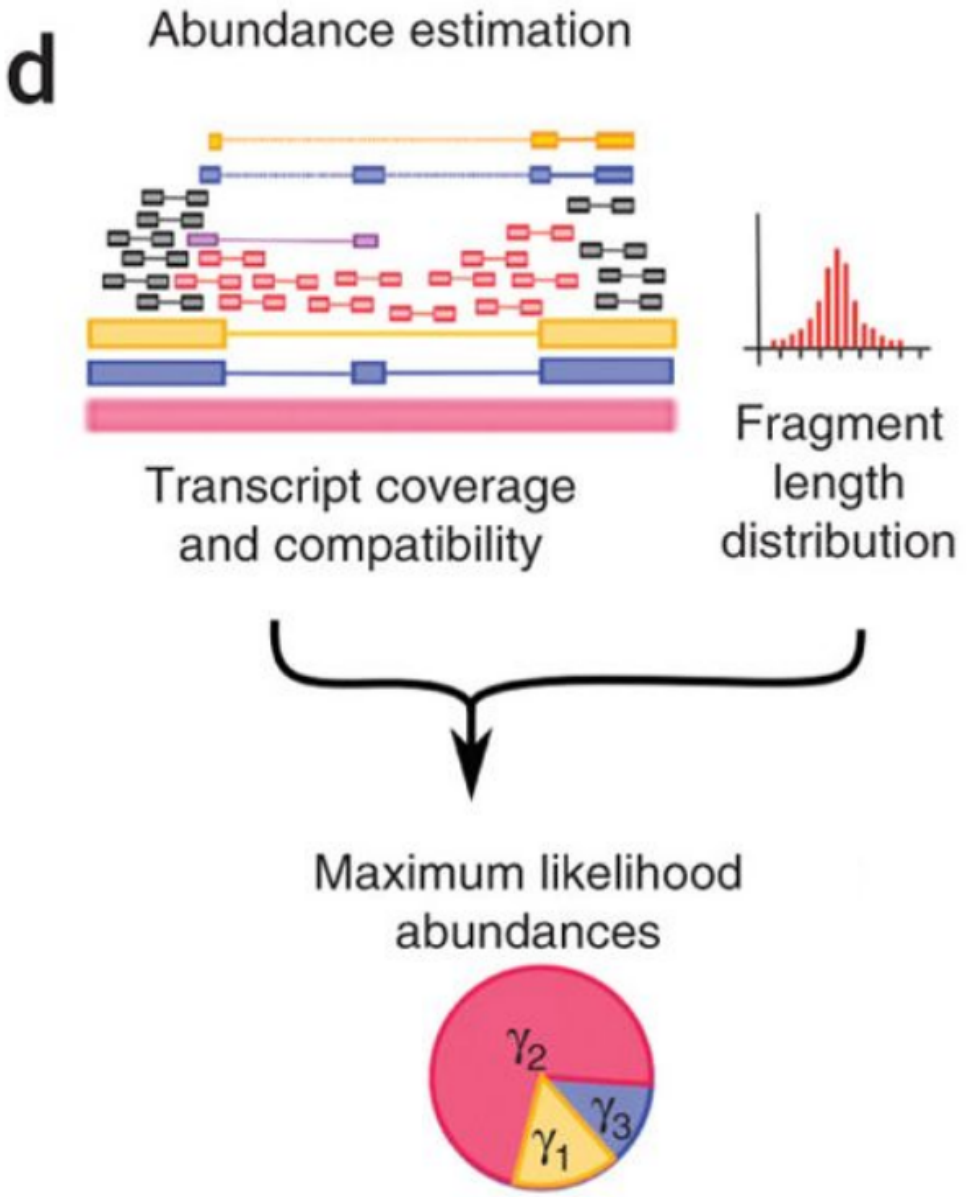
Pros estimation:

- 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,*]

[1]Bioinformatics Division, The Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Parkville, VIC 3052, [2]Department of Computing and Information Systems and [3]Department 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
  - Assignation 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 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

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)

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

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 = FPKM fragments of a 1kb-gene.

RPKM *vs* FPKM *vs* TPM

TMP:

- r_i = # reads on a gene *i*
- *si = size of the gene i*
- *cpbi = ri / si*
- *cpb = ∑ cpbi*
- *TMPi = cpbi / cpb × 106*

Remark:

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

Meaning:

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

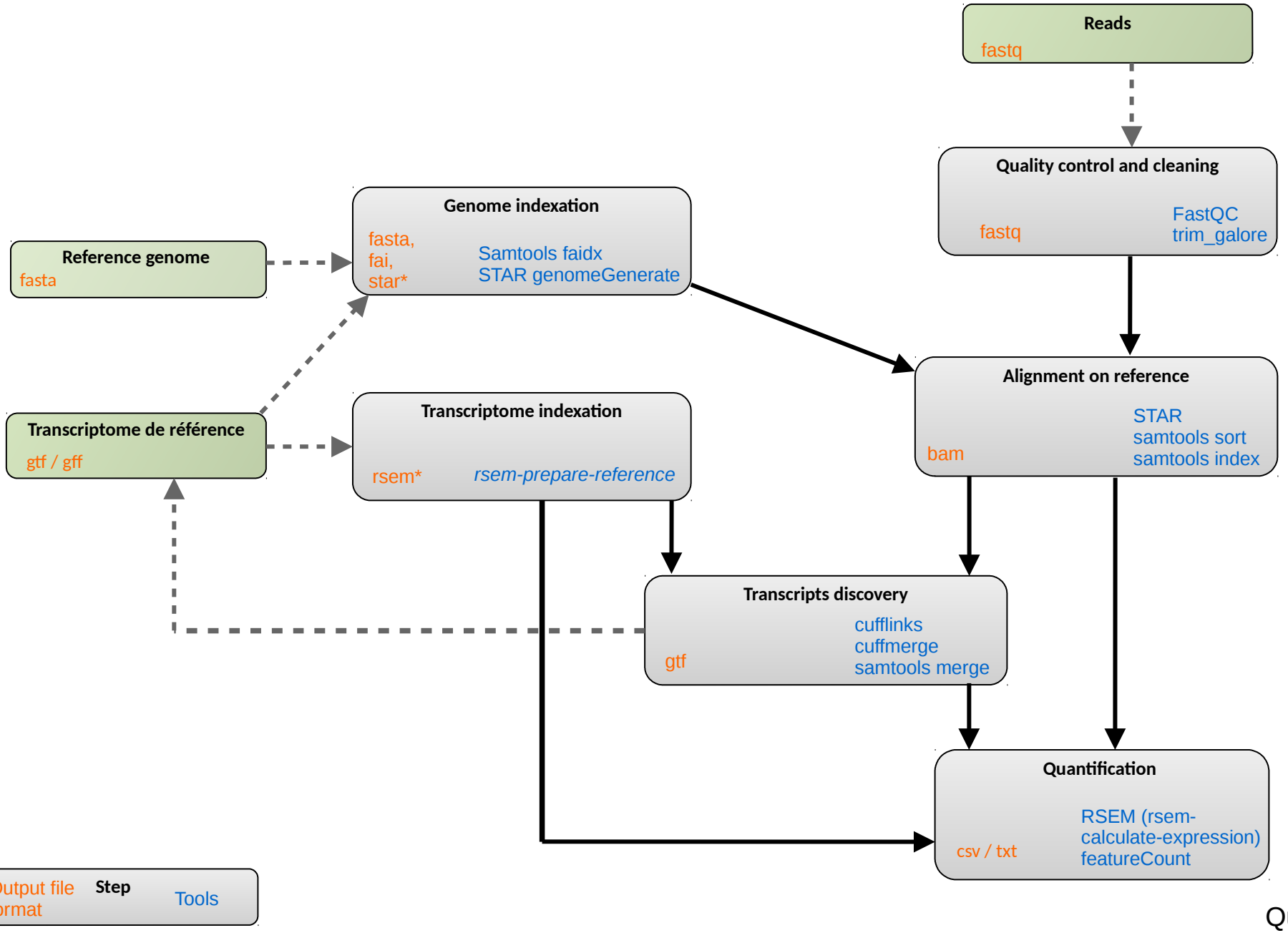
RPKM *vs* FPKM *vs* TPM

- These are refinement of library size normalization, with gene length effect.
- RPKM should not be used for PE reads.
- TMP 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.

Exercice 6

RNAseq pipeline : all steps



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