High-throughput sequencing: Alignment and related topic

Simon Anders
EMBL Heidelberg

HTS Platforms

- Established platforms
  - Illumina HiSeq, ABI SOLiD, Roche 454
- Newcomers: Benchtop machines
  - 454 GS Junior, Illumina MiSeq, IonTorrent PGM

Applications of HTS

- Sequencing of (genomic) DNA
  - de-novo sequencing
  - resequencing (variant finding)
  - enrichment sequencing (ChIP-Seq, MeDIP-Seq, ...)
  - targeted sequencing (exome sequencing, ...)
  - CCC-like (4C, HiC)
  - metagenomics
- Sequencing of RNA (actually: cDNA)

Applications of HTS

- Sequencing of (genomic) DNA
- Sequencing of RNA (actually: cDNA)
  - whole transcriptome*: RNA-Seq, Tag-Seq, ...
  - enriched fraction: HITS-CLIP, ...
  - labeled material: DTA, ...

* or: polyadenylated fraction

HTS: Bioinformatics challenges

Solutions specific to HTS are required for

- assembly
- alignment
- statistical tests (counting statistics)
- visualization
- segmentation
- ...

Two types of experiments

- Discovery experiments
  - finding all possible variant
  - getting an inventory of all transcripts
  - finding all binding sites of a transcription factor
- Comparative experiments
  - comparing tumour and normal samples
  - finding expression changes due to a treatment
  - finding changes in binding affinity

Assembly and Alignment

- First step in most analyses is the *alignment* of reads to a genome
- Except the point is to get the genome: *de-novo assembly*
- Special cases: Transcriptome assembly, metagenomics

The data funnel: ChIP-Seq, non-comparative

- Images
- Base calls
- Alignments
- Enrichment scores
- Location and scores of peaks (or of enriched regions)
- Summary statistics
- Biological conclusions

The data funnel: Comparative RNA-Seq

- Images
- Base calls
- Alignments
- Expression strengths of genes
- Differences between these
- Gene-set enrichment analyses
-

Where does Bioconductor come in?

- Processing of the images and determining of the read sequencest
  - typically done by core facility with software from the manufacturer of the sequencing machine
- Aligning the reads to a reference genome (or assembling the reads into a new genome)
  - Done with community-developed stand-alone tools.
- Downstream statistical analysis.
  - Write your own scripts with the help of Bioconductor infrastructure.

Alignment

Alignment

- Many different aligners:

  Eland, Maq, Bowtie, BWA, SOAP, SSAHA, TopHat, SpliceMap, GSNAP, Novoalign, ...

- Main differences:
  - Publication year, maturity, development after publication, popularity
  - usage of base-call qualities, calculation of mapping qualities
  - Burrows-Wheeler index or not
  - speed-vs-sensitivity trade-off
  - suitability for RNA-Seq (“spliced alignment”)
  - suitability for special tasks (e.g., color-space reads, bisulfite reads, variant injection, local re-alignment, ...)

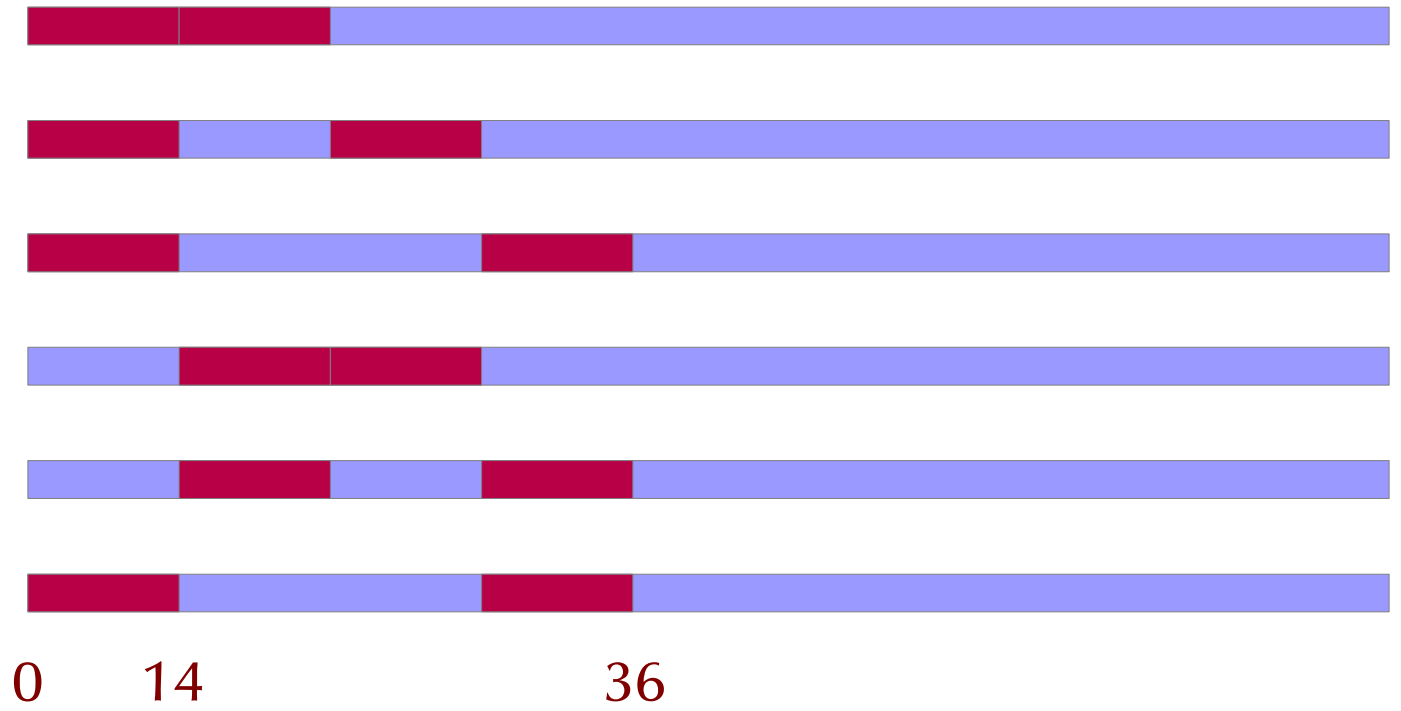
Short-read algorithms: Seed matches

Aligners often claim that they find all alignments with up to 2 mismatches and may find alignments with more than two mismtaches.

How does it work?

Spaced seeds

Maq prepares six hash table, each indexing 28 of the first 36 bases of the reads, selected as follows:



Hence, the aligner finds all alignments with at most 2 mismatches in the first 36 bases.

Alignment: Workflow

- Preparation: Generate an *index* from FASTA file with the genome.

- Input data: FASTQ files with raw reads (demultiplexed)
- Alignment
- Output file: SAM file with alignments

Raw reads: FASTQ format

"FASTA with Qualities"

Example:

```
@HWI-EAS225:3:1:2:854#0/1
GGGGGGAAGTCGGCAAAATAGATCCGTAACTTCGGG
+HWI-EAS225:3:1:2:854#0/1
a`abbbbabaabbbababb^``[aaa`_N]b^ab^``a
@HWI-EAS225:3:1:2:1595#0/1
GGGAAGATCTCAAAAACAGAAGTAAAACATCGAACG
+HWI-EAS225:3:1:2:1595#0/1
a`abbbababbbabbbbbbabb`aaababab\aa_`
```

FASTQ format

Each read is represented by four lines:

- '@', followed by read ID
- sequence
- '+', optionally followed by repeated read ID
- quality string:
  - same length as sequence
  - each character encodes the base-call quality of one base

FASTQ format: quality string

- If p is the probability that the base call is wrong,

  the Phred score is:

$$Q = -10 \log_{10} p$$

- The score is written with the character whose ASCII code is $Q+33$ (Sanger Institute standard).

- Before SolexaPipeline version 1.8, Solexa used instead the character with ASCII code $Q+64$.
- Before SolexaPipeline version 1.3, Solexa also used a different formula, namely $Q = -10 \log_{10} (p/(1-p))$

FASTQ: Phred base-call qualities

quality score Q_{phred}	error prob. p	characters
0 .. 9	1 .. 0.13	!"#$%&'()*
10 .. 19	0.1 .. 0.013	+,-./01234
20 .. 29	0.01 .. 0.0013	56789:;<=>
30 .. 39	0.001 .. 0.00013	?@ABCDEFGH
40	0.0001	I

Quality scales

```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
 .........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.....................
 ............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII.....................
 ..................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ.....................
 LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL.....................................................
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126

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)
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

[Wikipedia article “FASTQ format”]

FASTQ and paired-end reads

Convention for paired-end runs:

The reads are reported two FASTQ files, such that the n^{th} read in the first file is mate-paired to the n^{th} read in the second file. The read IDs must match.

Alignment output: SAM files

A SAM file consists of two parts:

- Header
  - contains meta data (source of the reads, reference genome, aligner, etc.)
  - Most current tools omit and/or ignore the header.
  - All header lines start with “@”.
  - Header fields have standardized two-letter codes for easy parsing of the information
- Alignment section
  - A tab-separated table with at least 11 columns
  - Each line describes one alignment

A SAM file

..]

HWI-EAS225_309MTAAXX:5:1:689:1485 0 XIII 863564 25 36M * 0 0
GAAATATATACGTTTTTATCTATGTTACGTTATATA CCCCCCCCCCCCCCCCCCCCCCCC4CCCB4CA?AAA< NM:i:0
X0:i:1 MD:Z:36

HWI-EAS225_309MTAAXX:5:1:689:1485 16 XIII 863766 25 36M * 0 0
CTACAATTTTGCACATCAAAAAAGACCTCCAACTAC =8A=AA784A9AA5AAAAAAAAAAA=AAAAAAAAAA NM:i:0
X0:i:1 MD:Z:36

HWI-EAS225_309MTAAXX:5:1:394:1171 0 XII 525532 25 36M * 0 0
GTTTACGGCGTTGCAAGAGGCCTACACGGGCTCATT CCCCCCCCCCCCCCCCCCCCCC?CCACCACA7?<??? NM:i:0
X0:i:1 MD:Z:36

HWI-EAS225_309MTAAXX:5:1:394:1171 16 XII 525689 25 36M * 0 0
GCTGTTATTTCTCCACAGTCTGGCAAAAAAAAGAAA 7AAAAAA?AA<AA?AAAAA5AAA<AAAAAAAAAAAA NM:i:0
X0:i:1 MD:Z:36

HWI-EAS225_309MTAAXX:5:1:393:671 0 XV 440012 25 36M * 0 0
TTTGGTGATTTTCCCGTCTTTATAATCTCGGATAAA AAAAAAAAAAAAAAA<AAAAAAAA<AAAA5<AAAA3 NM:i:0
X0:i:1 MD:Z:36

HWI-EAS225_309MTAAXX:5:1:393:671 16 XV 440188 25 36M * 0 0
TCATAGATTCCATATGAGTATAGTTACCCCATAGCC ?9A?A?CC?<ACCCCCCCCCCCCCCCCCCACCCCCCC NM:i:0
X0:i:1 MD:Z:36

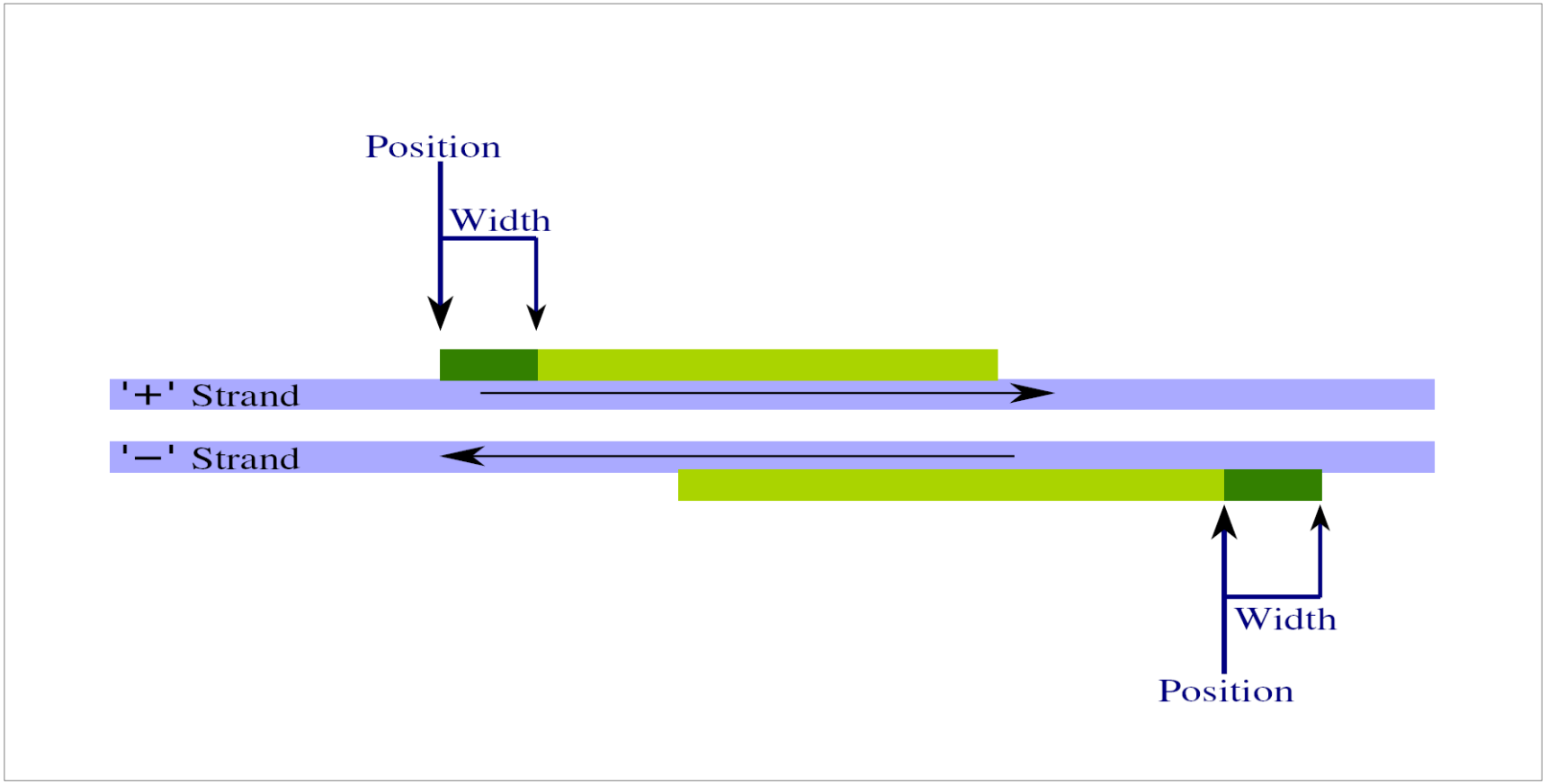
[...]

SAM format: Alignment section

The columns are:

- QNAME: ID of the read (“query”)
- FLAG: alignment flags
- RNAME: ID of the reference (typically: chromosome name)
- POS: Position in reference (1-based, left side)
- MAPQ: Mapping quality (as Phred score)
- CIGAR: Alignment description (gaps etc.) in CIGAR format
- MRNM: Mate reference sequence name [for paired end data]
- MPOS: Mate position [for paired end data]
- ISIZE: inferred insert size [for paired end data]
- SEQ: sequence of the read
- QUAL: quality string of the read
- extra fields

Reads and fragments



SAM format: Flag and extra fields

FLAG field: A number, encoding

- whether the read is from a paired-end run, and if so, which one
- if so, whether the read and/or its mate are mapped
- whether the read mapped to the forward or the reverse strand
- whether the read passed platform quality checks
- [and a few more things]

Extra fields:

- Always triples of the format TAG : VTYPE : VALUE
- may encode number of mismatches (“NM”), number of alignments for the same read, extra informations on quality, aligner-specific data etc.

SAM format: CIGAR strings

Alignments contain gaps (e.g., in case of an indel, or, in RNA-Seq, when a read straddles an intron).

Then, the CIGAR string gives details.

Example: “M10 I4 M4 D3 M12” means

- the first 10 bases of the read map (“M10”) normally (not necessarily perfectly)
- then, 4 bases are inserted (“I4”), i.e., missing in the reference
- then, after another 4 mapped bases (“M4”), 3 bases are deleted (“D4”), i.e., skipped in the query.
- Finally, the last 12 bases match normally.

There are further codes (N, S, H, P), which are rarely used.

SAM format: paired-end and multiple alignments

- Each line represents one *alignments*.
- Multiple alternative alignments for the same read take multiple lines. Only the read ID allows to group them.
- Paired-end alignments take two lines.

- All these reads are not necessarily in adjacent lines.

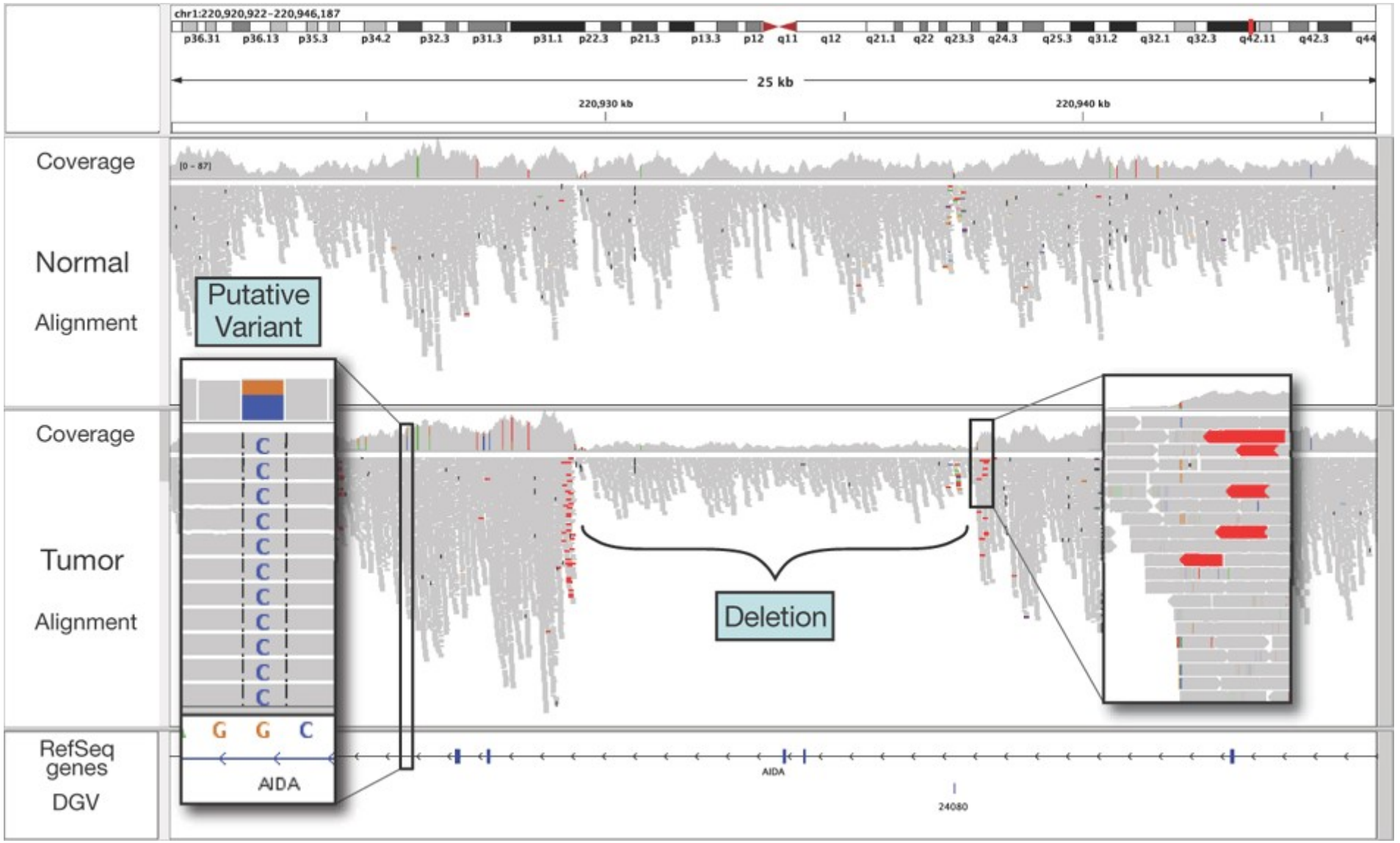
sorted SAM/BAM files

- Text SAM files (.sam): standard form
- BAM files (.bam): binary representation of SAM
  - more compact, faster to process, random access and indexing possible
- BAM index files (.bai) allow random access in a BAM file that is sorted by position.

SAMtools

- The SAMtools are a set of simple tools to
  - convert between SAM and BAM
  - sort and merge SAM files
  - index SAM and FASTA files for fast access
  - calculate tallies (“flagstat”)
  - view alignments (“tview”)
  - produce a “pile-up”, i.e., a file showing
    - local coverage
    - mismatches and consensus calls
    - indels
- The SAMtools C API facilitates the development of new tools for processing SAM files.

Visualization of SAM files

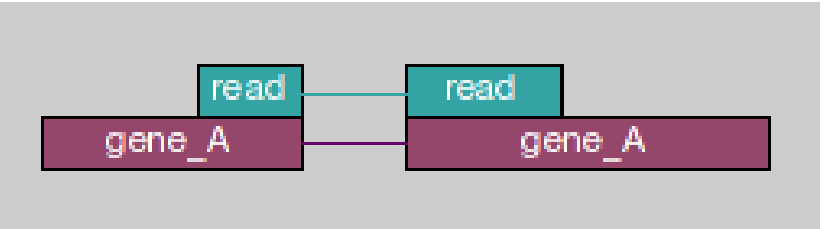


Integrative Genomics Viewer (IGV): Robinson et al., Broad Institute

Special considerations for RNA-Seq

RNA alignment

- Only few aligners (e.g., TopHat, GSNAP, SpliceMap) deal with spliced read.
- Use these for RNA-Seq data.

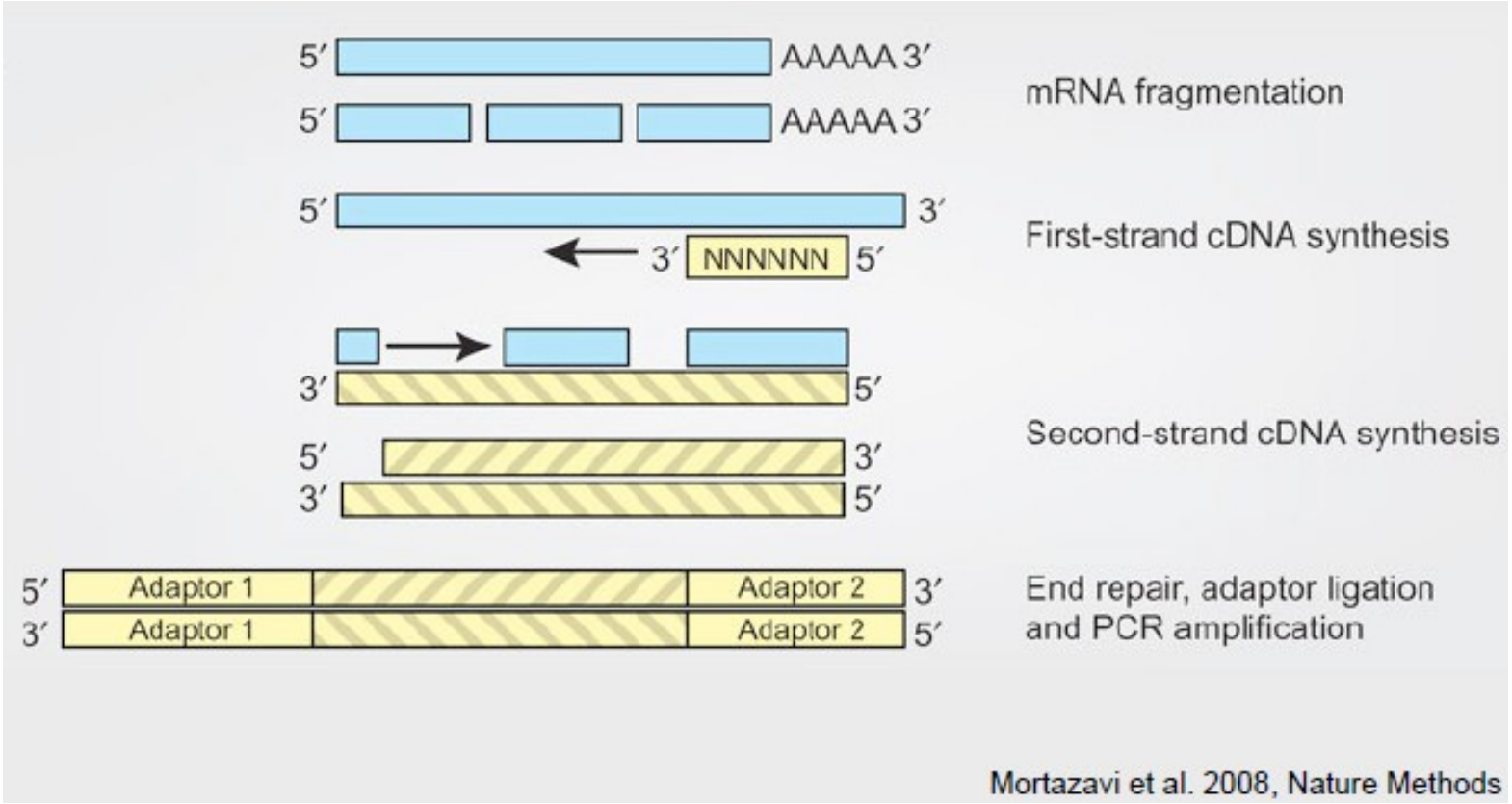


Strand-specific protocols

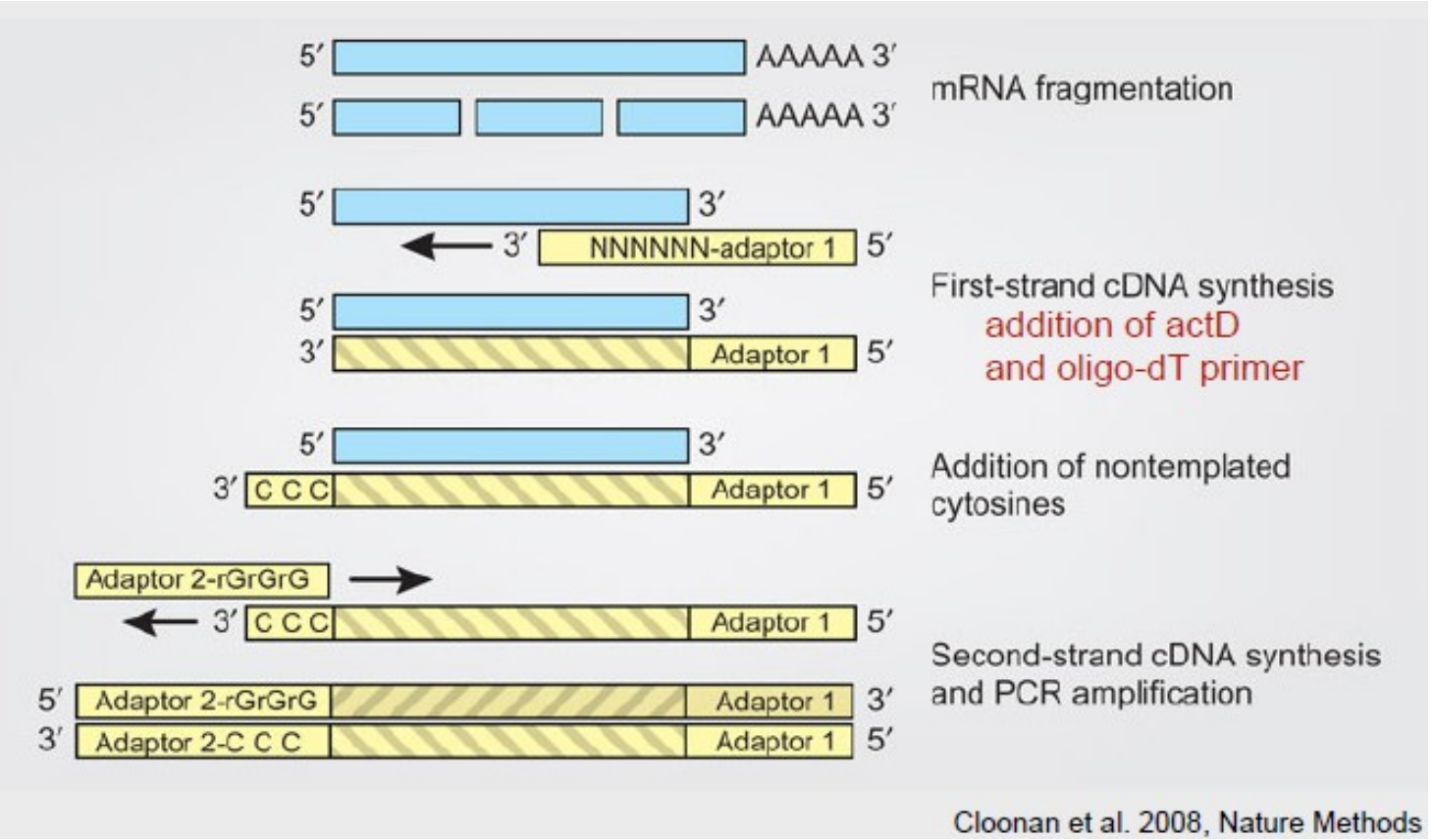
- Standard RNA-Seq loses strand information.
- If you want to distinguish sense from anti-sense transcripts, you need a strand-specific one.

- Make sure you know whether the library you analyse is strand-specific.

Solexa standard protocol for RNA-Seq



Strand-specific RNA-Seq with random hexamer priming

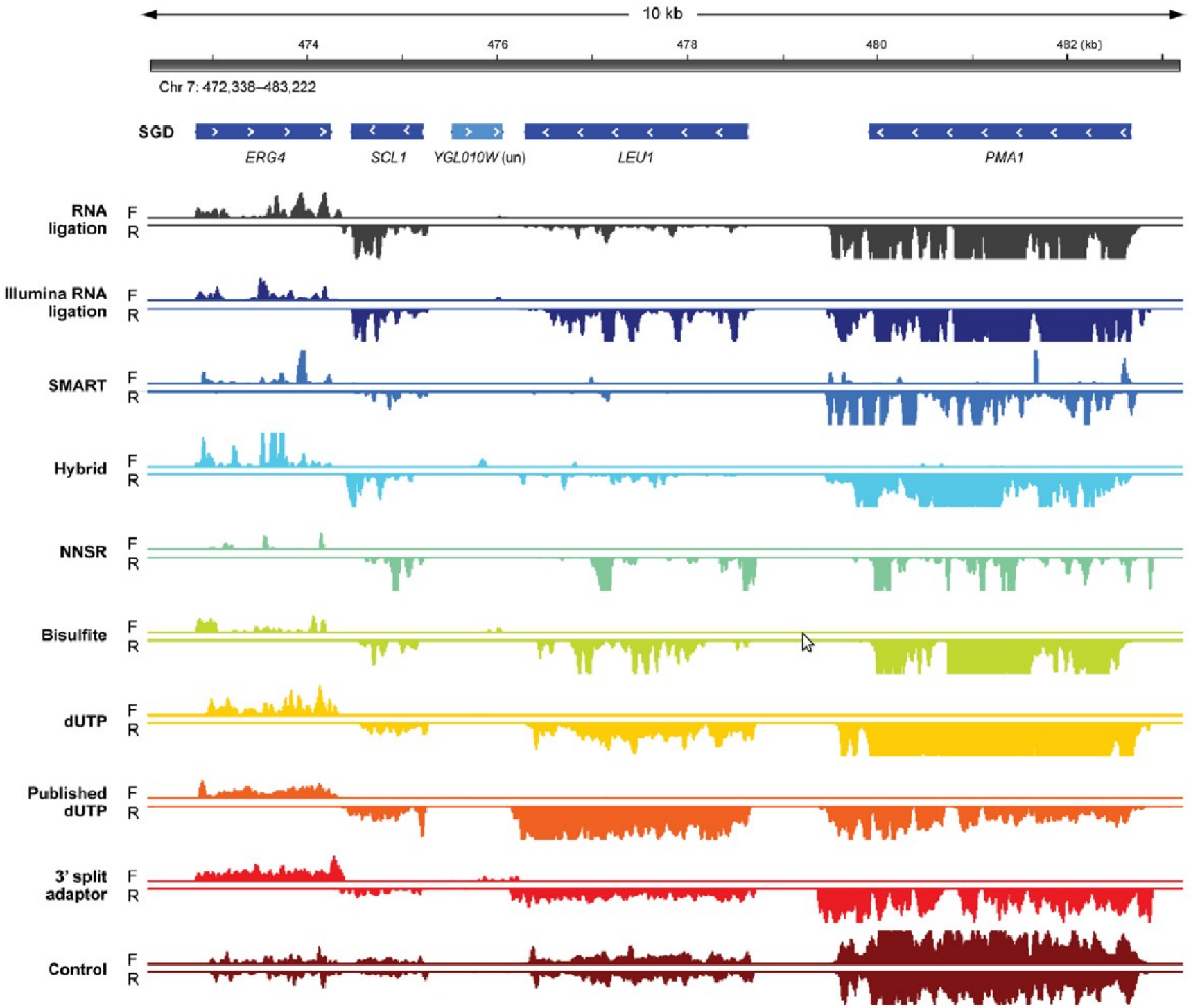


Cloonan et al. 2008, Nature Methods

Coverage in RNA-Seq

- When sequencing genomic DNA, the coverage seems reasonably even.

- In RNA-Seq, this quite different



Levin et al., Nature Methods, 2010

*