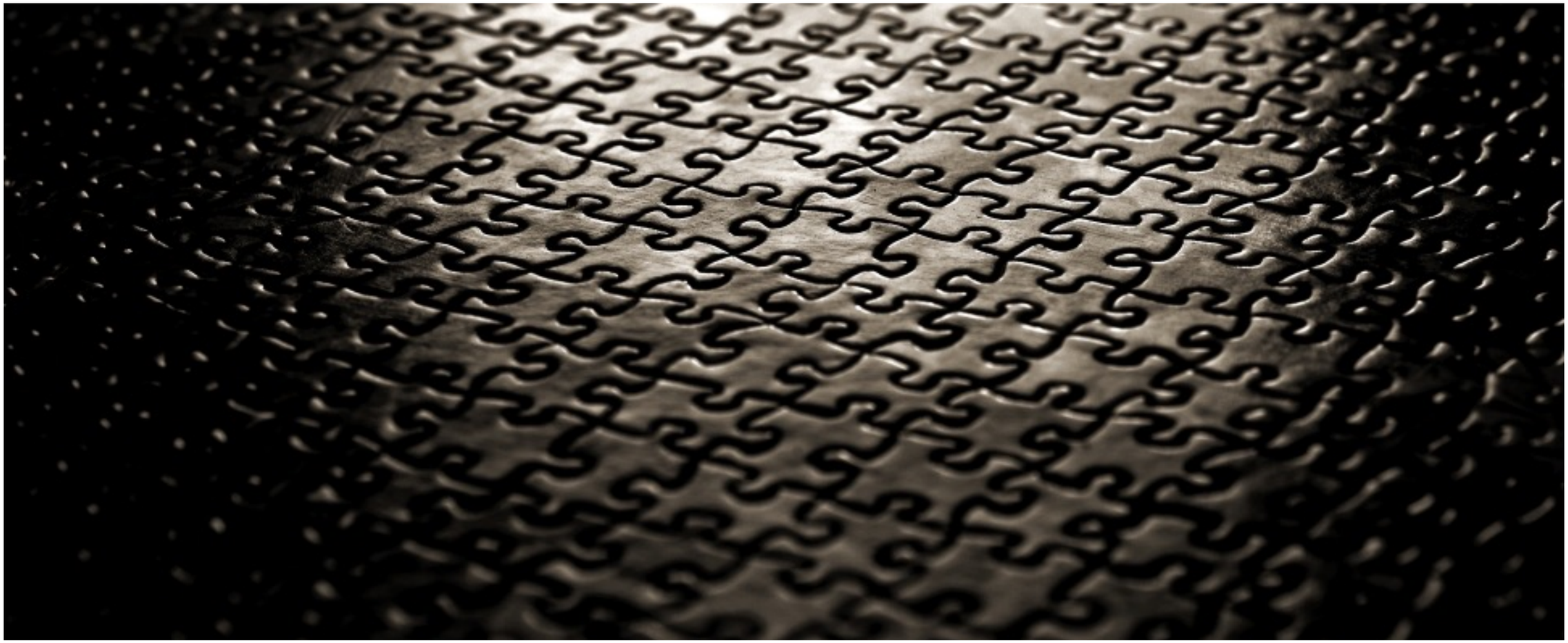
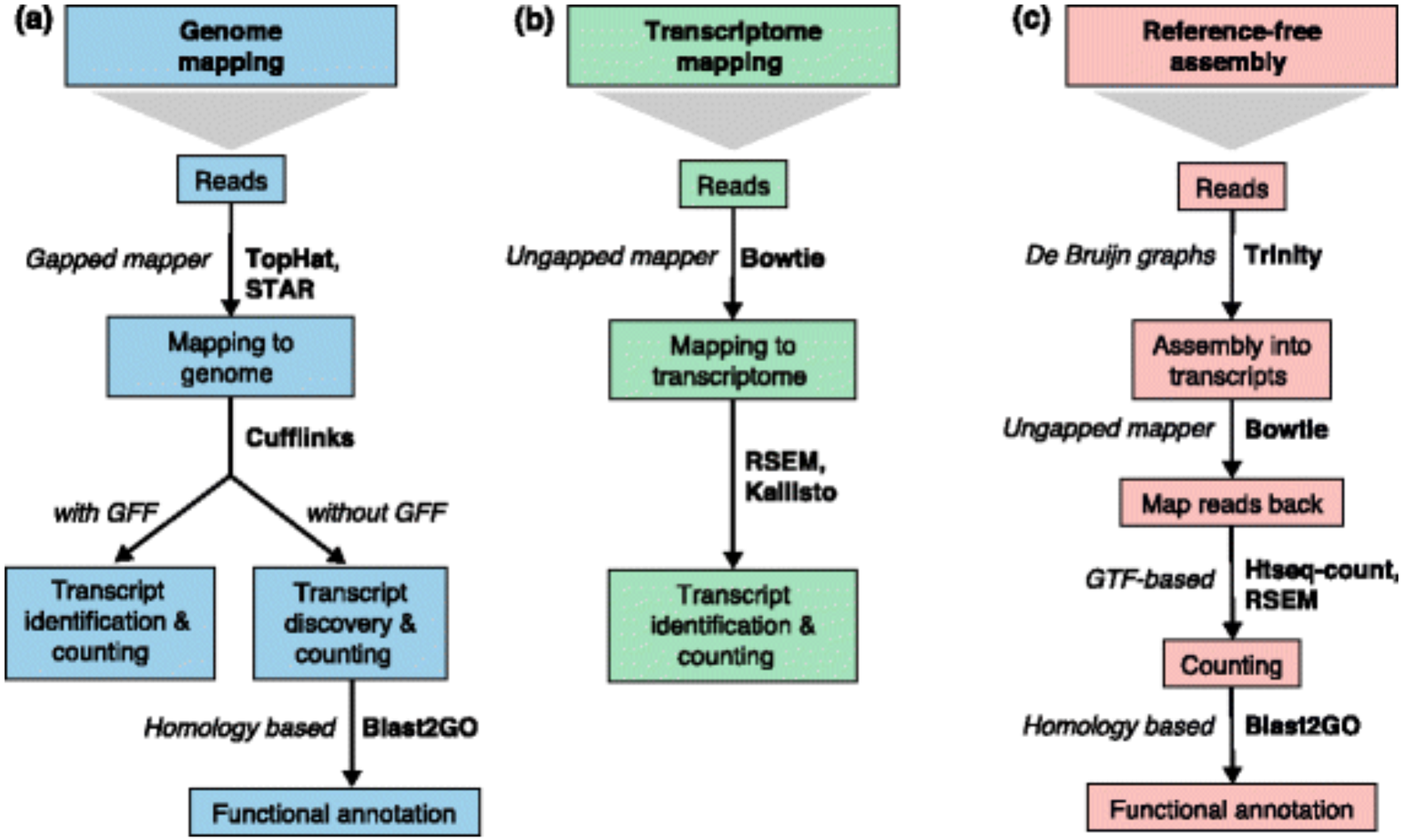
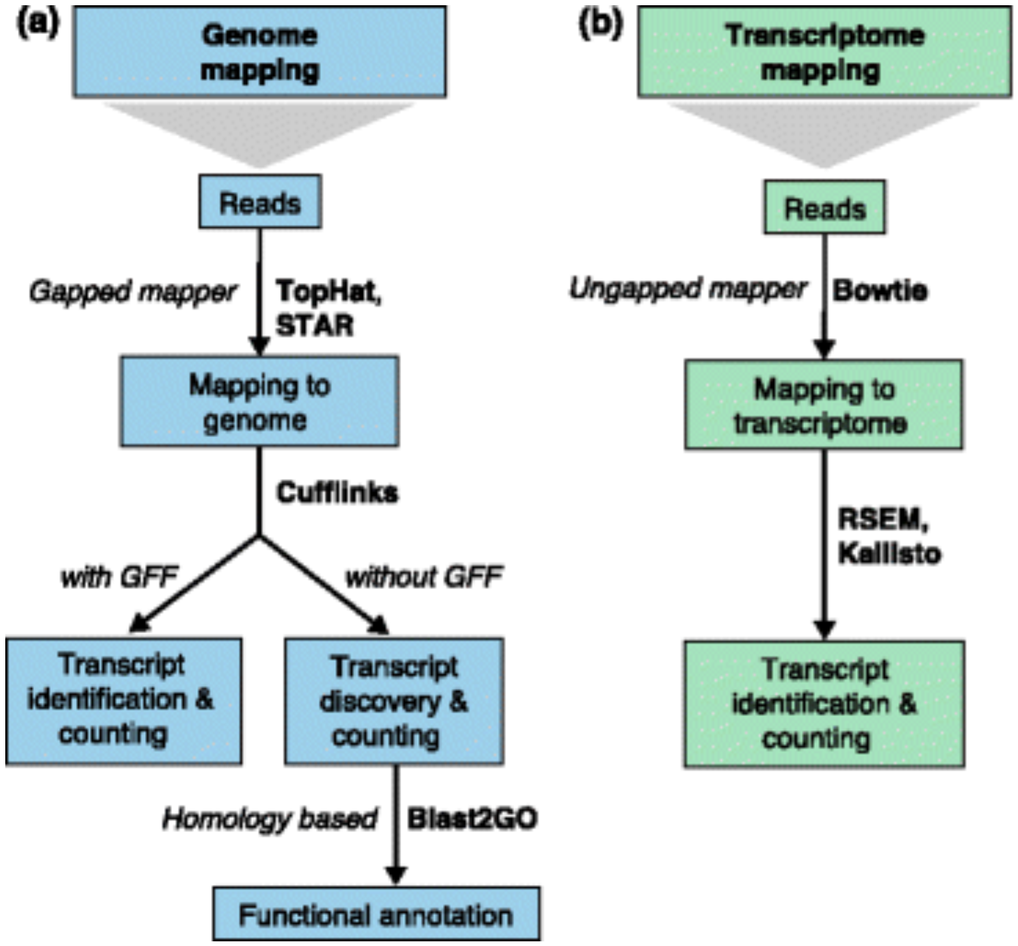


Aligning reads: tools and theory



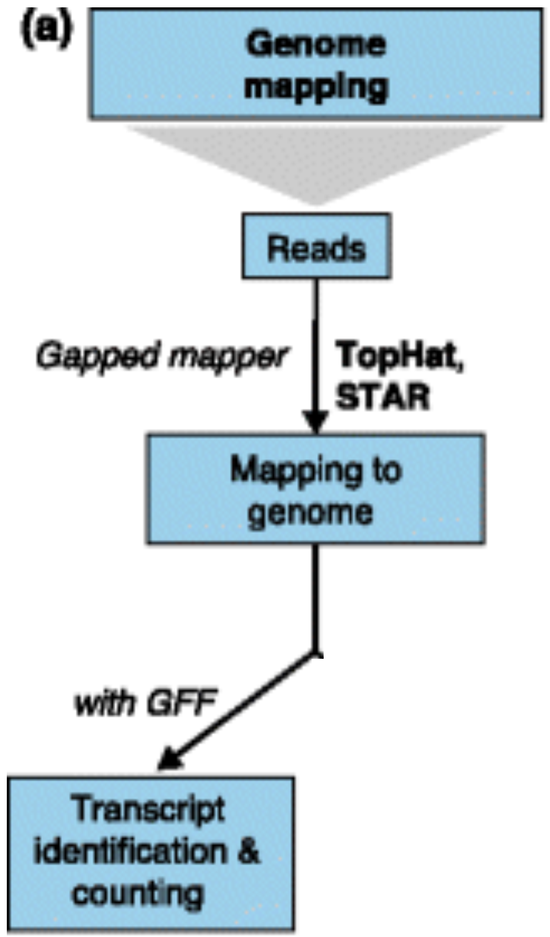
Strategies for read mapping with RNA-seq

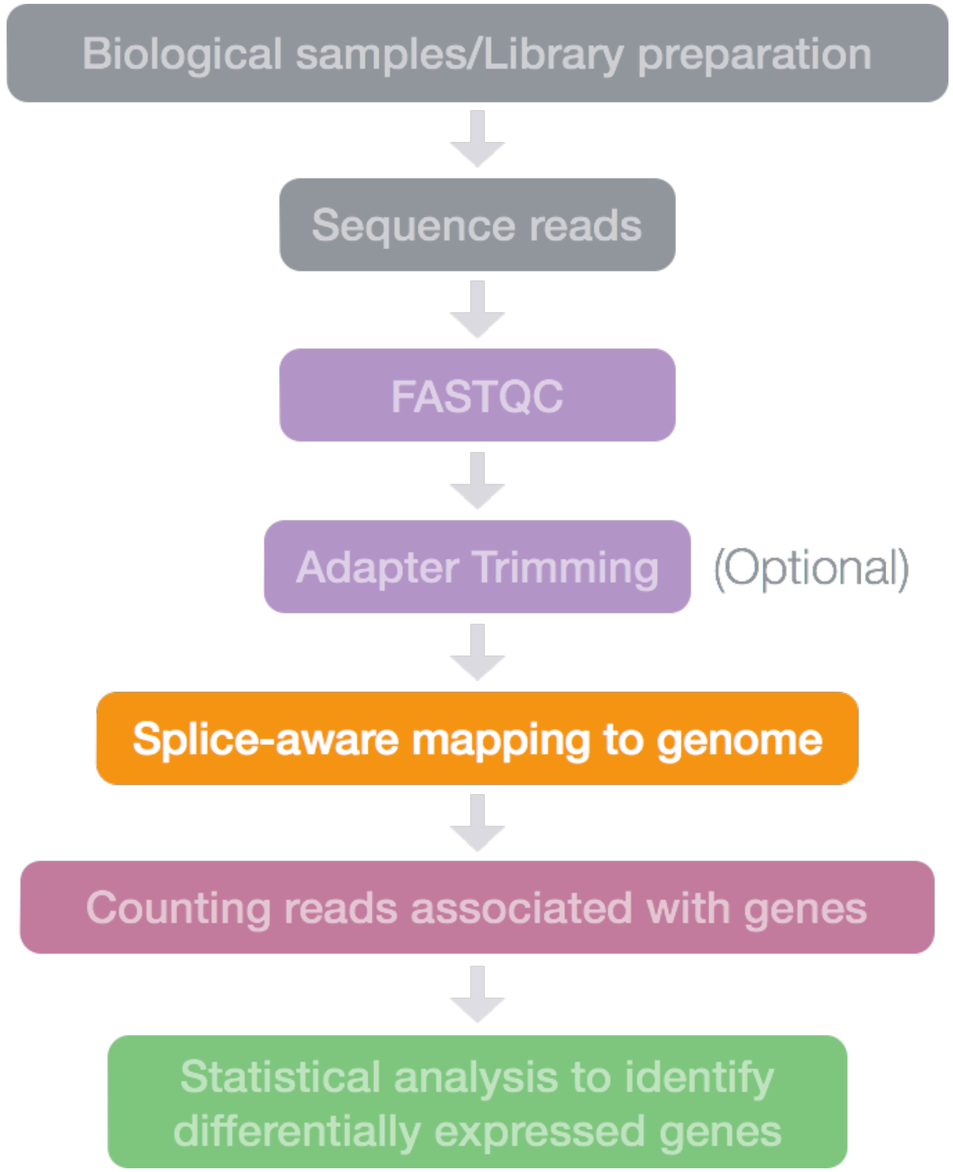
Conesa A., et al, Genome Biology 2016, 17:13 doi:10.1186/s13059-016-0881-8



Strategies for read mapping with RNA-seq

Conesa A., et al, Genome Biology 2016, 17:13 doi:10.1186/s13059-016-0881-8





Transcriptome quantification

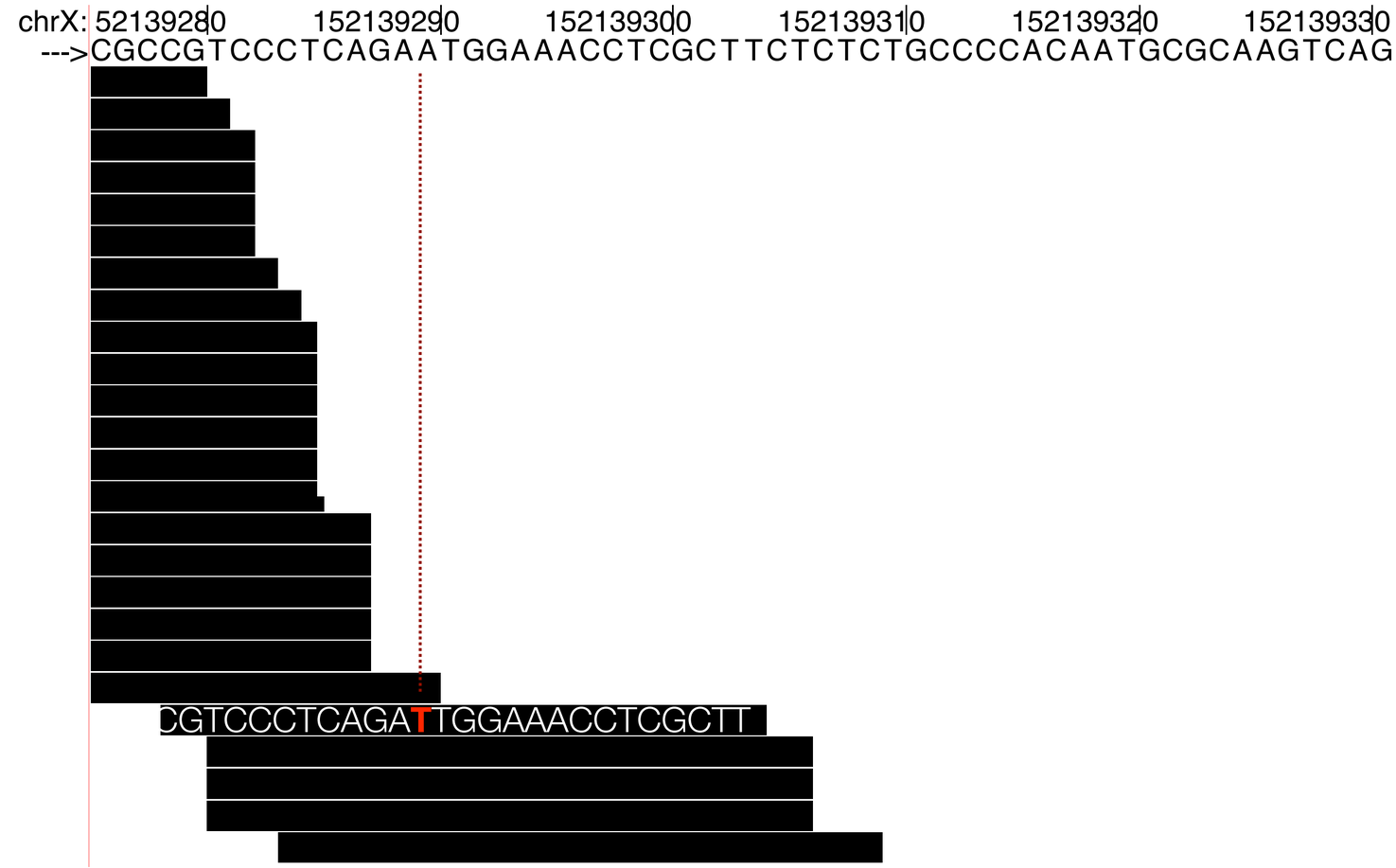
Genome

Sequence read

A simple case of string matching

Genome

Sequence reads



Difficult in practice

- Volume of data: ~3 Gbp
- ~50% of genome is repeat regions that cannot be covered by reads
  - – Simple repeats, tandem, interspersed
  - – Transposons
  - – Segmental duplications where mapping is unclear
- Gap or unfinished regions
  - – peri-centromere, sub-telomere
  - – ~5Mb unique to ethnic groups (e.g., African, Asian)
- Finishing errors(1/10,000bp), miscalled base incorporated

Challenges:

Human genome is large and complex

- Short reads: 50-150 bp (versus a very long reference)
  - Non-unique alignment
  - Sensitive to sequencing errors
- Massive amount of short reads: one lane produces ≥ 150 million 100 nucleotide reads
- Small insert size: 200-500 bp libraries

Challenges: short read NGS data

Reference	`ATCTCCATAGGACTAGAAGTAG`
Substitution	`ATCTCCATAGCACTAGAAGTAG`
Deletion	`ATCTCCATAGGAC-AGAAGTAG`
Insertion	`ATCTCCATAGGACTAGAAGTTAG`
3bp deletion	`ATCTC--AGGACTAGAAGTAG`

Challenges: non-exact matching

Local alignment vs Global alignment

- **Local alignment** matches the query with a *substring* (k-mer) of the reference
  - Tailored towards finding *regions of highly similar sequence* and aligning around those by working outwards to align the rest

Local Alignment

```
5' ACTACTAGATTACTTACGGATCAGGTACTTTAGAGGCTTGCAACCA 3'
           |||| |||||| ||||||||||||||||
      5' TACTCACGGATGAGGTACTTTAGAGGC 3'
```

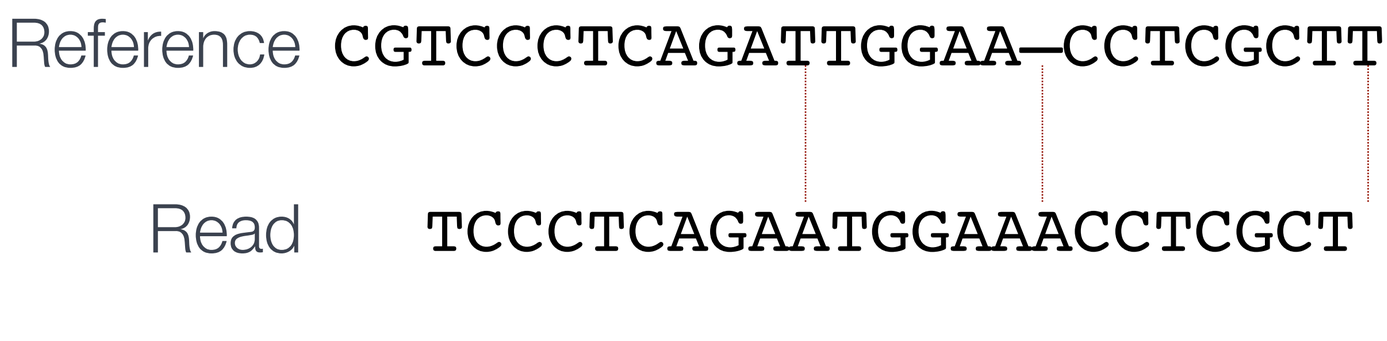
Global Alignment

```
5' ACTACTAGATTACTTACGGATCAGGTACTTTAGAGGCTTGCAACCA 3'
   ||||||||||    |||||||  |||||||||||||| |||||||
5' ACTACTAGATT----ACGGATC--GTACTTTAGAGGCTAGCAACCA 3'
```

- A **global alignment** performs end-to-end alignment between the query and the reference

Reference	`ATCTCCATAGGACTAGAAGTAG`
Substitution	`ATCTCCATAGCACTAGAAGTAG`
Deletion	`ATCTCCATAGGAC-AGAAGTAG`
Insertion	`ATCTCCATAGGACTAGAAGTTAG`
3bp deletion	`ATCTC--AGGACTAGAAGTAG`

General concepts: edit distance



Edit distance =3

General concepts: edit distance

Building an index

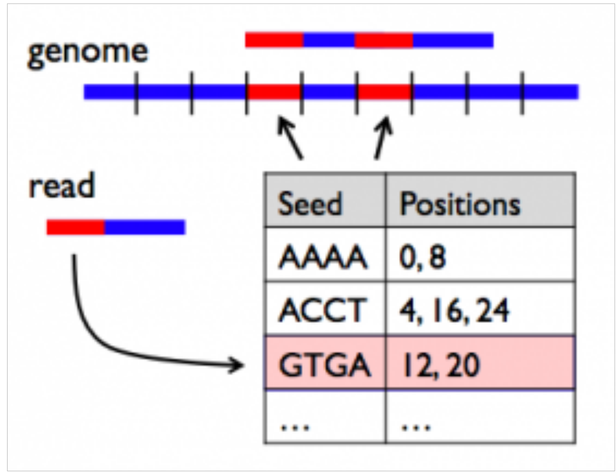
- For each read we need to scan the entire corpus as fast as possible
- Having an index of the reference genome provides an efficient way to search
- Once index is built, it can be queried any number of times
- Indexes are genome and tool-specific

Alignment tools can be grouped based on indexing method

- Some examples include:
  - Hash-based
  - Suffix arrays
  - Burrows-Wheeler Transform

Hash-based alignment (circa 1990)

- Pick k-mer size, build lookup of every k-mer in the reference mapped to its positions (the index)
- Break the query into k-mers
- Seed-and-extend strategy
- For BLAST, 100% match the query k-mer to reference then extend until score drops below 50%
- 0.1 - 1 sec per query; not feasible for NGS data



Hash-based alignment (present day)

- Need to make some concessions on sensitivity by making adaptations for use on NGS data:
  - allow for mismatches and/or gaps (ELAND, MAQ, SOAP)
  - using multiple seeds (BLAT, ELAND2)
- Memory intensive and slower (~16GB RAM required for hg19)
- Simpler in design but more sensitive

Suffix arrays

- A sorted table of all suffixes (substrings) of a given string
- A suffix array will contain integers that represent the starting indexes of the all the suffixes of a given string, after the aforementioned suffixes are sorted
- Requires large amount of memory to load the suffix array and genome sequence prior to alignment
- Popular Tools:

  STAR (2012)

Let the given string be "`mississippi`"

Suffixes	ID	Sorted Suffixes	Suffix Array
mississippi$	1	$	12
ississippi$	2	i$	11
ssissippi$	3	ippi$	8
sissippi$	4	issippi$	5
issippi$	5	ississippi$	2
ssippi$	6	mississippi$	1
sippi$	7	pi$	10
ippi$	8	ppi$	9
ppi$	9	sippi$	7
pi$	10	sissippi$	4
i$	11	ssippi$	6
$	12	ssissippi$	3

The suffix array will be:
{12, 11, 8, 5, 2, 1, 10, 9, 7, 4, 6, 3}

Burrows-Wheeler transform

- A compressed form of suffix arrays
- Tends to put runs of the same character together rather than alphabetically, which makes the compression work well

Suffixes ID	Sorted Suffixes	Suffix Array	Sorted Rotations (A_s matrix)	BWT Output (L)
mississippi$ 1	$	12	$mississippi	i
ississippi$ 2	i$	11	i$mississipp	p
ssissippi$ 3	ippi$	8	ippi$mississ	s
sissippi$ 4	issippi$	5	issippi$miss	s
issippi$ 5	ississippi$	2	ississippi$m	m
ssippi$ 6	mississippi$	1	mississippi$	$
sippi$ 7	pi$	10	pi$mississip	p
ippi$ 8	ppi$	9	ppi$mississi	i
ppi$ 9	sippi$	7	sippi$missis	s
pi$ 10	sissippi$	4	sissippi$mis	s
i$ 11	ssippi$	6	ssippi$missi	i
$ 12	ssissippi$	3	ssissippi$mi	i

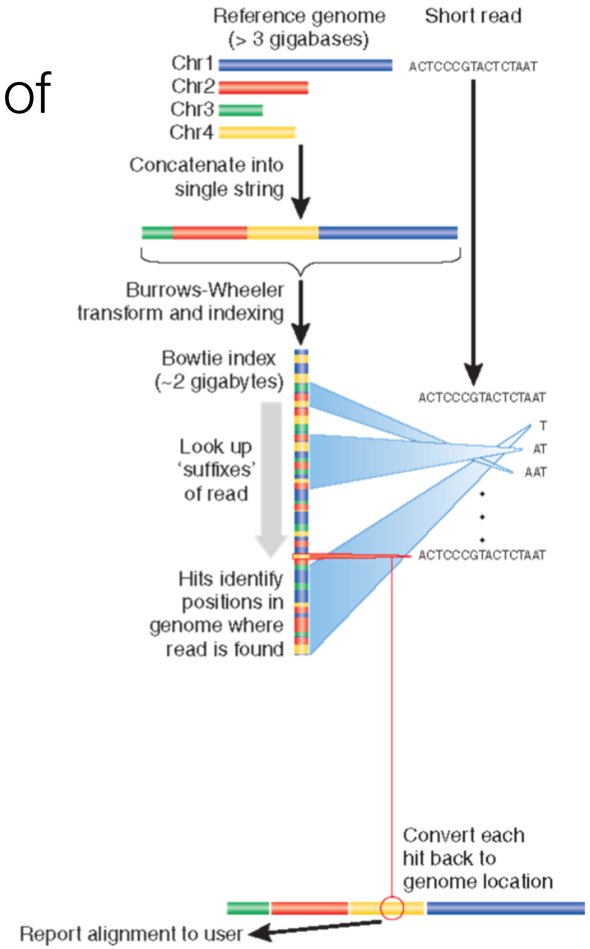
Burrows-Wheeler transform

- Much less memory because of compression; ~1.5 GB of RAM required for hg19 index
- But compression results in diminished efficiency of the string search operations
- Popular Tools:

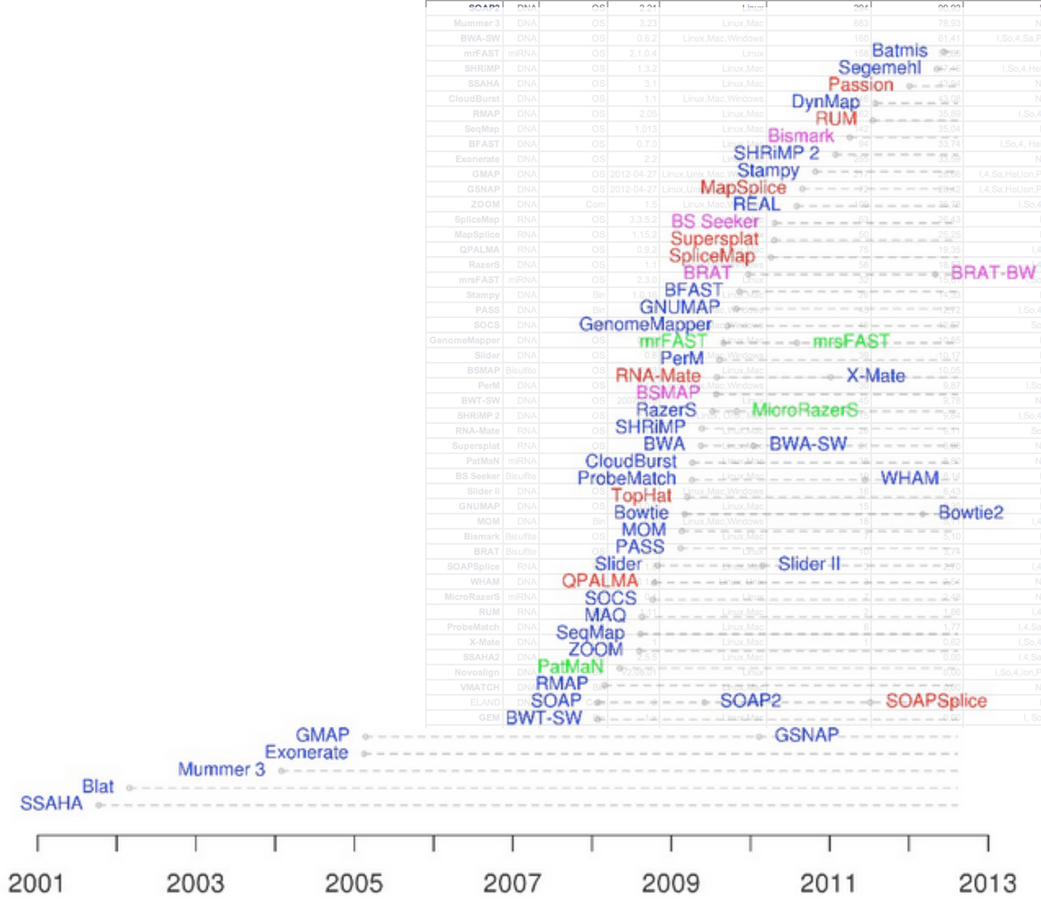
  Bowtie2 (2012)

  SOAP2

  BWA-MEM (2013)

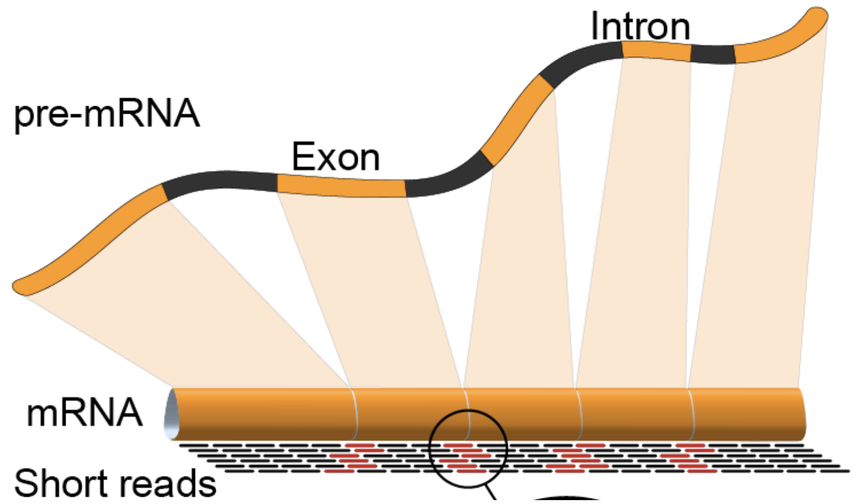


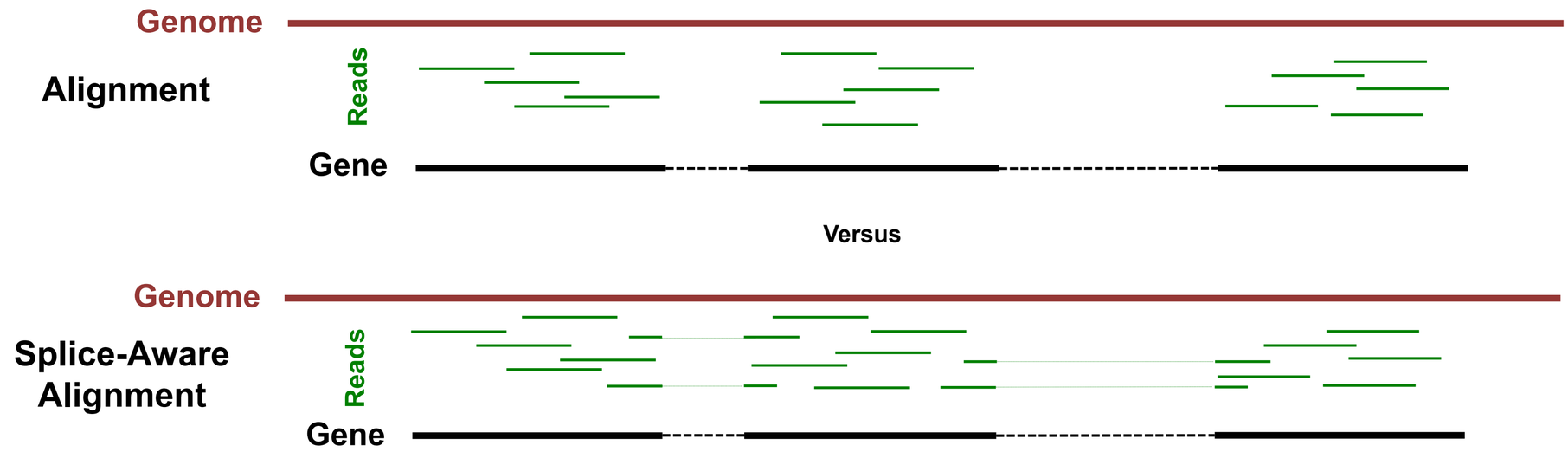
Mapper	Data	Availability	Version	O.S.	Number Citations	Citations/Years	Seq.Plat.	Input	Output	Min. RL	Max. RL	Mismatches	Indels	Gaps	Align. Reported	Alignment	Parallel	QA	PE	Splicing
Bowtie	DNA	OS	0.12.7	Linux,Mac,Windows	1168	335,04	I,So,4,Sa,P	(C)FAST(A/Q)	SAM TSV	4	1K	Score	Score	N	A,B,R,S	G L	SM	Y	Y	N
Blat	DNA	OS	34	Linux,Mac	2844	268,37	N	FASTA	TSV BLAST	11	5000K	Score	Score	Y	B	L	N	N	N	De novo
MAQ	DNA	OS	0.7.1	Linux,Mac	957	237,27	I,So	(C)FAST(A/Q)	TSV	8	63	Y	Y	N			N	Y	Y	N
BWA	DNA	OS	0.6.2	Linux,Mac,Windows	738	225,15	I,So,4,Sa,P	FASTA/Q	SAM	4	200	Y	8	Y	R,S	G	SM	Y	Y	N
TopHat	RNA	OS	1.4.1	Linux,Mac	389	112,66	I	FASTA/Q, GFF	BAM	-	-	2	0	N	B,S	-	SM	Y	Y	De novo
SOAP	DNA	OS	1.11	Linux,Mac	451	98,04	I	FASTA/Q	TSV	7	60	5	3	N	B,R S		SM	N	Y	N
SOAP3	DNA	OS	2.21	Linux	204	90,93	I	FASTA/Q	SAM TSV	27	1K	2	0	Y	A,B,R	L	SM	N	Y	N
Mummer 3	DNA	OS	3.23	Linux,Mac	683	78,93	N	FASTA	TSV	10	*	Y	Y	Y	A,B	G	N	N	N	N
BWA-SW	DNA	OS	0.6.2	Linux,Mac,Windows	160	61,41	I,So,4,Sa,P	FASTA/Q	SAM	4	1000K	0.1	0.1	Y	R,S	L	SM	Y	N	N
mrFAST	mRNA	OS	2.1.0.4	Linux	150	60,00	I	FASTA/Q	SAM	25	300	Score	6	N	A,B	G	N	N	Y	N
SHRiMP	DNA	OS	1.3.2	Linux,Mac	140	56,00	I,So,4,Hel	(C)FAST(A/Q)	TSV	14	1K	Score	Score	Y	B,S	G	SM	N	Y	N
SSAHA	DNA	OS	3.1	Linux,Mac	137	54,80	N	FASTA/Q	TSV	15	*	Y	Y	Y	B,S	G L	N	N	N	N
CloudBurst	DNA	OS	1.1	Linux,Mac,Windows	136	54,40	N	FASTA	TSV		1K	Y	Y	Y	A,B	G	Cloud	N	N	N
RMAP	DNA	OS	2.05	Linux,Mac	89	35,60	I,So,4	(C)FAST(A/Q)	BED	11	10K	Y	0	N	B,S		N	Y	Y	N
SeqMap	DNA	OS	1.013	Linux,Mac	112	35,04	I	FASTA	ELAND	15	500	5	3	N	A		SM	N	N	N
BFAST	DNA	OS	0.7.0	Linux,Mac,Windows	94	33,74	I,So,4, Hel	(C)FAST(A/Q)	SAM TSV		*	Y	Y	Y	B,R,U	G	SM	N	Y	N
Exonerate	DNA	OS	2.2	Linux,Mac,Windows	87	32,25	N	FASTA	TSV	20	*	Score	Score	Y	B,S	G L	N	N	N	De novo
GMAP	DNA	OS	2012-04-27	Linux,Unix,Mac,Windows	84	31,11	I,4,Sa,Hel,Ion,P	FASTA/Q	SAM, GFF	8	*	Y	Y	Y	B	G L	SM	N	N	De novo
GSNAP	DNA	OS	2012-04-27	Linux,Unix,Mac,Windows	84	31,11	I,4,Sa,Hel,Ion,P	FASTA/Q	SAM	8	250	Y	Y	Y	A,B,U,S	G L	SM	N	Y	Lib and de novo
ZOOM	DNA	Com	1.5	Linux,Mac	77	28,52	I,So,4	(C)FAST(A/Q)	SAM BED GFF	12	240	Y	Y	N	B,U,S	G	SM/DM	Y	Y	N
SpliceMap	RNA	OS	3.3.5.2	Linux,Mac	67	24,83	I	FASTA/Q	SAM BED	-	-	0.1		Y	A	-	SM	N	Y	Lib and/or de novo
MapSplice	RNA	OS	1.15.2	Linux,Mac	64	23,70	I	FASTA/Q	SAM BED	-	-	3		Y	B	-	SM	N	Y	De novo
QPALMA	RNA	OS	0.9.2	Linux,Mac	61	22,59	I,4	Specific	TSV	-	-	Y	Y	Y	B	L	N	Y	N	Lib and de novo
RazerS	DNA	OS	1.1	Linux,Mac,Windows	60	22,22	I	FASTQ	TSV ELAND	11	*	Score	Score	Y	A,B,S	G	N	N	Y	N
mrsFAST	mRNA	OS	2.3.0	Linux	59	21,85	I	FASTA/Q	SAM	25	200	Y	0	N	A	G	N	N	Y	N
Stampy	DNA	Bin	1.0.18	Linux,Mac,Windows	58	21,48	I	FASTA/Q	SAM TSV	4	4K	0.15	30	N	B,R,S	G	N	Y	Y	N
PASS	DNA			Linux,Mac,Windows	54	20,02	I,So,4	(C)FAST(A/Q)	SAM GFF3 BLAST	23	1K	Y	Y	Y	A,B	G	SM	Y	Y	De novo
SOCS	DNA			Linux,Mac,Windows	54	20,02	So	(C)FAST(A/Q)	TSV		64	Y	0	N	A,B		SM	Y	N	N
GenomeMapper	DNA	OS		Linux,Mac,Windows	52	19,26	I	FASTA/Q	BED TSV	12	2K	10	10	Y	A,B,R	G	SM	N	N	N
Slider	DNA	OS	0.9	Linux,Mac,Windows	51	18,89	I	Illumina	TSV		62	3	0	N	B,S		N	Y	Y	N
BSMAP	Bisulfite	OS		Linux,Mac	51	18,89	I	FASTA/Q	SAM TSV	8	144	15	0	N	B,U,S		SM	N	Y	N
PerM	DNA	OS		Linux,Mac,Windows	51	18,89	I,So	(C)FAST(A/Q)	SAM TSV	20	128	9	0	Y	A,U	G	DM	Y	Y	N
BWT-SW	DNA	OS	20	Linux,Mac	47	17,41	N	FASTA	TSV		1K	Score	Score	Y	A		N	N	N	N
SHRiMP 2	DNA	OS		Linux,Mac	46	17,04	I,So,4	FASTA/Q	SAM	30	1K	Y	Score	N	B,U,S	G	SM	Y	Y	N
RNA-Mate	RNA	OS		Linux,Mac	41	15,19	So	CFASTA	BED Counts	-	-	Y	0	N	S	-	DM	Y	N	Lib
Supersplat	RNA	OS		Linux,Mac	40	14,82	N	FASTA	TSV			0	0	Y	A,U	G	N	N	N	De novo
PatMaN	mRNA	OS		Linux,Mac	40	14,82	N	FASTA	TSV	1	*	Y	Y	N	A	G	N	N	N	N
BS Seeker	Bisulfite	OS		Linux,Mac	40	14,82	I	FASTA/Q	SAM	-	-	3	0	N	U	-	SM	Y	N	N
Slider II	DNA	OS		Linux,Mac,Windows	15	5,43	I	Illumina	TSV		93	Y		N	B,S		N	N	Y	N
GNUMAP	DNA	OS		Linux,Mac	10	3,70	I	FASTA/Q Illumina	SAM TSV	16	1K	Score	Score	Y	B	G	SM/DM	Y	N	N
MOM	DNA	Bin		Linux,Mac,Windows	10	3,70	I,4	FASTA	TSV			Y	0	N	A	L	SM	N	Y	N
Bismark	Bisulfite	OS		Linux,Mac	10	3,70	I	FASTA/Q	SAM	16	10K	Score	Score	N	U	-	SM	Y	Y	N
BRAT	Bisulfite	OS		Linux,Mac	10	3,70	I	FASTA/Q	TSV			Y	0	N			N	N	Y	N
SOAPSplice	RNA	OS		Linux,Mac	9	3,33	I,4	FASTA/Q	TSV	13	3K	5	2	Y	U	-	SM	Y	Y	De novo
WHAM	DNA	OS		Linux,Mac	9	3,33	N	FASTQ	SAM	5	128	5	3	N	A,B,R,U,S	G	N	Y	Y	De novo
MicroRazerS	mRNA	OS		Linux,Mac,Windows	9	3,33	N	FASTA	SAM TSV	10	*	Score	0	N	S	G	N	N	N	N
RUM	RNA	OS		Linux,Mac	5	1,86	I,4	FASTA/Q	SAM TSV BED	-	-	Y	Y	Y	B	-	SM	N	Y	De novo
ProbeMatch	DNA	OS		Linux,Mac	5	1,77	I,4,Sa	FASTA	ELAND	36	50	3	Y	N	A,B		N	N	N	N
X-Mate	DNA	OS		Linux,Mac	1	0,62	I,So,4	(C)FAST(A/Q)	SAM BED Counts	-	-	Y	0	N	S	-	DM	Y	N	Lib
SSAHA2	DNA	OS		Linux,Mac	0	0,00	I,4,Sa	FASTA/Q	SAM	15	48K	Score	Score	N	B,S	L	N	N	Y	N
Novoalign	DNA	OS	2.7.8	Linux	0	0,00	I,So,4,Ion,P	(C)FAST(A/Q) Illumina	SAM TSV	30	300	8	2	N	A,B,R,U,S	G	SM/DM/Cloud	Y	Y	Lib
VMATCH	DNA	OS		Linux,Mac,Windows	0	0,00	N	FASTA	TSV			Score	Score	Y	A,B,S	G L	N	N	N	N
ELAND	DNA	OS		Linux,Mac,Windows	0	0,00	I	FASTA	TSV		32	2	0	N	B		N	N	N	N
GEM	DNA	OS		Linux,Mac	0	0,00	I,So	FASTA/Q	SAM, Counts		*	Y	Y	Y	A,S	G	SM	Y	Y	Lib and de novo



http://wwwdev.ebi.ac.uk/fg/hts_mappers/

Short-read aligners: choices





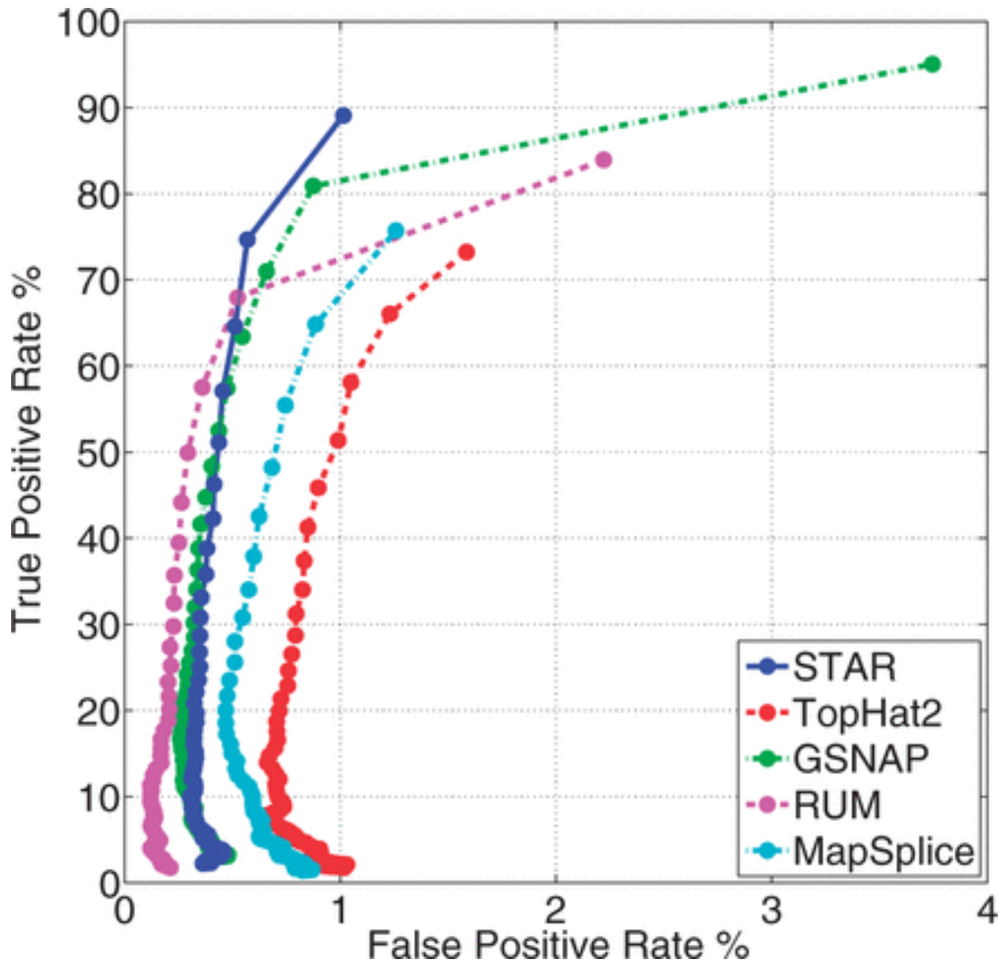
Splice-aware alignment

Splice-aware alignment tools:

HISAT2, STAR, MapSplice, SOAPSplice, Passion, SpliceMap, RUM, ABMapper, CRAC, GSNAP, HMMSplicer, Olego, BLAT

There are excellent aligners available that are not splice-aware. These are useful for aligning directly to genes. However, you will lose isoform information.

Bowtie2, BWA, Novoalign (not free), SOAPaligner



Aligner	Mapping speed: million read pairs/hour		Peak physical RAM, GB	
	6 threads	12 threads	6 threads	12 threads
STAR	309.2	549.9	27.0	28.4
STAR sparse	227.6	423.1	15.6	16.0
TopHat2	8.0	10.1	4.1	11.3
RUM	5.1	7.6	26.9	53.8
MapSplice	3.0	3.1	3.3	3.3
GSNAP	1.8	2.8	25.9	27.0

Bioinformatics (2013) 29 (1): 15-21

The RNA-Seq specific tools

Alignment for RNA-Seq

- Use the strategy that is most relevant based on the quality of your genome and GTF
- Choose an aligner that can allow for a read to be “split” across distant regions to account for splice events
- Evaluate your computational resources and use an aligner that would work best within the confines of the available memory and CPU