UNIVERSITY OF UTAH
SCHOOL OF MEDICINE | Department of Human Genetics

FASTQ format & seq. alignment overview

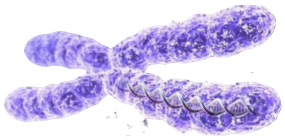
Andrew Farrell
USTAR Center for Genetic Discovery
University of Utah

USTAR Center for Genetic Discovery

1. What is a sequence read (a.k.a. “read”)?

Reads are the sequencer's best guess at what it saw for a given DNA molecule.

It's the "raw" data.



"base calling"
(mistakes happen)

cctaaccct
acccctgg
acccctgg
tcccctgg
cctaaccct
cctaaccct
ggcccctgg
acccctgg
acccctgg
actaaccct
cctaaccct
cctaaccct
acccctgg
cctaaccct
acccctgg
cctaaccct

Alignment is central to most genomics applications

FASTQ format

Align to genome

Quality control

Call SNPs

Find peaks

Quantify transcripts

Find SVs

.
.
.

etc.

Analysis, Interpretation, Discovery

Phase 0

Phase1

Phase 2-100

Alignment is central to most genomics applications

FASTQ format

Align to genome

Quality control

Call SNPs

Find peaks

Quantify transcripts

Find SVs

.
.
.

etc.

Analysis, Interpretation, Discovery

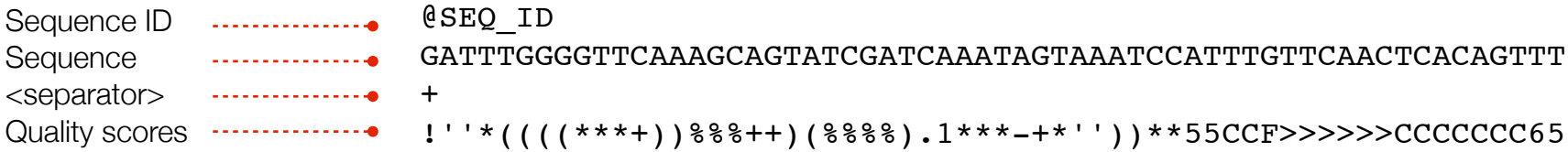
Phase 0

Phase1

Phase 2-100

The FASTQ format

A “standard” format for storing and defining sequences from next-generation sequencing technologies.



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

Sequence IDs

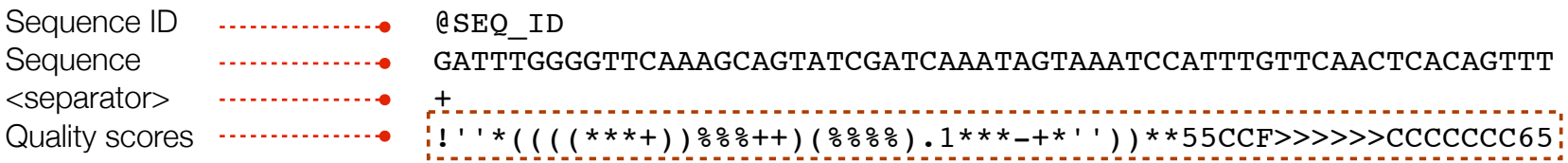
```
@HWUSI-EAS100R:6:73:941:1973#0/1
```

HWUSI-EAS100R	the unique instrument name
6	flowcell lane
73	tile number within the flowcell lane
941	'x'-coordinate of the cluster within the tile
1973	'y'-coordinate of the cluster within the tile
#0	index number for a multiplexed sample (0 for no indexing)
/1	the member of a pair, /1 or /2 *(paired-end or mate-pair reads only)*

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

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

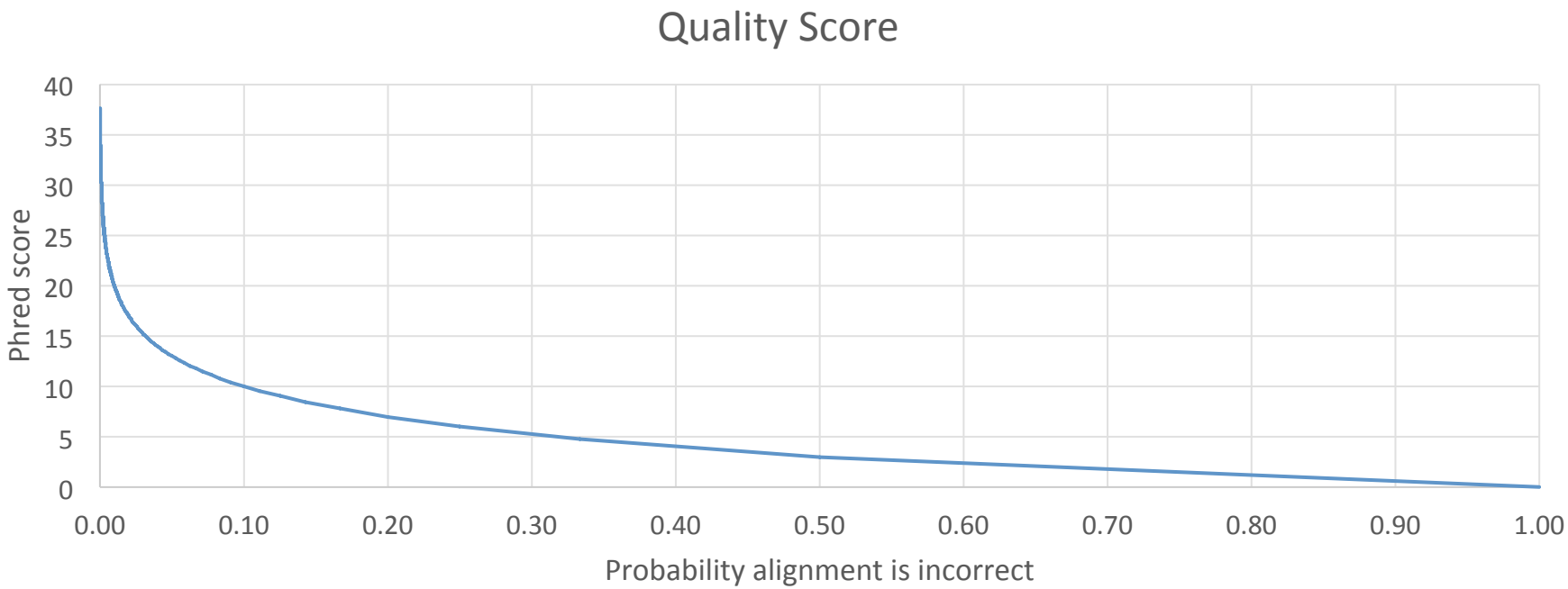
Quality scores

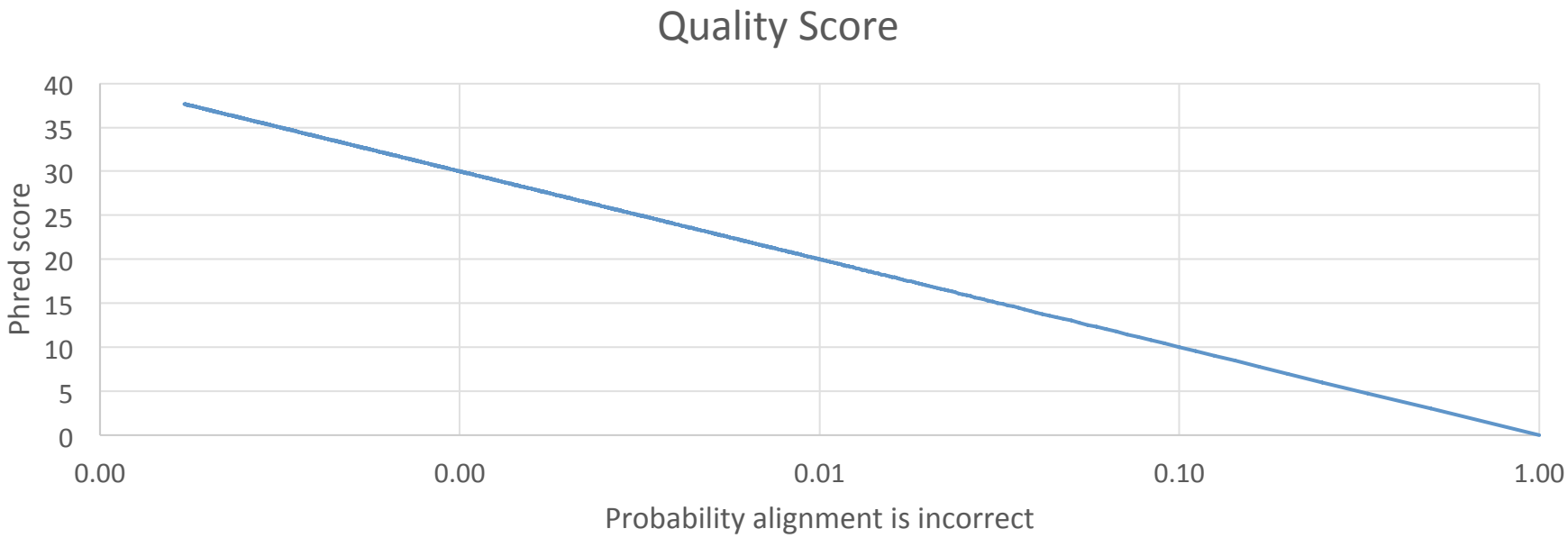


Qualities are based on the Phred scale and are *encoded*

$$Q = -10*\log_{10}(P_{err})$$

Q scores are derived differently for each technology.


Quality Score
Phred score
40
35
30
25
20
15
10
5
0
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.80
0.90
1.00
Probability alignment is incorrect


Quality Score
Phred score
40
35
30
25
20
15
10
5
0
0.00
0.00
0.01
0.10
1.00
Probability alignment is incorrect

Quality score encoding

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Formula for getting PHRED quality from encoded quality:

Q = ascii(char) - 33

Example:

!+EI

ASCII	33	43	69	73
-33	0	10	36	40

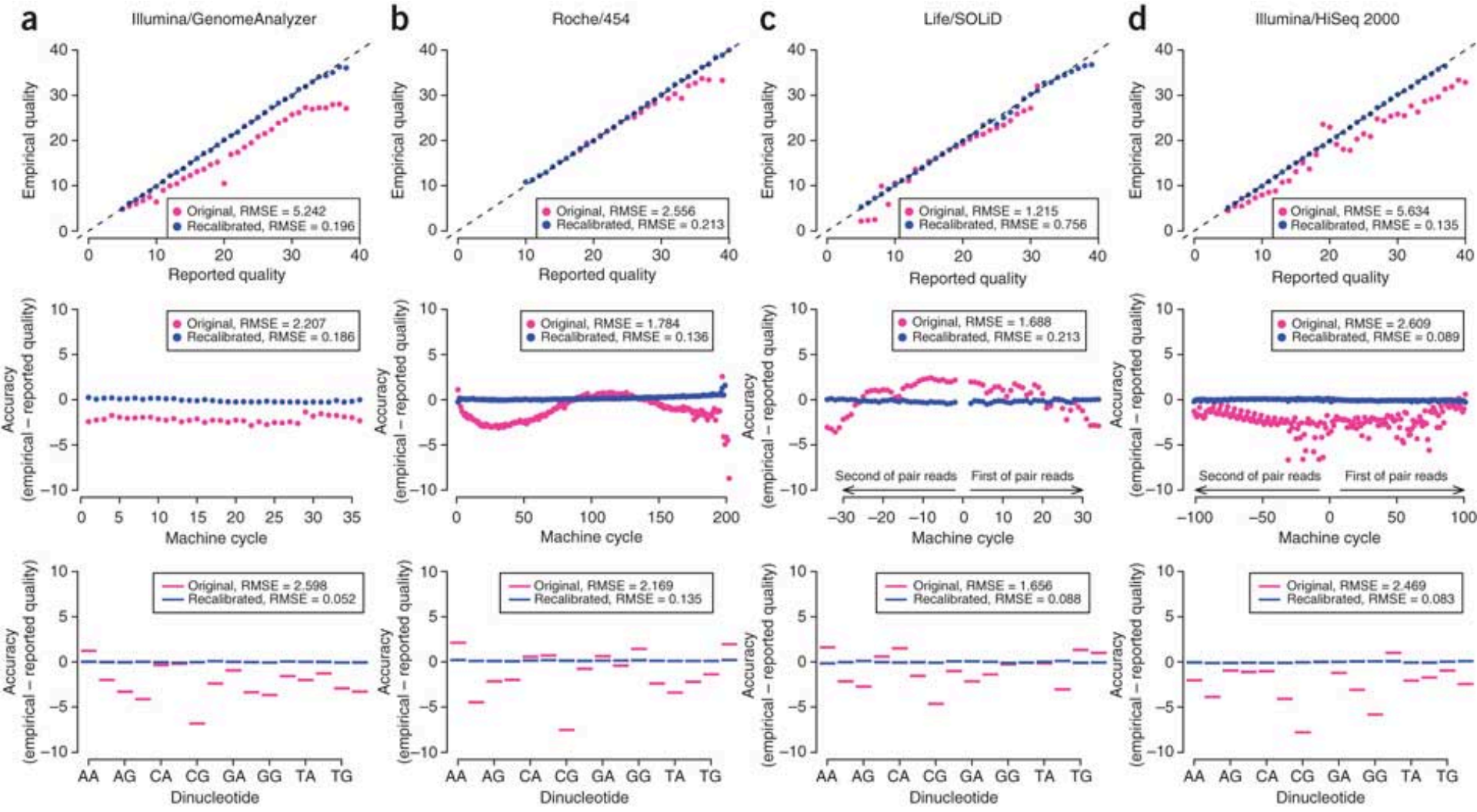
Quality score encoding

bonus question: why do we encode?

```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
 .........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.....................
 ..............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII..........................
 .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ.....................
 LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL.....................................................
 !"#$%&'()*+,-./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)
   (Note: See discussion above).
L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
```

Quality score are not always perfect



We have FASTQ files. Now what?

- Need to find a home for every read in the file.

- Must get the alignment just right. Else problems.

- Must choose the right tool for the experiment.

Alignment is central to most genomics applications

FASTQ format

Align to genome

Quality control

Call SNPs

Find peaks

Quantify transcripts

Find SVs

.
.
.

etc.

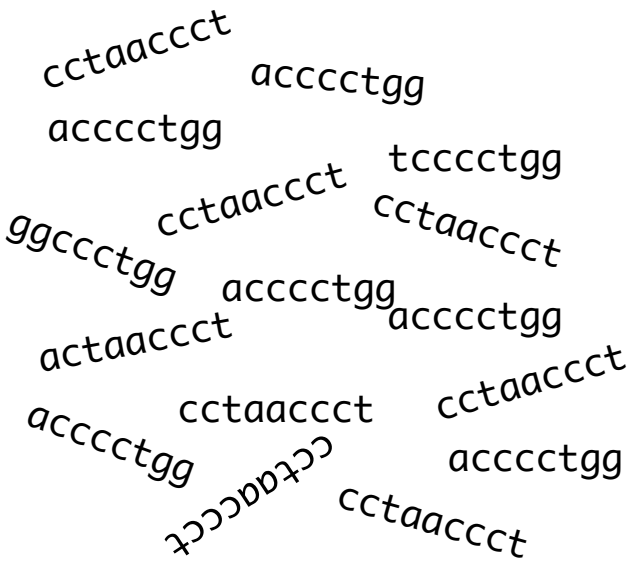
Analysis, Interpretation, Discovery

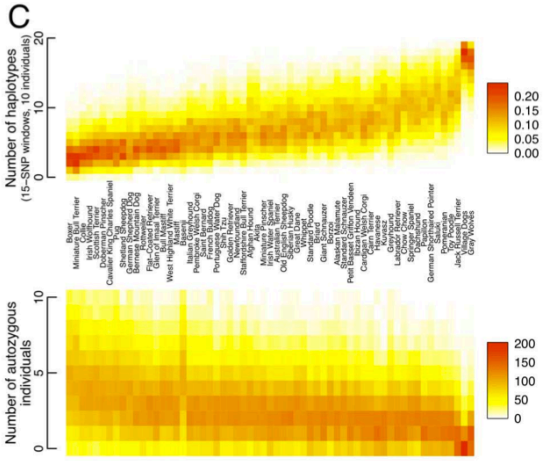
Phase 0

Phase1

Phase 2-100

The goal. Easy, right?

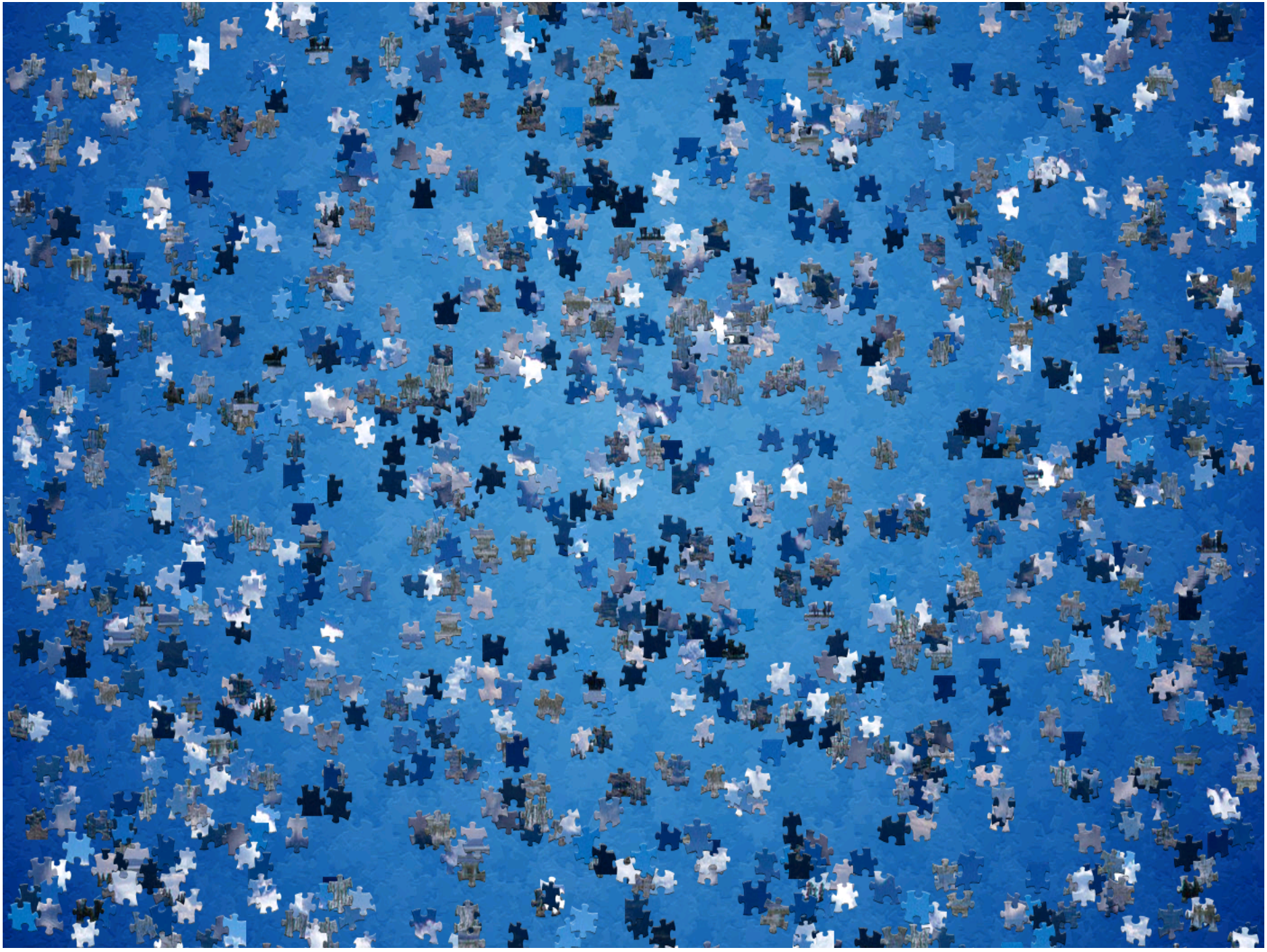




Sequence alignment is the crucial first step.

The problem.

- The human genome is big. Oh yeah, it's complex too.
- Sequencers can produce billions reads / run.
- But they make mistakes. Frequently.
- Accurate alignment takes time, but it's worth it.
  - Shortcuts lead to artifacts
- Alignment strategy is highly nuanced, depending on experimental context



Mapping to a reference genome

- This is like a jigsaw puzzle
- Compare reads to a reference genome, accounting for genetic differences
- Two major approaches:
  - Hashing the reference
  - Burrows-Wheeler transform

Best case scenario:

An error-free sequencing technology

ATTCGAAACA
TTCGCGCAAT
CTGGACTCAA

ATTCGAAACA
TTCGCGCAAT
CTGGACTCAA

→ Aligner →

TACCTCCAGGGGGCATCCTCCC
CCCCAATTCGAAACACAATCGTA
GCCCCTGGCACTACCTATGTGTG
TCAATTCGGAGAGAGAGAGATTC
ACGAAAAAAAAGTCTGGACTCAA
CTAGGATACACACATTCGGCTACA
GATACCAAAAAAAAAAAAAAAAA
ATTTTCACCATTGAGGCACCACCT
TCTCGTCGCTGCGTCGCTCTGCT
CGCTTCGGCTAAAAATTCGCGCA
ATACATTCGGCTACAGATACCAAA
AAAA

Computers are rather good at finding ***exact*** matches.
Think Google.

Reality

Errors happen - frequently; work harder.

ATTCGAAACA

→ ATTTGAAACA → Aligner

TACCTCCAGGGGGCATCCTCCC
CCCCAATTCGAAACACAATCGTA
GCCCCTGGCACTACCTATGTGTG
TCAATTCGGAGAGAGAGAGATTG
GAAACAAAAAAGTGCTACAGATA
CCACTAGGATACACACATTCGGC
TACAGATACCAAAAAAAAAAAAA
AAAAATTTTCACCATTGAGGCACC
ACCTTCTCGTCGCTGCGTCGCTC
TGCTCGCGGCTAAAAAATTAGAAA
CAACATTCGGCTACAGATACCAAA
ATTT

"Fuzzy" matching is much more computationally expensive.
Think Google's "Did you mean..."

Error types and rates modulate accuracy

Tech	Mode Error Type	Error Rate*
Illumina	Substitutions	Low
SOLiD	Substitutions	Low
454	Insertions & Deletions	Medium
Ion Torrent	Insertions & Deletions	Medium
PacBio	Deletions	High

There *are* optimal solutions.

Smith-Waterman, Needleman-Wunsch

Reference `cgggtatccaa`

Read `ccctaggtccca`

What is the best alignment?

Reference cgggtatccaa

Read ccctaggtccca

Reference cgggta--t-ccaa

Read ccc-taggtccc-a

Reference cgggtatccaa

Read ccctaggtccca

Reference cgggta--t-ccaa

Read ccc-taggtccc-a

Reference cgggta---tccaa

Read cc--ctaggtccca


Reference cgggtatccaa
Read ccctaggtccca
Reference cgggta--t-ccaa
Read ccc-taggtccc-a
Reference cgggta---tccaa
Read cc--ctaggtccca
Reference c-gggta--tccaa
Read cc--ctaggtccca

Smith Waterman algorithm

- Find the optimal alignment for each candidate.
- Maximise similarity measure between two sequences

Smith-Waterman example

- Generate a matrix with the sequences to compare
- Populate matrix with scores

$$M(i,0) = 0 \text{ for } 0 \le i \le m$$

$$M(0,j) = 0 \text{ for } 0 \le j \le n$$

$$M(i,j) = \max \begin{bmatrix} 0 \\ M(i-1, j-1) + s(a_i, b_j) \\ \max_{k \ge 1}\{M(i-k, j) + W_k\} \\ \max_{l \ge 1}\{M(i, j-l) + W_l\} \end{bmatrix}$$

	-	A	⋯	A
-	M(0,0)	M(1,0)	⋯	M(i,0)
A	M(0,1)	M(1,1)	⋯	M(i,1)
⋮	⋮	⋮	⋱	⋮
A	M(0,j)	M(1,j)	⋯	M(i,j)

Smith-Waterman example

$M(i,0) = 0 \text{ for } 0 \leq i \leq m$

$M(0,j) = 0 \text{ for } 0 \leq j \leq n$

	-	A	C	A	C	A	C	T	A
-	0	0	0	0	0	0	0	0	0
A	0								
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Smith-Waterman example

$$M(i,j) = \max \begin{bmatrix} 0 \\ M(i-1, j-1) + s(a_i, b_j) \\ \max_{k \geq 1}\{M(i-k, j) + W_k\} \\ \max_{l \geq 1}\{M(i, j-l) + W_l\} \end{bmatrix}$$

$$M(i-1, j-1) + s(a_i, b_j)$$

$s(a_i, b_j) = +2$ if $a = b$ Match

$s(a_i, b_j) = -1$ if $a \neq b$ Mismatch

$M(1, 1) = +2$

	-	A	C	A	C	A
-	0	0	0	0	0	0
A	0	M(1,1)				
G	0					
C	0					
A	0					
C	0					
A	0					
C	0					
A	0					

Smith-Waterman example

$$M(i,j) = \max \begin{bmatrix} 0 \\ M(i-1, j-1) + s(a_i, b_j) \\ \max_{k \geq 1}\{M(i-k, j) + W_k\} \\ \max_{l \geq 1}\{M(i, j-l) + W_l\} \end{bmatrix}$$

$$M(i-1, j-1) + s(a_i, b_j)$$

$s(a_i, b_j) = +2$ if $a = b$ Match

$s(a_i, b_j) = -1$ if $a \neq b$ Mismatch

$M(1, 1) = +2$

		-	A	C	A	C	A
	-	0	0	0	0	0	0
A		0	2				
G		0					
C		0					
A		0					
C		0					
A		0					
C		0					
A		0					

Smith-Waterman example

$$M(i,j) = \max \begin{bmatrix} 0 \\ M(i-1, j-1) + s(a_i, b_j) \\ \max_{k \geq 1}\{M(i-k, j) + W_k\} \\ \max_{l \geq 1}\{M(i, j-l) + W_l\} \end{bmatrix}$$

Insertion or deletion scoring

$W_i = -1$

		-	A	C	A	C	A
	-	0	0	0	0	0	0
A		0	2	M(2,1)			
G		0					
C		0					
A		0					
C		0					
A		0					
C		0					
A		0					

Smith-Waterman example

	-	A	C	A	C	A	C	T	A
-	0	0	0	0	0	0	0	0	0
A	0	2	1						
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Smith-Waterman example

	-	A	C	A	C	A	C	T	A
-	0	0	0	0	0	0	0	0	0
A	0	2	1	2					
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Smith-Waterman example

	-	A	C	A	C	A	C	T	A
-	0	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0	2	2	5	4	5	4	3	4
C	0	1	4	4	7	6	7	6	5
A	0	2	3	6	6	9	8	7	8
C	0	1	4	5	8	8	11	10	9
A	0	2	3	6	7	10	10	10	12

Traceback

- Start at highest value
- Diagonal line is a match/mismatch
- Up/down or left/right are indels

Sequence 1
A-CACACTA

Sequence 2
AGCACAC-A

	-	A	C	A	C	A	C	T	A
-	0	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0	2	2	5	4	5	4	3	4
C	0	1	4	4	7	6	7	6	5
A	0	2	3	6	6	9	8	7	8
C	0	1	4	5	8	8	11	10	9
A	0	2	3	6	7	10	10	10	12

$$\text{Max} \left\{ \begin{array}{c} 0 \\ M(i-1,j-1) + s(i,j) \\ M(i-1,j) + G \\ M(i,j-1) + G \end{array} \right\} \quad \{s(i,j) \text{ is M or X}\}$$

M =
X =
G =

		A	T	G	C	A
	0	0	0	0	0	0
A	0					
T	0					
C	0					
C	0					
A	0					

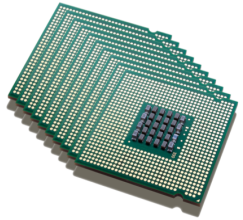
Mapping to a reference genome

- This is like a jigsaw puzzle
- Compare reads to a reference genome, accounting for genetic differences
- Two major approaches:
  - Hashing the reference
  - Burrows-Wheeler transform

Recall:

ATTCGAAACA
TTCGCGCAAT
CTGGACTCAA

ATTCGAAACA
TTCGCGCAAT
CTGGACTCAA



TACCTCCAGGGGGCATCCTCCCCCCC
AATTCGAAACACAATCGTAGCCCCTG
GCACTACCTATGTGTGTCAATTCGGA
GAGAGAGAGATTCACGAAAAAAAAGT
CTGGACTCAACTAGGATACACACATT
CGGCTACAGATACCAAAAAAAAAAAA
AAAAAAATTTTCACCATTGAGGCACC
ACCTTCTCGTCGCTGCGTCGCTCTGC
TCGCTTCGGCTAAAAATTCGCGCAAT
ACATTCGGCTACAGATACCAAAAAAA

Computers are rather good at finding ***exact*** matches.

This is the motivation behind hash-based aligners.

Hash-based alignment:

Step1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

Hash-based alignment:

Step 1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1

Hash-based alignment:

Step 1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1
ATG	2

Hash-based alignment:

Step 1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1
ATG	2
TGG	3

Hash-based alignment:

Step1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1
ATG	2
TGG	3
GGT	4

Hash-based alignment:

Step 1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1
ATG	2
TGG	3
GGT	4
GTC	5

Hash-based alignment:

Step1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1
ATG	2
TGG	3
GGT	4
GTC	5
TCA	6

Hash-based alignment:

Step1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3
GGT	4
GTC	5
TCA	6

Hash-based alignment:

Step1: hash/index the genome

Toy genome (16 bp) CATGGTCATTGGTTCC

k = 3

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

Complete hash/kmer index of our toy genome (forward strand only)

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

→ Read TGGTCA

kmer index is used to quickly find candidate alignment locations in genome.

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

→ Read TGGTCA

kmer index is used to quickly find candidate alignment locations in genome.

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

→

Read TGGTCA

3,10

kmer index is used to quickly find candidate alignment locations in genome.

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

→

Read TGGTCA

3,10

kmer index is used to quickly find candidate alignment locations in genome.

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

→

Read TGGTCA

3,10 6

kmer index is used to quickly find candidate alignment locations in genome.

Hash-based alignment:

Step2: use the index to seed alignments.

Toy genome CATGGTCATTGGTTCC

Read TGGTCA

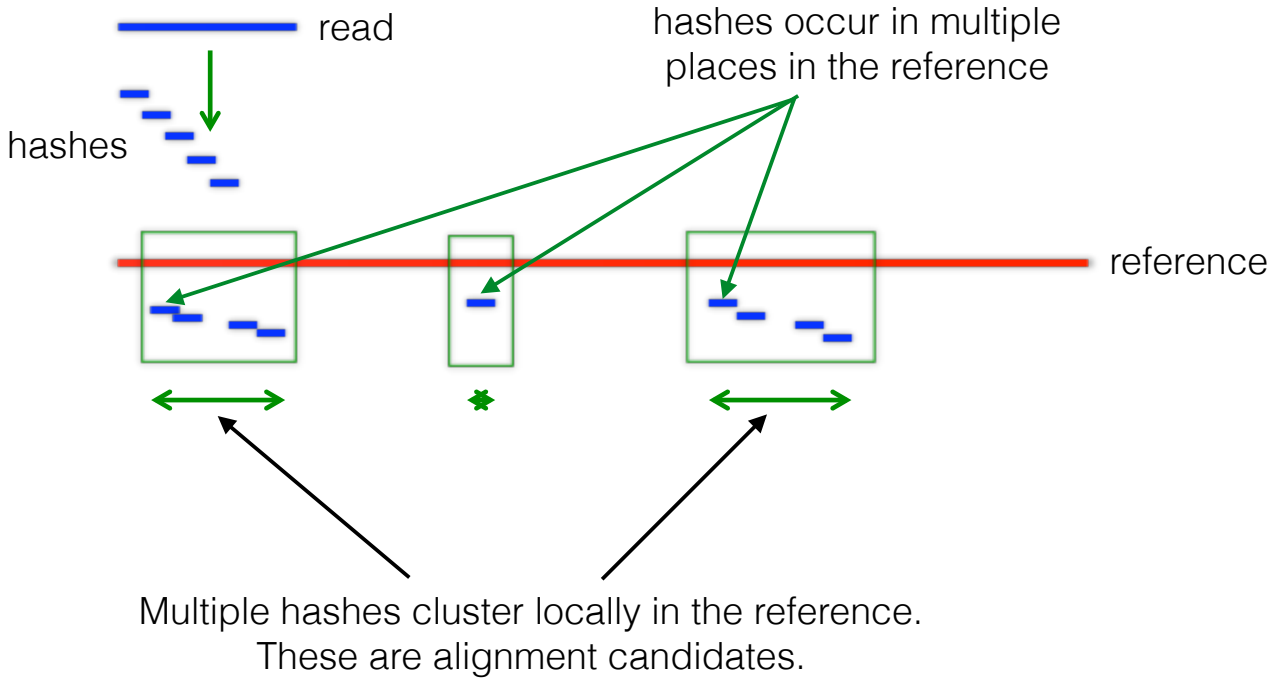
3,10 6

Kmer/Hash	Positions
CAT	1,7
ATG	2
TGG	3,10
GGT	4,11
GTC	5
TCA	6
ATT	8
TTG	9
GTT	12
TTC	13
TCC	14

kmer index is used to quickly find candidate alignment locations in genome.

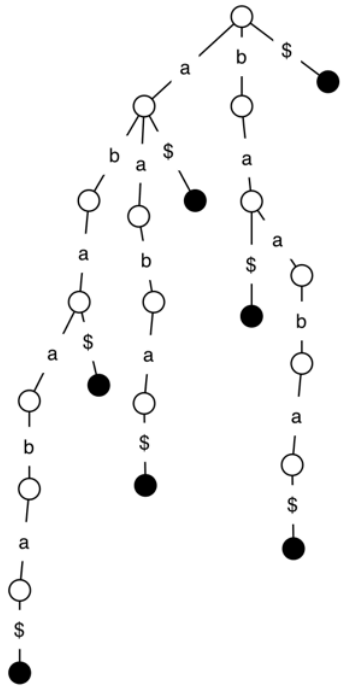
Compare read to reference

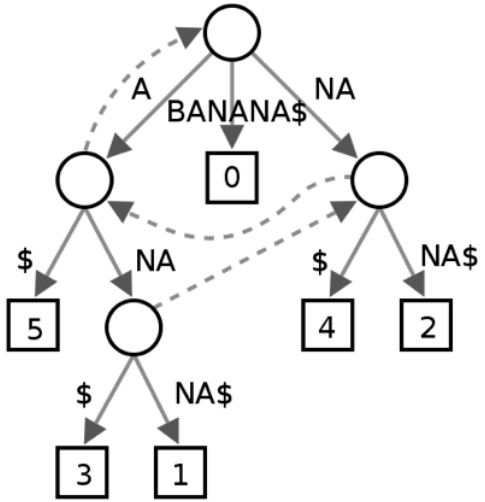
- Find where each hash lands in the reference:



Burrows-Wheeler Transform

The latest fad. More involved computationally, but requires much less memory.





6	$
5	A$
3	ANA$
1	ANANA$
0	BANANA$
4	NA$
2	NANA$

$ B A N A N A
A $ B A N A N
A N A $ B A N
A N A N A $ B
B A N A N A $
N A $ B A N A
N A N A $ B A

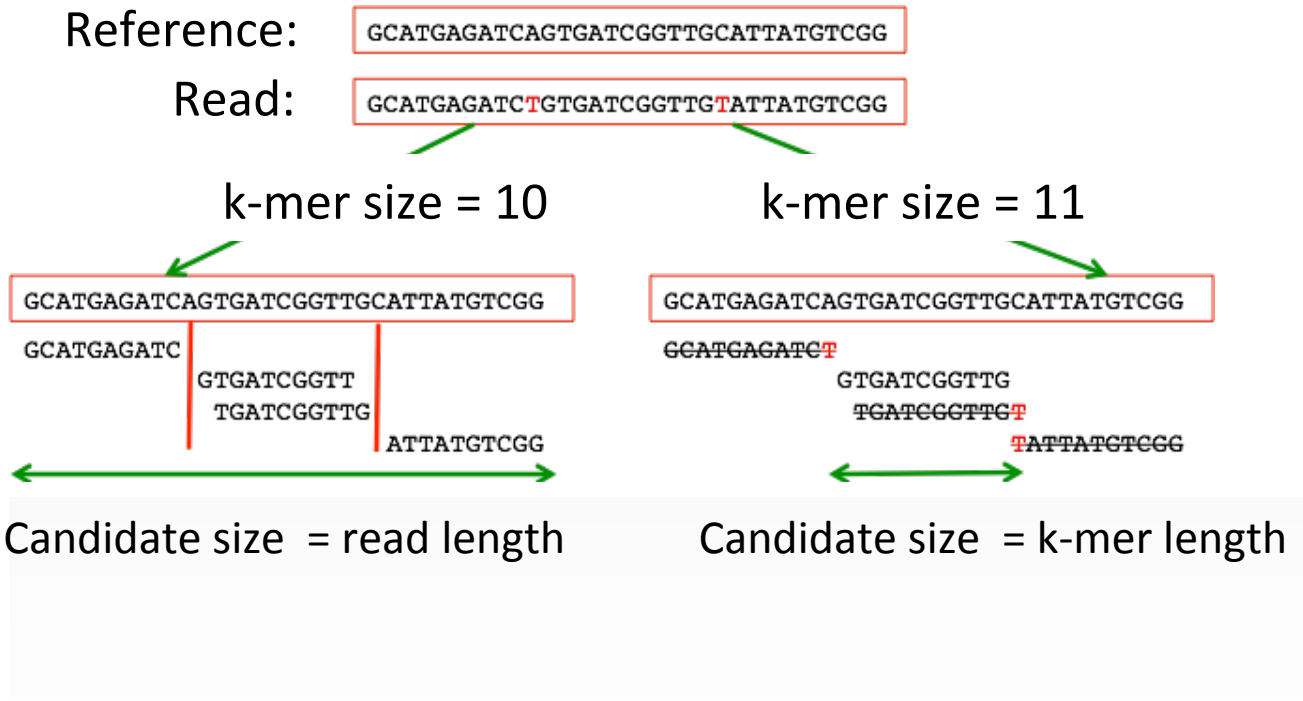
Bowtie1 / Bowtie2
SOAP2
BWA

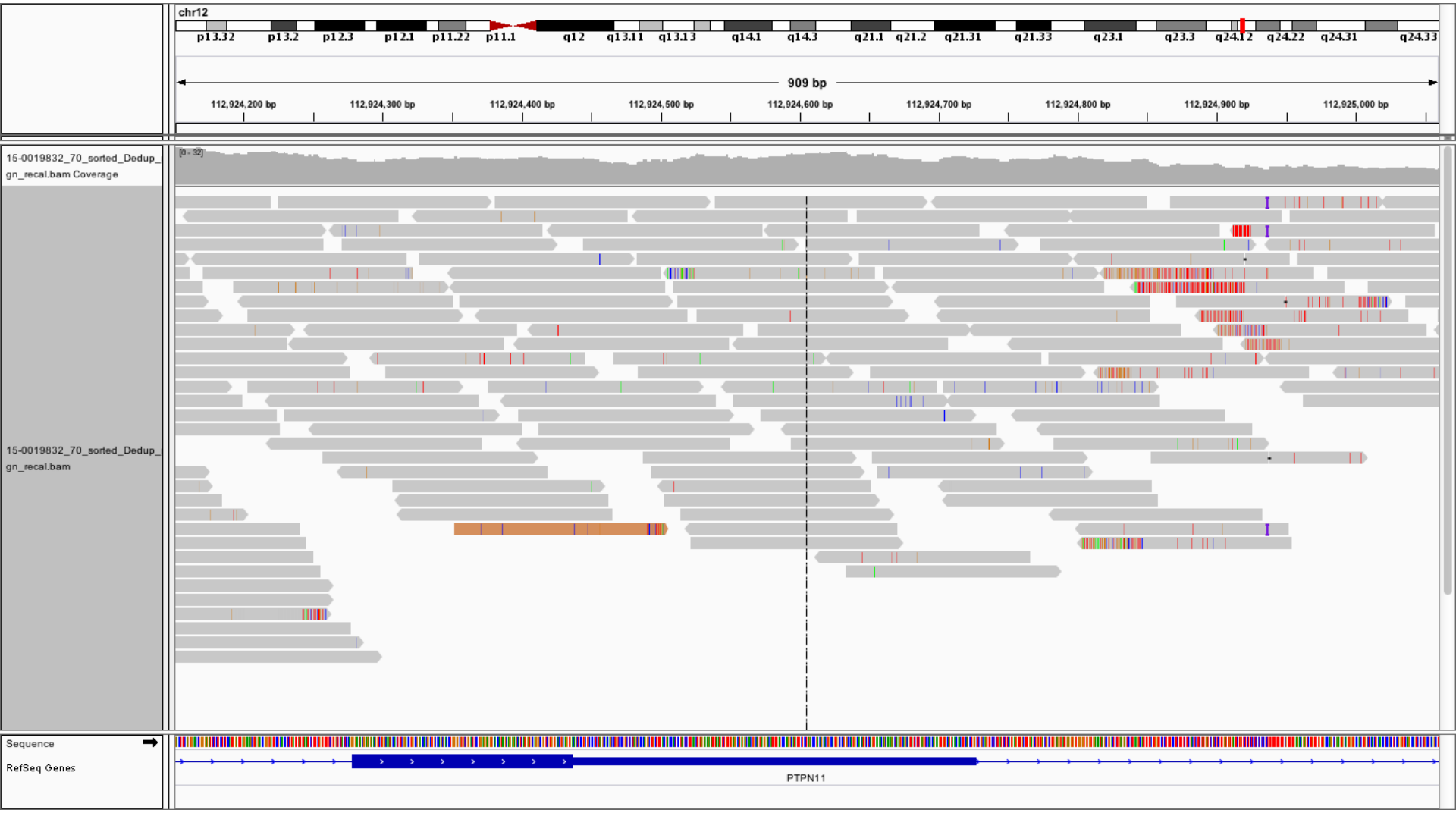
Mapping pros and cons

- Vast majority of sample sequence can be accurately placed
- Problems with:
  - Large scale differences - structural variation
  - Reference bias
  - Repetitive DNA
- How can we address these shortcomings?

Parameters - k-mer size

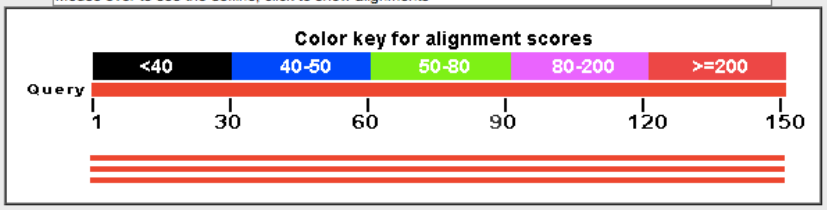
Short reads - choice of k-mer size is important




chr12
p13.32
p13.2
p12.3
p12.1
p11.22
p11.1
q12
q13.11
q13.13
q14.1
q14.3
q21.1
q21.2
q21.31
q21.33
q23.1
q23.3
q24.12
q24.22
q24.31
q24.33
909 bp
112,924,200 bp
112,924,300 bp
112,924,400 bp
112,924,500 bp
112,924,600 bp
112,924,700 bp
112,924,800 bp
112,924,900 bp
112,925,000 bp
15-0019832_70_sorted_Dedup_gn_recal.bam Coverage
[0 - 32]
15-0019832_70_sorted_Dedup_gn_recal.bam
Sequence
RefSeq Genes
PTPN11

Distribution of 3 Blast Hits on the Query Sequence

Mouse over to see the defline, click to show alignments



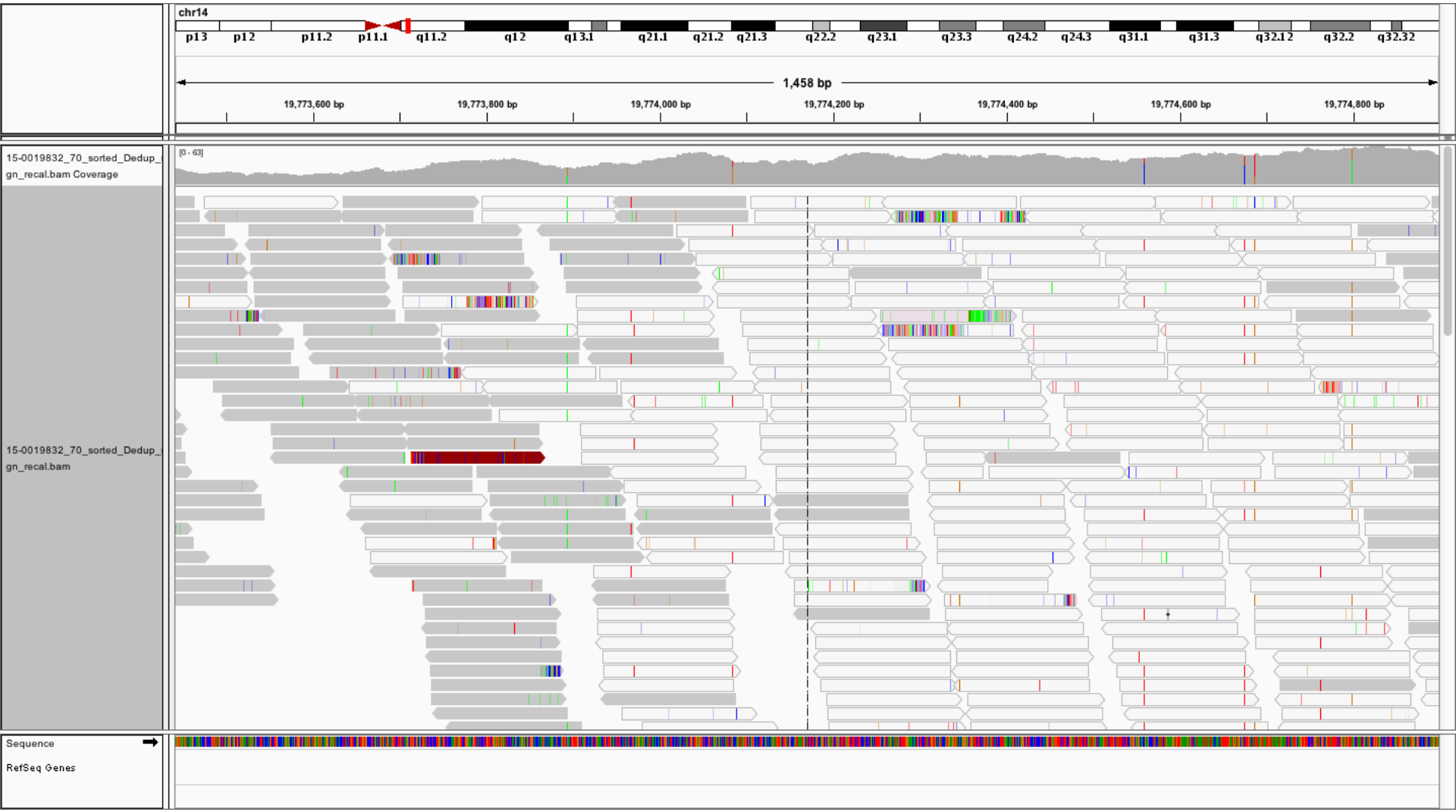
Descriptions

Sequences producing significant alignments:

Select: All None Selected:0

Alignments Download GenBank Graphics Distance tree of results

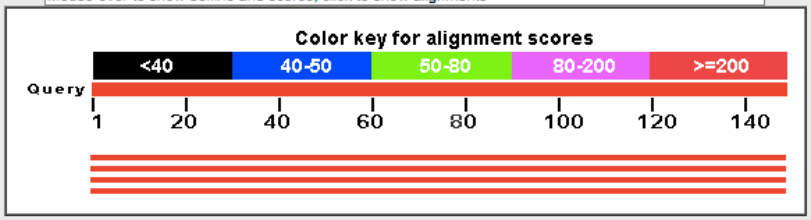
Description	Max score	Total score	Query cover	E value	Ident	Accession
Transcripts						
Homo sapiens protein tyrosine phosphatase, non-receptor type 11 (PTPN11), transcript variant 2, mRNA	279	279	100%	4e-73	100%	NM_080601.1
Genomic sequences [show first]						
Homo sapiens chromosome 12, alternate assembly CHM1_1.1	279	279	100%	4e-73	100%	NC_018923.2
Homo sapiens chromosome 12, GRCh38.p2 Primary Assembly	279	279	100%	4e-73	100%	NC_000012.12


chr14
p13
p12
p11.2
p11.1
q11.2
q12
q13.1
q21.1
q21.2
q21.3
q22.2
q23.1
q23.3
q24.2
q24.3
q31.1
q31.3
q32.12
q32.2
q32.32
1,458 bp
19,773,600 bp
19,773,800 bp
19,774,000 bp
19,774,200 bp
19,774,400 bp
19,774,600 bp
19,774,800 bp
15-0019832_70_sorted_Dedup_
gn_recal.bam Coverage
[0 - 63]
15-0019832_70_sorted_Dedup_
gn_recal.bam
Sequence
RefSeq Genes

Graphic Summary

Distribution of 4 Blast Hits on the Query Sequence

Mouse-over to show defline and scores, click to show alignments



Descriptions

Sequences producing significant alignments:

Select: All None Selected:0

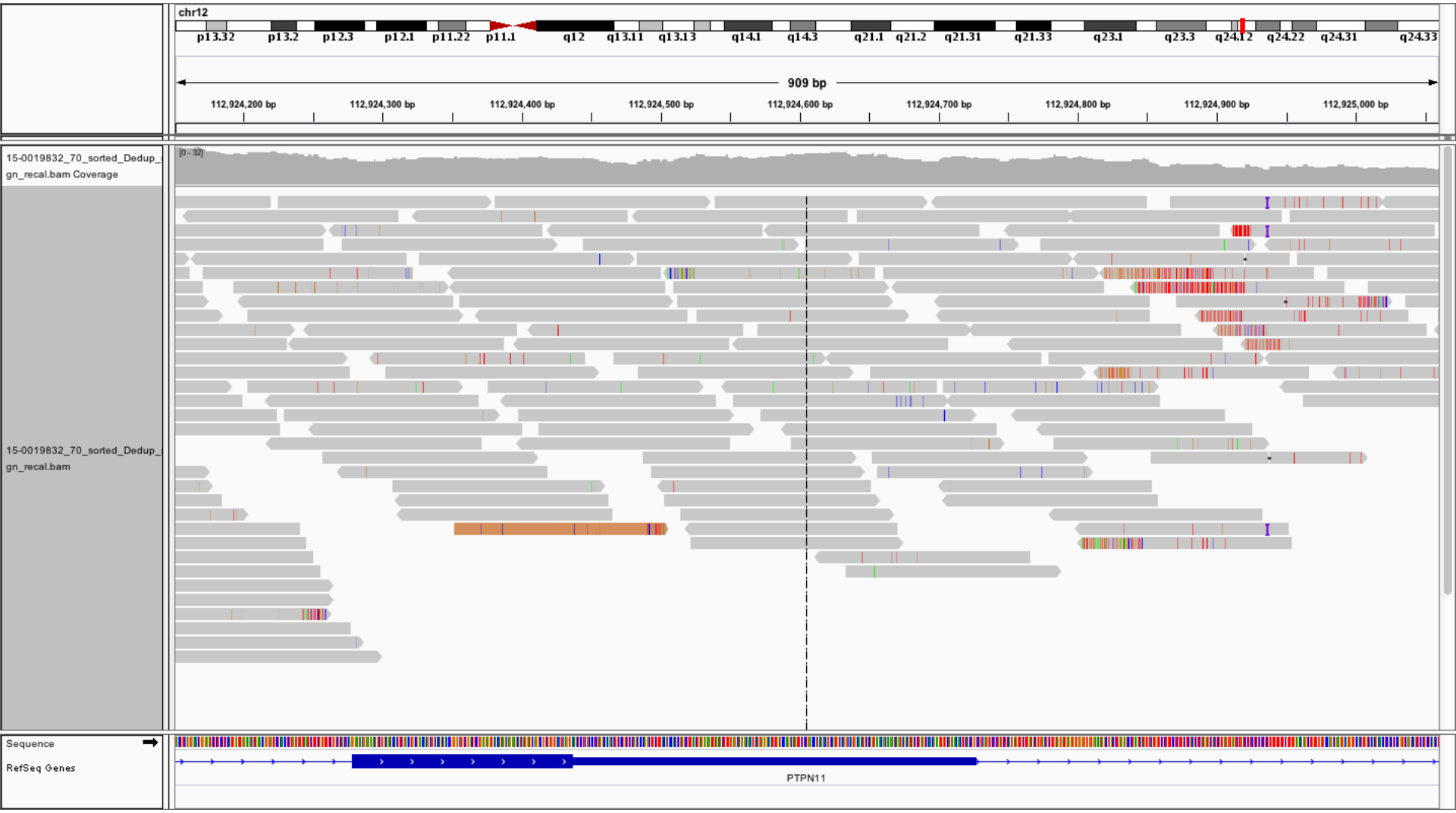
Alignments Download GenBank Graphics Distance tree of results

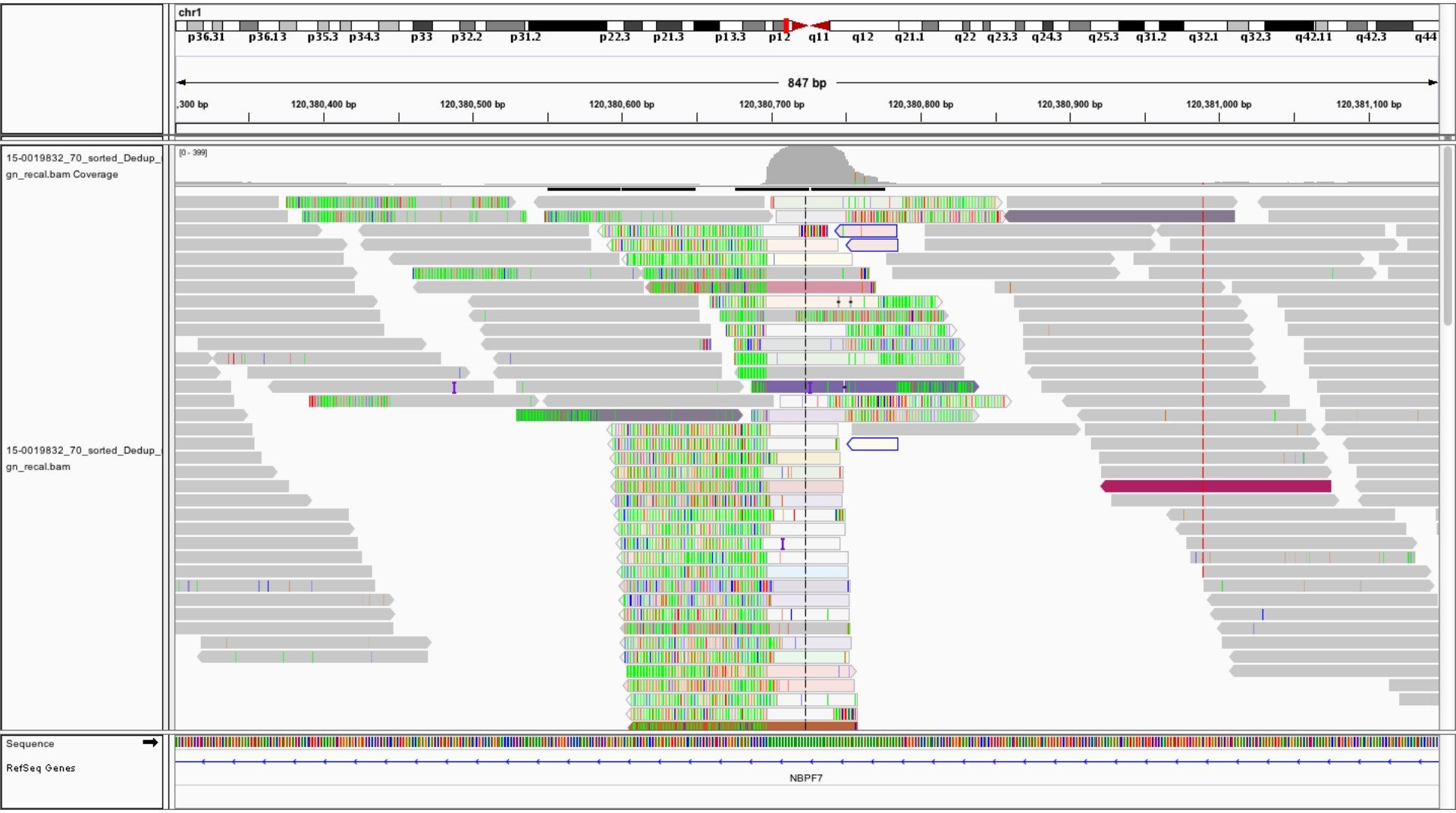
	Description	Max score	Total score	Query cover	E value	Ident	Accession
	Genomic sequences						
☐	Homo sapiens chromosome 14, alternate assembly CHM1_1.1	269	269	100%	4e-70	100%	NC_018925.2
☐	Homo sapiens chromosome 22, alternate assembly CHM1_1.1	269	269	100%	4e-70	100%	NC_018933.2
☐	Homo sapiens chromosome 14, GRCh38.p2 Primary Assembly	269	269	100%	4e-70	100%	NC_000014.9
☐	Homo sapiens chromosome 22, GRCh38.p2 Primary Assembly	269	269	100%	4e-70	100%	NC_000022.11

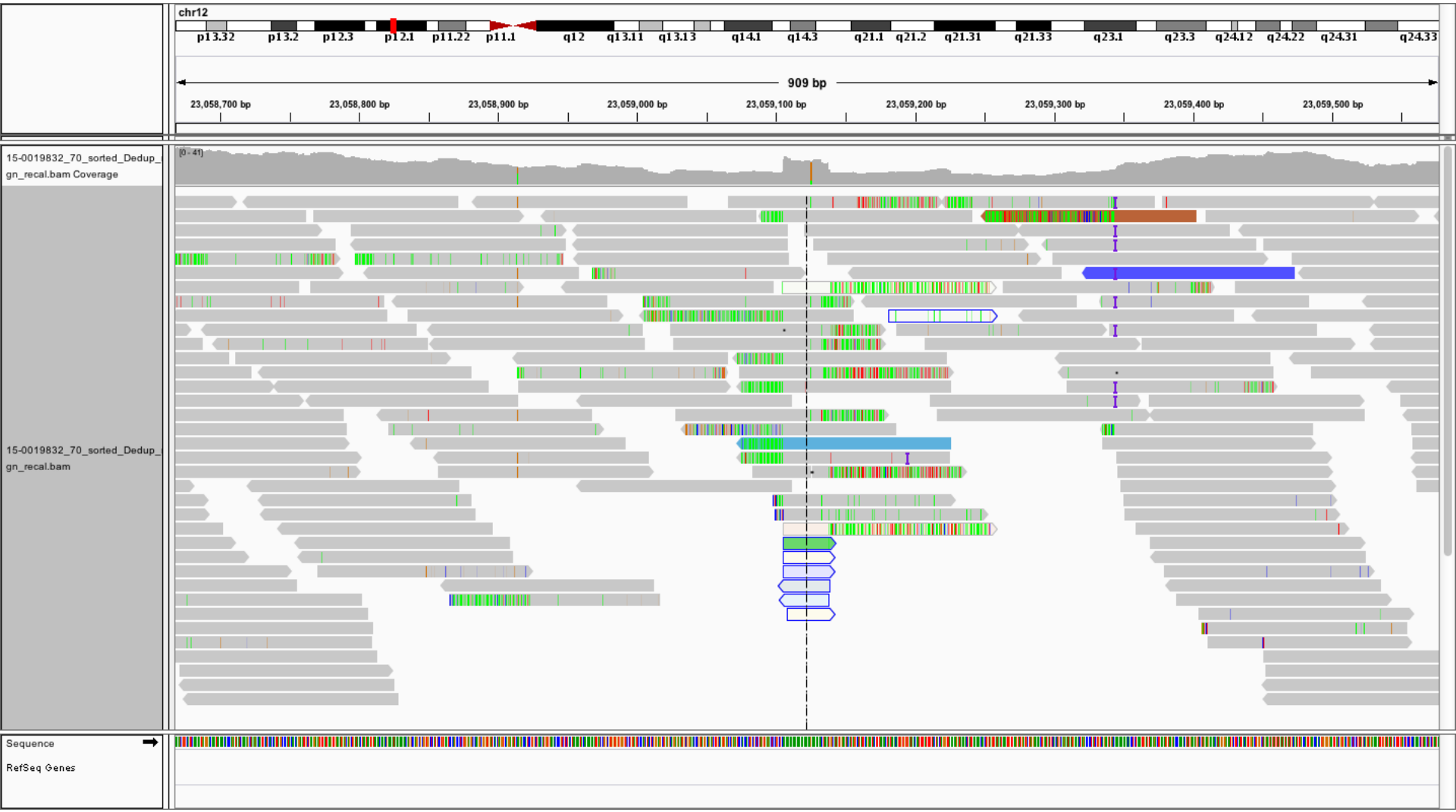
Alignments

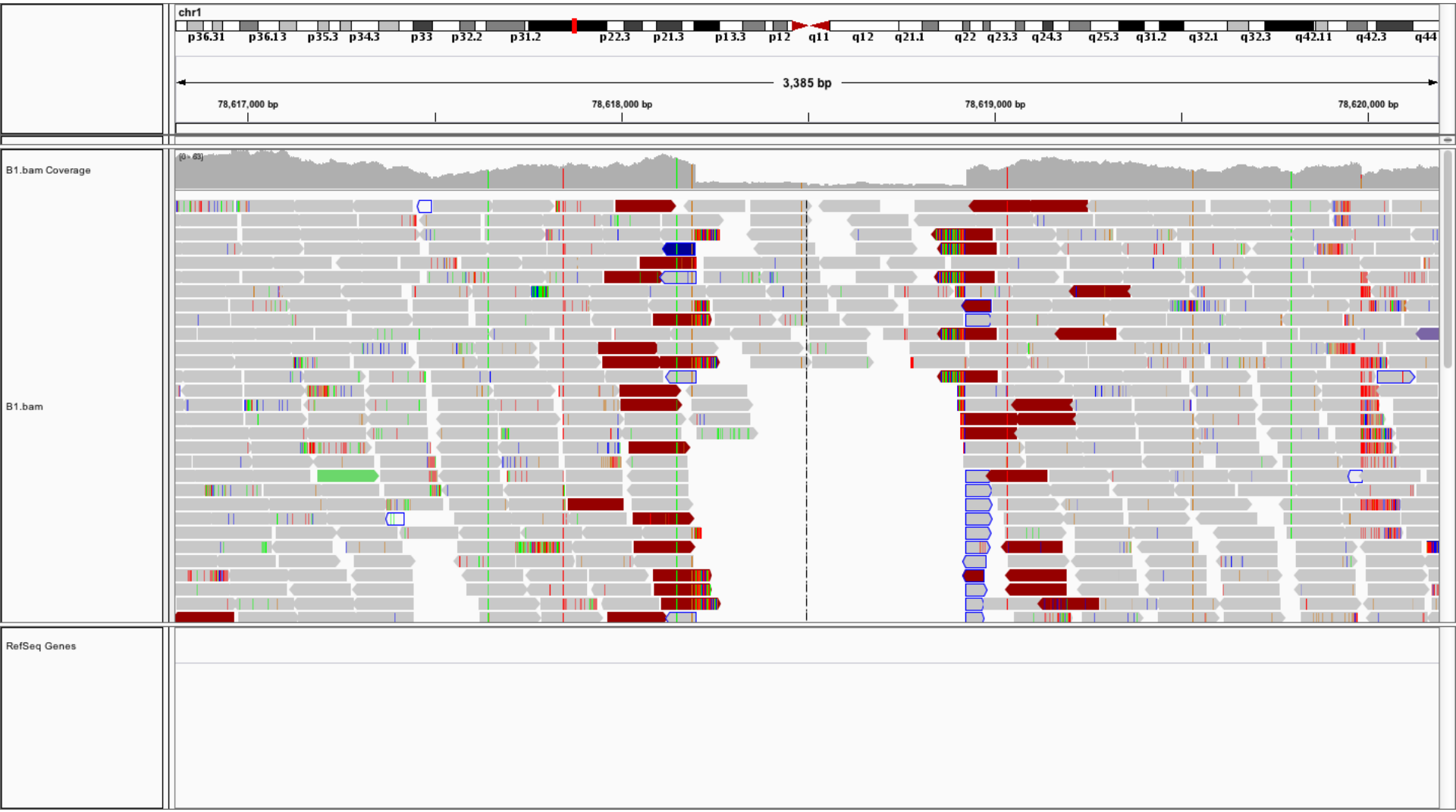
$$MAPPING\_QUALITY = -log_{10}\left(1.0 - \frac{10^{-SUM\_BASE\_Q(best)}}{\sum_i 10^{-SUM\_BASE\_Q(i)}}\right)$$

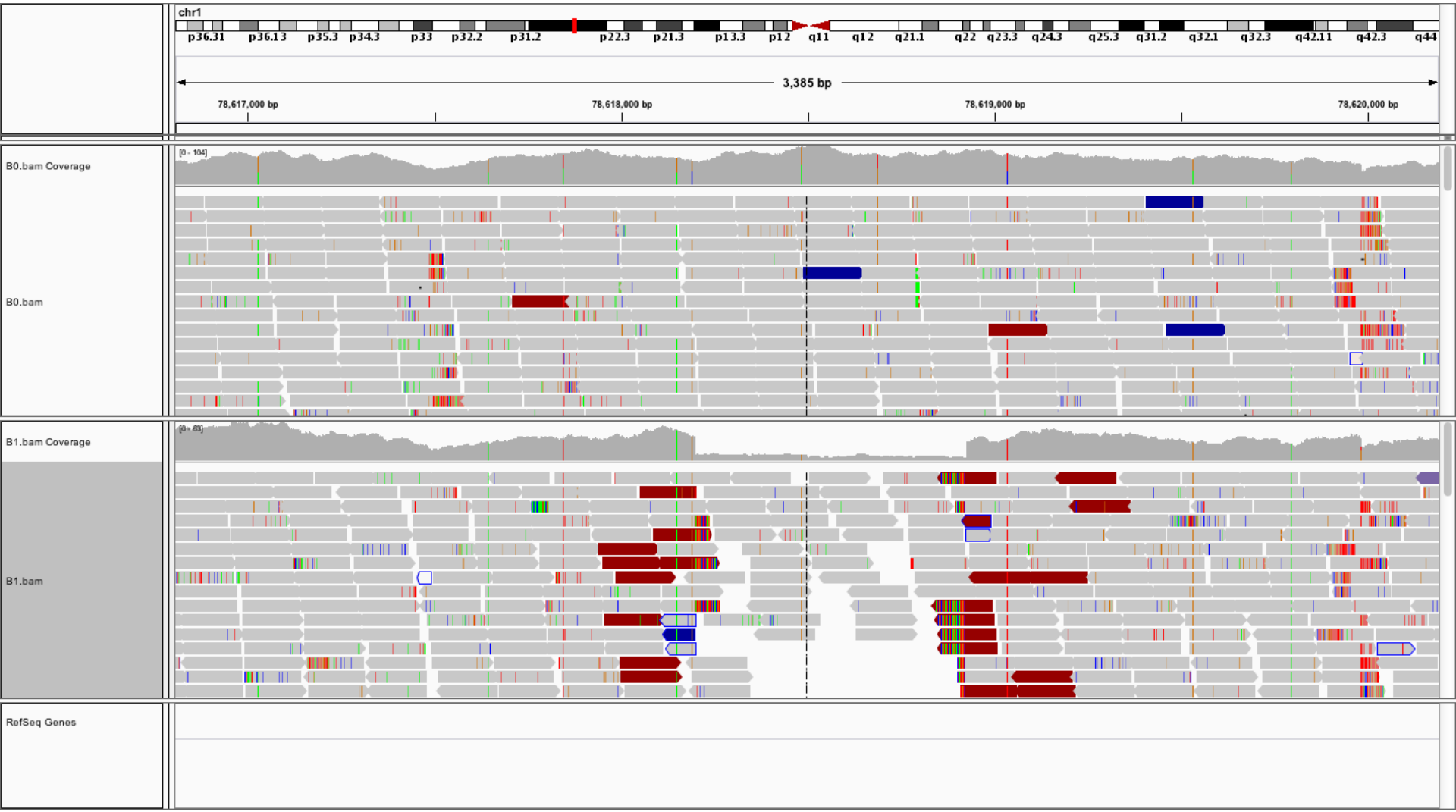
- For a particular short sequence read, consider its best alignment in the genome. For this alignment, calculate the sum of base quality scores at mismatched bases and define a quantity *SUM_BASE_Q(best)*. Also, consider all other possible alignments for the read. For the alignment *i*, define *SUM_BASE_Q(i)* as the sum of base quality scores at mismatched bases for that alignment.
-


chr12
p13.32
p13.2
p12.3
p12.1
p11.22
p11.1
q12
q13.11
q13.13
q14.1
q14.3
q21.1
q21.2
q21.31
q21.33
q23.1
q23.3
q24.12
q24.22
q24.31
q24.33
909 bp
112,924,200 bp
112,924,300 bp
112,924,400 bp
112,924,500 bp
112,924,600 bp
112,924,700 bp
112,924,800 bp
112,924,900 bp
112,925,000 bp
15-0019832_70_sorted_Dedup_gn_recal.bam Coverage
[0 - 32]
15-0019832_70_sorted_Dedup_gn_recal.bam
Sequence
RefSeq Genes
PTPN11

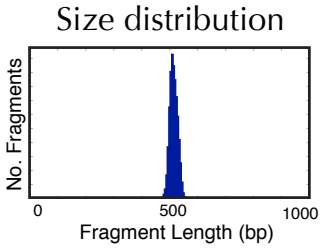

chr1
p36.31
p36.13
p35.3
p34.3
p33
p32.2
p31.2
p22.3
p21.3
p13.3
p12
q11
q12
q21.1
q22
q23.3
q24.3
q25.3
q31.2
q32.1
q32.3
q42.11
q42.3
q44
847 bp
,300 bp
120,380,400 bp
120,380,500 bp
120,380,600 bp
120,380,700 bp
120,380,800 bp
120,380,900 bp
120,381,000 bp
120,381,100 bp
15-0019832_70_sorted_Dedup_
gn_recal.bam Coverage
[0 - 399]
15-0019832_70_sorted_Dedup_
gn_recal.bam
Sequence
RefSeq Genes
NBPF7


chr12
p13.32
p13.2
p12.3
p12.1
p11.22
p11.1
q12
q13.11
q13.13
q14.1
q14.3
q21.1
q21.2
q21.31
q21.33
q23.1
q23.3
q24.12
q24.22
q24.31
q24.33
909 bp
23,058,700 bp
23,058,800 bp
23,058,900 bp
23,059,000 bp
23,059,100 bp
23,059,200 bp
23,059,300 bp
23,059,400 bp
23,059,500 bp
15-0019832_70_sorted_Dedup_
gn_recal.bam Coverage
[0 - 41]
15-0019832_70_sorted_Dedup_
gn_recal.bam
Sequence
RefSeq Genes


chr1
p36.31
p36.13
p35.3
p34.3
p33
p32.2
p31.2
p22.3
p21.3
p13.3
p12
q11
q12
q21.1
q22
q23.3
q24.3
q25.3
q31.2
q32.1
q32.3
q42.11
q42.3
q44
3,385 bp
78,617,000 bp
78,618,000 bp
78,619,000 bp
78,620,000 bp
B1.bam Coverage
B1.bam
RefSeq Genes


chr1
p36.31
p36.13
p35.3
p34.3
p33
p32.2
p31.2
p22.3
p21.3
p13.3
p12
q11
q12
q21.1
q22
q23.3
q24.3
q25.3
q31.2
q32.1
q32.3
q42.11
q42.3
q44
3,385 bp
78,617,000 bp
78,618,000 bp
78,619,000 bp
78,620,000 bp
B0.bam Coverage
[0 - 104]
B0.bam
B1.bam Coverage
[0 - 83]
B1.bam
RefSeq Genes

Single-end alignment



0. Size-selected DNA library (e.g., ~500bp)

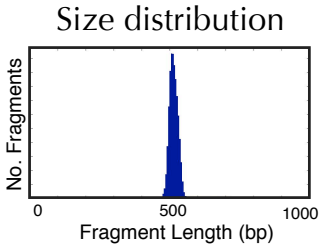
1. Sequence the ***entire length*** of each molecule the library

472bp 499bp 519bp 503bp 561bp

2. Align each ***contiguous*** molecule to the reference genome.

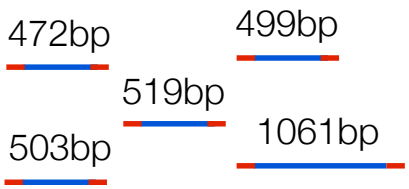
472bp 519bp 503bp 499bp 561bp

Paired-end alignment

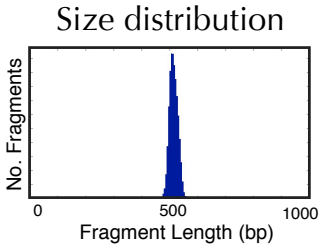


0. Size-selected DNA library (e.g., ~500bp)

1. Sequence ***solely the 5' ends*** of each molecule the library



Paired-end alignment



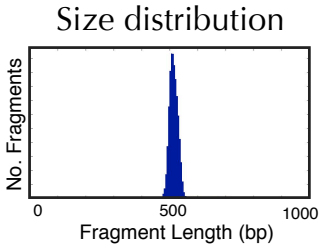
0. Size-selected DNA library (e.g., ~500bp)

1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.

Paired-end alignment

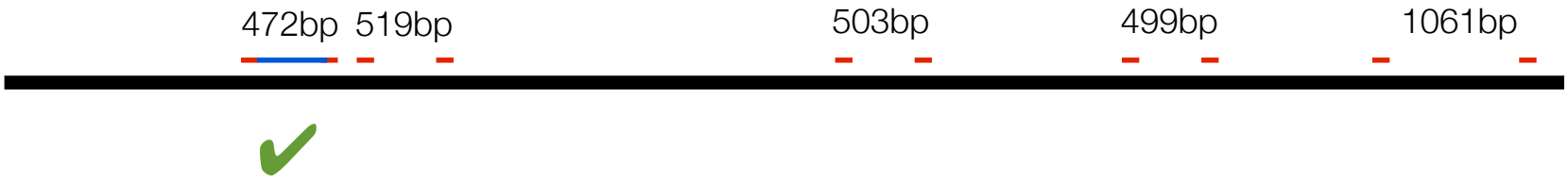


0. Size-selected DNA library (e.g., ~500bp)

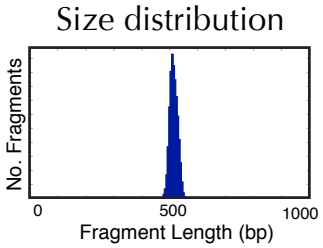
1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.



Paired-end alignment

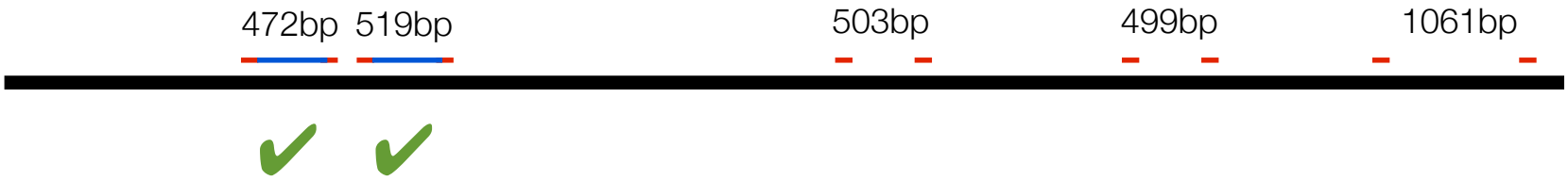


0. Size-selected DNA library (e.g., ~500bp)

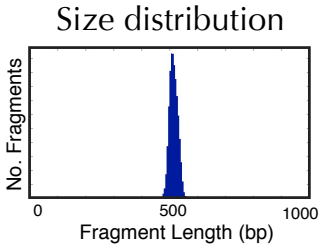
1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.



Paired-end alignment

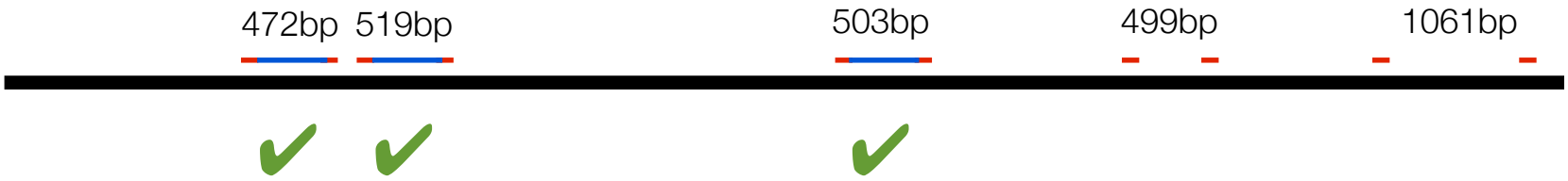


0. Size-selected DNA library (e.g., ~500bp)

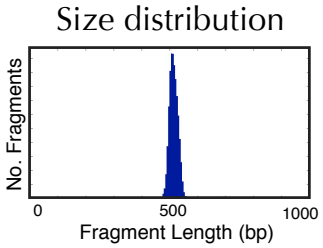
1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.



Paired-end alignment

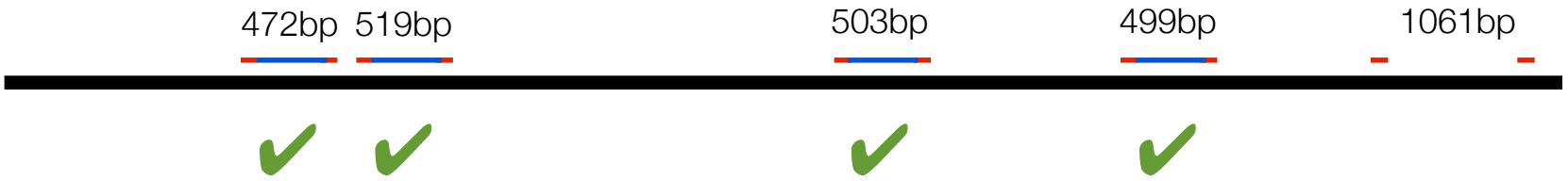


0. Size-selected DNA library (e.g., ~500bp)

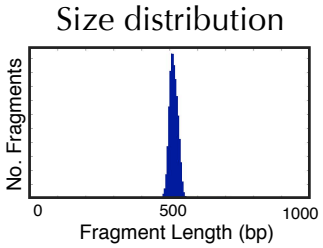
1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.



Paired-end alignment

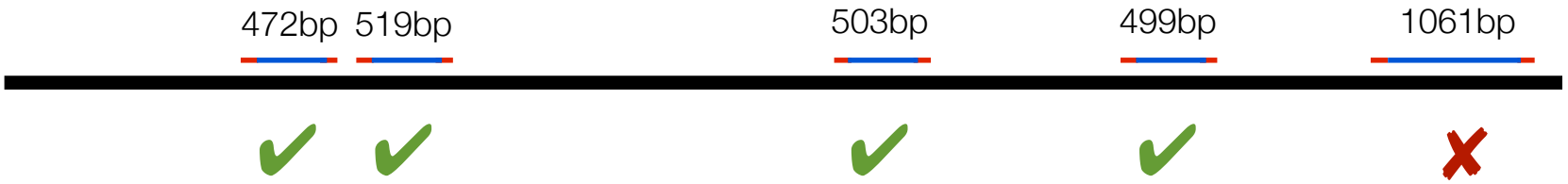


0. Size-selected DNA library (e.g., ~500bp)

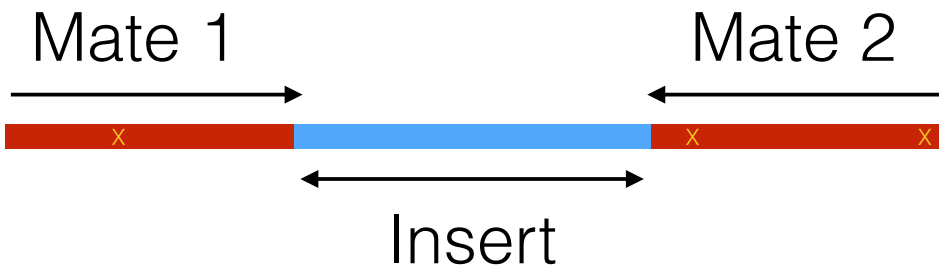
1. Sequence ***solely the 5' ends*** of each molecule the library

472bp
499bp
519bp
503bp
1061bp

2. Separately align ***each end of each*** molecule to the reference genome.



Paired end reads

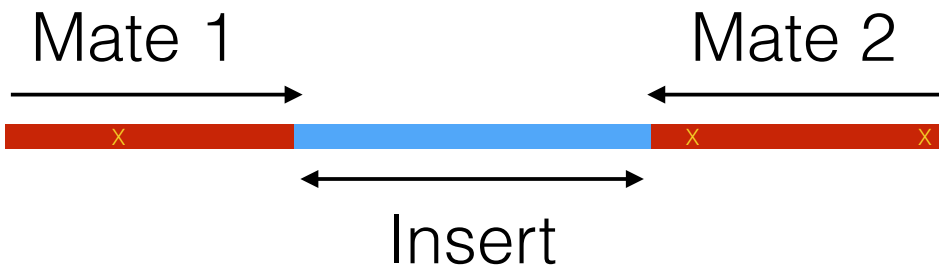


mapped uniquely

mapped to multiple locations

unmapped

Paired end reads



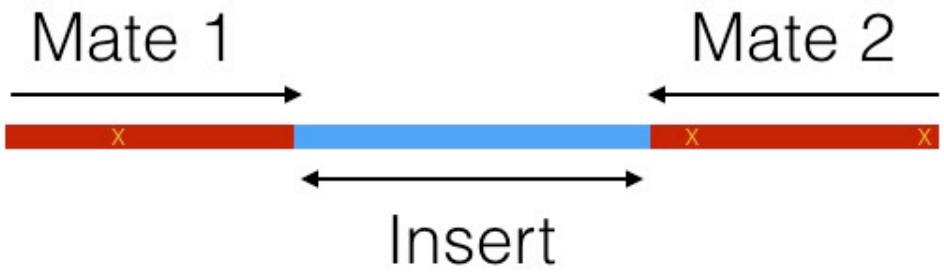
mapped uniquely

mapped to multiple locations

unmapped

Both mates map uniquely

Paired end reads



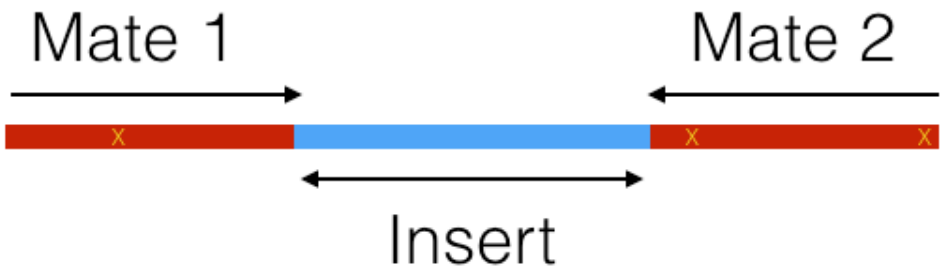
mapped uniquely

mapped to multiple locations

unmapped

One mate maps uniquely, the other is unmapped

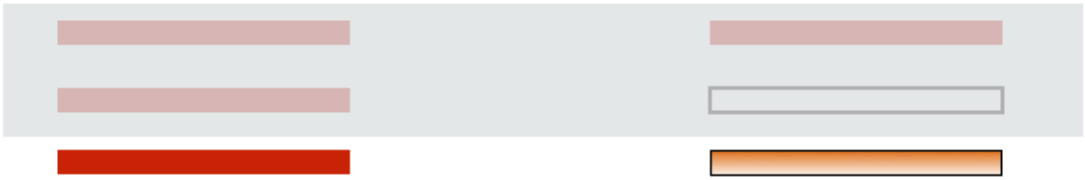
Paired end reads



mapped uniquely

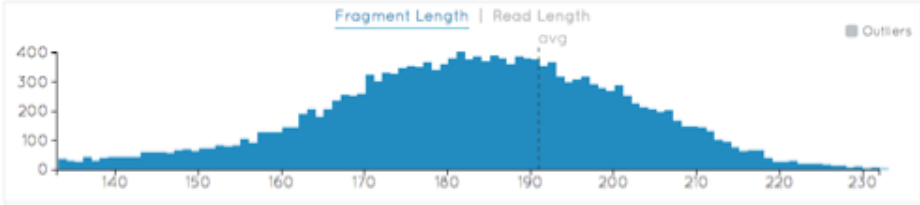
mapped to multiple locations

unmapped

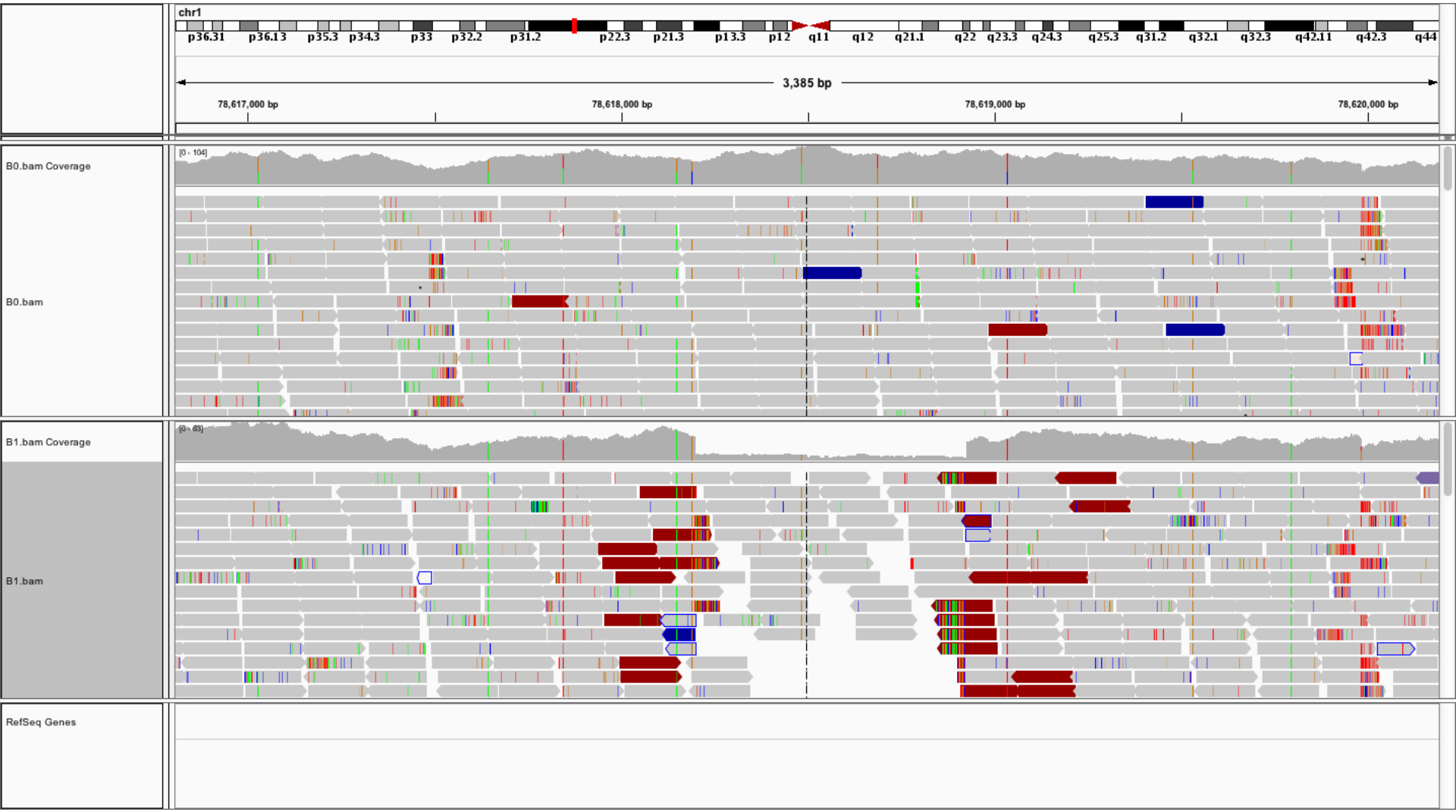


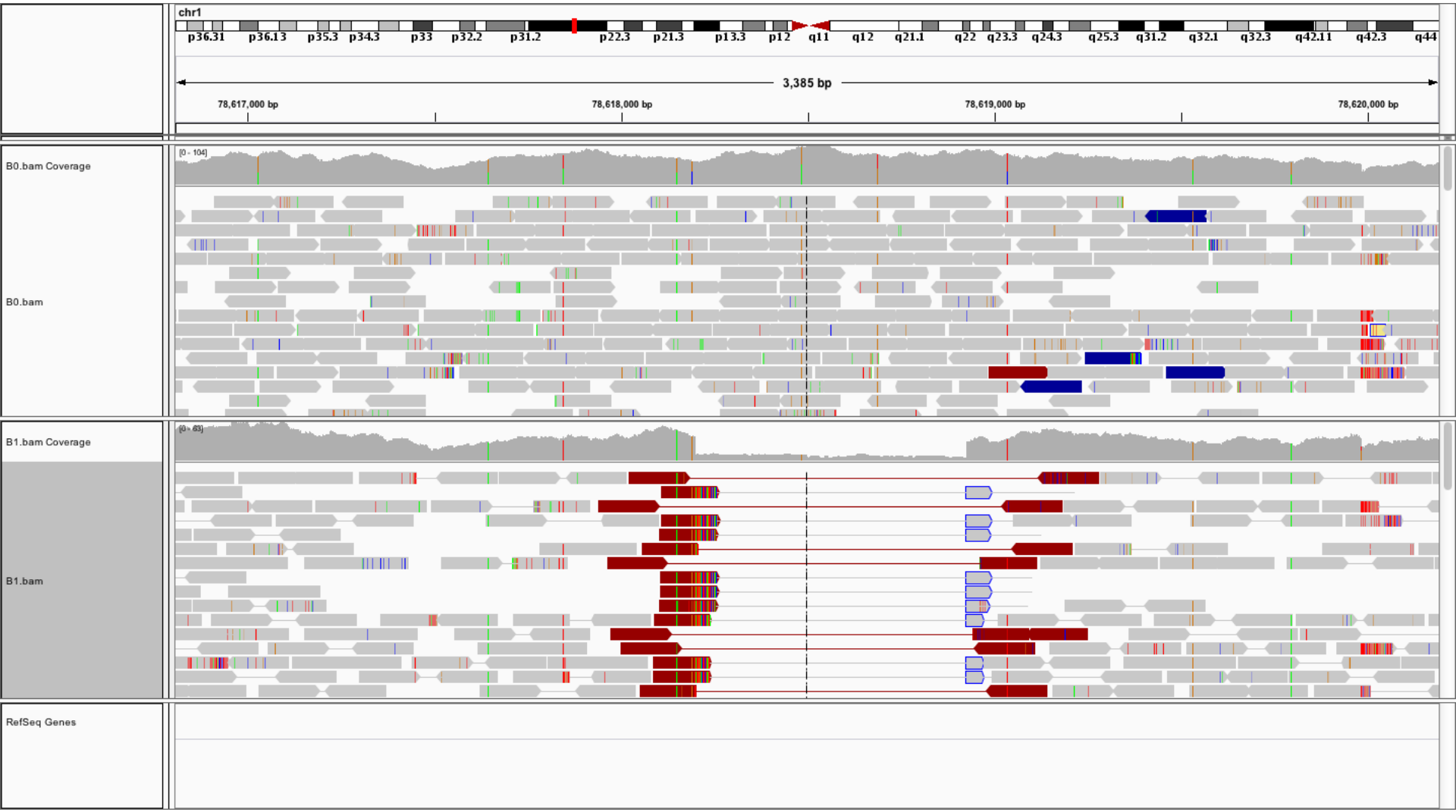
One mate maps uniquely, the other maps to multiple locations

Use fragment length distribution to determine most likely location



bam.iobio.io


chr1
p36.31
p36.13
p35.3
p34.3
p33
p32.2
p31.2
p22.3
p21.3
p13.3
p12
q11
q12
q21.1
q22
q23.3
q24.3
q25.3
q31.2
q32.1
q32.3
q42.11
q42.3
q44
3,385 bp
78,617,000 bp
78,618,000 bp
78,619,000 bp
78,620,000 bp
B0.bam Coverage
[0 - 104]
B0.bam
B1.bam Coverage
[0 - 83]
B1.bam
RefSeq Genes


chr1
p36.31 p36.13 p35.3 p34.3 p33 p32.2 p31.2 p22.3 p21.3 p13.3 p12 q11 q12 q21.1 q22 q23.3 q24.3 q25.3 q31.2 q32.1 q32.3 q42.11 q42.3 q44
3,385 bp
78,617,000 bp
78,618,000 bp
78,619,000 bp
78,620,000 bp
B0.bam Coverage
[0 - 104]
B0.bam
B1.bam Coverage
[0 - 83]
B1.bam
RefSeq Genes

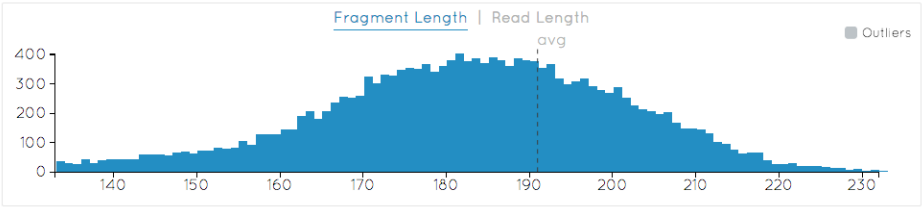
Split read mapping

mate 1

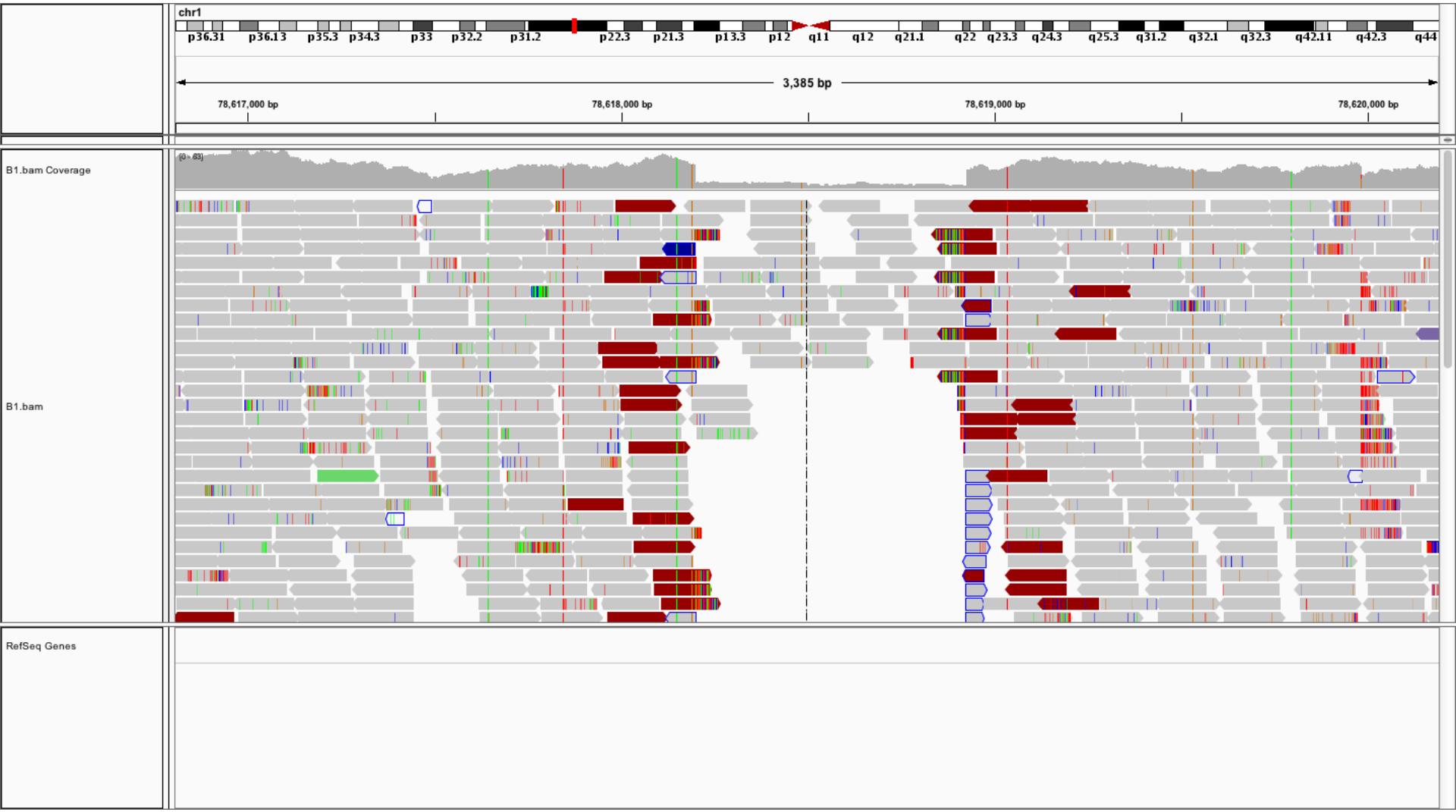
mate 2

Unmapped

Anchor - uniquely mapped read

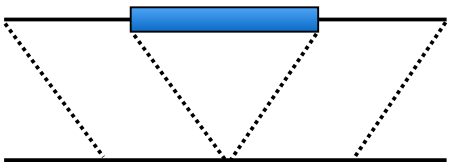


Estimate of fragment length - we have an idea of where to look


chr1
p36.31
p36.13
p35.3
p34.3
p33
p32.2
p31.2
p22.3
p21.3
p13.3
p12
q11
q12
q21.1
q22
q23.3
q24.3
q25.3
q31.2
q32.1
q32.3
q42.11
q42.3
q44
3,385 bp
78,617,000 bp
78,618,000 bp
78,619,000 bp
78,620,000 bp
B1.bam Coverage
B1.bam
RefSeq Genes

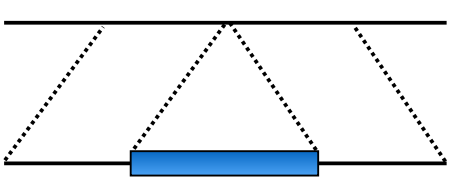
SV types

Deletion



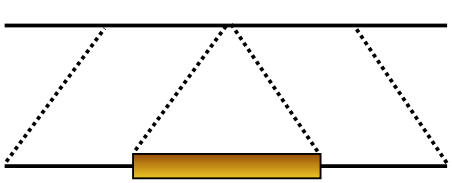
If inserted sequence/deletion ≦ 50bp
-> indel

Novel insertion



sequence ≧ 50bp
-> Copy number variation

Mobile element insertion



The mobile element sequence is ubiquitous in the genome

Aligner	Strategy	Availability
BFAST	hash-based (genome)	open-source
Bowtie2*	Suffix array / BWT	open-source
BWA*	Suffix array / BWT	open-source
MAQ	hash-based (reads)	open-source
Mosaik*	hash-based (genome)	open-source
Novoalign*	hash-based (genome)	free for academic use
RMAP	hash-based (reads)	open-source
SOAP2	Suffix array / BWT	free for academic use
Eland	hash-based (reads)	commercial (Illumina)

Unresolved problems

- Mapping tries to match the reference, so inherently introduces a bias towards the reference
- We have to modify parameters based on the read content (e.g. deletions)
- Mapping to repetitive DNA is still problematic
- What if there is no or an incomplete reference for the sequenced organism?