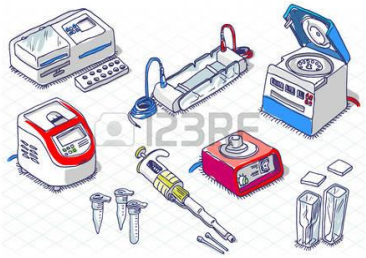
Human Genome Sequencing and Interpretation

Lesson 1 - 20/01/2020
Lesson 2 - 21/01/2020
Lesson 3 - 27/01/2020
(Lesson 4 - 28/01/2020)

Prof. Massimo Delledonne
Functional Genomics lab

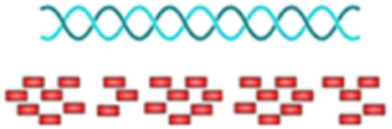
Library preparation



Sequencing

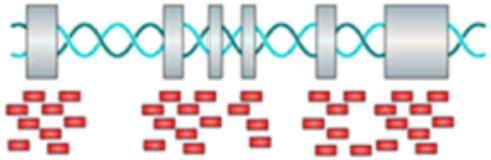
Bioinformatic analysis

Whole genome sequencing



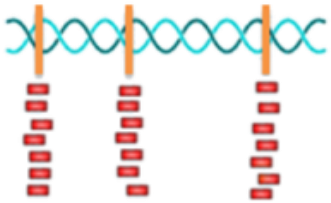
- Sequencing region : whole genome
- Sequencing Depth: >30X
- Covers everything – can identify all kinds of variants including SNPs, INDELs and SV.

Whole exome sequencing



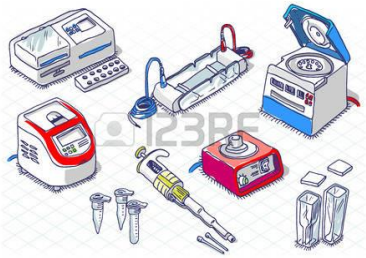
- Sequencing region: whole exome
- Sequencing Depth : >50X ~ 100X
- Identify all kinds of variants including SNPs, INDELs and SV in coding region.
- Cost effective

Targeted sequencing



- Sequencing region: specific regions (could be customized)
- Sequencing Depth : >500X
- Identify all kinds of variants including SNPs, INDELs in specific regions
- Most Cost effective

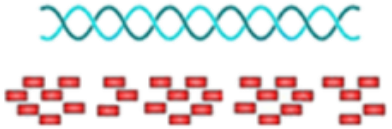
Library preparation



Sequencing

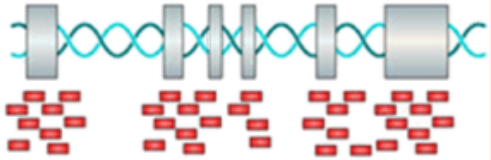
Bioinformatic analysis

Whole genome sequencing



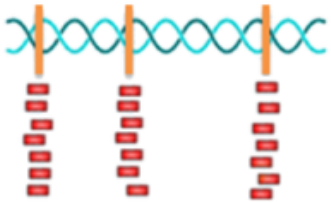
- Sequencing region : whole genome
- Sequencing Depth: >30X
- Covers everything – can identify all kinds of variants including SNPs, INDELs and SV.

Whole exome sequencing



- Sequencing region: whole exome
- Sequencing Depth : >50X ~ 100X
- Identify all kinds of variants including SNPs, INDELs and SV in coding region.
- Cost effective

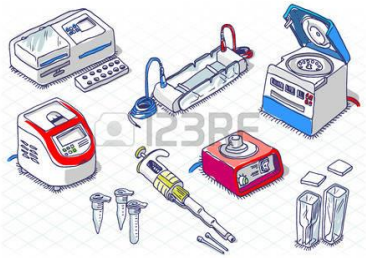
Targeted sequencing



- Sequencing region: specific regions (could be customized)
- Sequencing Depth : >500X
- Identify all kinds of variants including SNPs, INDELs in specific regions
- Most Cost effective

Library preparation

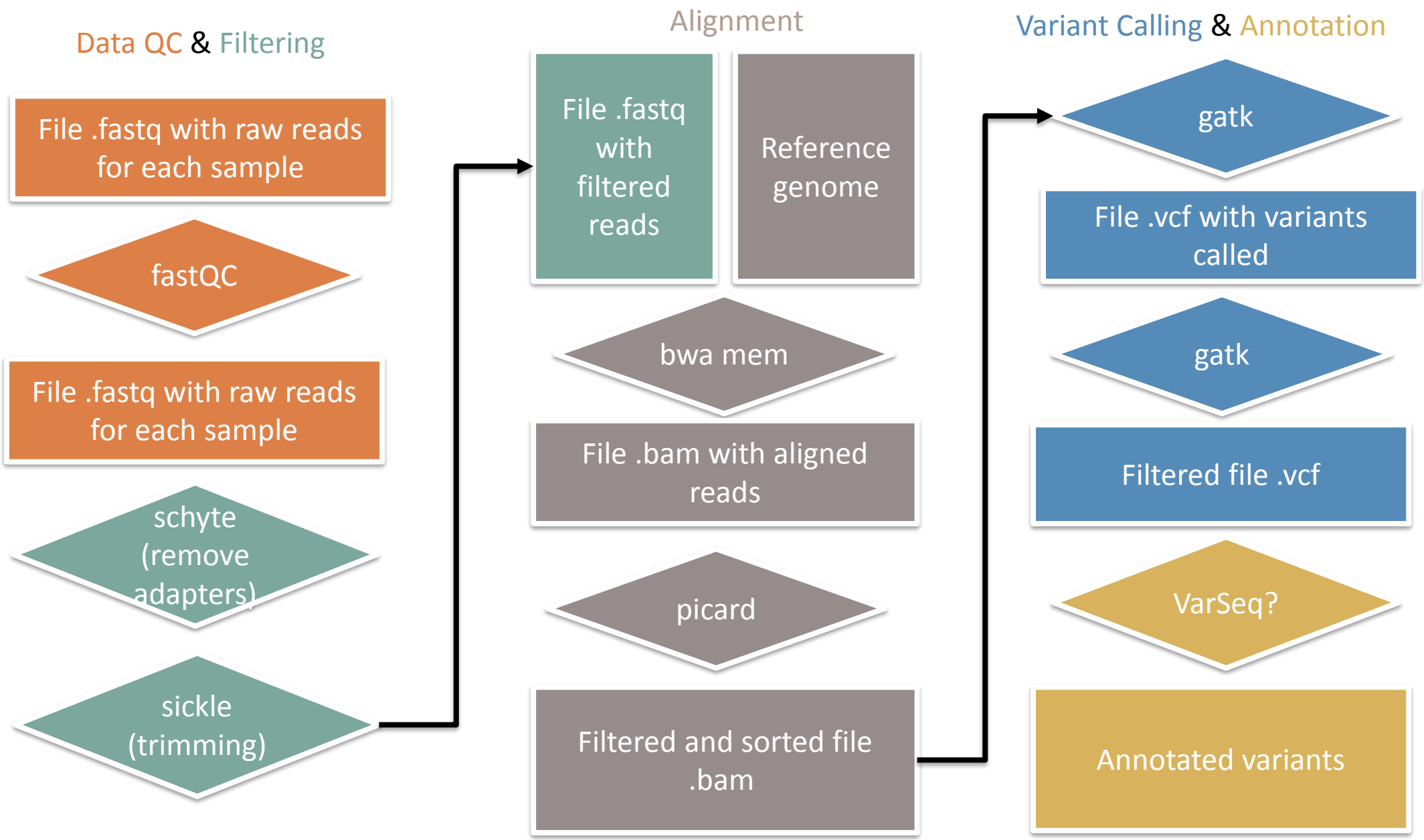
Bioinformatic analysis



Sequencing

Data QC → Alignment → Variant calling → Variant prioritization

Pipeline


Data QC & Filtering
File .fastq with raw reads for each sample
fastQC
File .fastq with raw reads for each sample
schyte (remove adapters)
sickle (trimming)
Alignment
File .fastq with filtered reads
Reference genome
bwa mem
File .bam with aligned reads
picard
Filtered and sorted file .bam
Variant Calling & Annotation
gatk
File .vcf with variants called
gatk
Filtered file .vcf
VarSeq?
Annotated variants

Shell

Windows:

- https://mobaxterm.mobatek.net/download.html

MAC & Linux:

- Open terminal

Connect to the server

1. Enter in the server:
   a. ssh HGSI2020@157.27.26.214
   b. Password: hgsi2020
2. Enter in the folder: cd /attachedvolume/HGSI2020
3. Create your folder: mkdir your_name
4. Enter in the created folder: cd your_name

Work on the server

1. Create a symbolic link of the files in your folder:

   ln -s ../example/samples/1351S/R*.fastq.gz .

2. Check you have copied the files: ls

3. Open the file to see what is inside:

   less R1.fastq.gz

4. Close the visualization: q

.fq / .fastq file

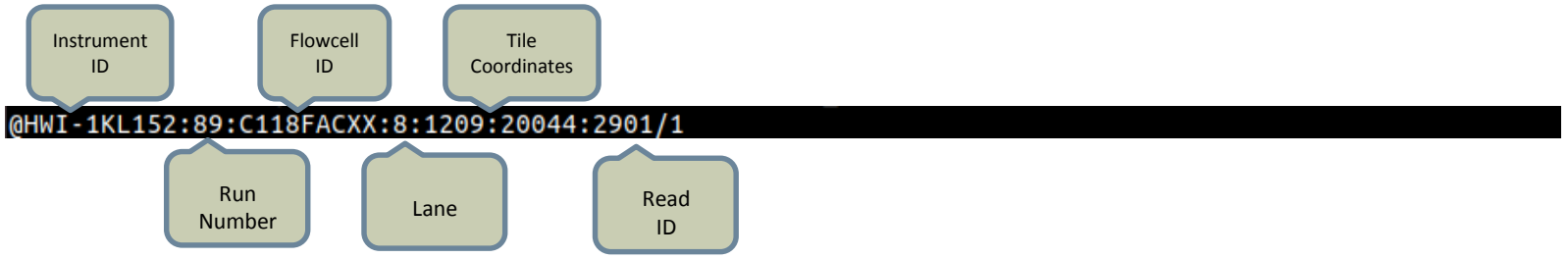
For each sample we obtain 2 fastq files containing all the sequences generated

.fq / .fastq file

Each reads is represented by four rows...

```
@HWI-1KL152:89:C118FACXX:8:1209:20044:2901/1
CTCTGTGGCTGGGAGAGGAGTCTGGGGGGGCCCCGGGCGCCAGCCAGGGATAGCCTGATCTCTGCTCCAGTCGACAGATCCTTAACGGATTTTCTTTCTCT
+
/%53@DDAFGIDCCD;EBCE:C>DDCBB&:9<FD5=02&4?8=<CCCA<BB>FFDD=B@C98,>7:<9-9/0&74>CEBEHHBCDHA?=;&A9DHB#####
```

First row identify the sequence:



Second row contains the sequence:

```
CTCTGTGGCTGGGAGAGGAGTCTGGGGGGGCCCCGGGCGCCAGCCAGGGATAGCCTGATCTCTGCTCCAGTCGACAGATCCTTAACGGATTTTCTTTCTCT
```

Thirs row contains a delimiter:

```
+
```

Fourth row indicate the quality of each base:

```
CCTGTGGCTGGGAGAGGAGTCTGGGGGGGCCCCGGGCGCCAGCCAGGGATAGCCTGATCTCTGCTCCAGTCGACAGATCCTTAACGGATTTTCTTTCTCT
/%53@DDAFGIDCCD;EBCE:C>DDCBB&:9<FD5=02&4?8=<CCCA<BB>FFDD=B@C98,>7:<9-9/0&74>CEBEHHBCDHA?=;&A9DHB#####
```

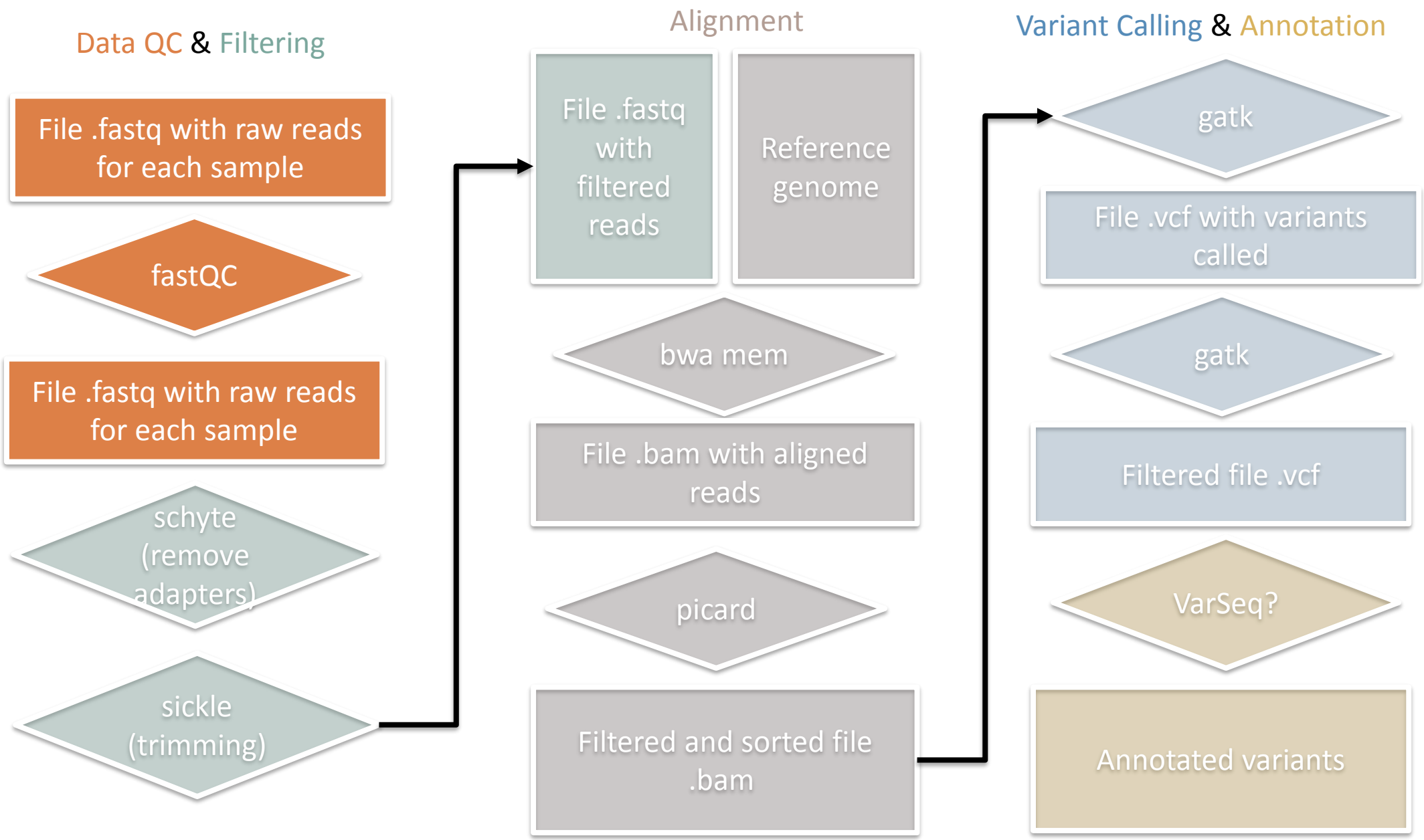
Q score as ASCII chars: "/" = 47

ASCII CODE

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Pipeline


Data QC & Filtering
File .fastq with raw reads for each sample
fastQC
File .fastq with raw reads for each sample
schyte (remove adapters)
sickle (trimming)
Alignment
File .fastq with filtered reads
Reference genome
bwa mem
File .bam with aligned reads
picard
Filtered and sorted file .bam
Variant Calling & Annotation
gatk
File .vcf with variants called
gatk
Filtered file .vcf
VarSeq?
Annotated variants

Check quality of your fastq files with fastQC

Fastqc command

1. In your folder, create a folder for fastqc output:

```
mkdir fastqc
```

2. Launch fastQC on both files:

```
fastqc R*.fastq.gz -o fastqc
```

Fastqc command

On the server, we don't have a graphical vision, so..

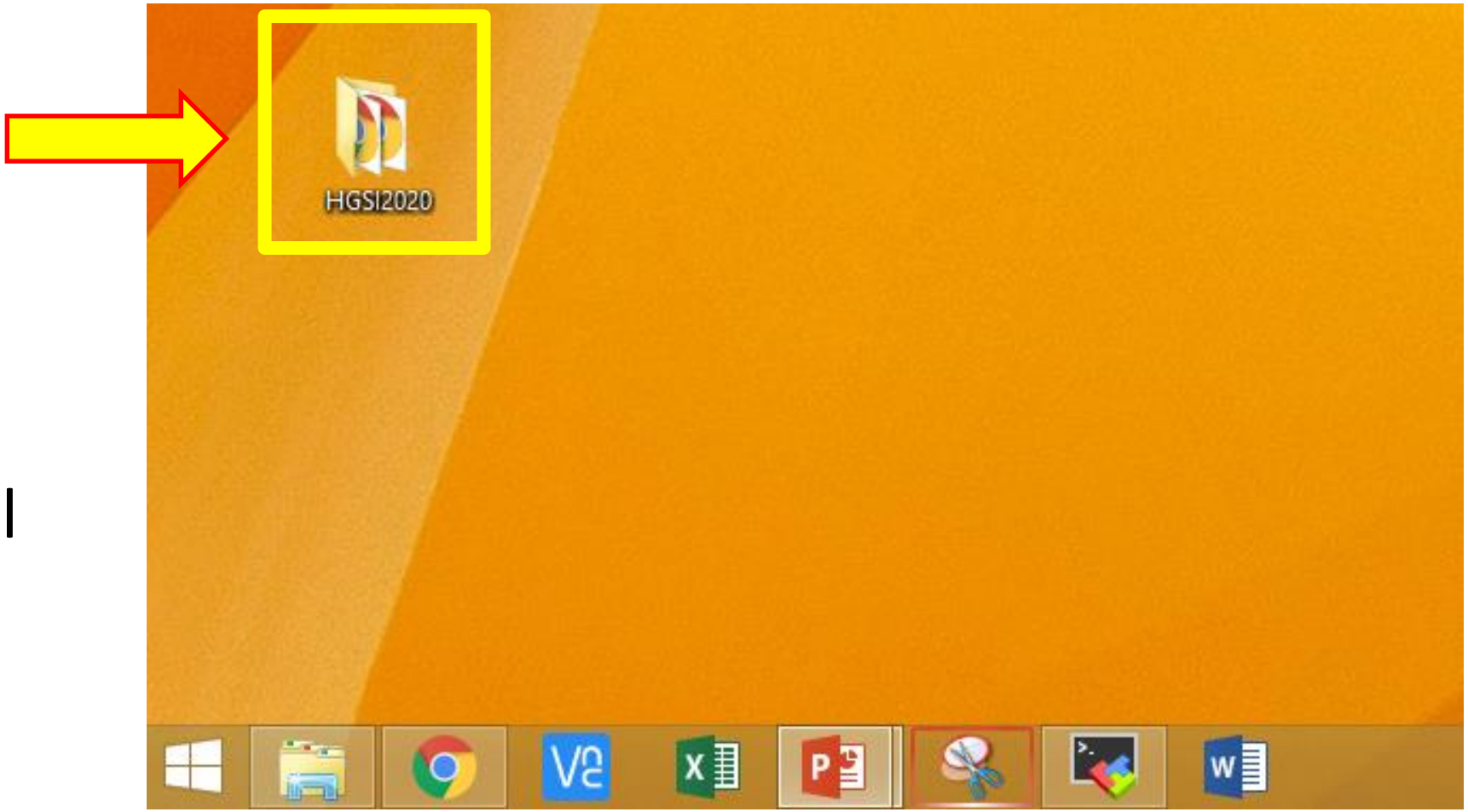
1. Open a new terminal
2. Create a folder on your PC for the course: mkdir Desktop/HGSI2020
3. Enter in the folder: cd Desktop/HGSI2020
4. Download the results here:

   rsync -auv HGSI2020@157.27.26.214:/attachedvolume/HGSI2020/Denise/fastqc/R*_fastqc.html .

   Pass: hgsi2020
5. Check you have downloaded: ls
6. Close the shell

Download the files

1. On your desktop, open the file «HGSI2020»



2. Open the file html

FastQC

FastQC software allows to do some quality control checks on raw sequence data coming from high throughput sequencing.

HTML Report

FastQC Report

Thu 21 Dec 2017
good_sequence_short.txt

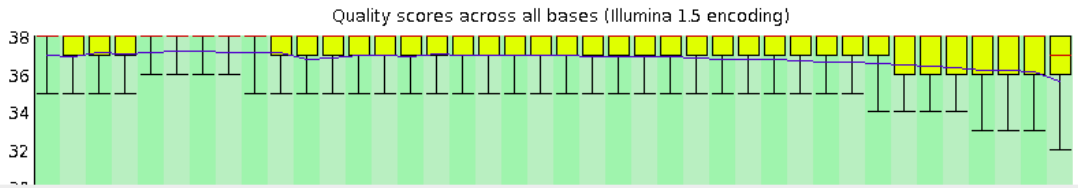
Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

Basic Statistics

Measure	Value
Filename	good_sequence_short.txt
File type	Conventional base calls
Encoding	Illumina 1.5
Total Sequences	250000
Sequences flagged as poor quality	0
Sequence length	40
%GC	45

Per base sequence quality



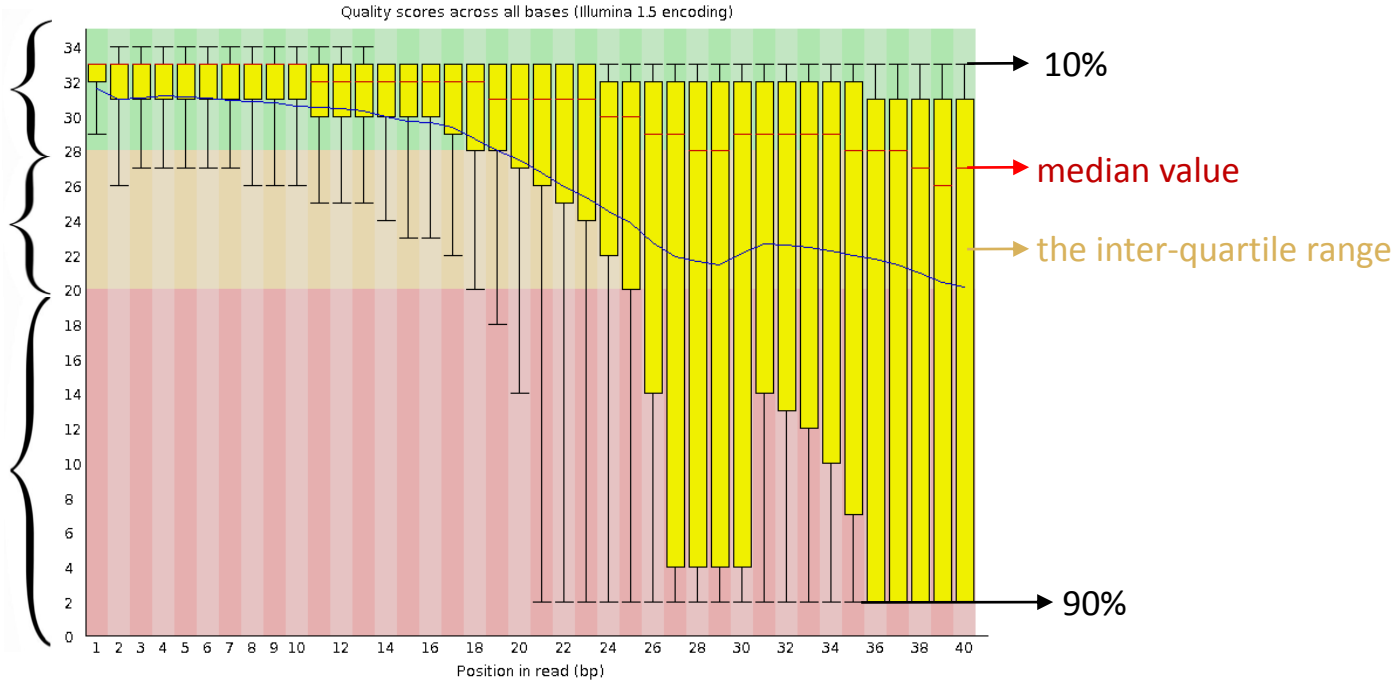
Produced by FastQC (version 0.11.6)

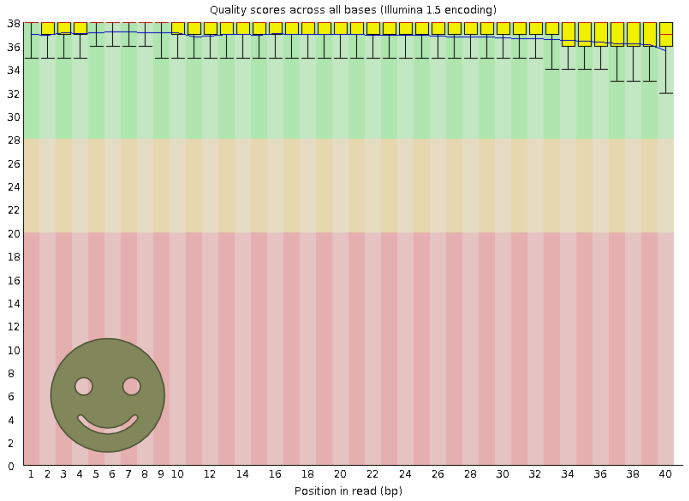
FastQC – Per base sequence quality

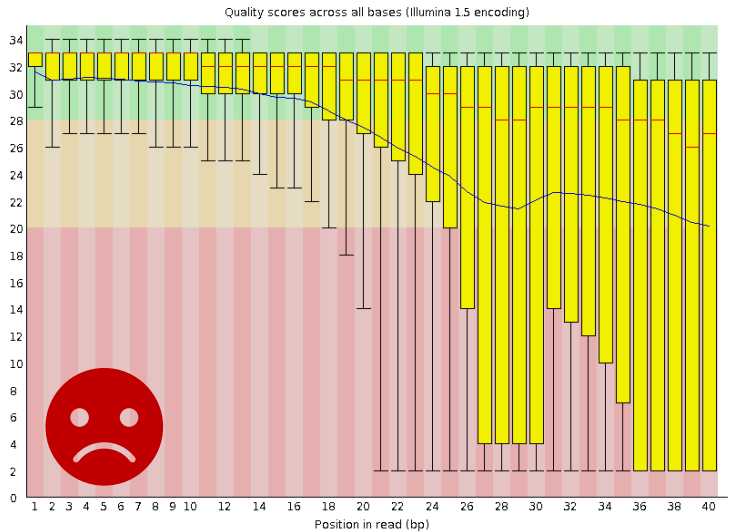
Very good quality

Reasonable quality

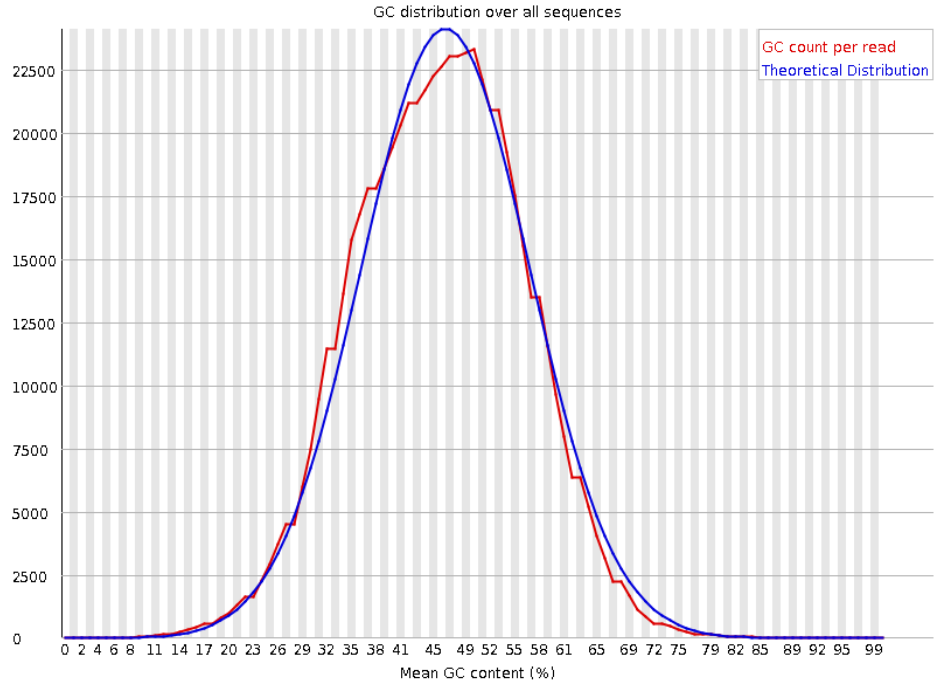
Poor quality



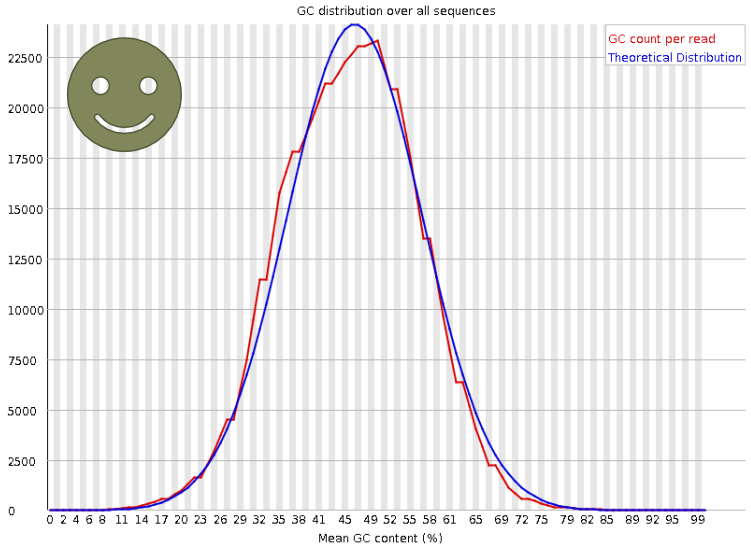


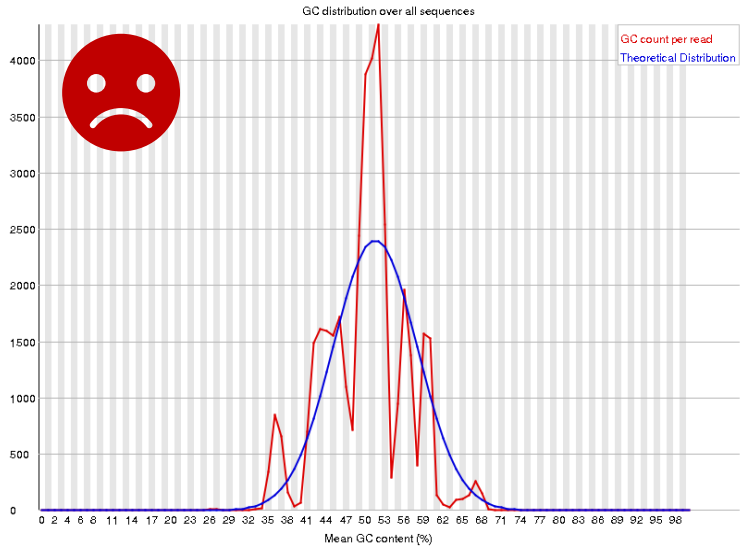


FastQC – Per sequence GC content

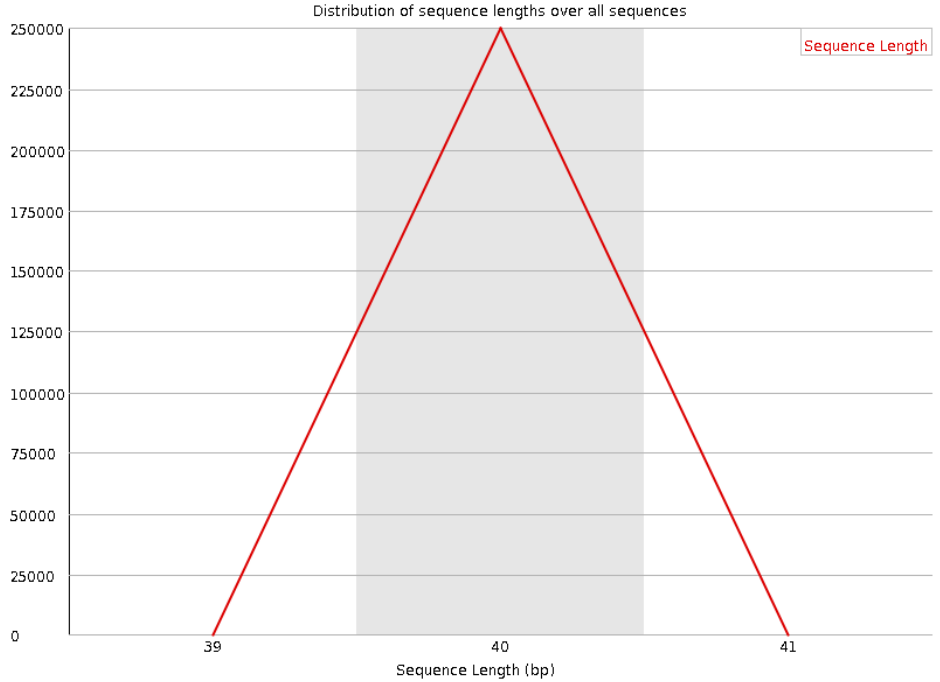


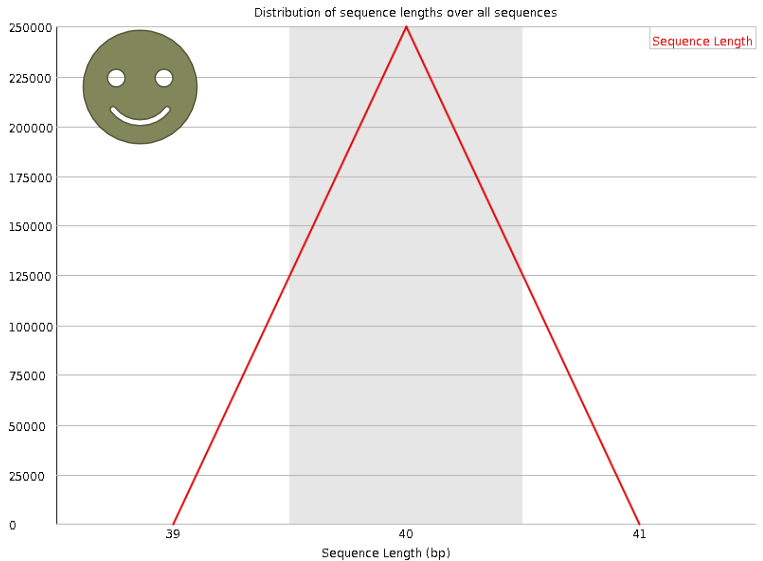
Measure of the GC content across the whole length of each sequence and compares it to a modelled normal distribution of GC content



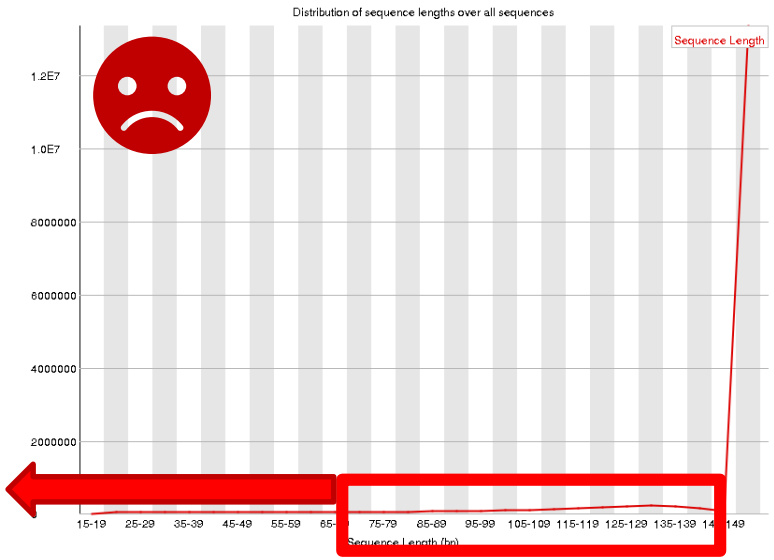


FastQC – Sequence Length Distribution

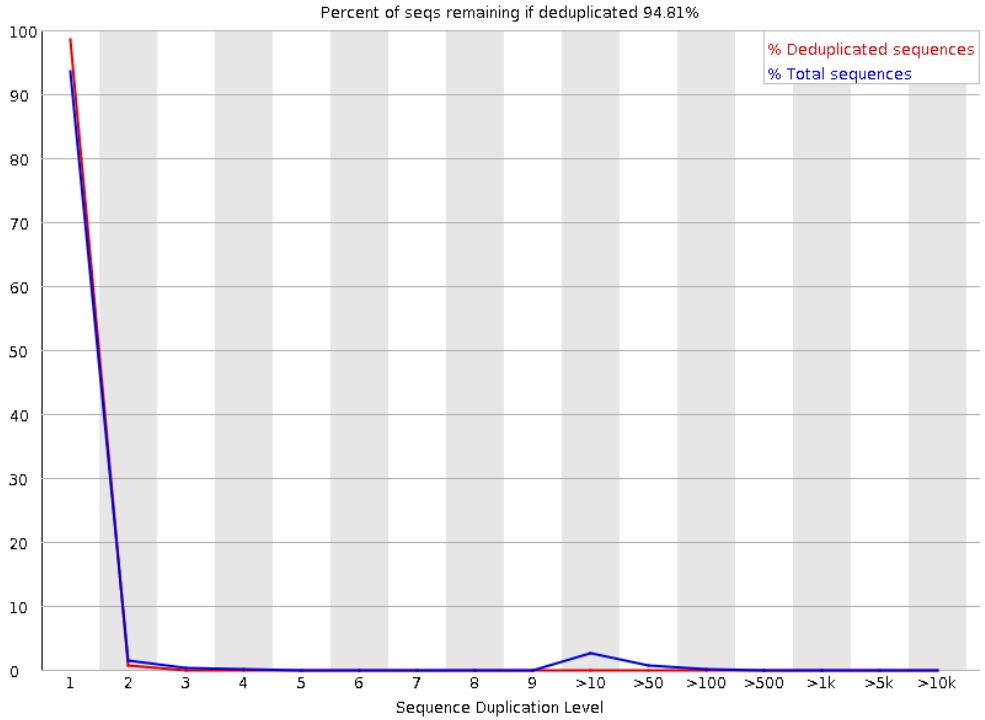


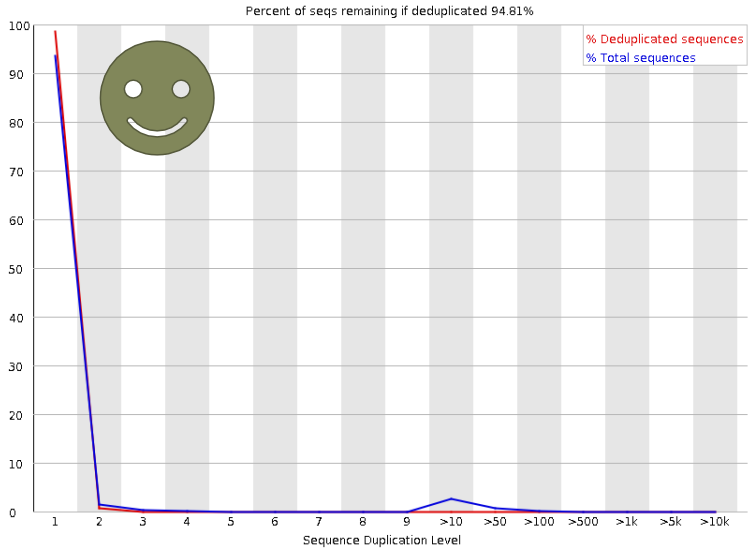


Some sequences have different length or zero length

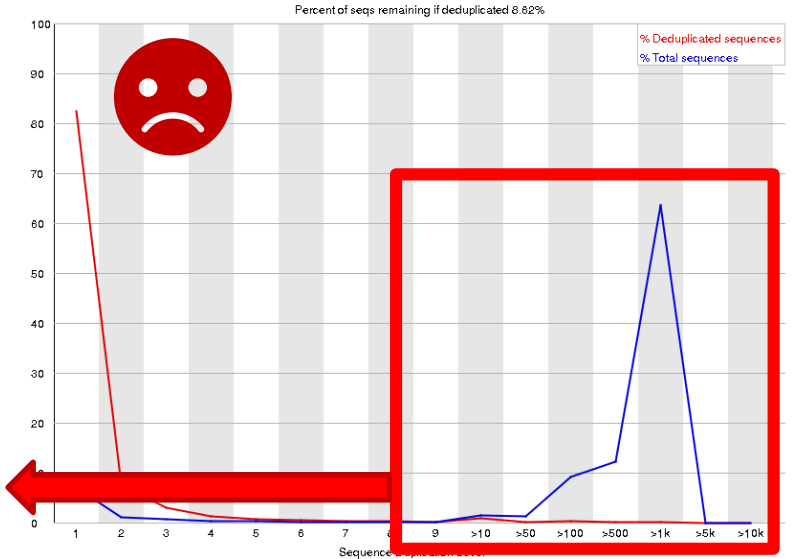


FastQC – Sequence Duplication Levels

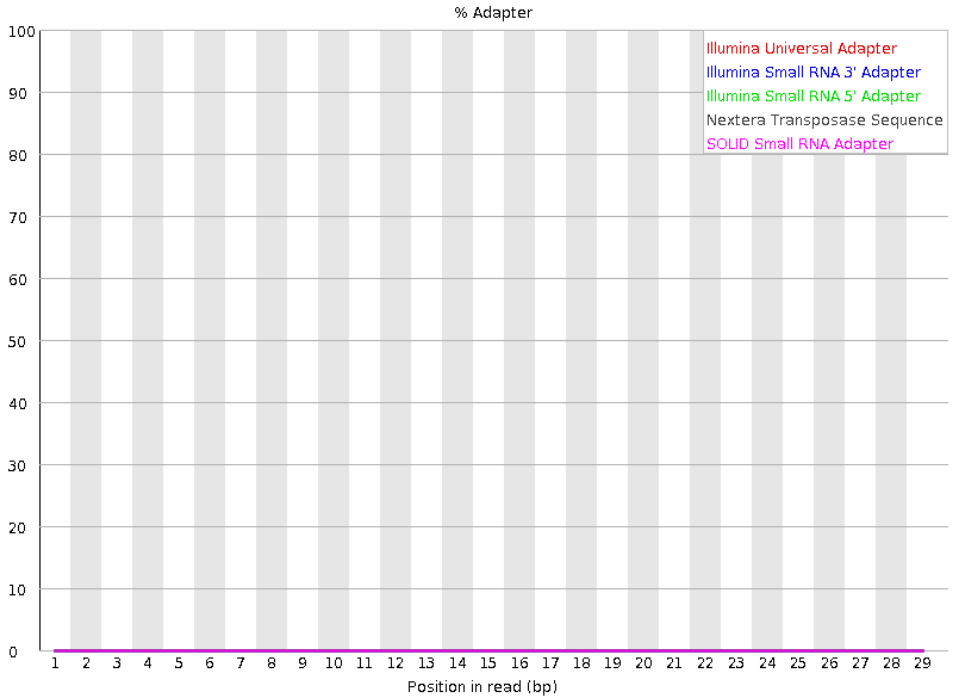


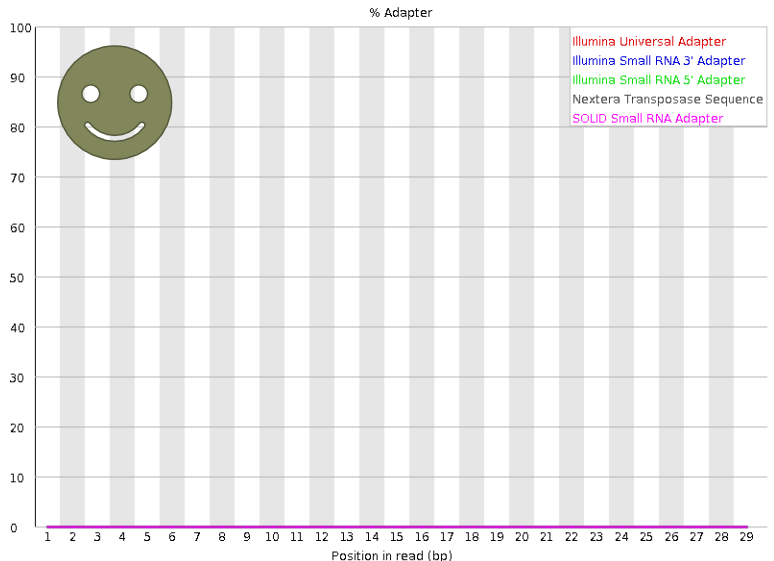


Very large number of sequences with high levels of duplication

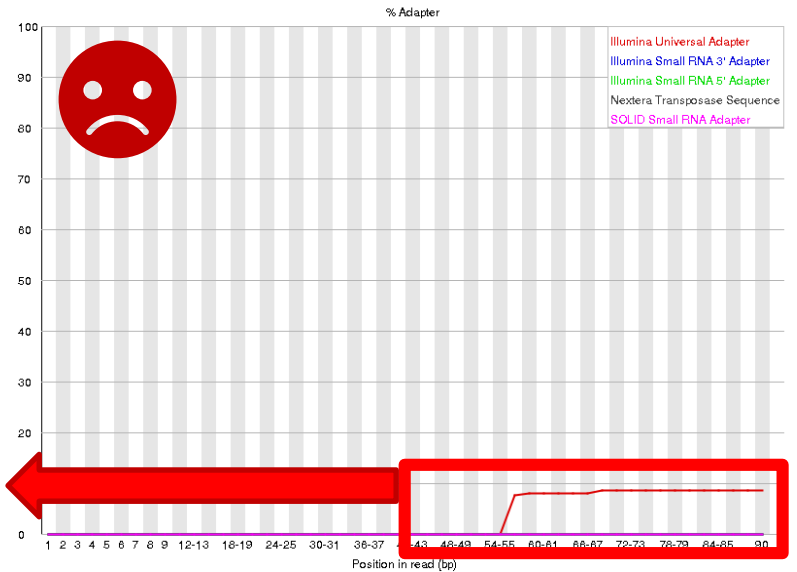


FastQC – Adapter Content

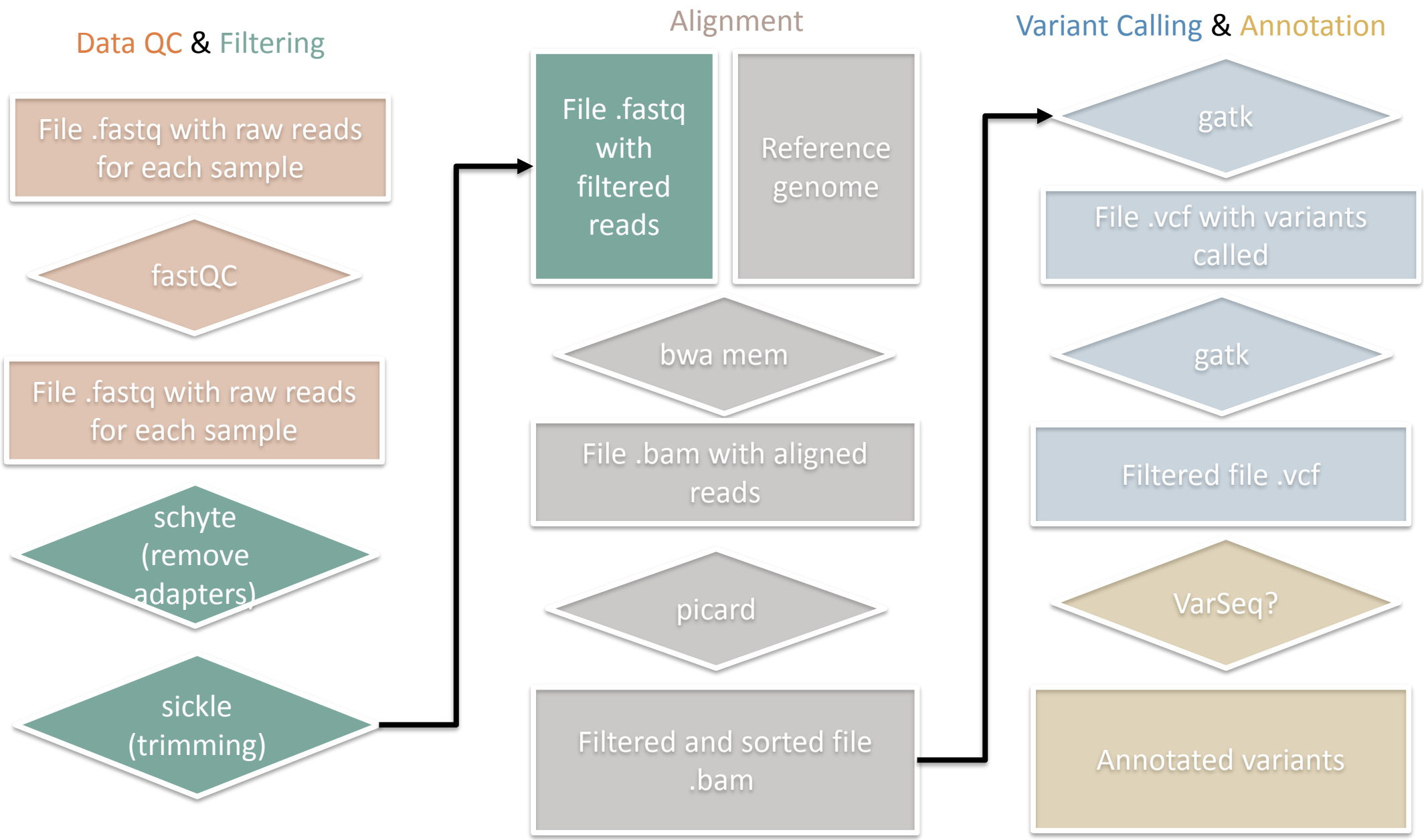




Some sequences contains adapters

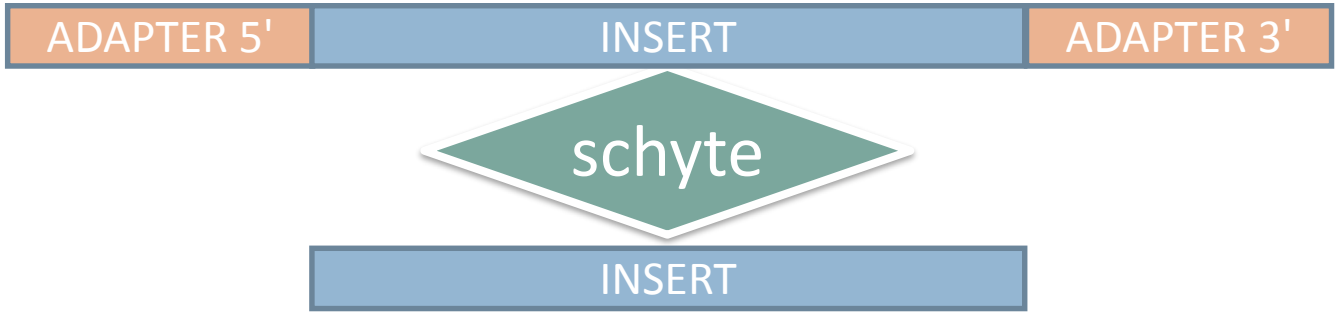


Pipeline


Data QC & Filtering
File .fastq with raw reads for each sample
fastQC
File .fastq with raw reads for each sample
schyte (remove adapters)
sickle (trimming)
Alignment
File .fastq with filtered reads
Reference genome
bwa mem
File .bam with aligned reads
picard
Filtered and sorted file .bam
Variant Calling & Annotation
gatk
File .vcf with variants called
gatk
Filtered file .vcf
VarSeq?
Annotated variants

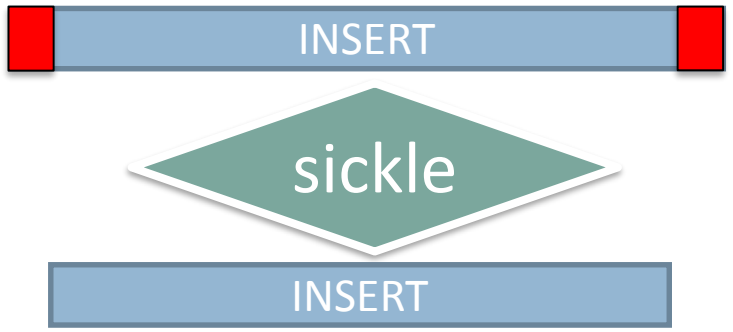
Remove adapters

Input: RAW fastQ read

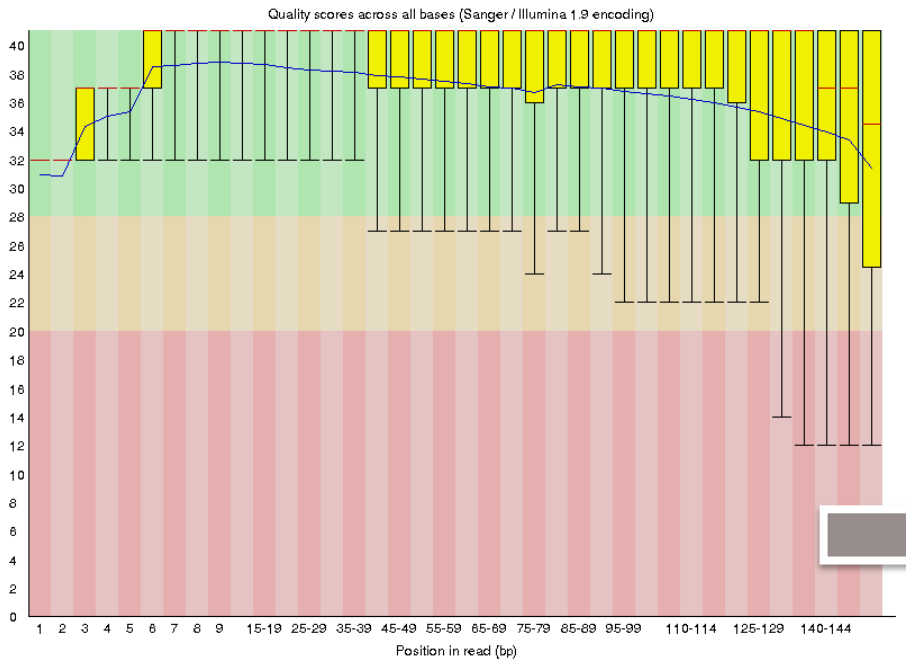


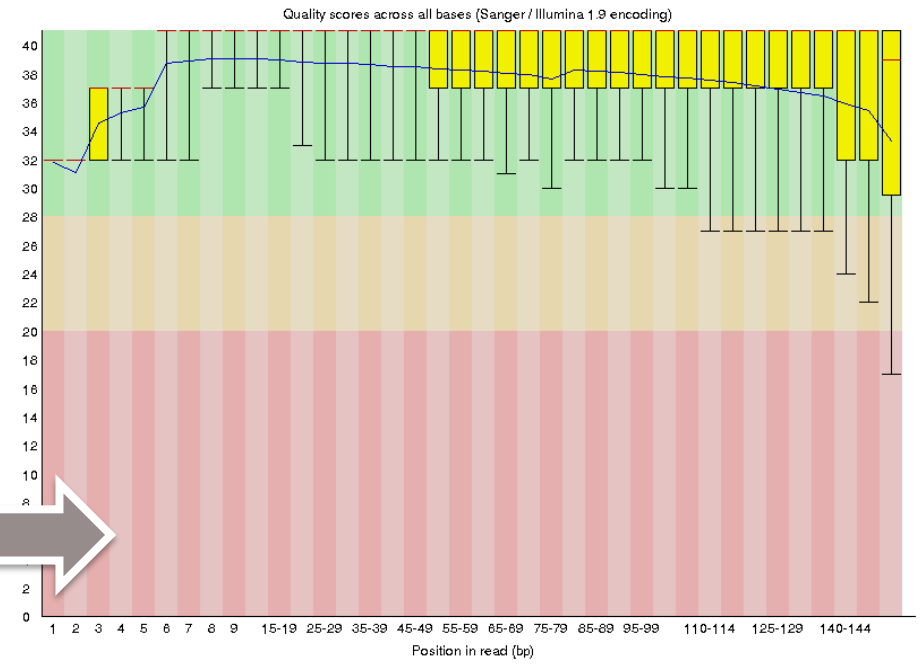
Reads trimming

Input: RAW fastQ read without adapters



Low quality bases





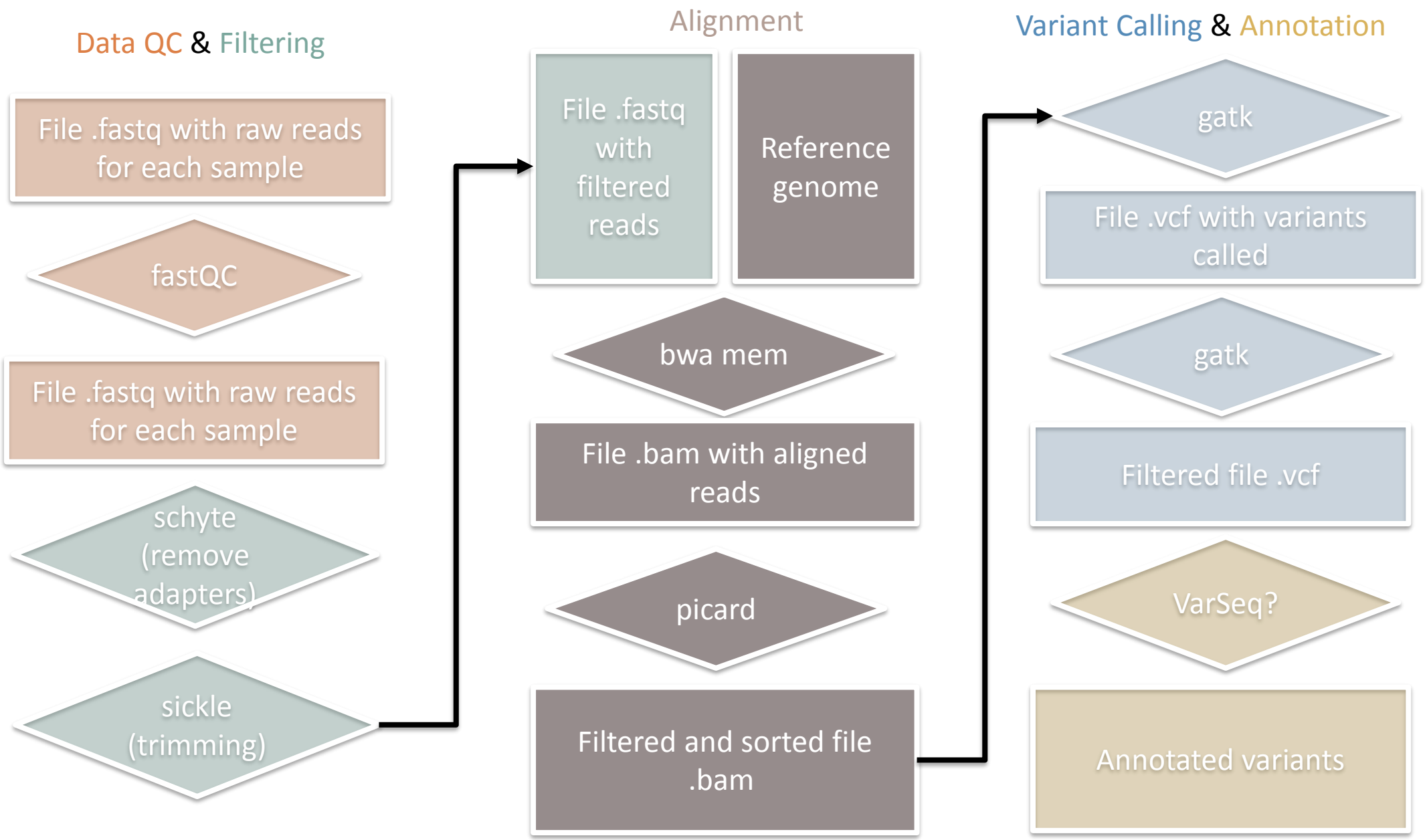
Filter command

Remove adapters from both reads and trimm reads:

```
sickle pe -g -t sanger
-f <( scythe -a ../example/reference/illumina_adapters.fa -q sanger R1.fastq.gz)
-r <( scythe -a ../example/reference/illumina_adapters.fa -q sanger R2.fastq.gz )
-o trimmed1.fastq.gz -p trimmed2.fastq.gz -s /dev/null
```

Execute the script: bash /attachedvolume/HGSI2020/example/scripts/step0.trimming.sh

Pipeline


Data QC & Filtering
File .fastq with raw reads for each sample
fastQC
File .fastq with raw reads for each sample
schyte (remove adapters)
sickle (trimming)
Alignment
File .fastq with filtered reads
Reference genome
bwa mem
File .bam with aligned reads
picard
Filtered and sorted file .bam
Variant Calling & Annotation
gatk
File .vcf with variants called
gatk
Filtered file .vcf
VarSeq?
Annotated variants

Alignment

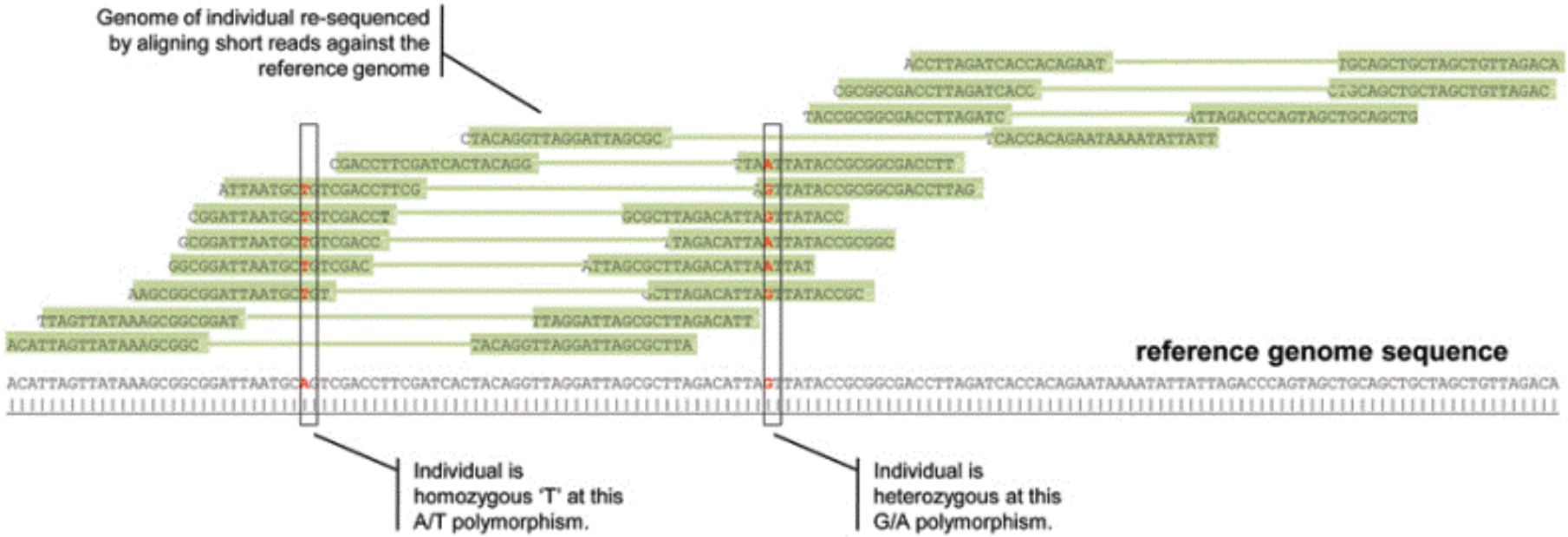
Many of the next-generation sequencing projects begin with a known, or so-called ‘reference’, genome. In this case, to make sense of the reads, their positions within the reference sequence must be determined. This process is known as aligning or ‘mapping’ the read to the reference.

Computationally difficult

- Short Reads
- Lots of repeats
- Presence of mismatch

Different algorithm solution:

- Bowtie
- BWA
- ISAAC Aligner



Alignment command

- Align your reads to the reference genome (chr6 hg38):

  /attachedvolume/HGSI2020/example/bin/bwa-0.7.12 mem /attachedvolume/HGSI2020/example/reference/chr6.hg38.fa trimmed1.fastq.gz trimmed2.fastq.gz > sample.sam

- Turn your file sam into file bam:

  samtools view -bT /attachedvolume/HGSI2020/example/reference/chr6.hg38.fa -o sample.bam sample.sam

Alignment command

- Sort your file:

  samtools sort sample.bam -o sample.sorted.bam

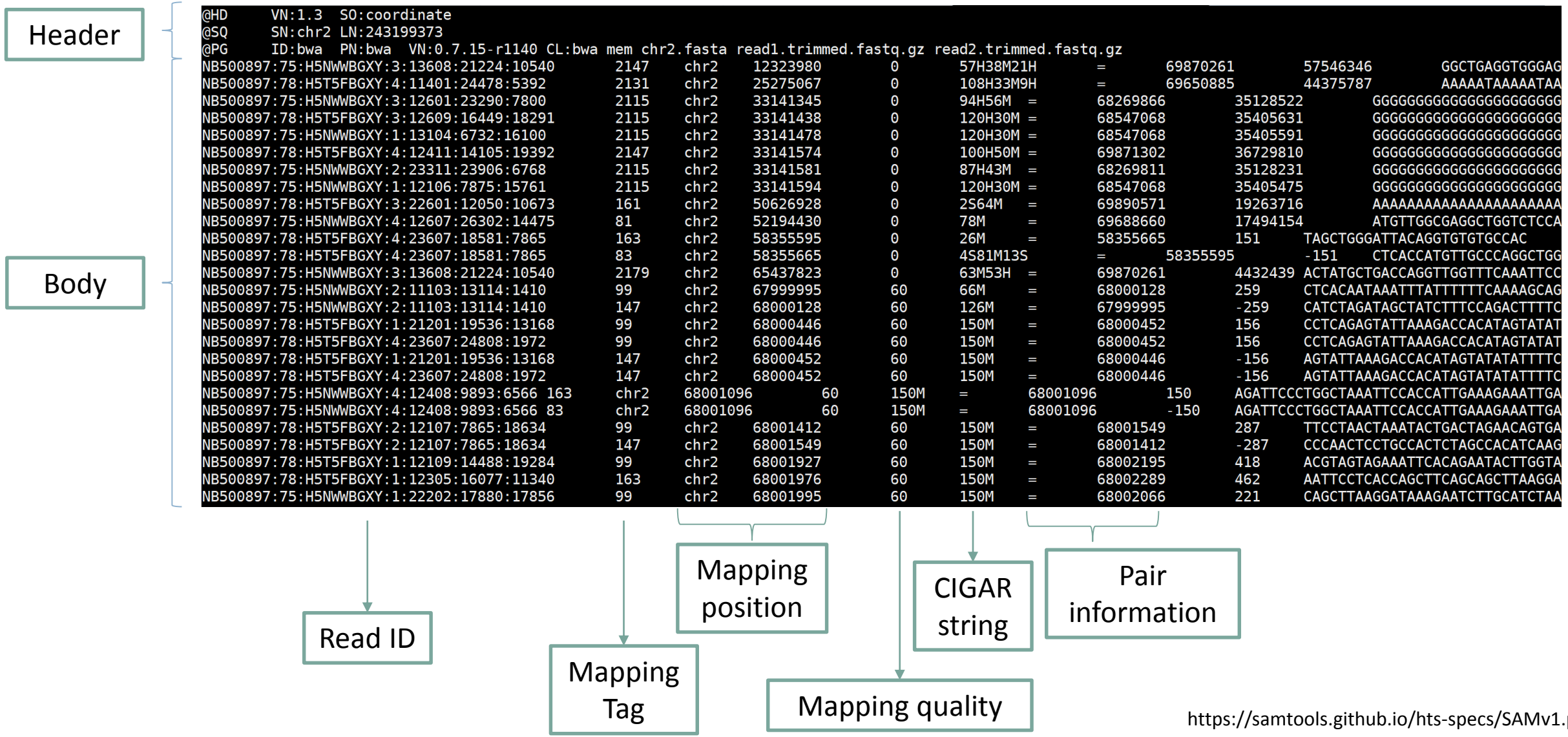
- Create index for your bam file:

  samtools index sample.sorted.bam

- Open the file:

  samtools view sample.sorted.bam | less -S

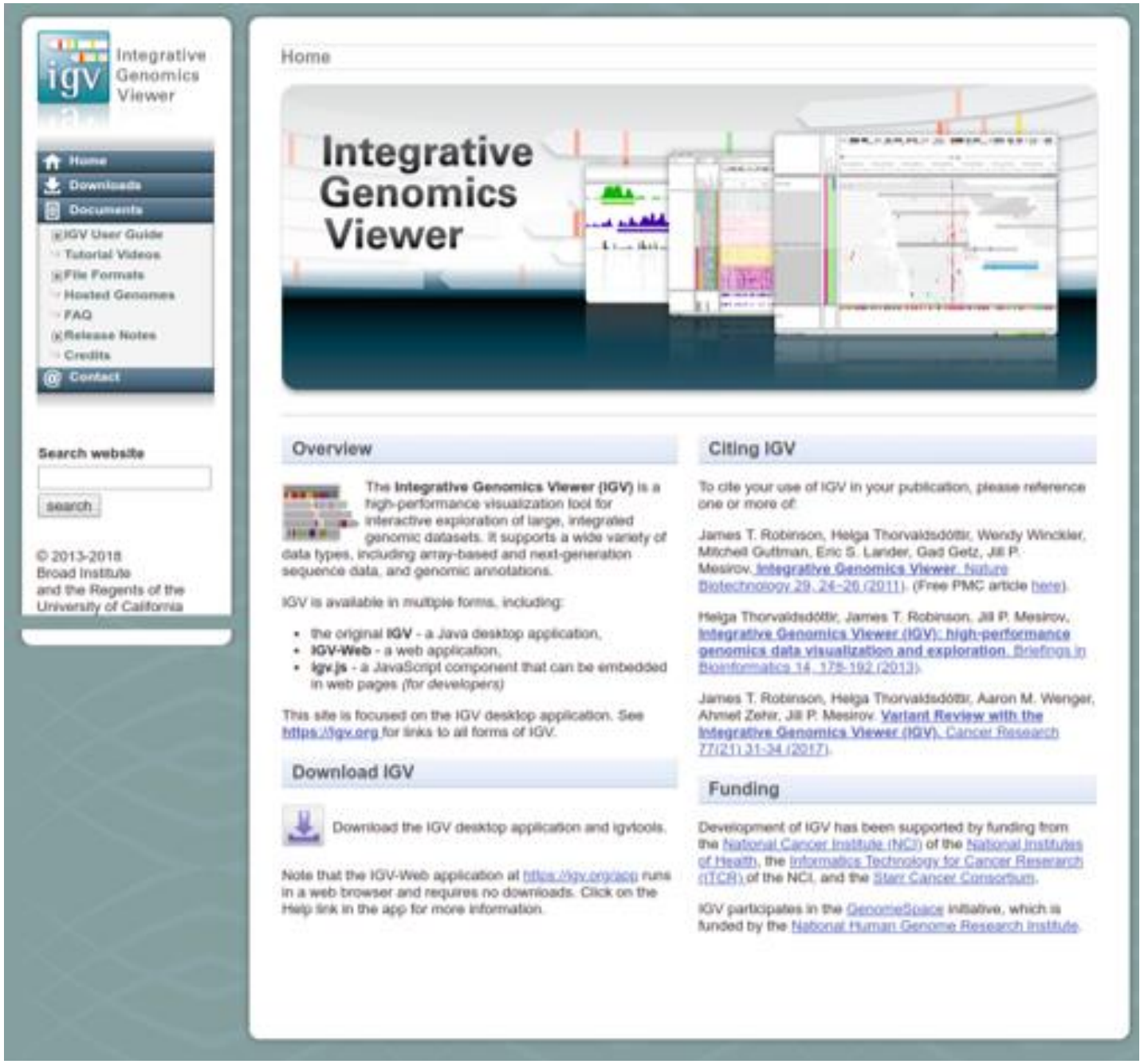
Alignment output – BAM file



https://samtools.github.io/hts-specs/SAMv1.pdf

Cigar

Op	Description
M	alignment match (can be a sequence match or mismatch)
I	insertion to the reference
D	deletion from the reference
N	skipped region from the reference
S	soft clipping (clipped sequences present in SEQ)
H	hard clipping (clipped sequences NOT present in SEQ)
P	padding (silent deletion from padded reference)
=	sequence match
X	sequence mismatch



IGV

https://software.broadinstitute.org/software/igv/

Download IGV

Install IGV 2.8.x

See the Release Notes for what's new in each release.

IGV Mac App

Download and unzip the Mac App Archive, then double-click the IGV application to run it.
You can move the app to the *Applications* folder, or anywhere else.

MacOS Catalina users: We sign our Mac App as a trusted Apple developer, but it is not yet notarized by Apple (a new requirement in Catalina). To run it, right-click on the downloaded IGV app; select "Open" from the menu; and click the "Open" button in the window that pops up. After that, double-clicking on the app will also work.

IGV for Windows

Download and run the installer.
An IGV shortcut will be created on the Desktop; double-click it to run the application.

IGV for Linux

Download and unzip the Archive.
See the downloaded *readme.txt* for further instructions.

IGV and igvtools to run on the command line (all platforms)

Download and unzip the Archive. ***Requires Java 11.***
See the downloaded *readme.txt* and *igvtools_readme.txt* for further instructions.

Download the bam and the bai

- Download the bam file and the index file on your pc:
- Open new terminal:

  cd Desktop/HGSI2020

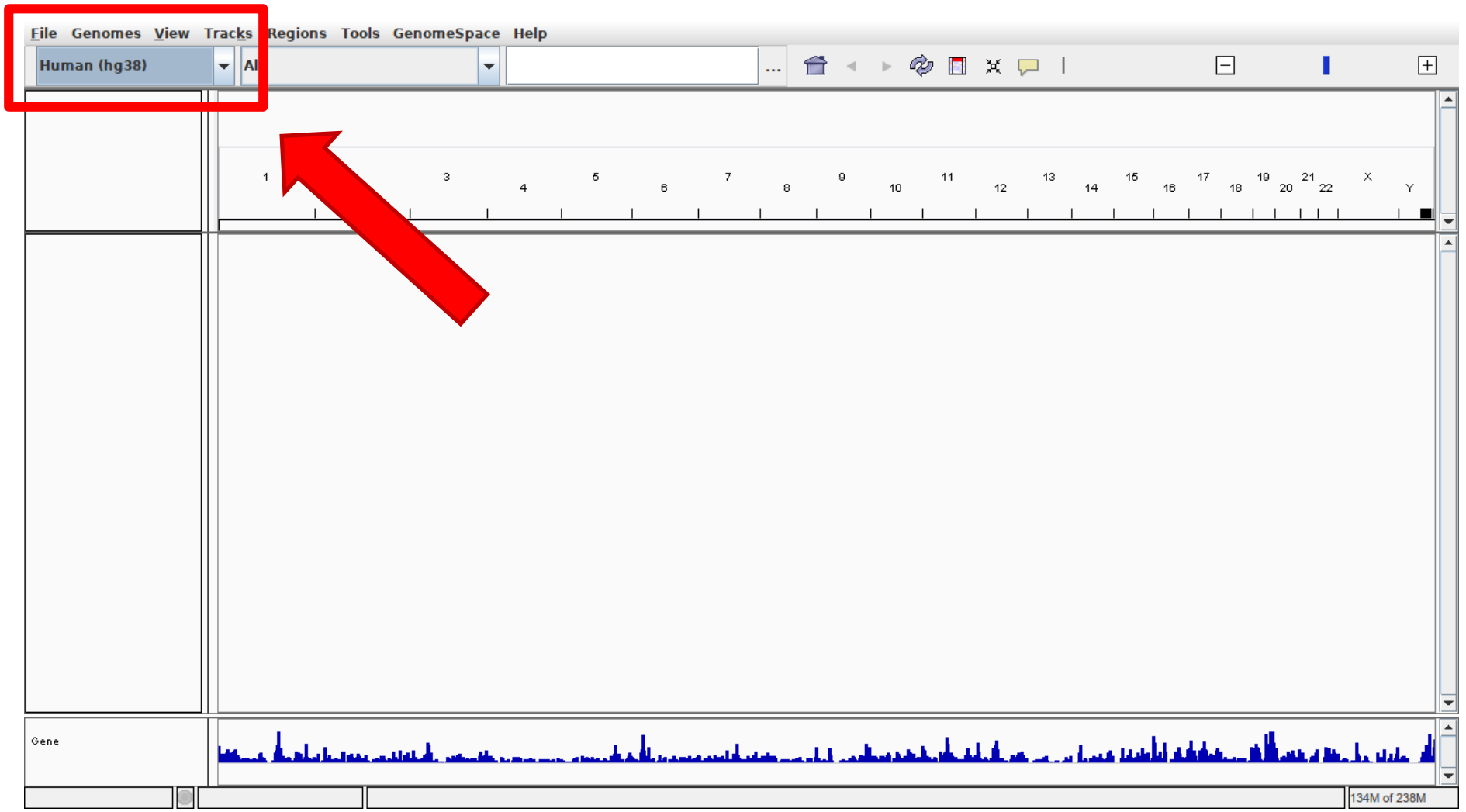
  rsync -auv HGSI2020@157.27.26.214:/attachedvolume/HGSI2020/Denise/sample.sorted.bam* .

- Check if you have downloaded: ls

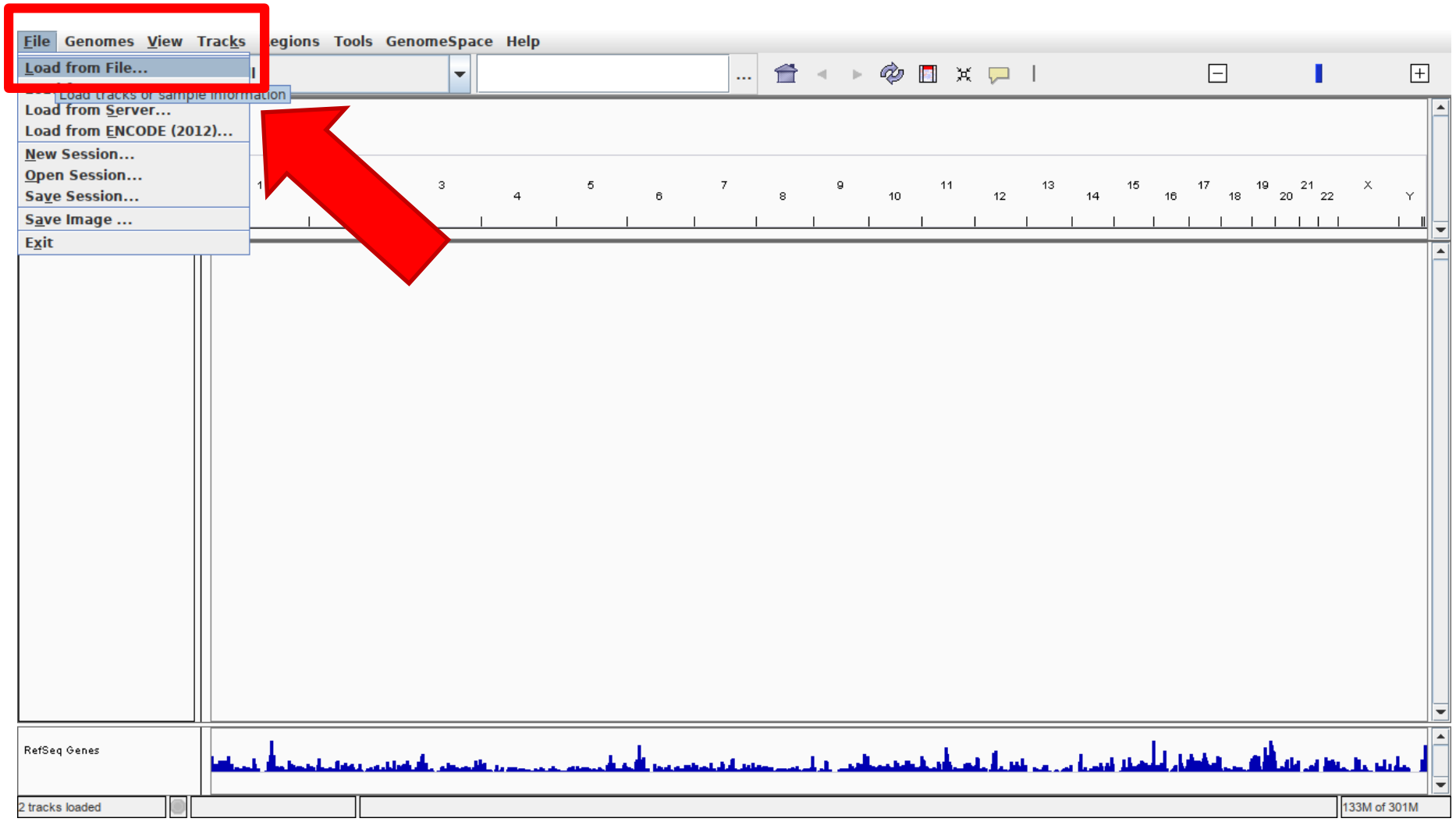
- Open IGV

  ./igv.sh for Ubuntu

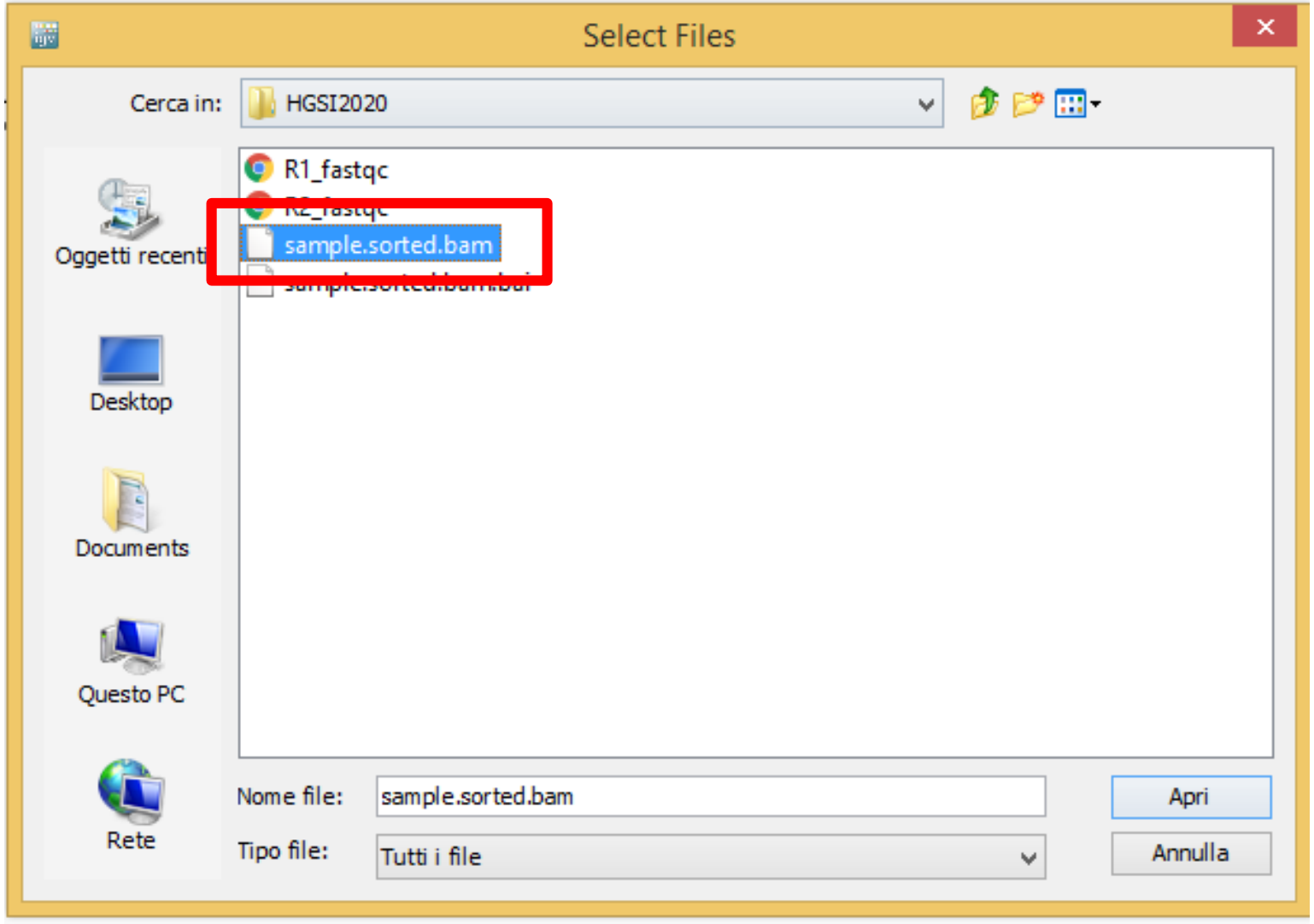
Choose the right genome



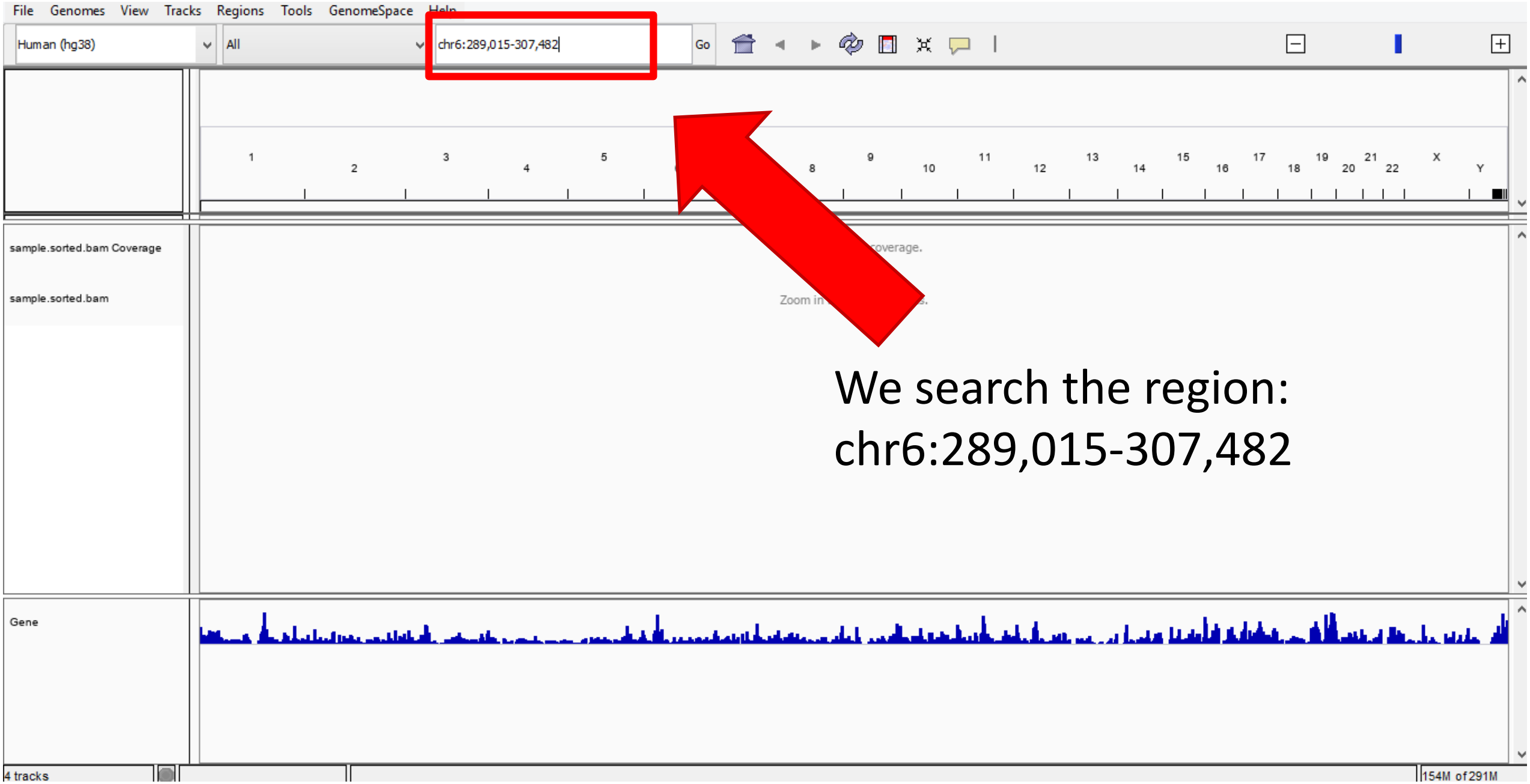
Upload the bam



Go into the folder "HGSI2020", choose the file bam and open it



Search a specific region



We search the region:
chr6:289,015-307,482

Results


File Genomes View Tracks Regions Tools GenomeSpace Help
Human (hg38)
chr6
chr6:289,015-307,482
Go
p25.1 p24.1 p22.3 p22.2 p21.32 p21.1 p12.3 p12.1 q11.1 q13 q14.1 q14.3 q16.1 q16.3 q21 q22.1 q22.31 q22.33 q23.3 q24.2 q25.1 q25.3 q27
18 kb
290 kb 292 kb 294 kb 296 kb 298 kb 300 kb 302 kb 304 kb 306 kb
sample.sorted.bam Coverage
[0 - 146]
sample.sorted.bam
Gene
DUSP22

Difference between genome and exome sequencing