

# Human Genome Sequencing and Interpretation

**Lesson 1 - 20/01/2020**

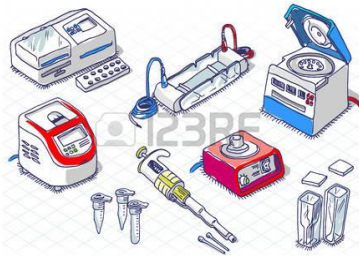
Lesson 2 - 21/01/2020

Lesson 3 - 27/01/2020

(Lesson 4 - 28/01/2020)

Prof. Massimo Delledonne  
Functional Genomics lab

Library  
preparation

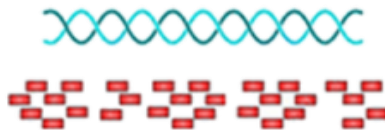


Sequencing

Bioinformatic  
analysis

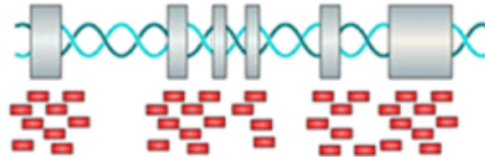


Whole genome sequencing



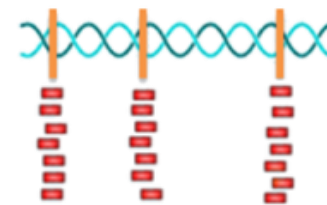
- Sequencing region : whole genome
- Sequencing Depth: >30X
- Covers everything – can identify all kinds of variants including SNPs, INDELs and SV.

Whole exome sequencing



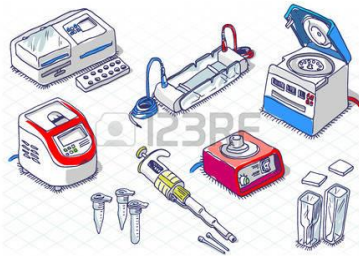
- Sequencing region: whole exome
- Sequencing Depth : >50X ~ 100X
- Identify all kinds of variants including SNPs, INDELs and SV in coding region.
- Cost effective

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

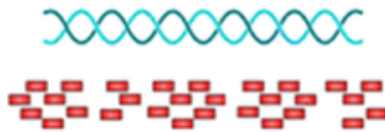


Sequencing

Bioinformatic  
analysis

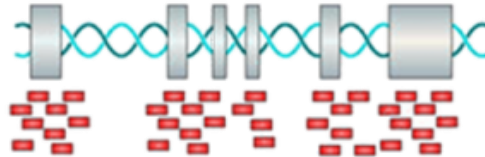


Whole genome sequencing



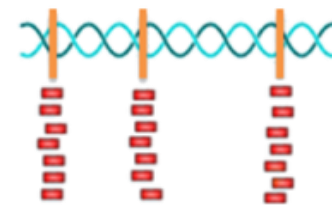
- Sequencing region : whole genome
- Sequencing Depth: >30X
- Covers everything – can identify all kinds of variants including SNPs, INDELs and SV.

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

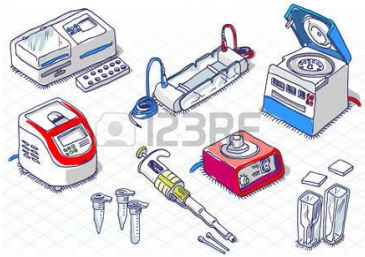


- Sequencing region: specific regions (could be customized)
- Sequencing Depth : >500X
- Identify all kinds of variants including SNPs, INDELs in specific regions
- Most Cost effective

Library  
preparation



Bioinformatic  
analysis



Sequencing



Data QC

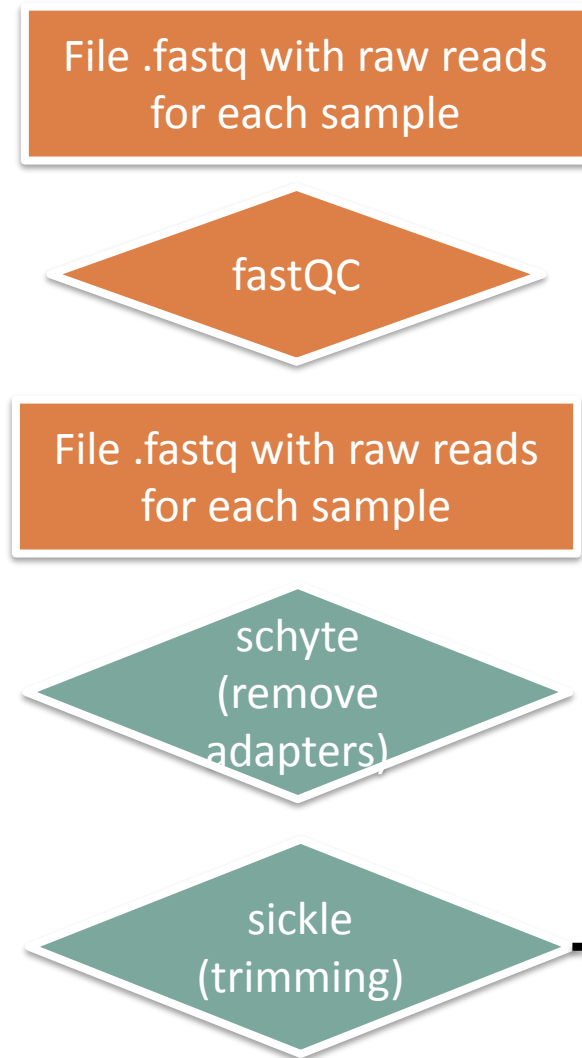
Alignment

Variant  
calling

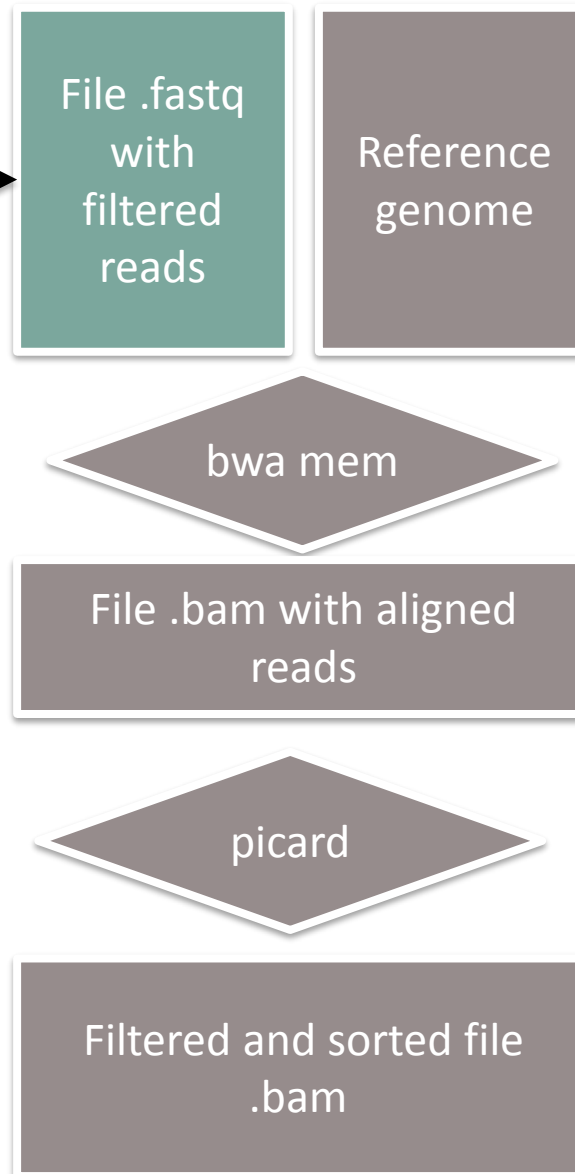
Variant  
prioritization

# Pipeline

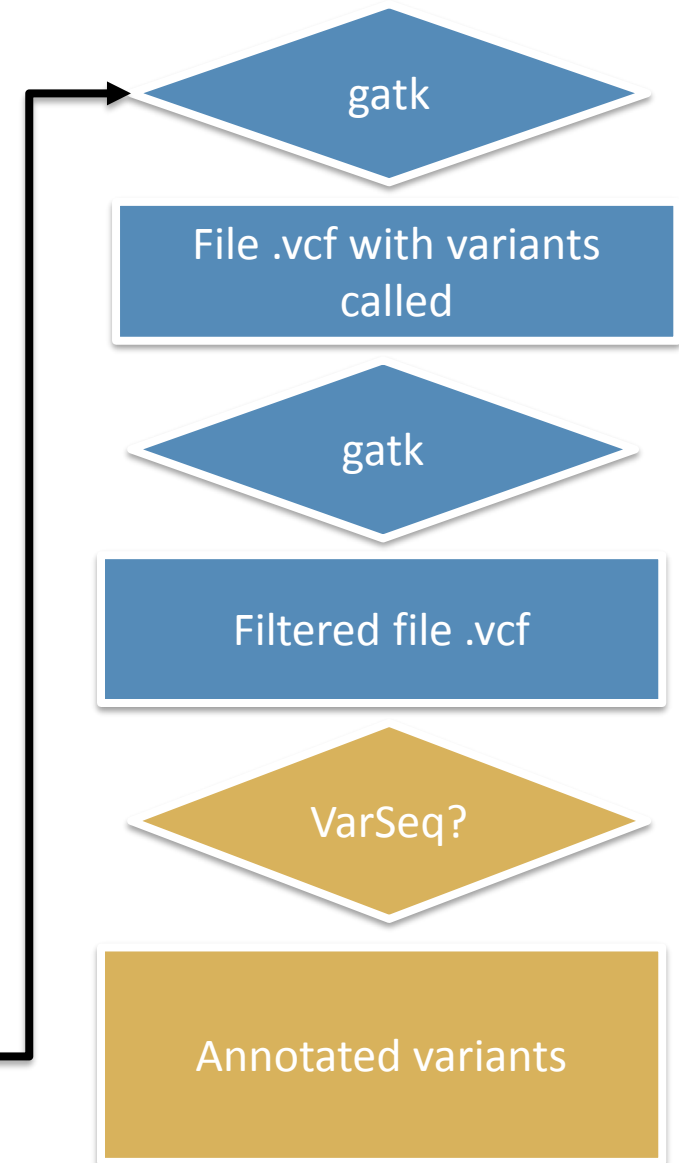
## Data QC & Filtering



## Alignment



## Variant Calling & Annotation



# 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

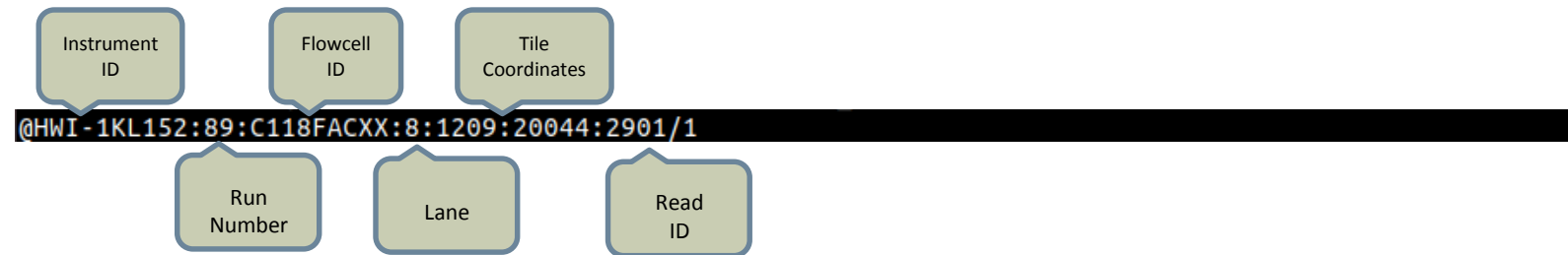


# .fq / .fastq file

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:



The first row of a FASTQ file is a header line starting with '@'. It contains several fields separated by colons. The diagram shows callouts for the following fields:

- Instrument ID: @HWI-1KL152
- Flowcell ID: 89
- Tile Coordinates: C118FACXX
- Run Number: 8
- Lane: 1209
- Read ID: 20044:2901/1

```
@HWI-1KL152:89:C118FACXX:8:1209:20044:2901/1
```

Second row contains the sequence:

```
CTCTGTGGCTGGGAGAGGAGTCTGGGGGGGCCCCGGGCGCCAGCCAGGGATAGCCTGATCTCTGCTCCAGTCGACAGATCCTTAACGGATTTTCTTTCTCT
```

Thirs row contains a delimiter:

```
+
```

Fourth row indicate the quality of each base:

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

# 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

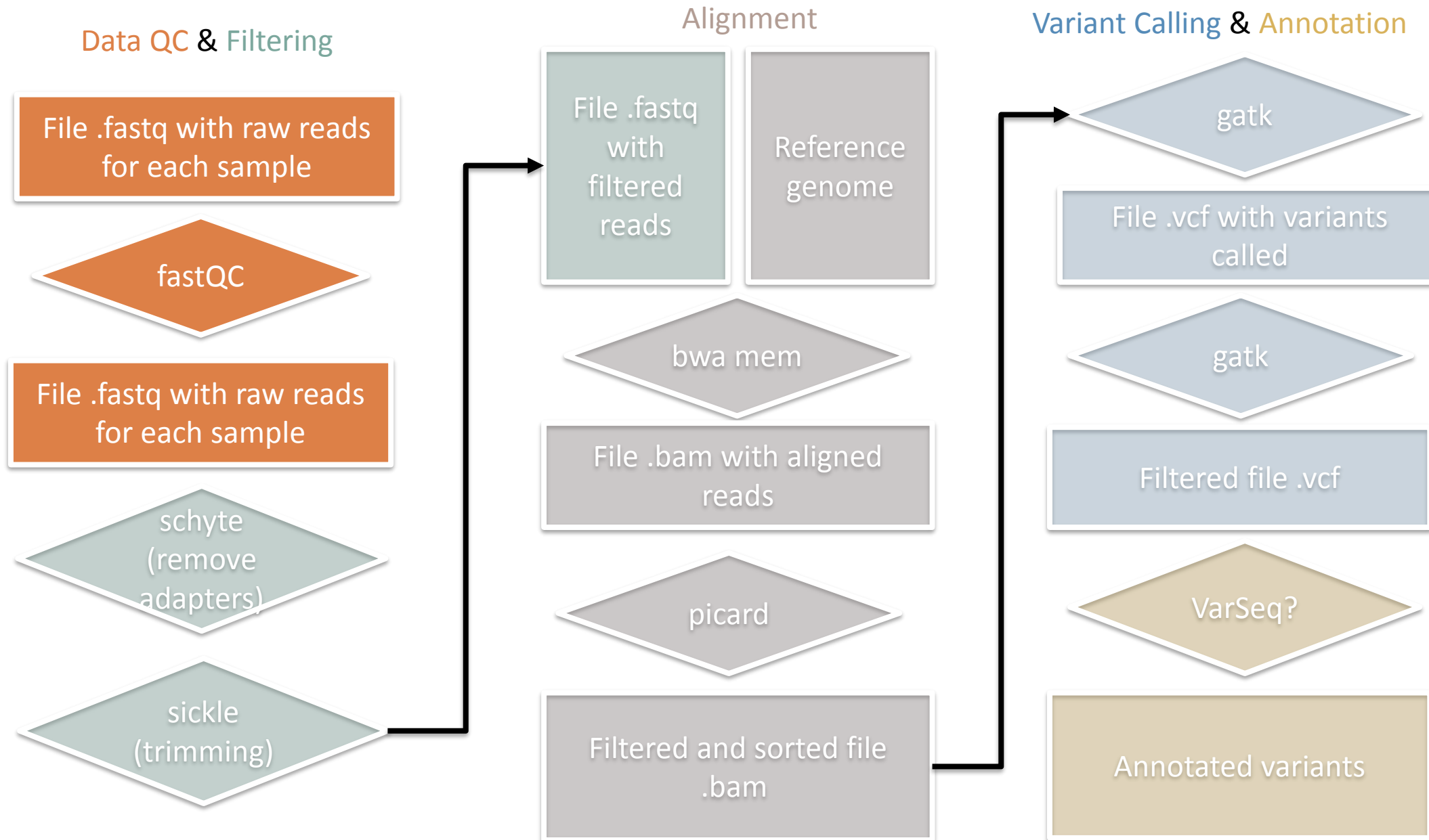
File .vcf with variants  
called

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:

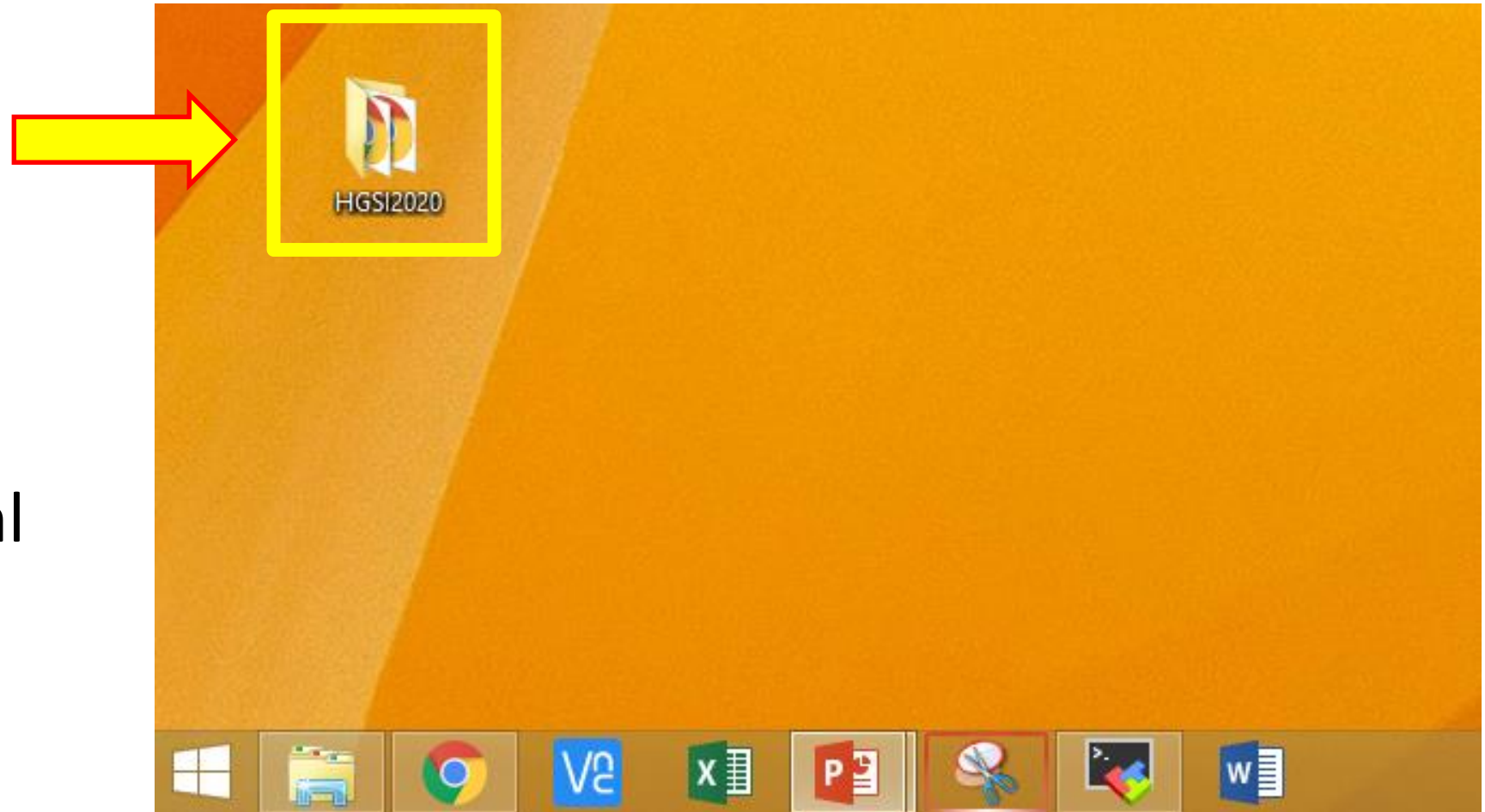
```
rsync -auv HGSI2020@157.27.26.214:/attachedvolume/HGSI2020/Denise/fastqc/R*_fastqc.html .
```

Pass: `hgsi2020`

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

# Download the files

1. On your desktop, open the file «HGSi2020»



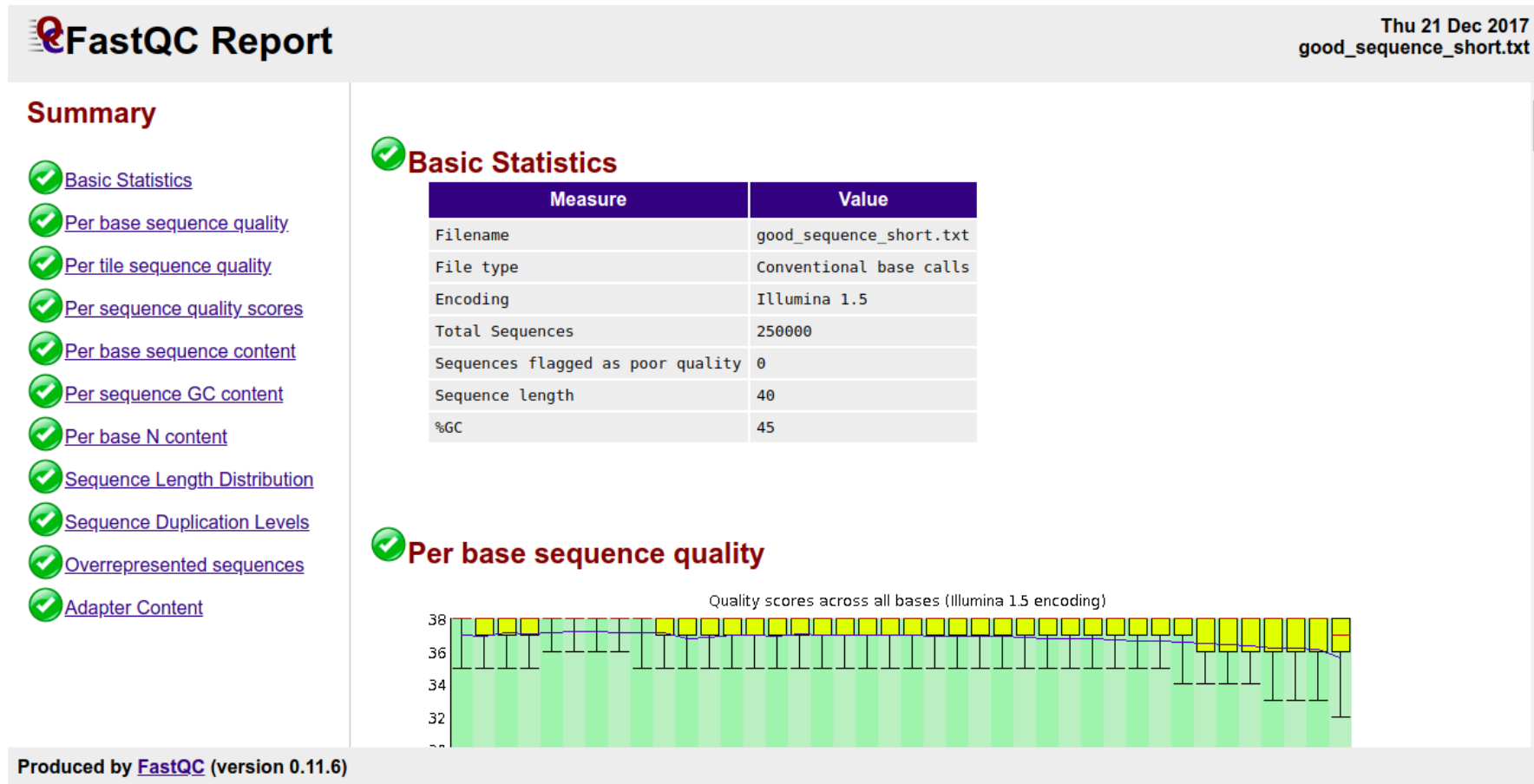
2. Open the file html



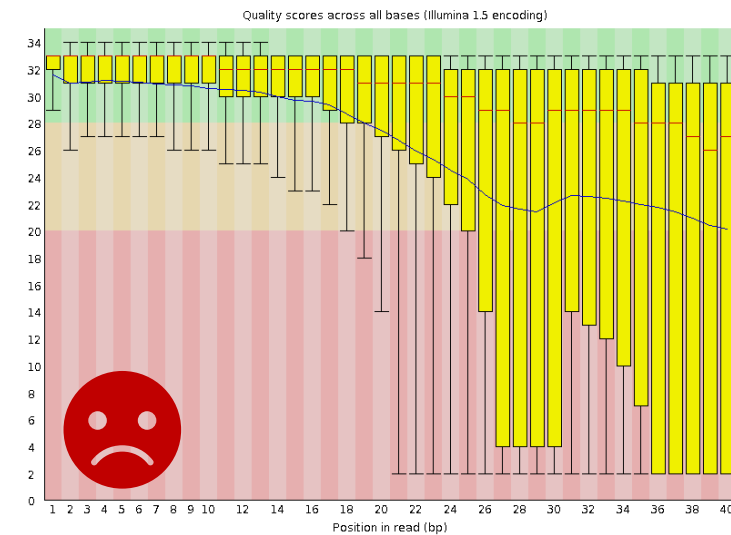
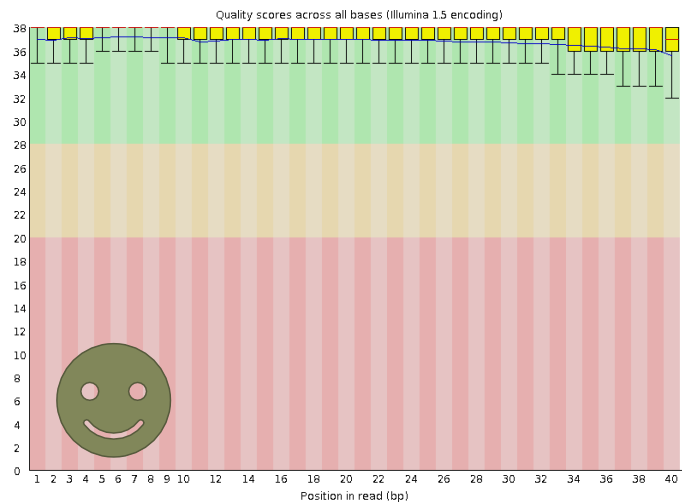
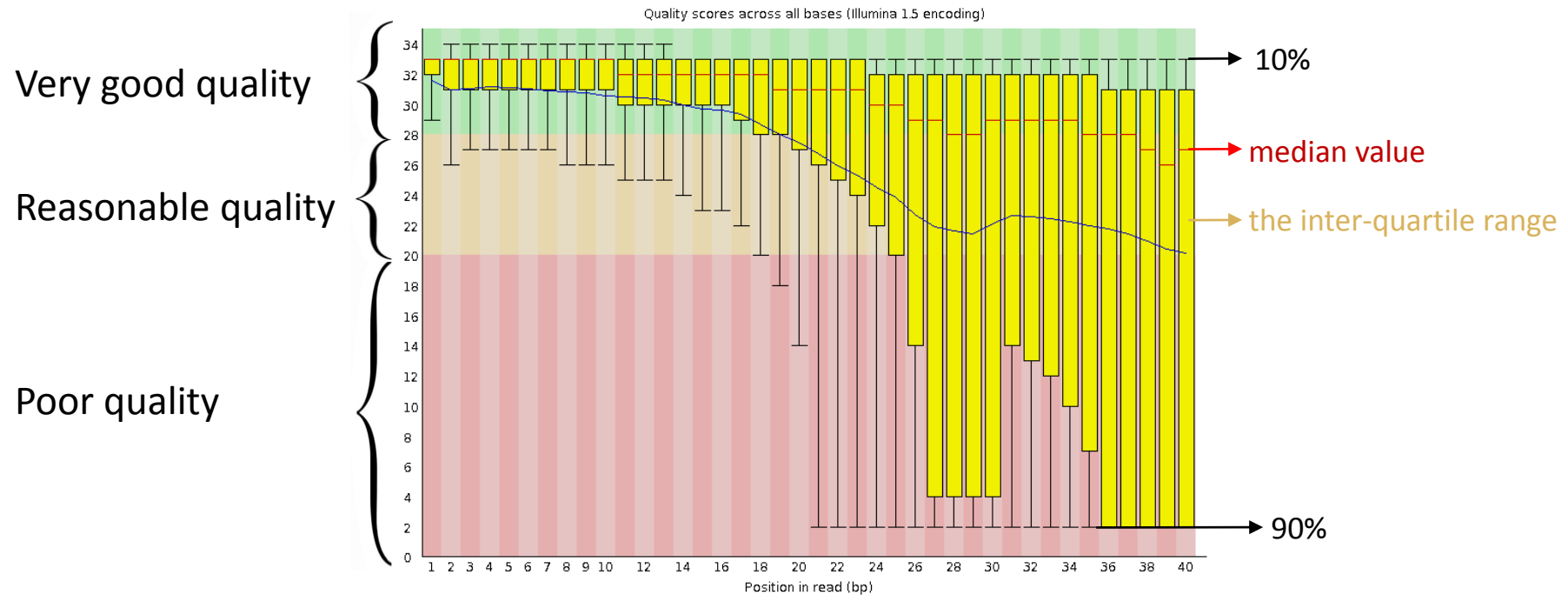
# FastQC

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

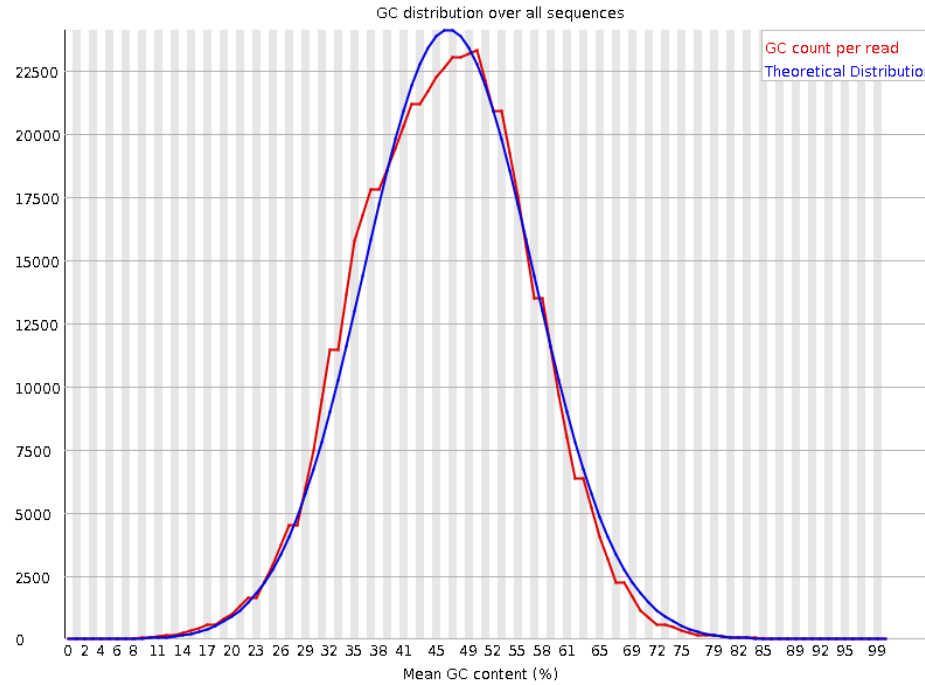
## HTML Report



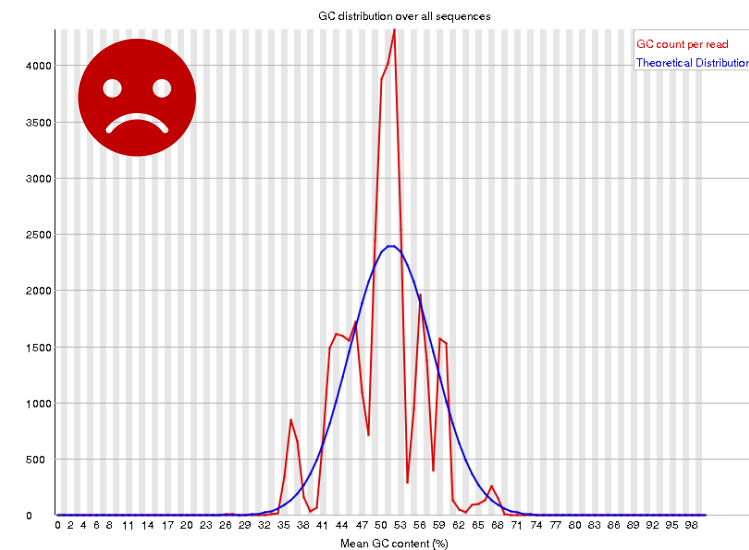
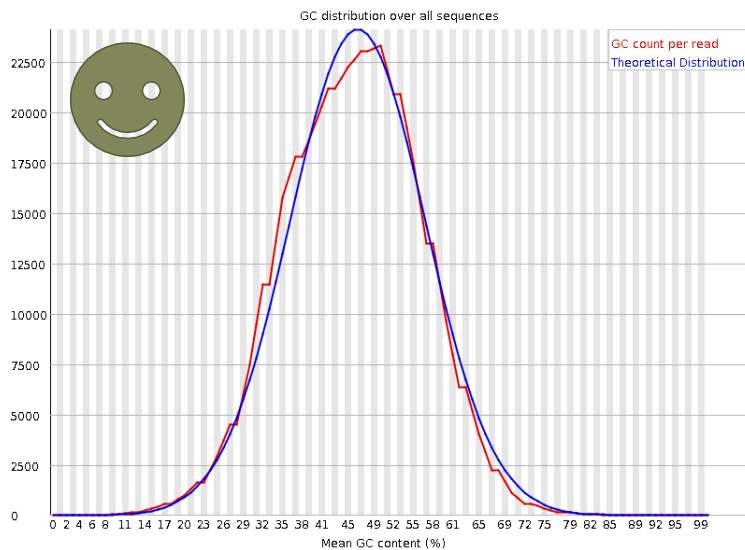
# FastQC – Per base sequence quality



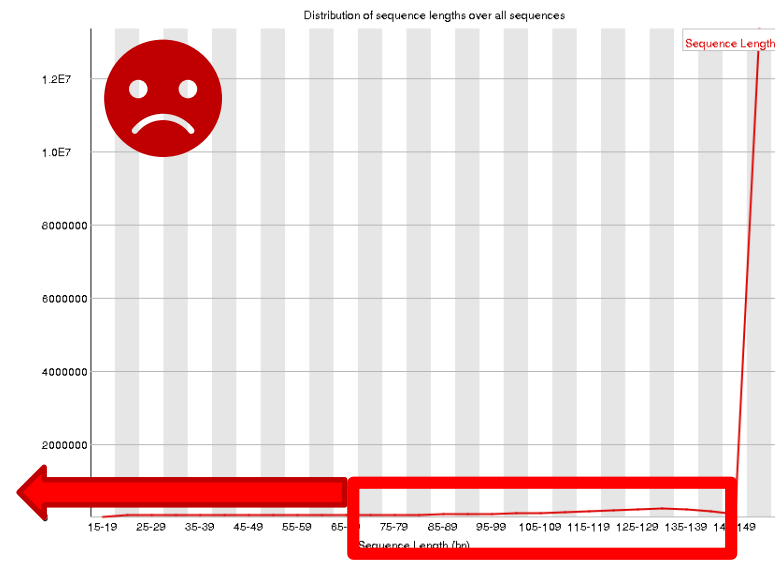
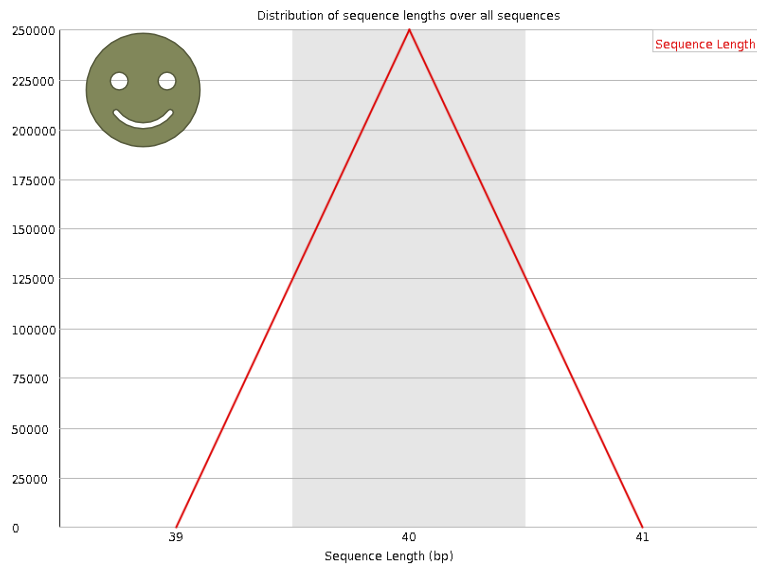
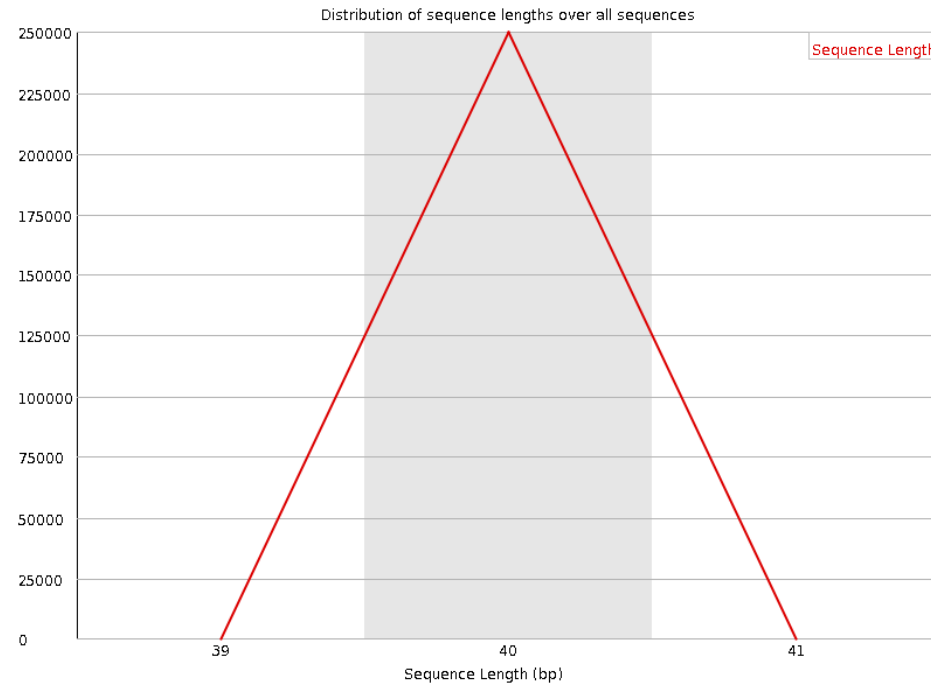
# FastQC – Per sequence GC content



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

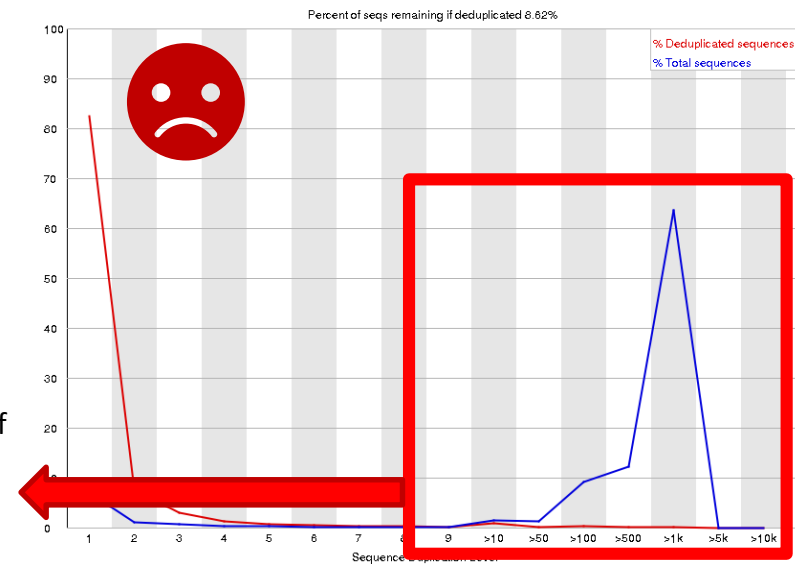
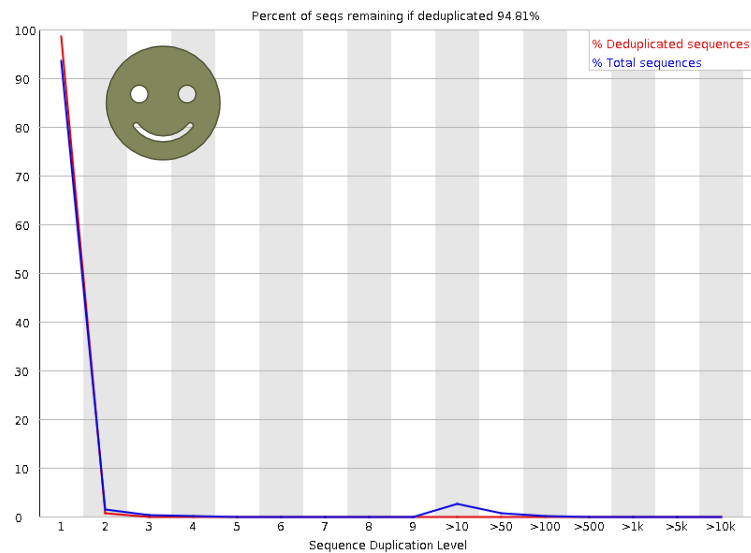
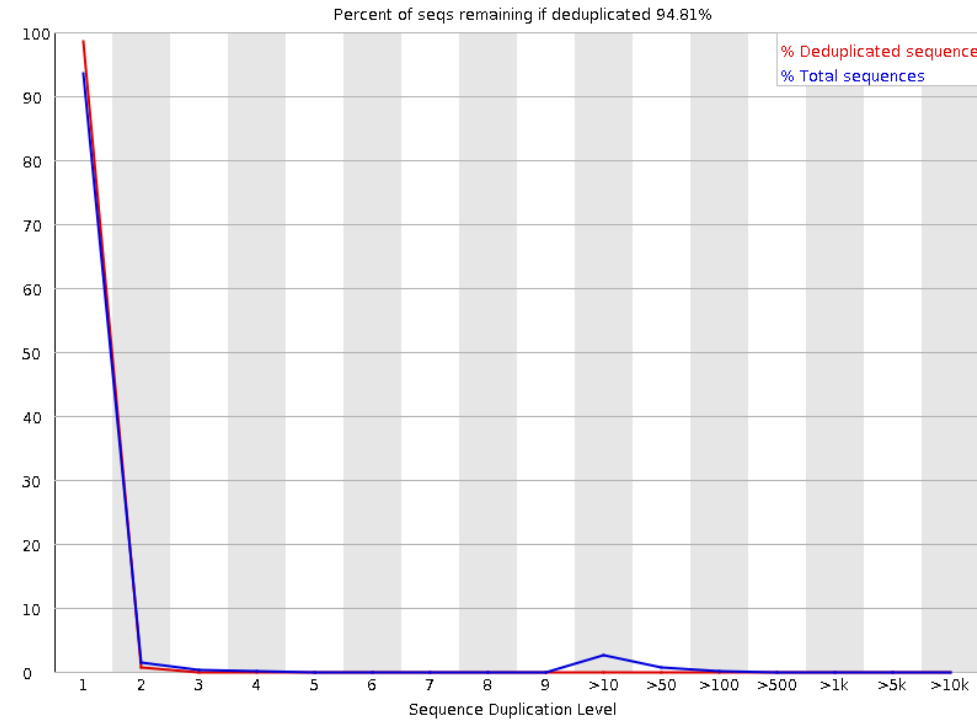


# FastQC – Sequence Length Distribution



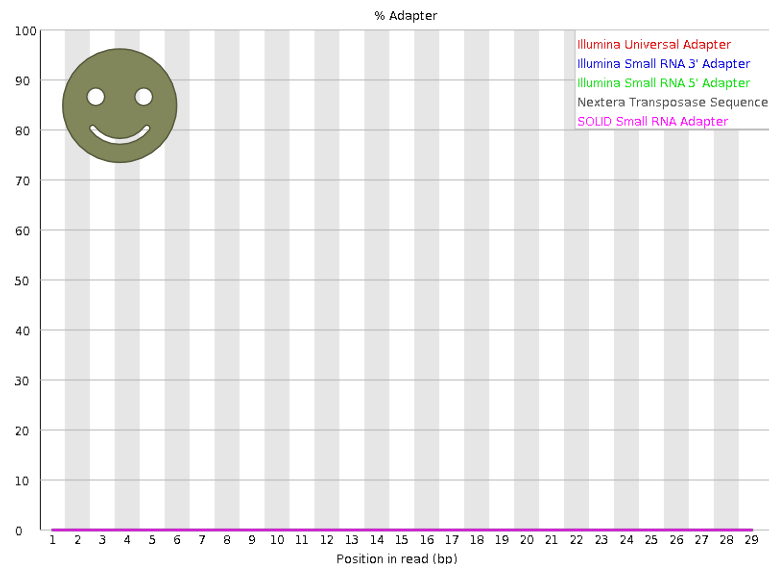
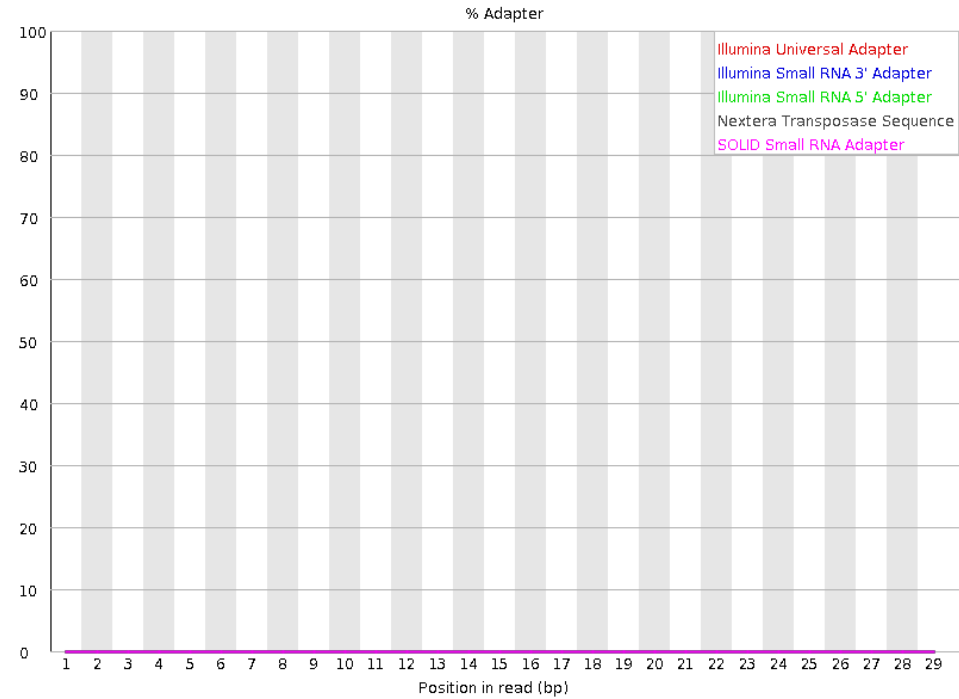
Some sequences have different length or zero length

# FastQC – Sequence Duplication Levels

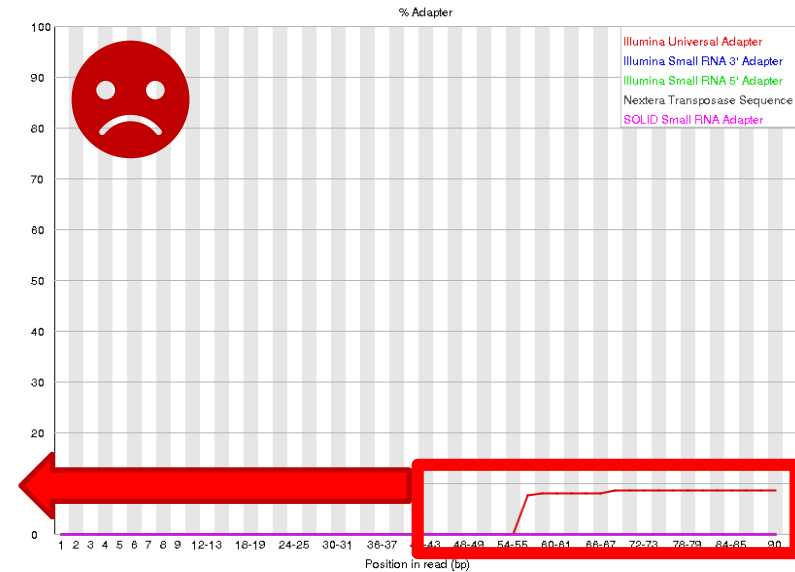


Very large number of sequences with high levels of duplication

# 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

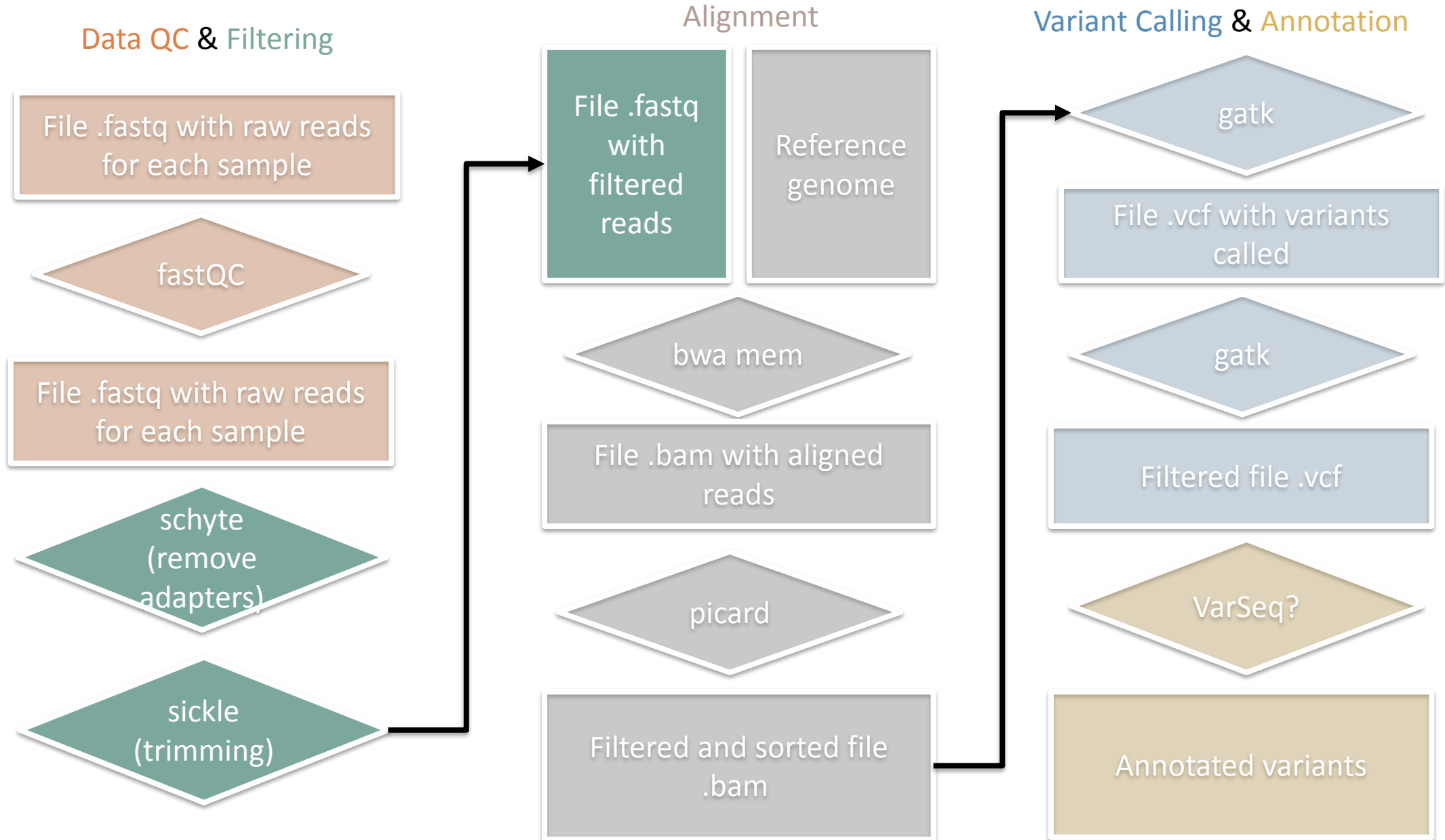
File .vcf with variants  
called

gatk

Filtered file .vcf

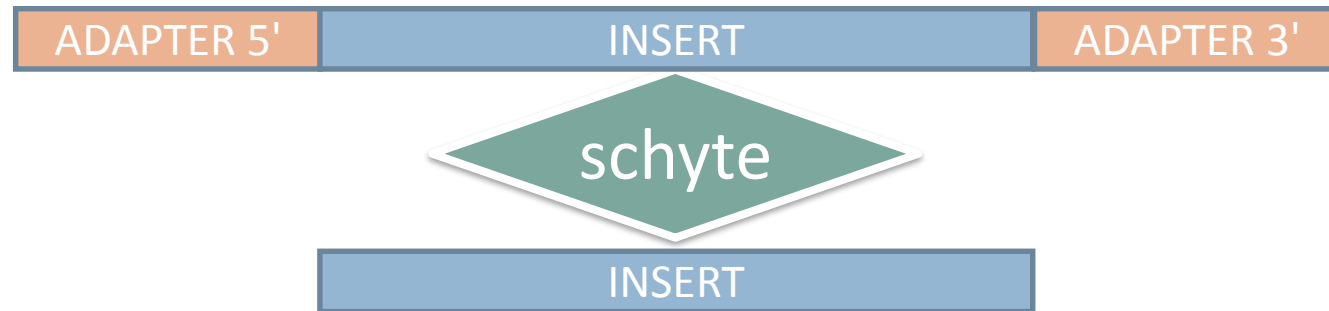
VarSeq?

Annotated variants



# Remove adapters

Input: RAW fastQ read




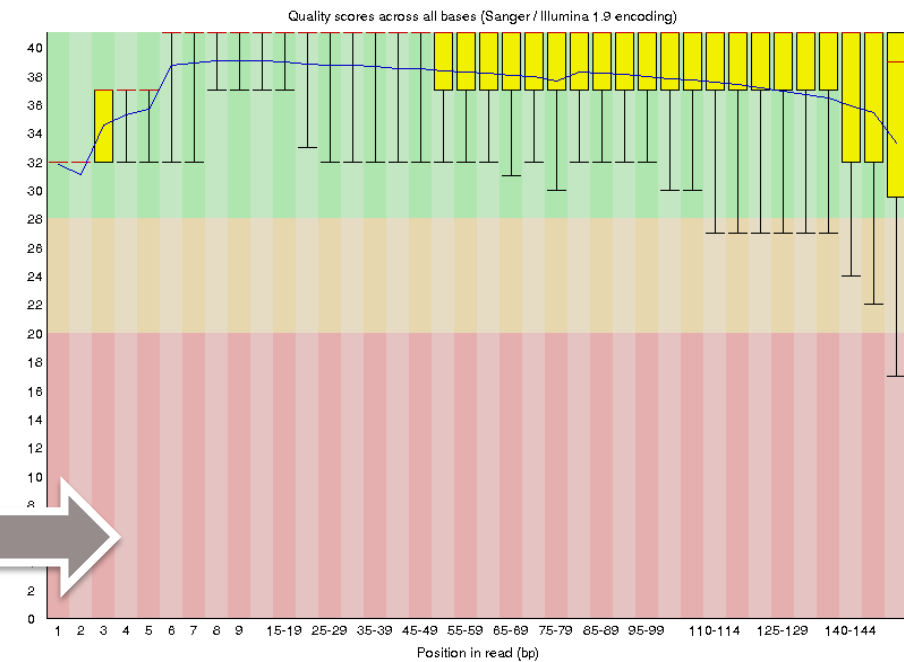
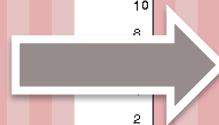
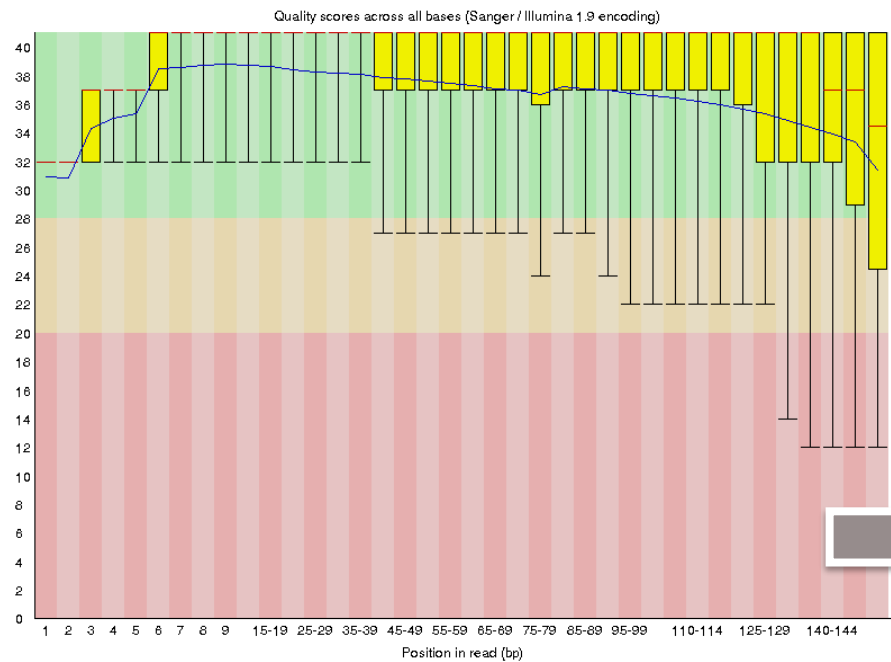


# Reads trimming

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

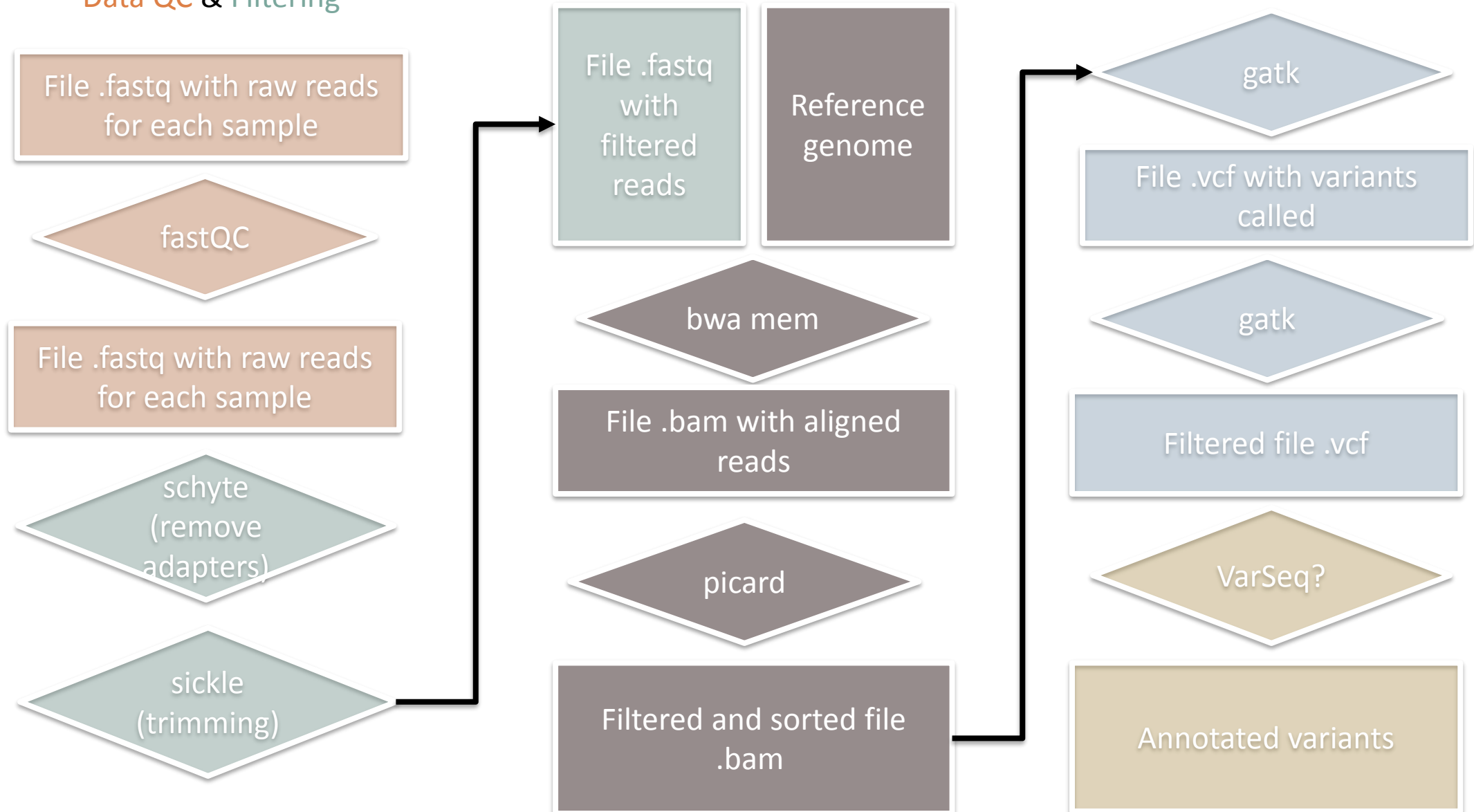
File .vcf with variants  
called

gatk

Filtered file .vcf

VarSeq?

Annotated variants



# Alignment

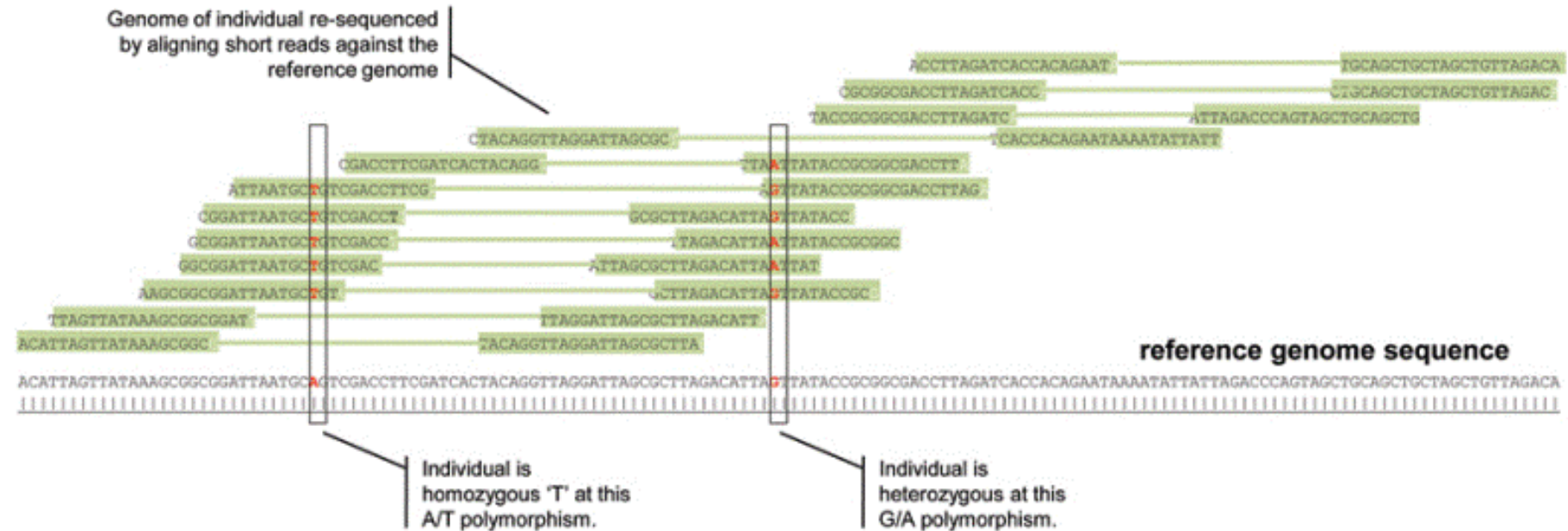
Many of the next-generation sequencing projects begin with a known, or so-called 'reference', genome. In this case, to make sense of the reads, their positions within the reference sequence must be determined. This process is known as aligning or 'mapping' the read to the reference.

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

Header

Body

```
@HD VN:1.3 SO:coordinate
@SQ SN:chr2 LN:243199373
@PG ID:bwa PN:bwa VN:0.7.15-r1140 CL:bwa mem chr2.fasta read1.trimmed.fastq.gz read2.trimmed.fastq.gz
NB500897:75:H5NWWBGXY:3:13608:21224:10540 2147 chr2 12323980 0 57H38M21H = 69870261 57546346 GGCTGAGGTGGGAG
NB500897:78:H5T5FBGXY:4:11401:24478:5392 2131 chr2 25275067 0 108H33M9H = 69650885 44375787 AAAAATAAAAAATAA
NB500897:75:H5NWWBGXY:3:12601:23290:7800 2115 chr2 33141345 0 94H56M = 68269866 35128522 GGGGGGGGGGGGGGGGGGGGGG
NB500897:78:H5T5FBGXY:3:12609:16449:18291 2115 chr2 33141438 0 120H30M = 68547068 35405631 GGGGGGGGGGGGGGGGGGGGGG
NB500897:75:H5NWWBGXY:1:13104:6732:16100 2115 chr2 33141478 0 120H30M = 68547068 35405591 GGGGGGGGGGGGGGGGGGGGGG
NB500897:78:H5T5FBGXY:4:12411:14105:19392 2147 chr2 33141574 0 100H50M = 69871302 36729810 GGGGGGGGGGGGGGGGGGGGGG
NB500897:75:H5NWWBGXY:2:23311:23906:6768 2115 chr2 33141581 0 87H43M = 68269811 35128231 GGGGGGGGGGGGGGGGGGGGGG
NB500897:75:H5NWWBGXY:1:12106:7875:15761 2115 chr2 33141594 0 120H30M = 68547068 35405475 GGGGGGGGGGGGGGGGGGGGGG
NB500897:78:H5T5FBGXY:3:22601:12050:10673 161 chr2 50626928 0 2S64M = 69890571 19263716 AAAAAAAAAAAAAAAAAAAAAA
NB500897:75:H5NWWBGXY:4:12607:26302:14475 81 chr2 52194430 0 78M = 69688660 17494154 ATGTTGGCGAGGCTGGTCTCCA
NB500897:78:H5T5FBGXY:4:23607:18581:7865 163 chr2 58355595 0 26M = 58355665 151 TAGCTGGGATTACAGGTGTGTGCCAC
NB500897:78:H5T5FBGXY:4:23607:18581:7865 83 chr2 58355665 0 4S81M13S = 58355595 -151 CTCACCATGTTGCCAGGCTGG
NB500897:75:H5NWWBGXY:3:13608:21224:10540 2179 chr2 65437823 0 63M53H = 69870261 4432439 ACTATGCTGACCAGTTGGTTTCAAATTCC
NB500897:75:H5NWWBGXY:2:11103:13114:1410 99 chr2 67999995 60 66M = 68000128 259 CTCACAATAAATTTATTTTTTCAAAGCAG
NB500897:75:H5NWWBGXY:2:11103:13114:1410 147 chr2 68000128 60 126M = 67999995 -259 CATCTAGATAGCTATCTTTCCAGACTTTTC
NB500897:78:H5T5FBGXY:1:21201:19536:13168 99 chr2 68000446 60 150M = 68000452 156 CCTCAGAGTATTAAGACCACATAGTATAT
NB500897:78:H5T5FBGXY:4:23607:24808:1972 99 chr2 68000446 60 150M = 68000452 156 CCTCAGAGTATTAAGACCACATAGTATAT
NB500897:78:H5T5FBGXY:1:21201:19536:13168 147 chr2 68000452 60 150M = 68000446 -156 AGTATTAAGACCACATAGTATATATTTTC
NB500897:78:H5T5FBGXY:4:23607:24808:1972 147 chr2 68000452 60 150M = 68000446 -156 AGTATTAAGACCACATAGTATATATTTTC
NB500897:75:H5NWWBGXY:4:12408:9893:6566 163 chr2 68001096 60 150M = 68001096 150 AGATTCCTGGCTAAATTCACCATTGAAAGAAATTGA
NB500897:75:H5NWWBGXY:4:12408:9893:6566 83 chr2 68001096 60 150M = 68001096 -150 AGATTCCTGGCTAAATTCACCATTGAAAGAAATTGA
NB500897:78:H5T5FBGXY:2:12107:7865:18634 99 chr2 68001412 60 150M = 68001549 287 TTCCTAACTAAATACTGACTAGAACAGTGA
NB500897:78:H5T5FBGXY:2:12107:7865:18634 147 chr2 68001549 60 150M = 68001412 -287 CCCAACTCCTGCCACTCTAGCCACATCAAG
NB500897:78:H5T5FBGXY:1:12109:14488:19284 99 chr2 68001927 60 150M = 68002195 418 ACGTAGTAGAAATTCACAGAATACTTGTA
NB500897:78:H5T5FBGXY:1:12305:16077:11340 163 chr2 68001976 60 150M = 68002289 462 AATTCCTCACCAGCTTCAGCAGCTTAAGGA
NB500897:75:H5NWWBGXY:1:22202:17880:17856 99 chr2 68001995 60 150M = 68002066 221 CAGCTTAAGGATAAAGAATCTTGCATCTAA
```

Read ID

Mapping  
Tag

Mapping  
position

Mapping quality

CIGAR  
string

Pair  
information

# 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





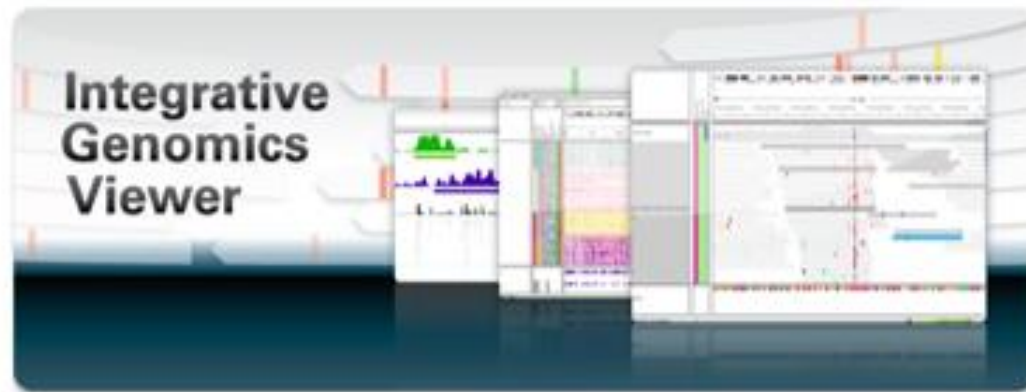
- Home
- Downloads
- Documents
  - IGV User Guide
  - Tutorial Videos
  - File Formats
  - Hosted Genomes
  - FAQ
  - Release Notes
  - Credits
- Contact

Search website

search

© 2013-2018  
Broad Institute  
and the Regents of the  
University of California

Home



## Overview

The **Integrative Genomics Viewer (IGV)** is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets. It supports a wide variety of data types, including array-based and next-generation sequence data, and genomic annotations.

IGV is available in multiple forms, including:

- the original **IGV** - a Java desktop application,
- IGV-Web** - a web application,
- igv.js** - a JavaScript component that can be embedded in web pages (for developers)

This site is focused on the IGV desktop application. See <https://igv.org> for links to all forms of IGV.

## Download IGV



Download the IGV desktop application and igvtools.

Note that the IGV-Web application at <https://igv.org/app> runs in a web browser and requires no downloads. Click on the Help link in the app for more information.

## Citing IGV

To cite your use of IGV in your publication, please reference one or more of:

James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. [Integrative Genomics Viewer](#). *Nature Biotechnology* 29, 24-26 (2011). (Free PMC article [here](#)).

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. [Integrative Genomics Viewer \(IGV\): high-performance genomics data visualization and exploration](#). *Briefings in Bioinformatics* 14, 178-192 (2013).

James T. Robinson, Helga Thorvaldsdóttir, Aaron M. Wenger, Ahmet Zehir, Jill P. Mesirov. [Variant Review with the Integrative Genomics Viewer \(IGV\)](#). *Cancer Research* 77(21):31-34 (2017).

## Funding

Development of IGV has been supported by funding from the [National Cancer Institute \(NCI\)](#) of the [National Institutes of Health](#), the [Informatics Technology for Cancer Research \(ITCR\)](#) of the NCI, and the [Star Cancer Consortium](#).

IGV participates in the [GenomeSpace](#) initiative, which is funded by the [National Human Genome Research Institute](#).

# 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
```

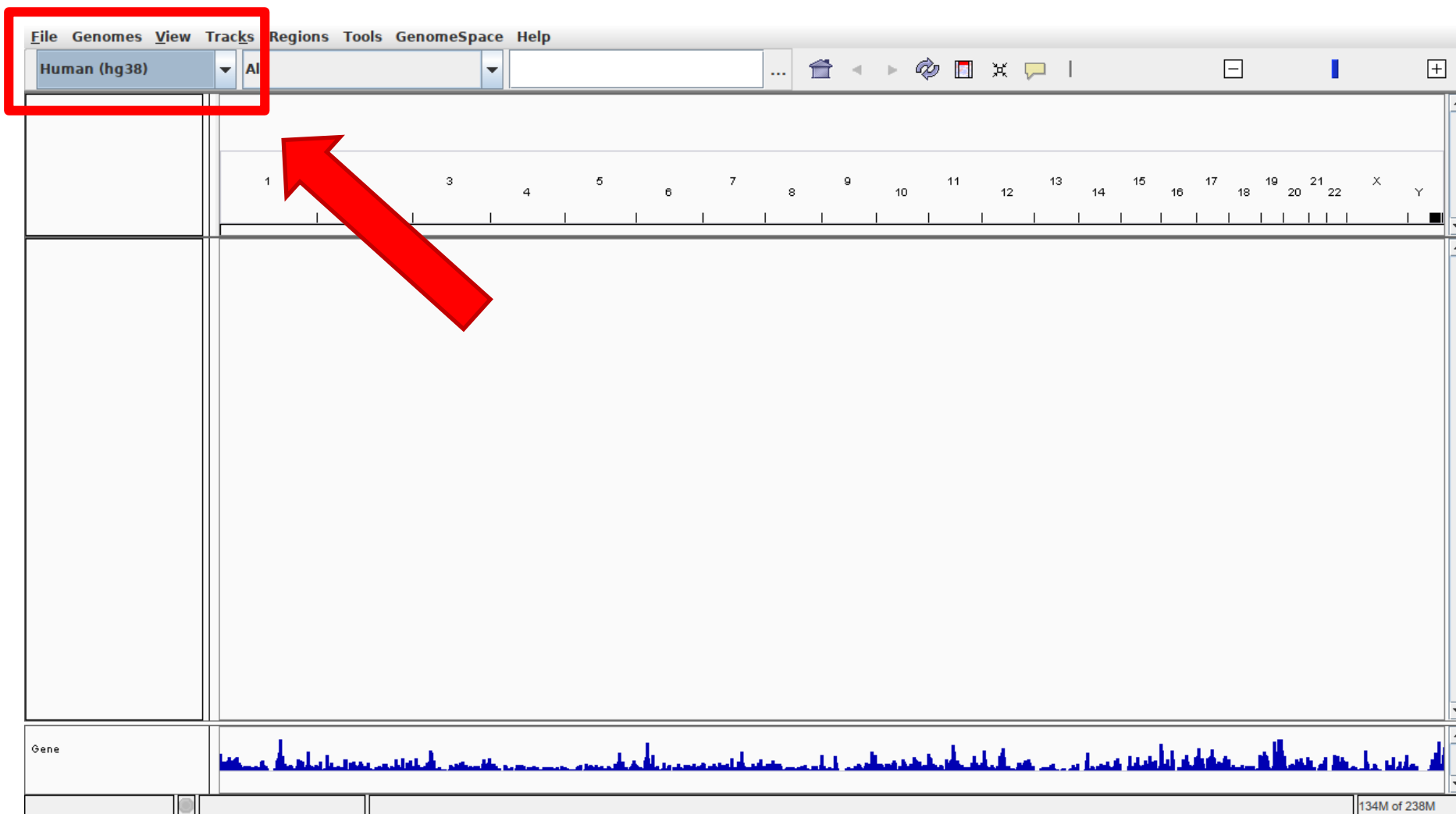
```
rsync -auv HGSI2020@157.27.26.214:/attachedvolume/HGSI2020/Denise/sample.sorted.bam* .
```

- Check if you have downloaded: `ls`

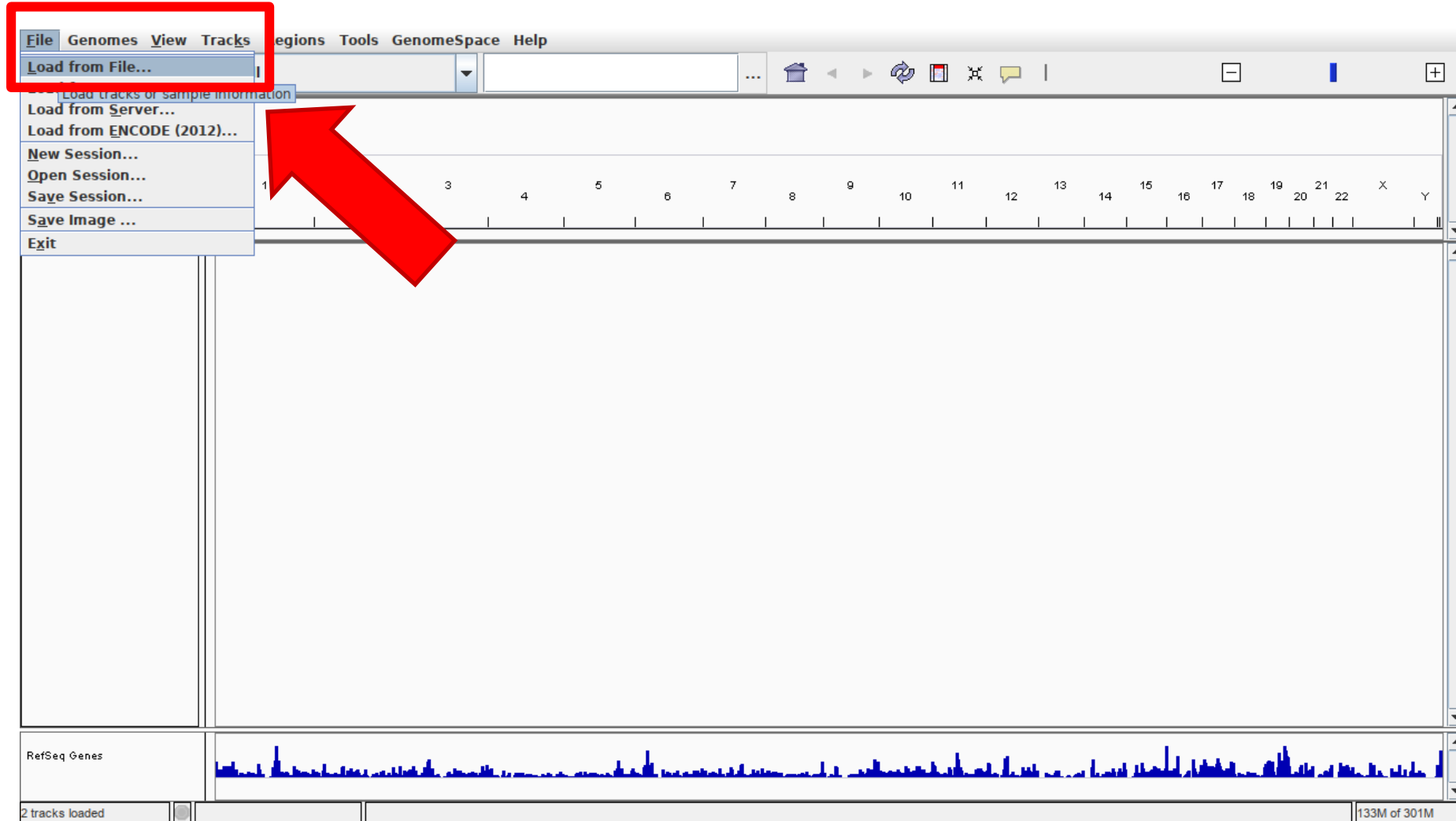
- Open IGV

```
./igv.sh for Ubuntu
```

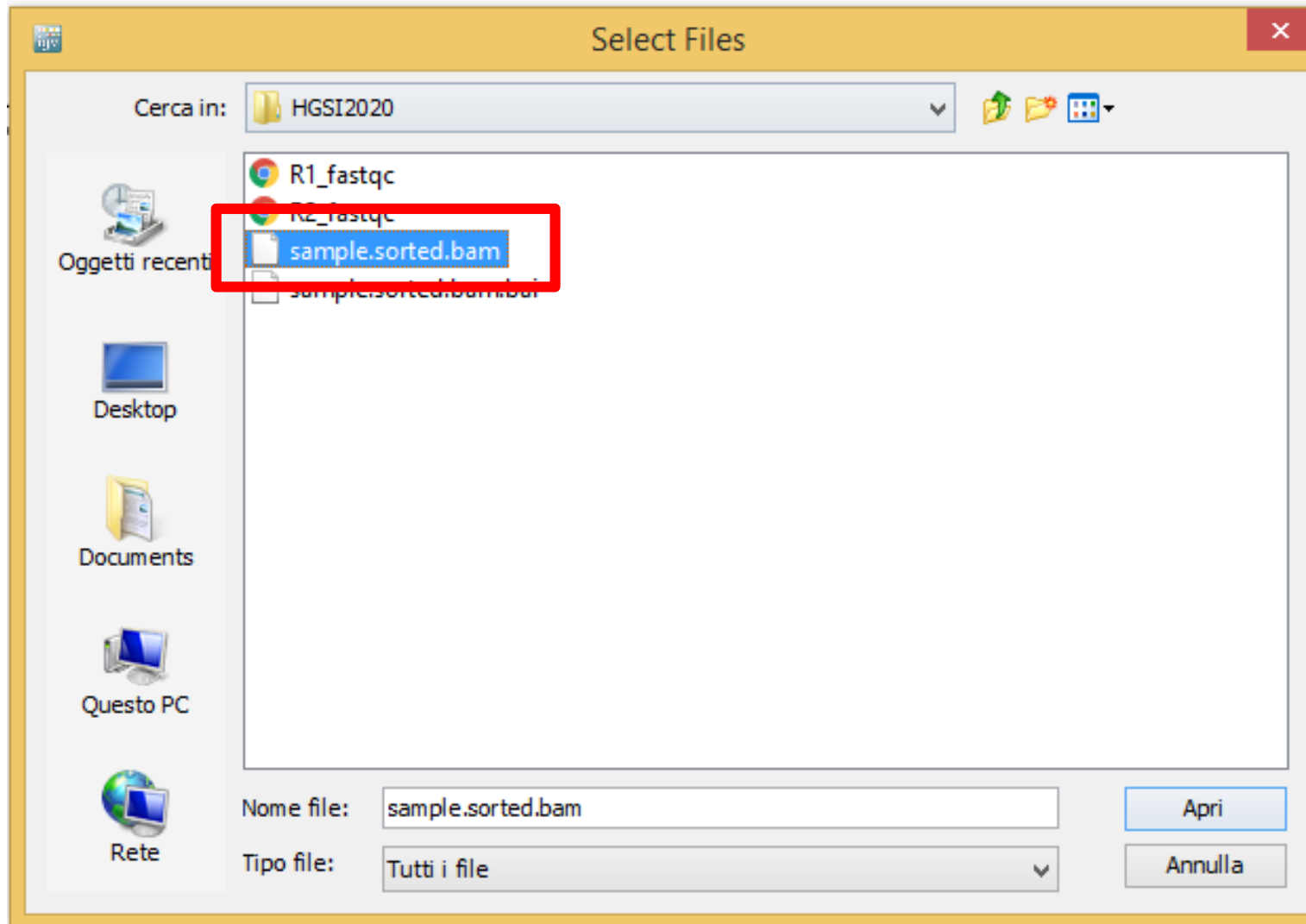
# Choose the right genome



# Upload the bam



Go into the folder "HGS12020",  
choose the file bam and open it



# Search a specific region

The screenshot displays a genomic browser interface with a menu bar at the top containing 'File', 'Genomes', 'View', 'Tracks', 'Regions', 'Tools', 'GenomeSpace', and 'Help'. Below the menu, a dropdown menu is set to 'Human (hg38)', and another dropdown is set to 'All'. A text input field contains the search query 'chr6:289,015-307,482', which is highlighted by a red rectangular box. To the right of the input field is a 'Go' button and a series of navigation icons. Below the navigation bar is a chromosome ideogram showing chromosomes 1 through 22, X, and Y. A large red arrow points from the text 'We search the region: chr6:289,015-307,482' to the search input field. The main panel of the browser is divided into three tracks: 'sample.sorted.bam Coverage' (showing a coverage plot), 'sample.sorted.bam' (showing raw sequencing data), and 'Gene' (showing gene models). The bottom status bar indicates '4 tracks' on the left and '154M of 291M' on the right.

Human (hg38) All chr6:289,015-307,482 Go

1 2 3 4 5 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 X Y

sample.sorted.bam Coverage

sample.sorted.bam

Gene

4 tracks 154M of 291M

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

# Results





# Difference between genome and exome sequencing

