SPAAM

Standards, Precautions & Advances in Ancient Metagenomics

Practical 4B: Genome mapping

Alexander Herbig & Alina Hiss

Intro: What is mapping?

- Alignment of DNA sequencing reads to a reference genome
- Suitable for comparing closely related genomes
- Identification of variation/differences between genomes
- Identified variants serve as input for down-stream analyses (such as phylogenomics or functional analyses)

Intro: Alignment Basics

Sequence 1:

ACGAAGTAGCAGACGATATAGC

Sequence 2:

ACGCAGTAGAGGATAGCGTACC

Alignment:

```
ACGAAGTAGCAGACGATA---TAGC
||| ||||   || ||||   || |
ACGCAGTA---GAGGATAGCGTACC
```

9 Modifications
(Edit Distance)

Intro: Reference Mapping Concept

ACATCGACGA

GACGACATAC

GCTAGACAT

AGGCTACGCTA

ATACCTAGGC

GCTAGCTAGCGTAG

Intro: Reference Mapping Concept

...GCTAGACATCGACGACATACCTAGGCTACGCTAGCTAGCGTAG...

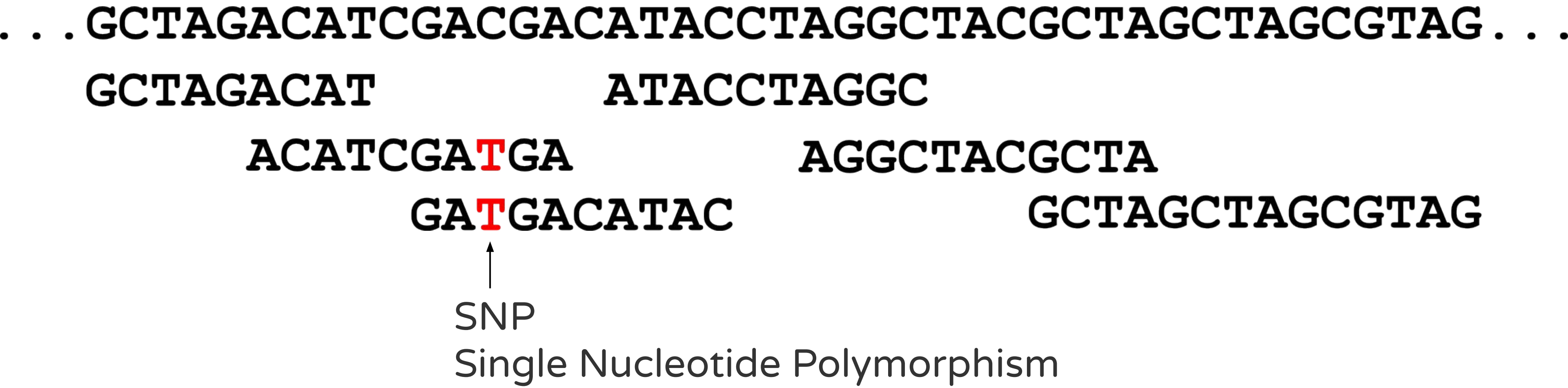
Intro: Reference Mapping Concept

```
...GCTAGACATCGACGACATACCTAGGCTACGCTAGCTAGCGTAG...
   GCTAGACAT
```

Intro: Reference Mapping Concept

```
...GCTAGACATCGACGACATACCTAGGCTACGCTAGCTAGCGTAG...
   GCTAGACAT       ATACCTAGGC
                         AGGCTACGCTA
                                GCTAGCTAGCGTAG
```

Intro: Reference Mapping Concept

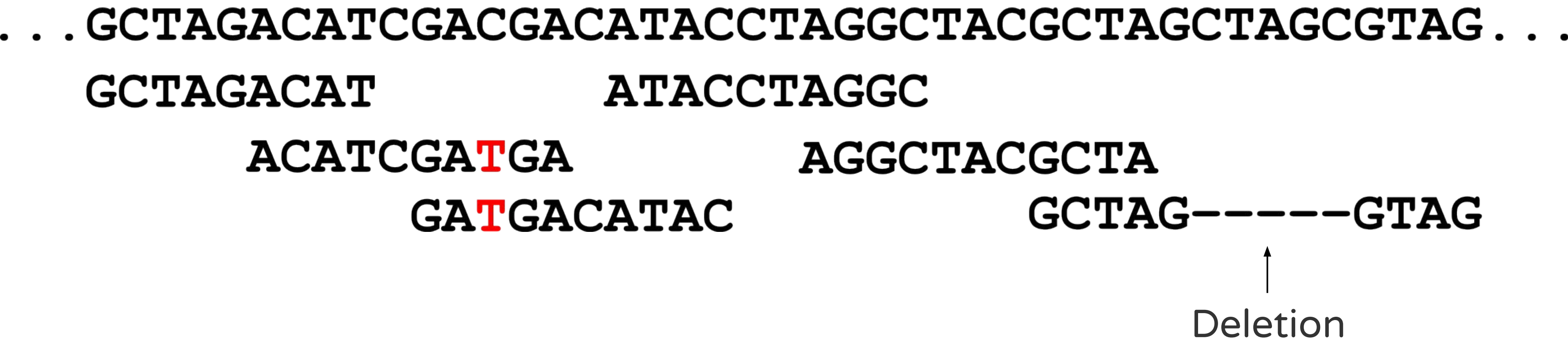


```
...GCTAGACATCGACGACATACCTAGGCTACGCTAGCTAGCGTAG...
   GCTAGACAT       ATACCTAGGC
        ACATCGATGA       AGGCTACGCTA
             GATGACATAC         GCTAGCTAGCGTAG
```

SNP

Single Nucleotide Polymorphism

Intro: Reference Mapping Concept



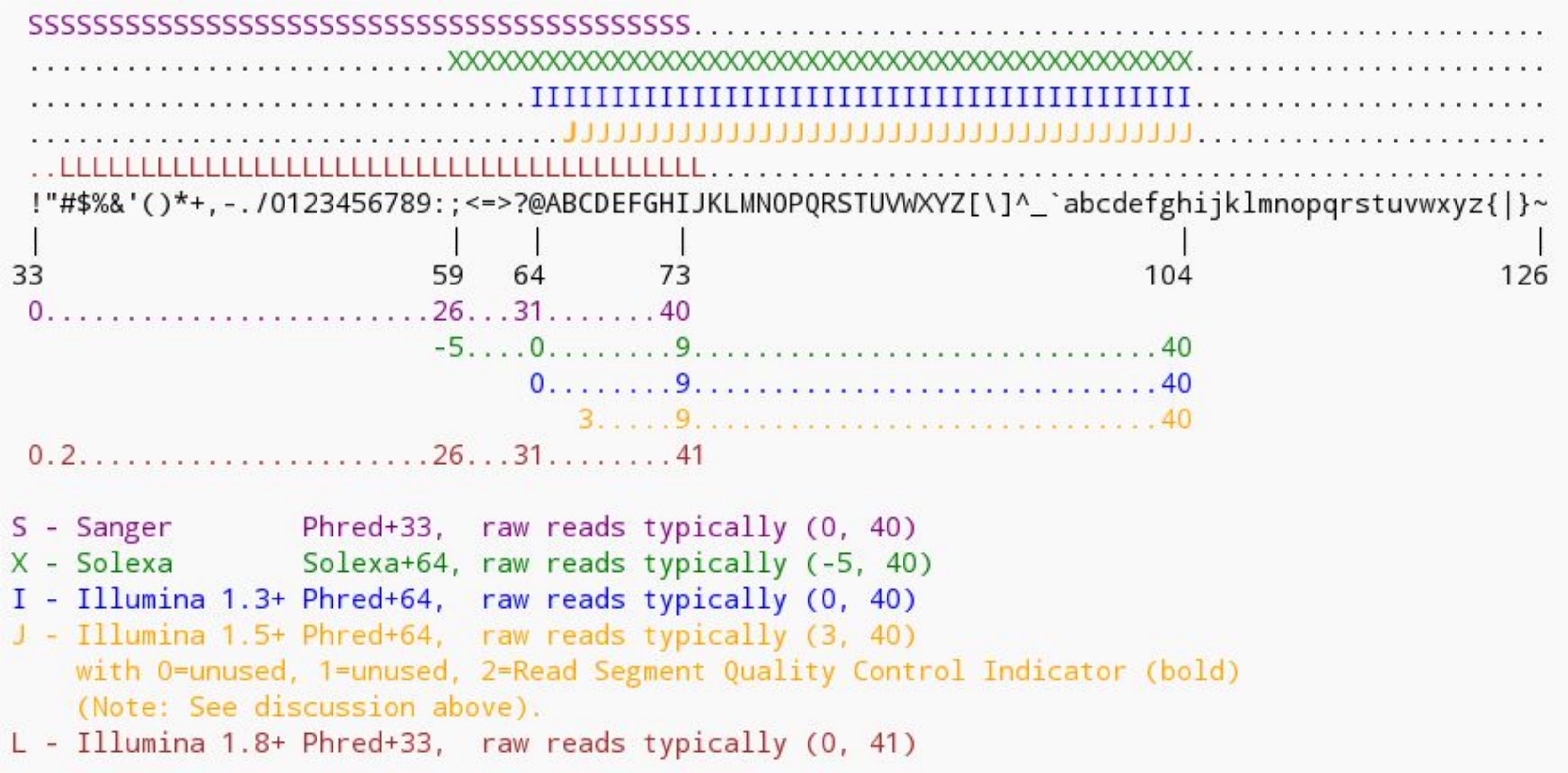
```
...GCTAGACATCGACGACATACCTAGGCTACGCTAGCTAGCGTAG...
   GCTAGACAT       ATACCTAGGC
        ACATCGATGA       AGGCTACGCTA
             GATGACATAC         GCTAG-----GTAG
```

Intro: Input Format

- FASTQ (Sequence and Quality)

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

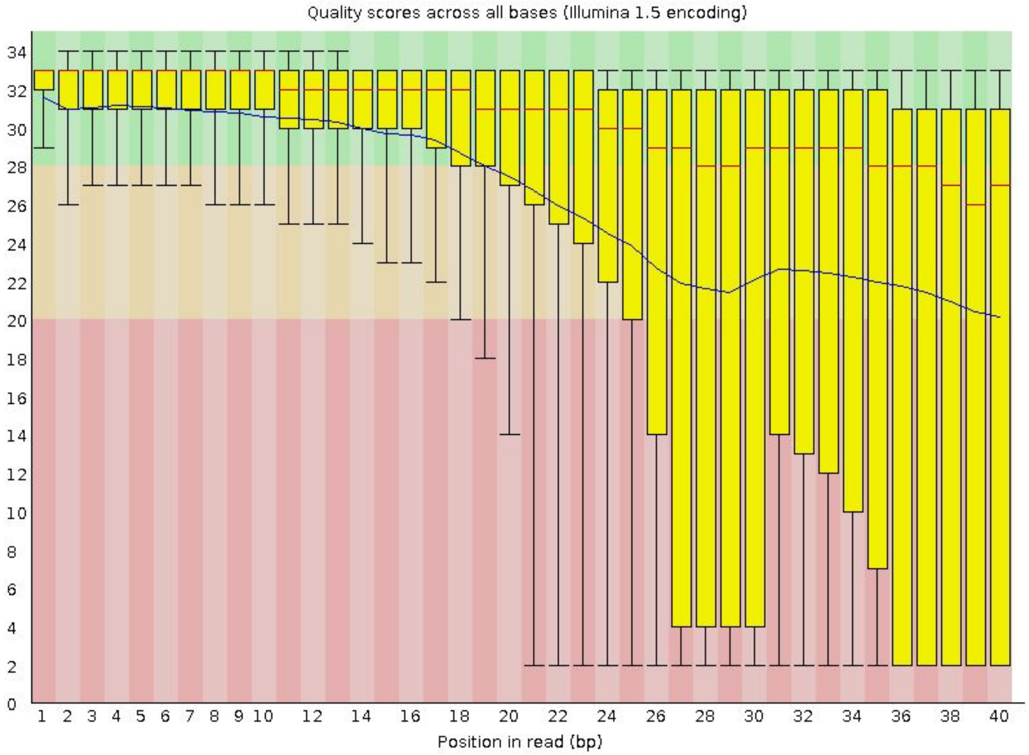
Intro: FASTQ - Quality Encoding



```
 SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS.....................................................
 ..........................XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX......................
 ...............................IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII......................
 .................................JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ......................
 ..LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL....................................................
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126
 0........................26...31.......40
                          -5....0........9.............................40
                                0........9.............................40
                                   3.....9.............................40
 0.2......................26...31........41

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)
```

Intro: FASTQ - Quality Encoding

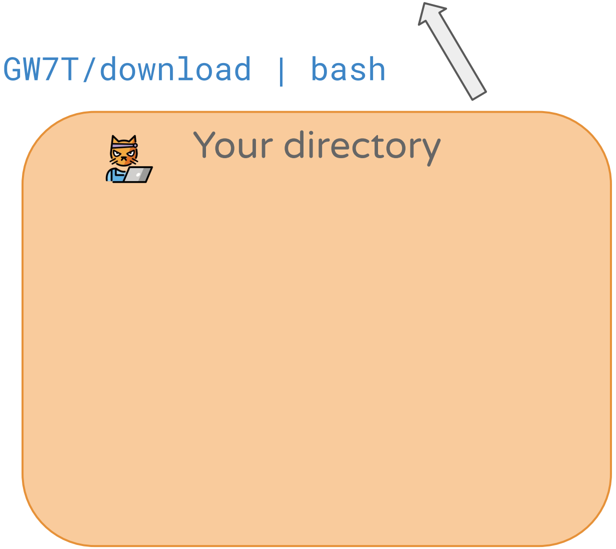


Practical part: Preparation

“Your directory” in this case is “4b-genome-mapping”

- Recover data (everyone)

```
curl -s https://share.eva.mpg.de/index.php/s/gfeY84DHWFJGW7T/download | bash
```



- Go into directory for this session

```
cd /vol/volume/4b-genome-mapping
```

- Get the most recent file with the commands…

```
wget https://share.eva.mpg.de/index.php/s/p9HY4W2aiGD5xk5/download -O commands_UPDATED.txt
```

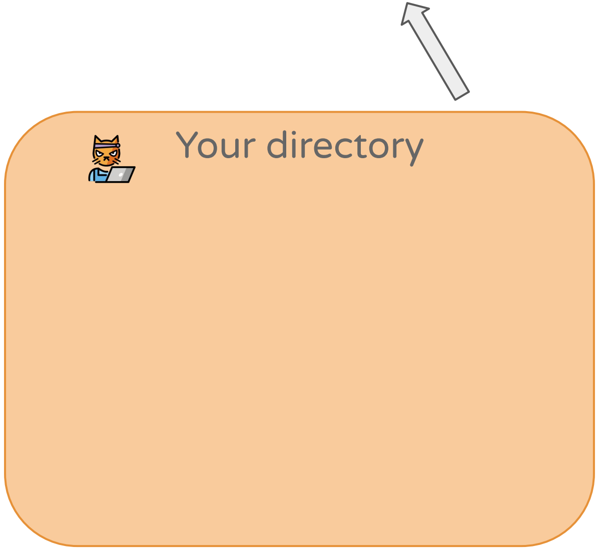
Practical part: Preparation

- Now you’re all set, just one more thing:

  activate conda environment

```
conda activate microbial-genomics
```

“Your directory” in this case is “4b-genome-mapping”



Intro: Burrows-Wheeler Alignment

- We will be using the **Burrows-Wheeler Aligner**
- Li et al. 2009 – http://bio-bwa.sourceforge.net/
- Different algorithms implemented for different types of data (diff. read lengths)
    - Here: BWA backtrack (bwa aln) → suitable for Illumina sequences up to 100bp
    - Others: bwa mem and bwa sw for longer reads

Intro: prior mapping: Reference genome

- We need a reference genome in FASTA format
- Ideally of the organism we want to map to, if not available, closely related species
- Download reference from database, e.g. NCBI

Practical part: Reference genome - Indexing

- In your directory, you can find 2 samples and your reference (and the commands file).
- First step: index reference genome (make sure you are inside your directory)

```
bwa index YpestisCO92.fa

samtools faidx YpestisCO92.fa

picard CreateSequenceDictionary R=YpestisCO92.fa
```

Your directory

reference.fa

sample1

sample2

Intro: Parameters

- We will be using bwa aln, but which parameters are specifically relevant?
  - Seed length
  - Maximum edit distance

- Rest can be set to default for now
- Parameters settings depend on the type of data, we generally differentiate between strict and lenient mapping parameters

Intro: Parameters - Seed length

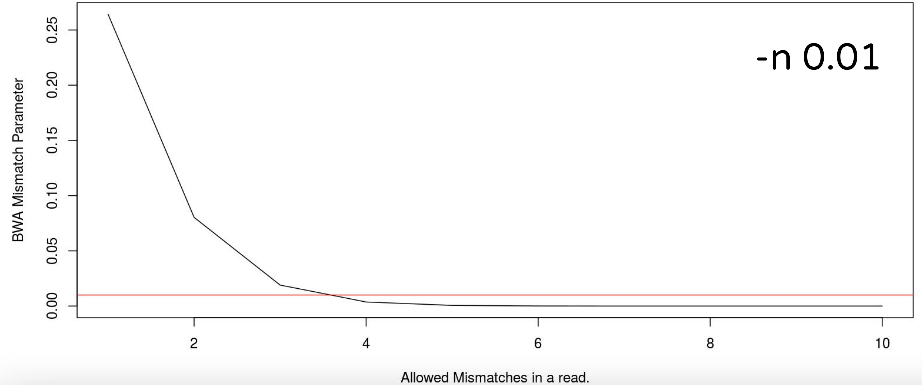
- “Seed-and-Extend” algorithm
- Sequence of first N bases used to find hit in ref. genome
- Seeding speeds up alignment
- Can be disabled by setting a long seed (e.g. -l 1024 → longer than reads)

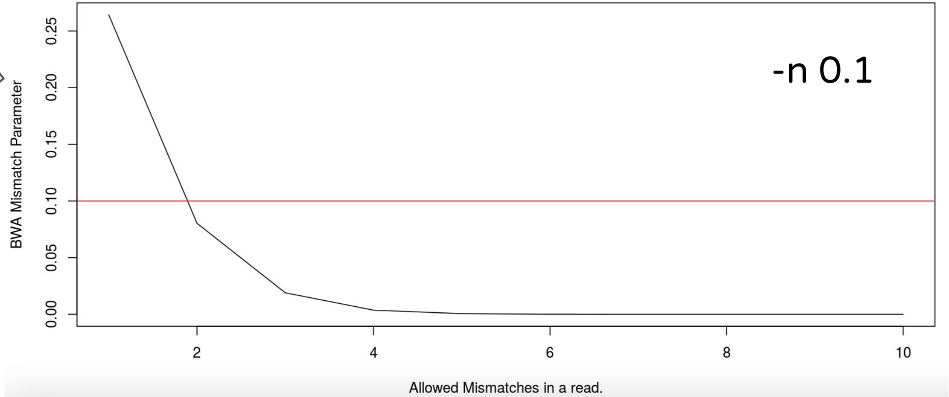
Intro: Parameters - Seed length

- Short seed:
    - maps to more possible positions in ref genome
    - less accurate, allows for more differences
    - longer run time
- Long seed:
    - maps to less positions
    - more accurate but chance of missing less “perfect” mapping positions in genome
    - faster

Intro: Parameters - Maximum Edit Distance

- How many mismatches allowed in a read
- Top example, 3 mismatches allowed → lenient mapping
- Bottom example, 2 mismatches allowed → strict mapping
- Scales with read length, here set to 50





https://apeltzer.shinyapps.io/bwa-mismatches/

Intro: Parameters - lenient vs. strict mapping

- Lenient
  - Allow for more mismatches → -n 0.01
  - Short seed length → -l 16

- Strict
  - Allow for less mismatches → -n 0.1
  - Long seed length → -l 32

Practical part: mapping to a *Y. pestis* genome

- We will be working with pre-processed files: quality-filtered and adapters are already removed
- 2 input files
  - sample1
  - sample2
- One is an ancient genome, one is modern
- Following parameters (2 alignments for each sample)
  - Lenient: -n 0.01 -l 16
  - Strict: -n 0.1 -l 32

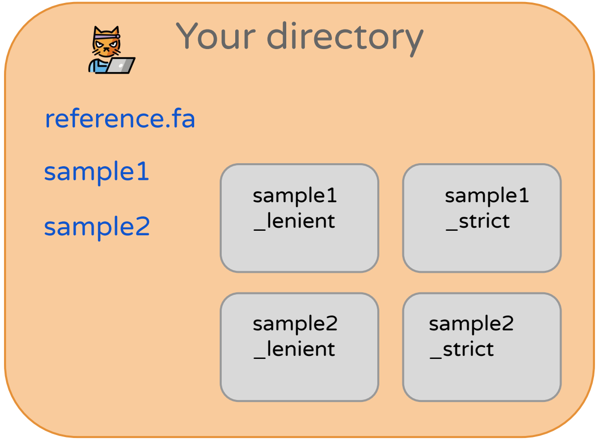
Practical part: preparation

- We will be doing 4 mappings:
  - Genome 1 lenient
  - Genome 2 lenient
  - Genome 1 strict
  - Genome 2 strict

For this, we will make 4 separate directories, to avoid mixing up files. This is not necessary if you always name the output files in an informative way, however, there is also an advantage to having different directories for a later step.

Practical part: preparation

- Make the following 4 directories (you can use other names, just make sure they are informative):
  - sample1_lenient
  - sample2_lenient
  - sample1_strict
  - sample2_strict



Your directory

reference.fa

sample1

sample2

sample1_lenient

sample1_strict

sample2_lenient

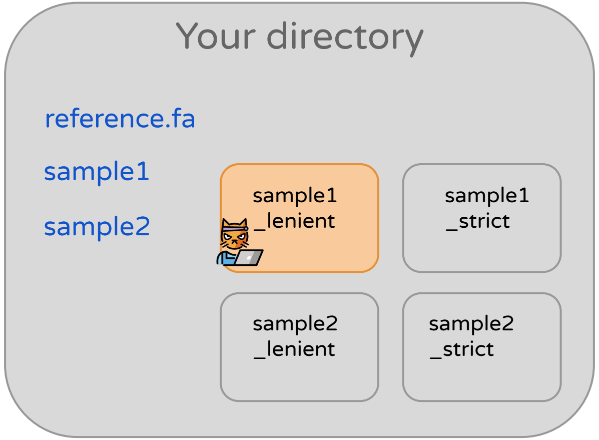
sample2_strict

```
mkdir sample1_lenient sample2_lenient sample1_strict sample2_strict
```

Practical part: mapping to a *Y. pestis* genome

- Let’s begin with a lenient mapping of sample1:
- Go into the corresponding folder

```
cd sample1_lenient
```



Your directory

reference.fa

sample1

sample2

sample1_lenient

sample1_strict

sample2_lenient

sample2_strict

- Create file for bwa alignment, here sample1, specify lenient mapping parameters

```
bwa aln -n 0.01 -l 16 ../YpestisCO92.fa ../sample1.fastq.gz > reads_file.sai
```

Practical part: mapping to a *Y. pestis* genome

- Proceed with the actual mapping, using the created file

```
bwa samse -r '@RG\tID:all\tLB:NA\tPL:illumina\tPU:NA\tSM:NA' ../YpestisCO92.fa
reads_file.sai ../sample1.fastq.gz > reads_mapped.sam
```

- Explanation:
  - -r specifies read group in a certain format

Practical part: mapping to a *Y. pestis* genome

- Convert SAM file to binary format (BAM file)

```
samtools view -b -S reads_mapped.sam > reads_mapped.bam
```

- Background: SAMtools – Exploration, handling and post-processing of SAM files (Li et al. 2009 – http://samtools.sourceforge.net/)
    - -b specifies to output in BAM format
    - (-S to specify input is SAM, can be omitted in recent versions)

Practical part: After mapping - Sorting

- Sort bam file → Sort alignments by leftmost coordinates

```
samtools sort reads_mapped.bam > reads_mapped_sorted.bam
```

- Sorted bam file can be indexed → more efficient for further processing

```
samtools index reads_mapped_sorted.bam
```

Practical part: After mapping - Deduplication

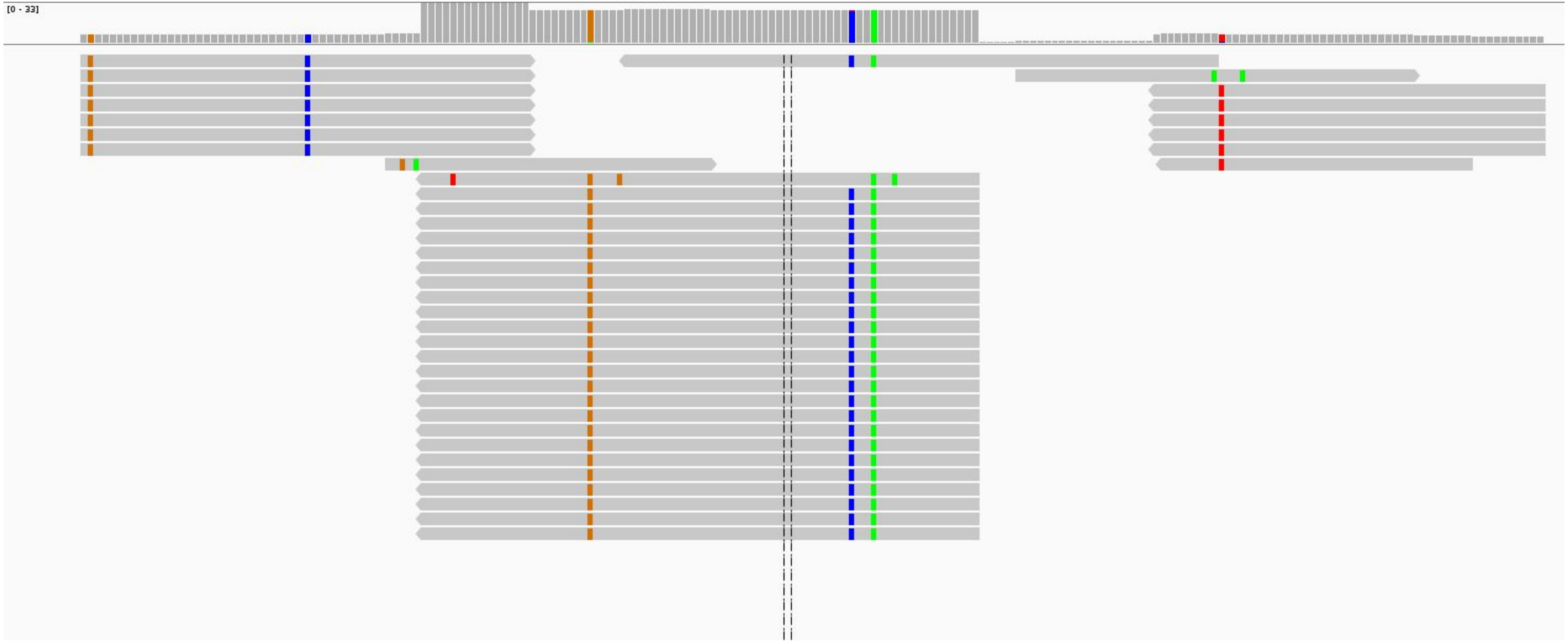
- Deduplication: Removal of reads from duplicated fragments

```
samtools rmdup -s reads_mapped_sorted.bam reads_mapped_sorted_dedup.bam

samtools index reads_mapped_sorted_dedup.bam
```

- Explanation:
  - -s → remove duplicates for single-end reads. By default, the command works for paired-end reads only.
  - Duplicated reads are usually a consequence of amplification of the DNA fragments in the lab, therefore not biologically meaningful

Practical part: After mapping



Practical part: After mapping

- Deduplication: Removal of reads from duplicated fragments

```
samtools rmdup -s reads_mapped_sorted.bam reads_mapped_sorted_dedup.bam

samtools index reads_mapped_sorted_dedup.bam
```

- Let's have a look…

```
samtools view reads_mapped_sorted_dedup.bam | less -S
```

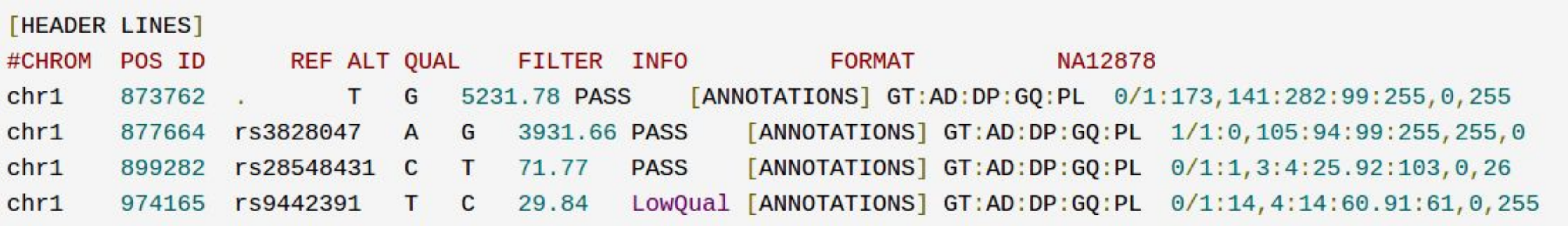
(exit with Q)

```
samtools idxstats reads_mapped_sorted_dedup.bam
```

Intro: Genotyping

- Identification of all SNPs that differentiate a genome from the reference
- Based on read mapping
- GATK – Genome Analysis Toolkit
- DePristo et al. 2011 – http://www.broadinstitute.org/gatk/
- Input: reference genome (fasta); mapping (bam)
- Output: Variant Call Format (vcf)

Intro: VCF Format



```
[HEADER LINES]
#CHROM  POS ID       REF ALT QUAL    FILTER  INFO          FORMAT          NA12878
chr1    873762  .        T   G   5231.78 PASS    [ANNOTATIONS] GT:AD:DP:GQ:PL  0/1:173,141:282:99:255,0,255
chr1    877664  rs3828047   A   G   3931.66 PASS    [ANNOTATIONS] GT:AD:DP:GQ:PL  1/1:0,105:94:99:255,255,0
chr1    899282  rs28548431  C   T   71.77   PASS    [ANNOTATIONS] GT:AD:DP:GQ:PL  0/1:1,3:4:25.92:103,0,26
chr1    974165  rs9442391   T   C   29.84   LowQual [ANNOTATIONS] GT:AD:DP:GQ:PL  0/1:14,4:14:60.91:61,0,255
```

Practical part: Genotyping

- Perform genotyping on mapping file

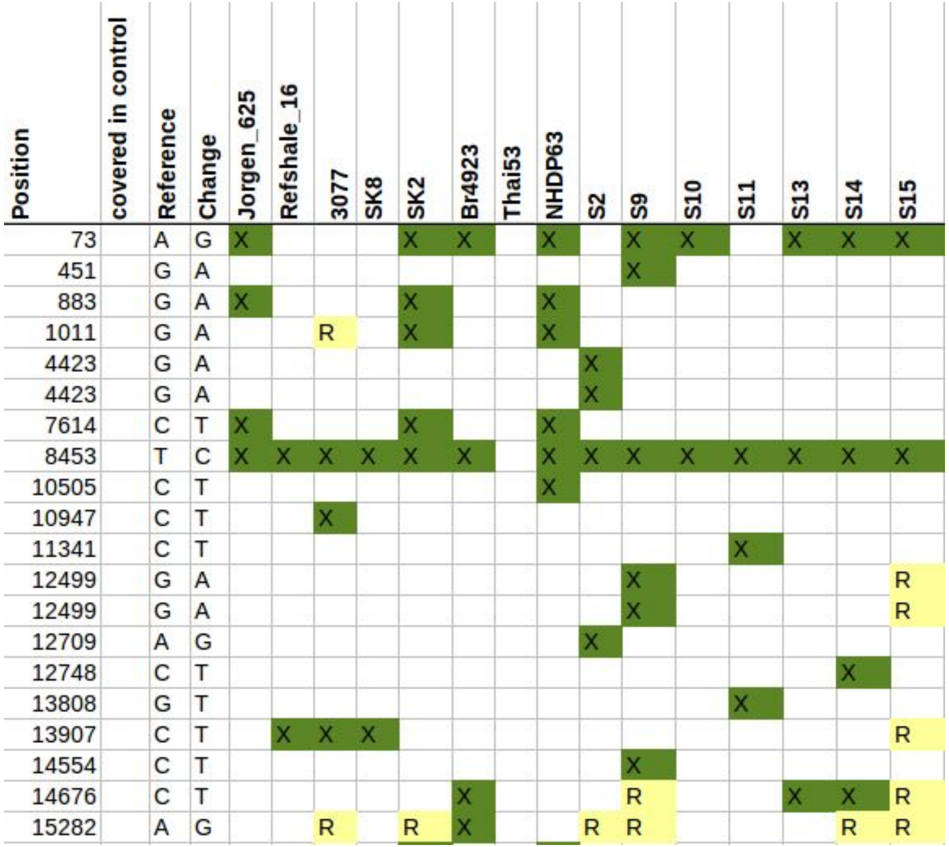
```
gatk3 -T UnifiedGenotyper -R ../YpestisCO92.fa -I reads_mapped_sorted_dedup.bam
--output_mode EMIT_ALL_SITES -o mysnps.vcf
```

- Let’s have a look…

```
cat mysnps.vcf | less -S
```

(exit with Q)

Intro: Comparative SNP Analysis



Position	covered in control	Reference	Change	Jorgen_625	Refshale_16	3077	SK8	SK2	Br4923	Thai53	NHDP63	S2	S9	S10	S11	S13	S14	S15
73		A	G	X				X	X		X		X	X		X	X	X
451		G	A										X					
883		G	A	X				X			X							
1011		G	A			R		X			X							
4423		G	A									X						
4423		G	A									X						
7614		C	T	X				X			X							
8453		T	C	X	X	X	X	X	X		X	X	X	X	X	X	X	X
10505		C	T								X							
10947		C	T			X												
11341		C	T												X			
12499		G	A										X					R
12499		G	A										X					R
12709		A	G									X						
12748		C	T														X	
13808		G	T												X			
13907		C	T		X	X	X											R
14554		C	T										X					
14676		C	T						X				R			X	X	R
15282		A	G			R		R	X			R	R				R	R

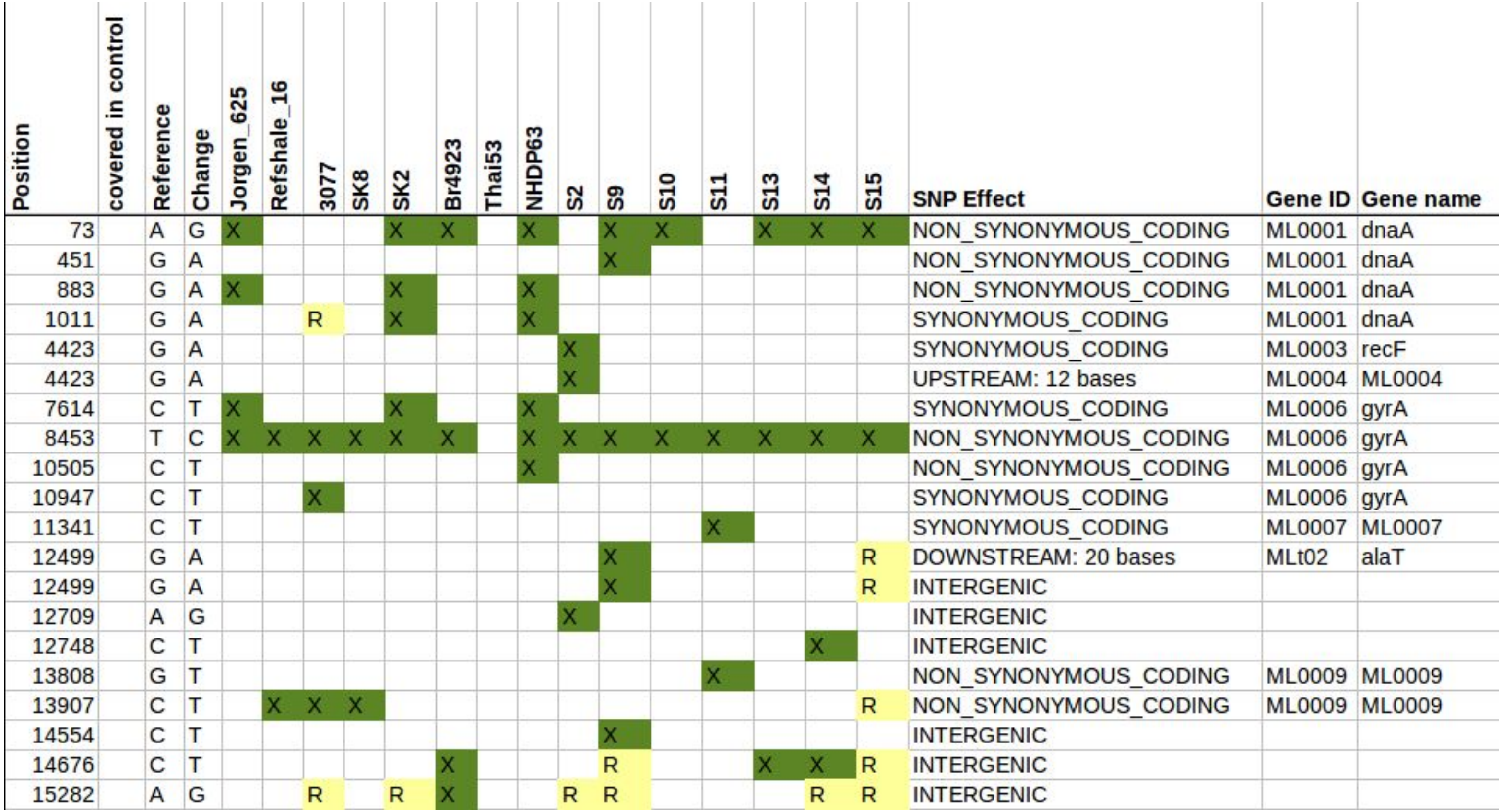
Schuenemann et al. 2013

Intro: Comparative SNP Analysis

- MultiVCFAnalyzer (https://github.com/alexherbig/MultiVCFAnalyzer)
- Gathering SNPs from multiple VCFs for comparative analysis
- Various output formats and summary statistics
- Can integrate gene annotation for SNP effect analysis
- SnpEff – Genetic variant annotation and effect prediction toolbox

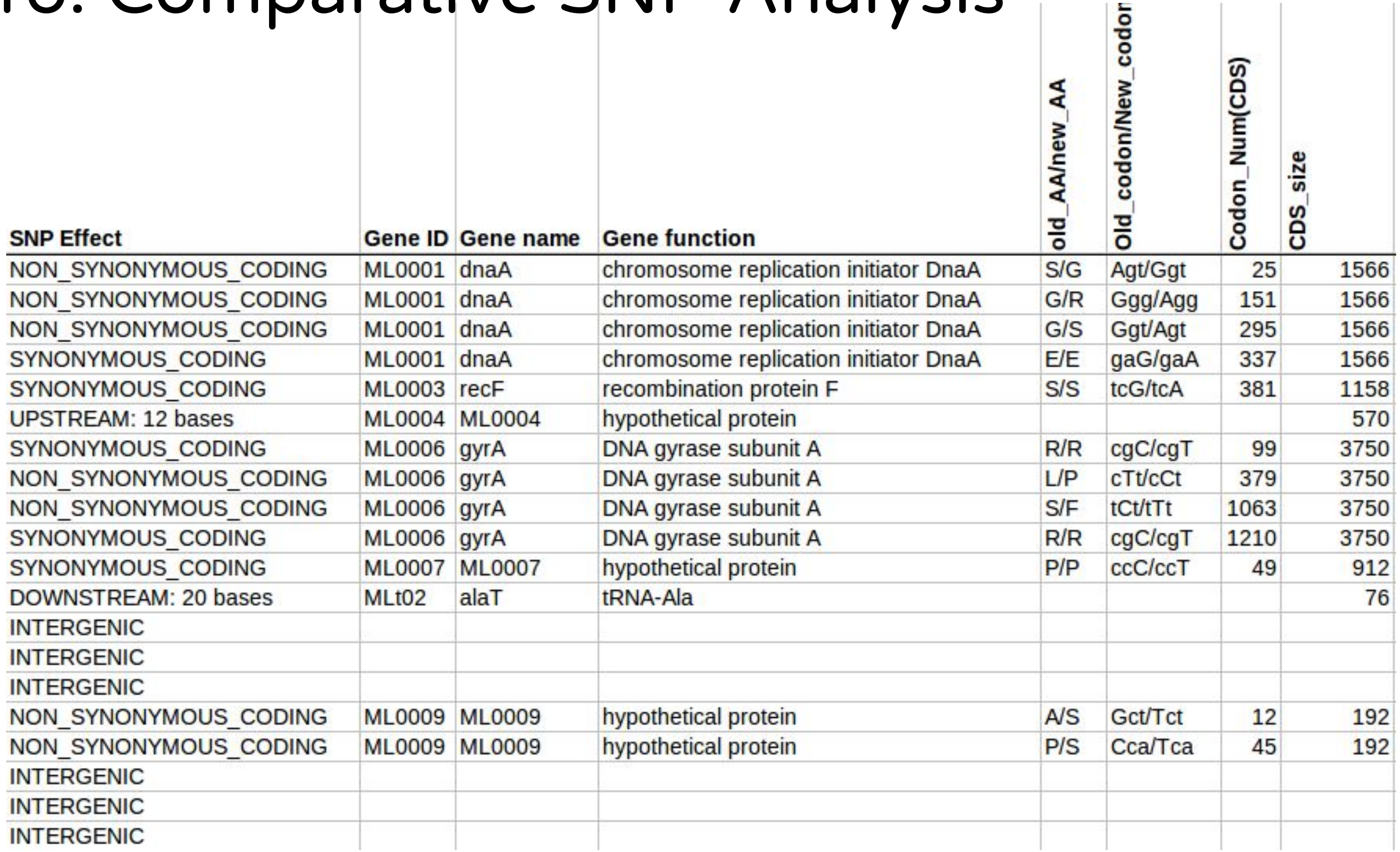
  Cingolani et al. 2012 – http://snpeff.sourceforge.net/

Intro: Comparative SNP Analysis



Position	covered in control	Reference	Change	Jorgen_625	Refshale_16	3077	SK8	SK2	Br4923	Thai53	NHDP63	S2	S9	S10	S11	S13	S14	S15	SNP Effect	Gene ID	Gene name
73		A	G	X				X	X		X		X	X		X	X	X	NON_SYNONYMOUS_CODING	ML0001	dnaA
451		G	A										X						NON_SYNONYMOUS_CODING	ML0001	dnaA
883		G	A	X				X			X								NON_SYNONYMOUS_CODING	ML0001	dnaA
1011		G	A			R		X			X								SYNONYMOUS_CODING	ML0001	dnaA
4423		G	A									X							SYNONYMOUS_CODING	ML0003	recF
4423		G	A									X							UPSTREAM: 12 bases	ML0004	ML0004
7614		C	T	X				X			X								SYNONYMOUS_CODING	ML0006	gyrA
8453		T	C	X	X	X	X	X	X		X	X	X	X	X	X	X	X	NON_SYNONYMOUS_CODING	ML0006	gyrA
10505		C	T								X								NON_SYNONYMOUS_CODING	ML0006	gyrA
10947		C	T			X													SYNONYMOUS_CODING	ML0006	gyrA
11341		C	T												X				SYNONYMOUS_CODING	ML0007	ML0007
12499		G	A										X					R	DOWNSTREAM: 20 bases	MLt02	alaT
12499		G	A										X					R	INTERGENIC		
12709		A	G									X							INTERGENIC		
12748		C	T														X		INTERGENIC		
13808		G	T												X				NON_SYNONYMOUS_CODING	ML0009	ML0009
13907		C	T		X	X	X											R	NON_SYNONYMOUS_CODING	ML0009	ML0009
14554		C	T										X						INTERGENIC		
14676		C	T						X				R			X	X	R	INTERGENIC		
15282		A	G			R		R	X			R	R				R	R	INTERGENIC		

Intro: Comparative SNP Analysis



SNP Effect	Gene ID	Gene name	Gene function	old_AA/new_AA	Old_codon/New_codor	Codon_Num(CDS)	CDS_size
NON_SYNONYMOUS_CODING	ML0001	dnaA	chromosome replication initiator DnaA	S/G	Agt/Ggt	25	1566
NON_SYNONYMOUS_CODING	ML0001	dnaA	chromosome replication initiator DnaA	G/R	Ggg/Agg	151	1566
NON_SYNONYMOUS_CODING	ML0001	dnaA	chromosome replication initiator DnaA	G/S	Ggt/Agt	295	1566
SYNONYMOUS_CODING	ML0001	dnaA	chromosome replication initiator DnaA	E/E	gaG/gaA	337	1566
SYNONYMOUS_CODING	ML0003	recF	recombination protein F	S/S	tcG/tcA	381	1158
UPSTREAM: 12 bases	ML0004	ML0004	hypothetical protein				570
SYNONYMOUS_CODING	ML0006	gyrA	DNA gyrase subunit A	R/R	cgC/cgT	99	3750
NON_SYNONYMOUS_CODING	ML0006	gyrA	DNA gyrase subunit A	L/P	cTt/cCt	379	3750
NON_SYNONYMOUS_CODING	ML0006	gyrA	DNA gyrase subunit A	S/F	tCt/tTt	1063	3750
SYNONYMOUS_CODING	ML0006	gyrA	DNA gyrase subunit A	R/R	cgC/cgT	1210	3750
SYNONYMOUS_CODING	ML0007	ML0007	hypothetical protein	P/P	ccC/ccT	49	912
DOWNSTREAM: 20 bases	MLt02	alaT	tRNA-Ala				76
INTERGENIC							
INTERGENIC							
INTERGENIC							
NON_SYNONYMOUS_CODING	ML0009	ML0009	hypothetical protein	A/S	Gct/Tct	12	192
NON_SYNONYMOUS_CODING	ML0009	ML0009	hypothetical protein	P/S	Cca/Tca	45	192
INTERGENIC							
INTERGENIC							
INTERGENIC							

Practical part: MultiVCFAnalyzer

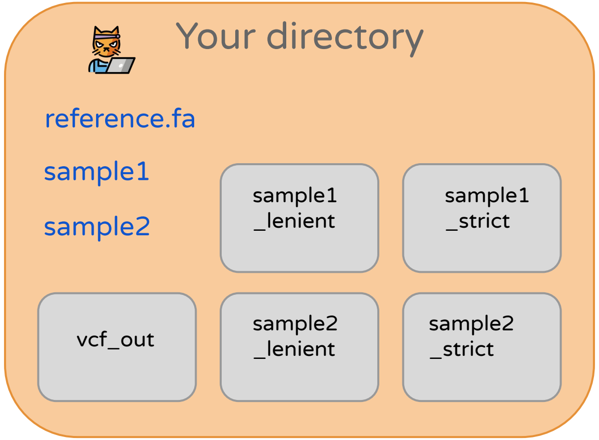
- Run MultiVCFAnalyzer on all 4 files at once
  - First cd one level up (if you type `ls` you should see your 4 directories, reference, etc.)

```
cd ..
```

  - Then make a new directory…

```
mkdir vcf_out
```

  - …and run the programme


Your directory

reference.fa

sample1

sample2

```
multivcfanalyzer NA YpestisCO92.fa NA vcf_out F 30 3 0.9 0.9 NA
sample1_lenient/mysnps.vcf sample1_strict/mysnps.vcf sample2_lenient/mysnps.vcf
sample2_strict/mysnps.vcf
```

Practical part: MultiVCFAnalyzer

- Let’s have a look in the ‘vcf_out’ directory (cd into it)
  - Check the parameters we set earlier:

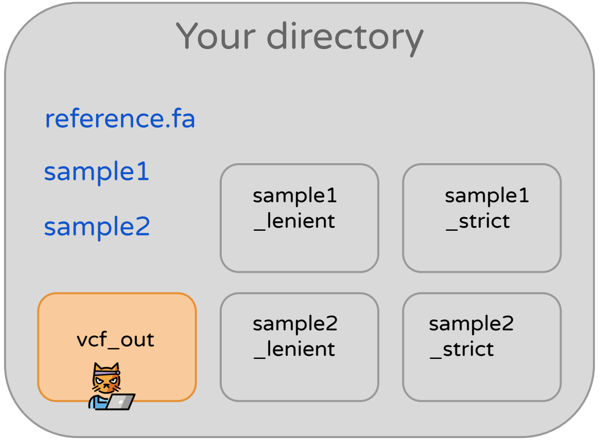
    ```
    less -S info.txt
    ```

    (exit with Q)

  - Check results:

    ```
    less -S snpStatistics.tsv
    ```

    (exit with Q)

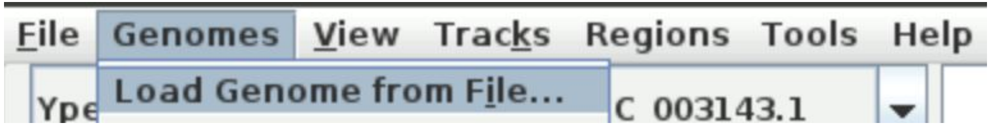


Practical part: IGV

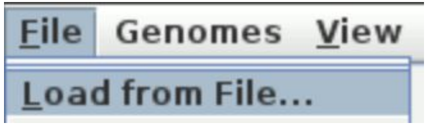
- Let’s have a look at our bam files with IGV (Integrative Genomics Viewer)
  - To open IGV, simply type the following command and the app will open:

```
igv    #(beware that you cannot use the terminal while IGV is open. If you want
to use it anyways, open a second terminal via the bar on the bottom)
```

- Load your reference (YpestisCO92.fa)
  - → Genomes - Load Genome from File



- Load your vcf files (do this 4 times, for all 4 mappings)
  - → File - Load from File



Practical part: Assignments

- What differences do you observe between the samples and parameters?
  - Differences in number of mapped reads, coverage, number of SNPs
  - Do you see any global patterns?
  - Which sample is more affected by changing the parameters?
  - Which of the two samples might be ancient, which is modern?
- Let’s examine some SNPs
  - Have a look at `snpTable.tsv`
  - Can you identify SNPs that were called with lenient but not with strict parameters?
  - Let’s check out some of these in IGV.
  - Do you observe certain patterns in these genomic regions?

Practical part: Clean up

- Deconnect from your conda environment

```
conda deactivate
```