Basics of genomics & quality control: I

1st EvoGenomics Methods Workshop

Anne Chambers (eachambers@berkeley.edu)

Outline for today

“lost in a field of genomics and code CGI style”

My background: largely in RADseq methods

What the structure of today is (focused on WGS and red-rep data)

Key is to look at your data in as many ways as possible!

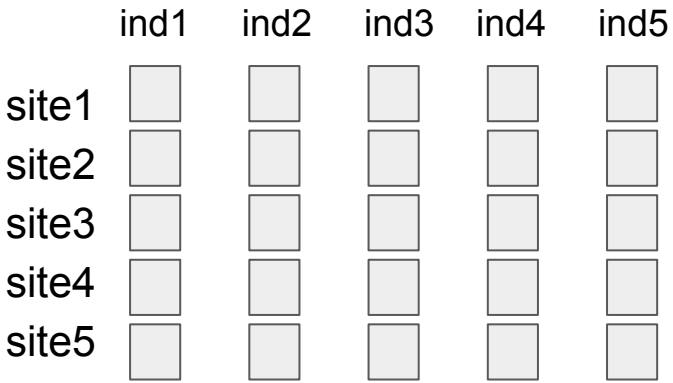
Outline for today

“lost in a field of genomics and code CGI style”

My background: largely in RADseq methods

What the structure of today is (focused on WGS and red-rep data)

Key is to look at your data in as many ways as possible!



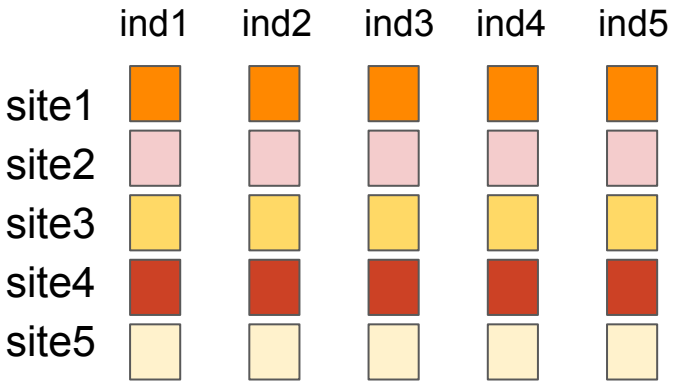
Outline for today

“lost in a field of genomics and code CGI style”

My background: largely in RADseq methods

What the structure of today is (focused on WGS and red-rep data)

Key is to look at your data in as many ways as possible!



Exercise 1: across sites

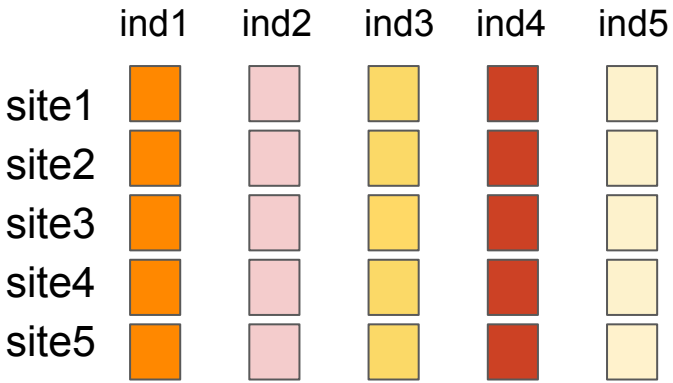
Outline for today

"lost in a field of genomics and code CGI style"

My background: largely in RADseq methods

What the structure of today is (focused on WGS and red-rep data)

Key is to look at your data in as many ways as possible!



Exercise 1: across sites

Exercise 2: across individuals

An overview of ‘omics approaches

Genomics

- The study of the genome
- We’ll be talking about whole genome sequencing (WGS) and reduced-representation sequencing (sequencing a subset of the genome)

An overview of ‘omics approaches

Genomics

- The study of the genome
- We’ll be talking about whole genome sequencing (WGS) and reduced-representation sequencing (sequencing a subset of the genome)

Metagenomics

- The study of nucleotide sequences from all organisms in a sample
- Sometimes called “community genomics”, alluding to the study of specific communities
- Can involve sequencing a specific marker from all organisms of a specific type (e.g., 16S from all bacteria) or sequencing all extracted DNA or RNA

An overview of ‘omics approaches

Genomics

- The study of the genome
- We’ll be talking about whole genome sequencing (WGS) and reduced-representation sequencing (sequencing a subset of the genome)

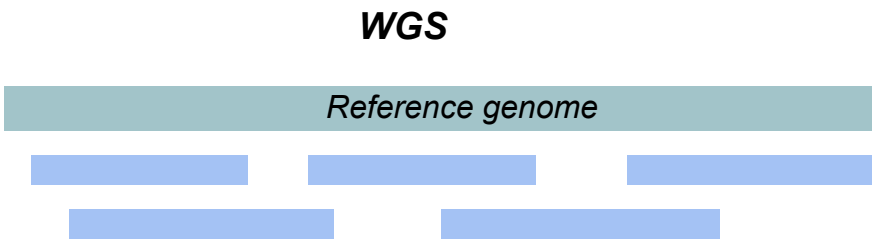
Metagenomics

- The study of nucleotide sequences from all organisms in a sample
- Sometimes called “community genomics”, alluding to the study of specific communities
- Can involve sequencing a specific marker from all organisms of a specific type (e.g., 16S from all bacteria) or sequencing all extracted DNA or RNA

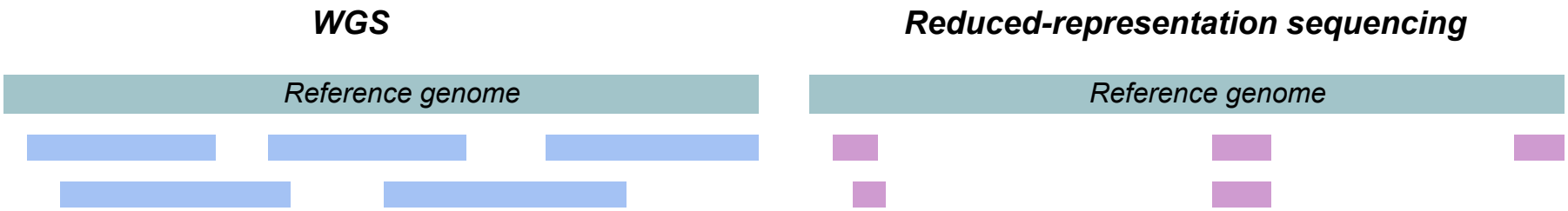
Transcriptomics

- The study of the transcriptome - the complete set of RNA transcripts that are produced by the genome
- Often the goal is to compare what genes are expressed in specific environments, circumstances, or tissue/cell types

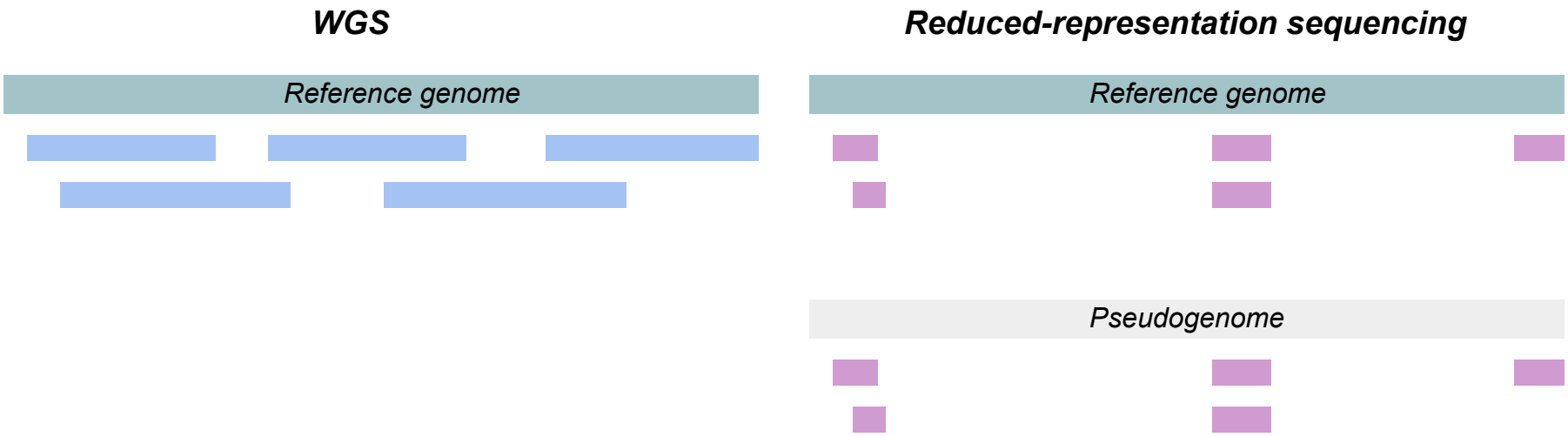
WGS and reduced-representation sequencing



WGS and reduced-representation sequencing



WGS and reduced-representation sequencing



Practical considerations

- Financial & computational resources
- Sample quality (high-quality DNA is critical for sequencing methods that require long, contiguous DNA; low-quality DNA is fragmented)
- Sample selection (outgroups, etc.)
- Number of loci
- Number of reads (depth of coverage)
  - 0.1–3X for probabilistic genotyping (takes into account uncertainty)
  - 10–30X for “hard genotyping”

Adapted (with permission) from R.D. Tarvin

Reference genomes: the (very) basics

Reference genomes: the (very) basics

Genome

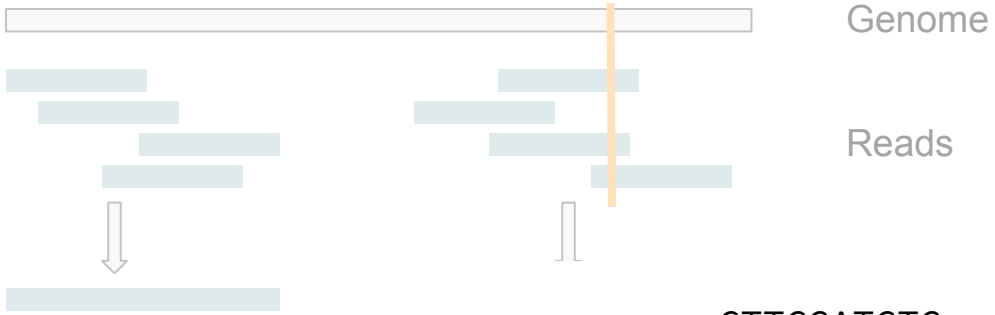
Reads

***Coverage**: number of reads we have overlapping any given position, or:*
Average sequenced bp / total genome length

Contigs

Scaffold (chromosome?)

Reference genomes: the (very) basics



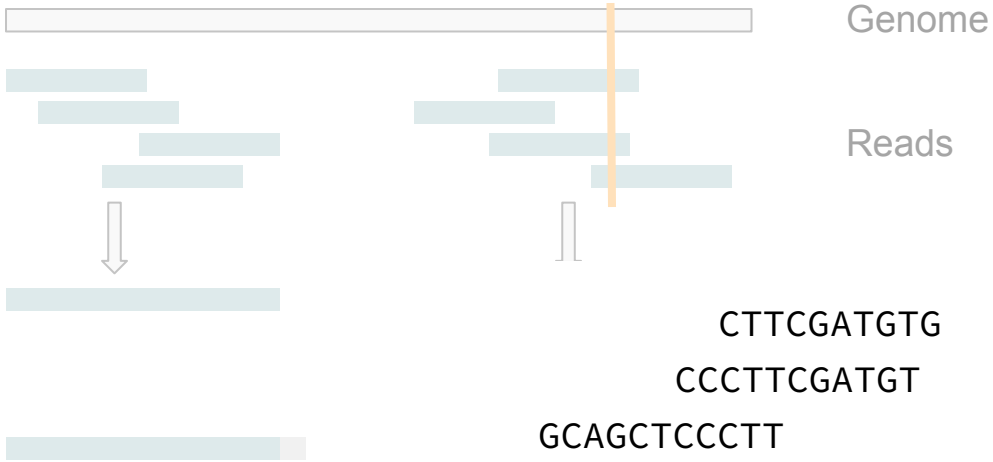
***Coverage**: number of reads we have overlapping any given position, or:*
Average sequenced bp / total genome length

```
                    CTTCGATGTG
                  CCCTTCGATGT
            GCAGCTCCCTT
        CGGCAGCTCC
  TGGATTCGGC
  TGGATTCGG
```

Total sequenced bases = 61

genome **TGGATTCGGCAGCTCCCTTCGATGTG** Length = 26

Reference genomes: the (very) basics



***Coverage**: number of reads we have overlapping any given position, or: Average sequenced bp / total genome length*

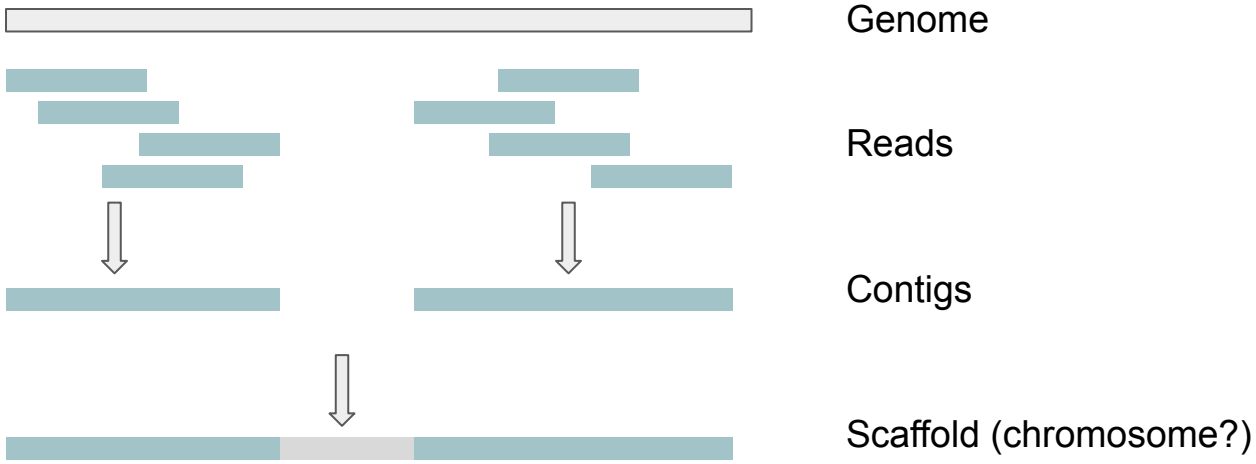
```
                    CTTCGATGTG
                  CCCTTCGATGT
            GCAGCTCCCTT
        CGGCAGCTCC
TGGATTCGGC
TGGATTCGG
```

genome **TGGATTCGGCAGCTCCCTTCGATGTG** Length = 26

Total sequenced bases = 61

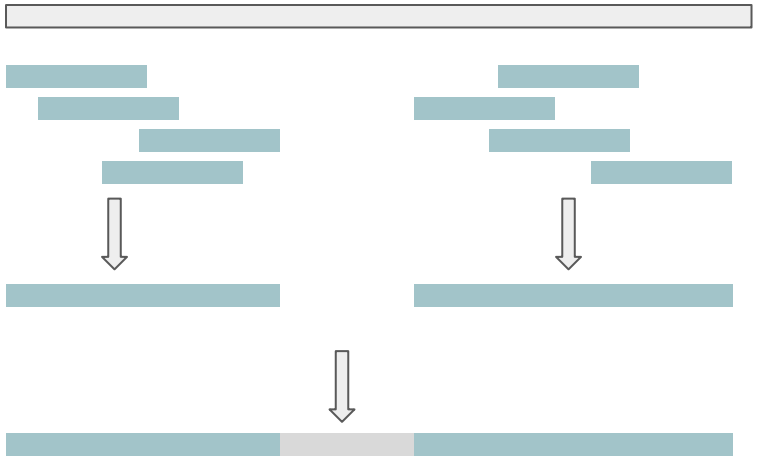
Coverage: 61 / 26 = 2.3X

Reference genomes: the (very) basics



Comparing genome assemblies: **N50** and **L50** usually referred to

Reference genomes: the (very) basics

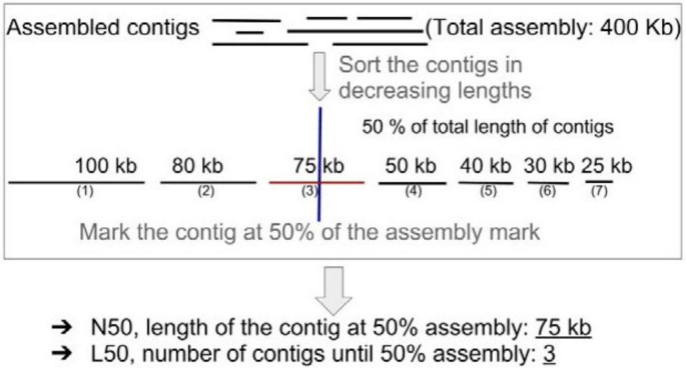


Genome

Reads

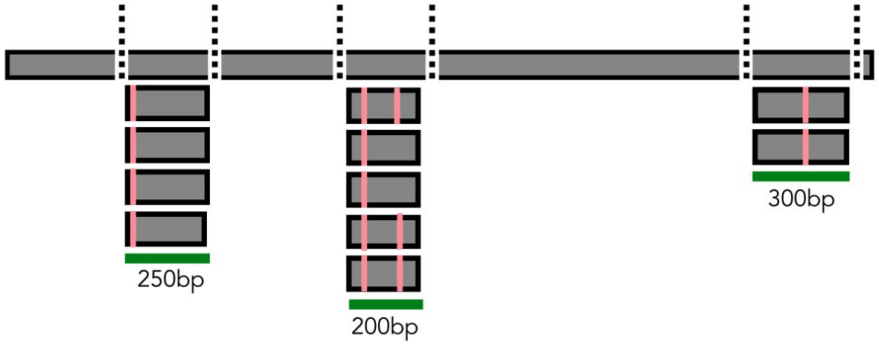
Contigs

Scaffold (chromosome?)

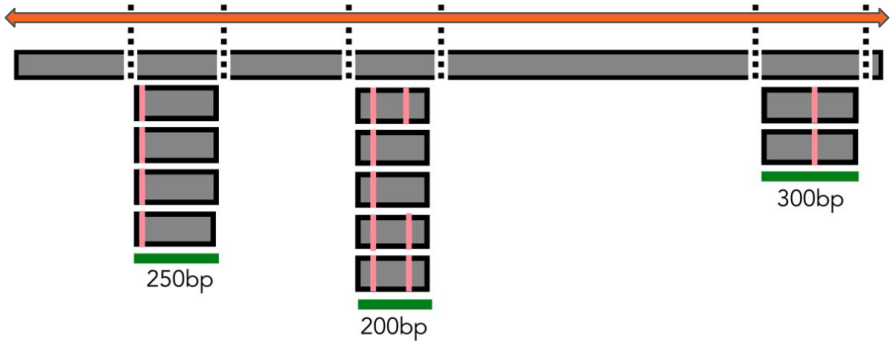


Comparing genome assemblies: **N50** and **L50** usually referred to

Calculating your sequencing needs



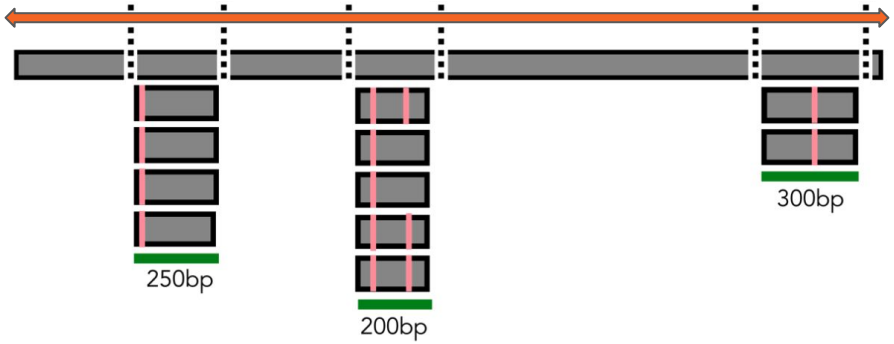
Calculating your sequencing needs



For RADseq libraries:

Genome size (bp)

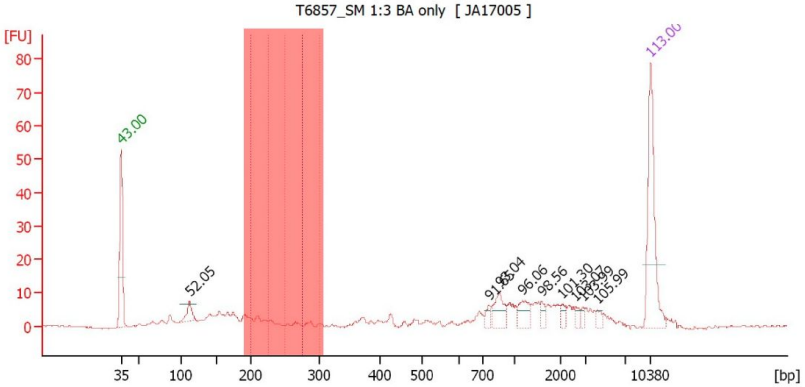
Calculating your sequencing needs



For RADseq libraries:

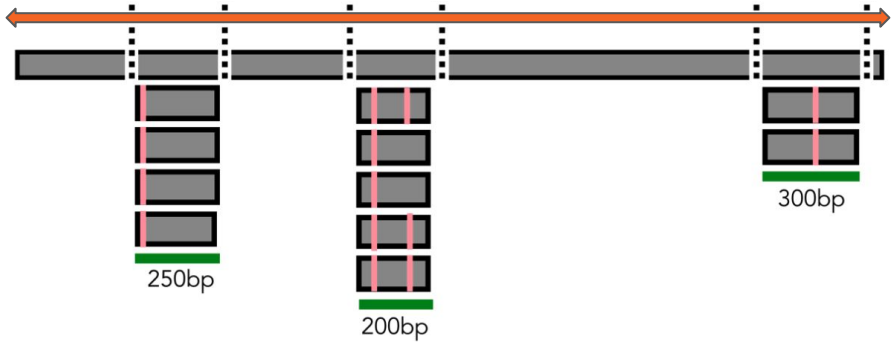
Genome size (bp) x **% genome in size range**

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

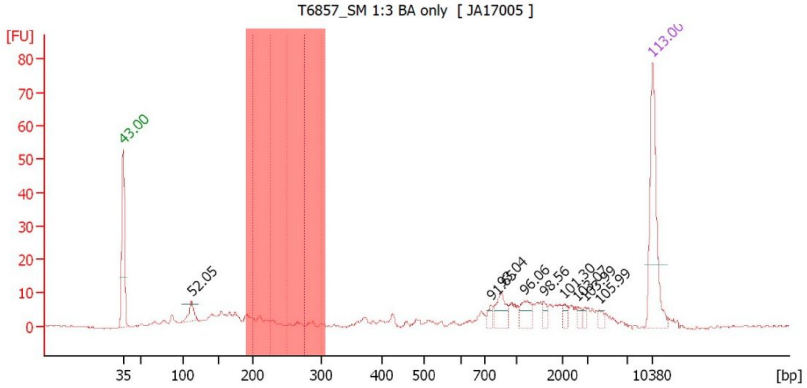
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

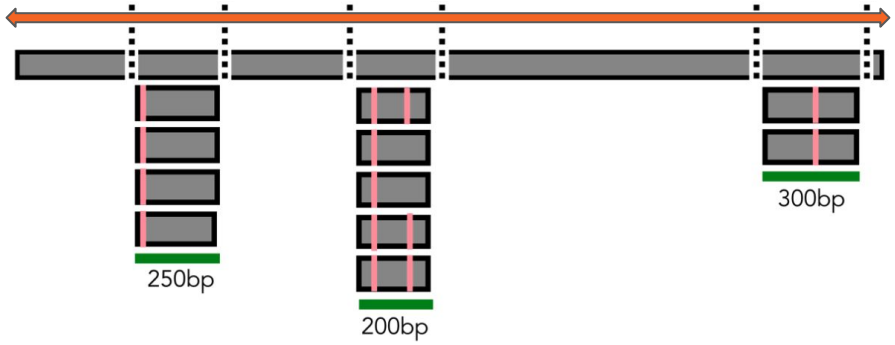
Genome size (bp) x **% genome in size range**

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

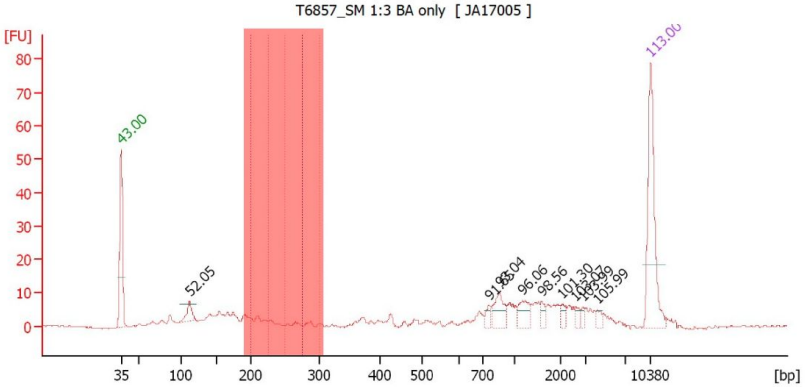
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

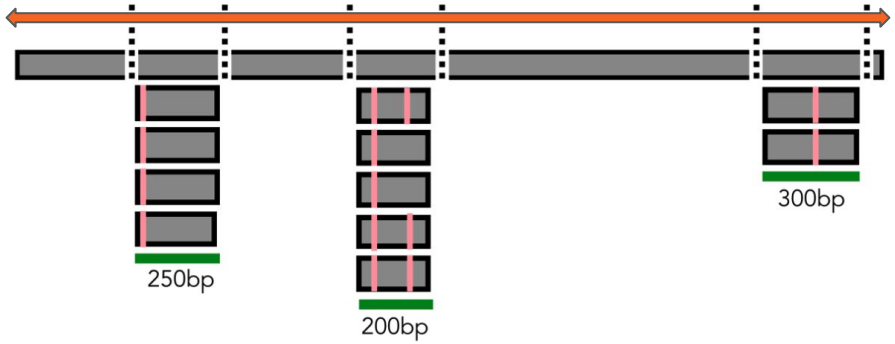
Genome size (bp) x **% genome in size range**

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

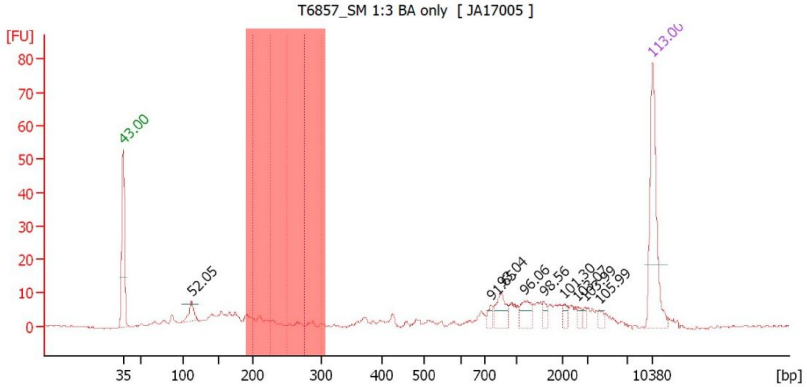
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

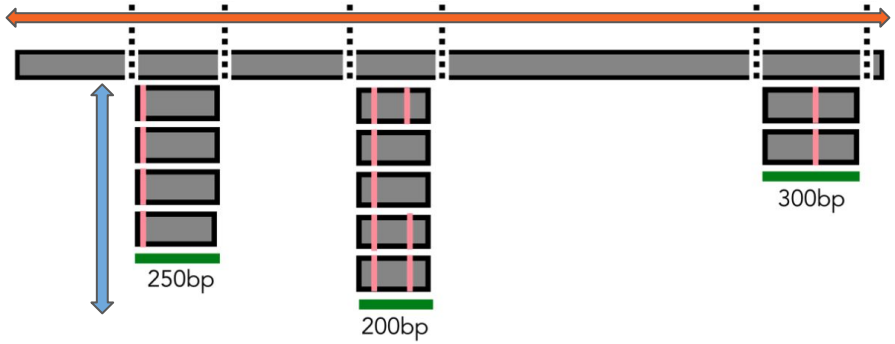
$$\frac{\text{Genome size (bp)} \times \text{\% genome in size range}}{\text{Average fragment length (bp)}}$$

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

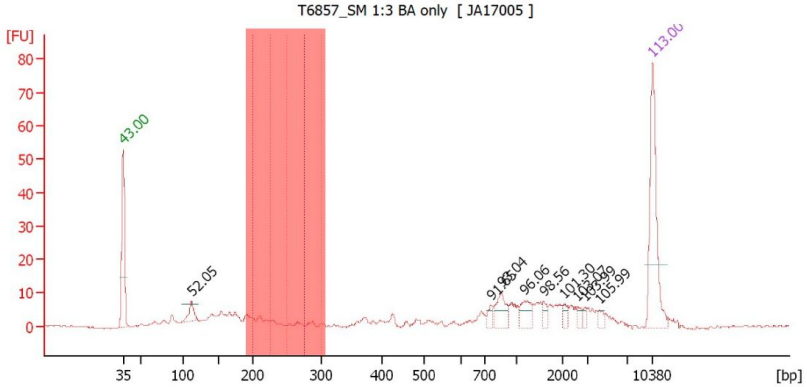
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

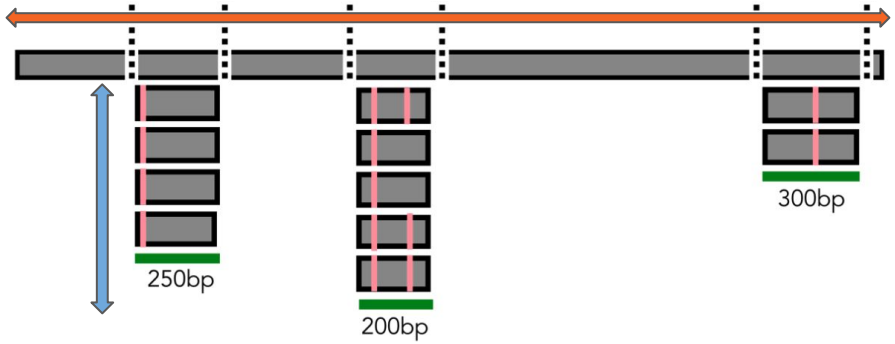
$$\frac{\text{Genome size (bp)} \times \text{\% genome in size range}}{\text{Average fragment length (bp)}} \times \text{desired coverage}$$

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

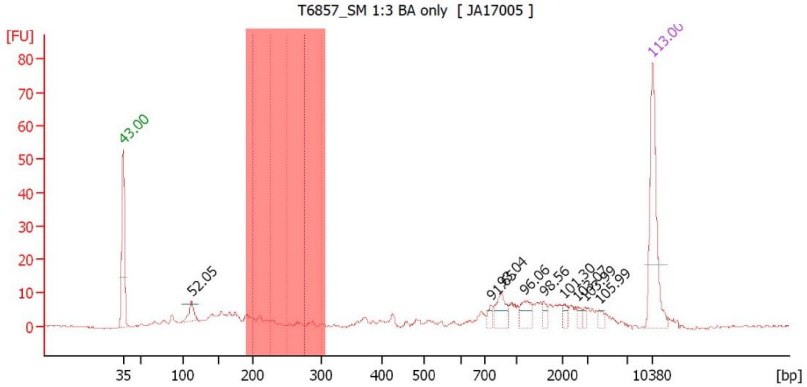
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

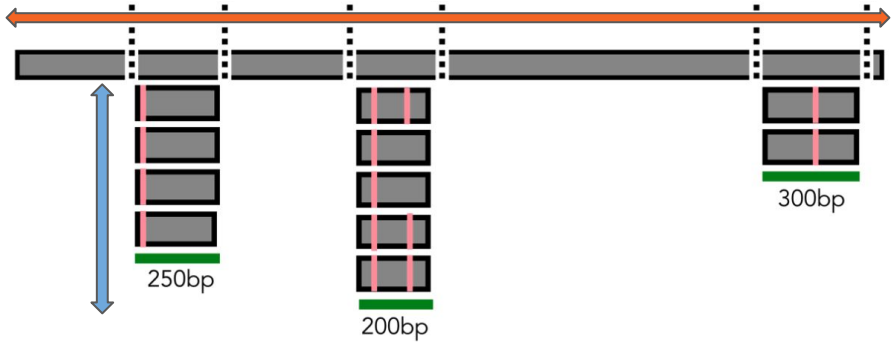
$$\frac{\text{2Gb genome} \times \text{2\%}}{\text{275bp avg fragment length}} \times \text{10X coverage}$$

Calculating your sequencing needs



Region table for sample 3 : T6857_SM 1:3 BA only

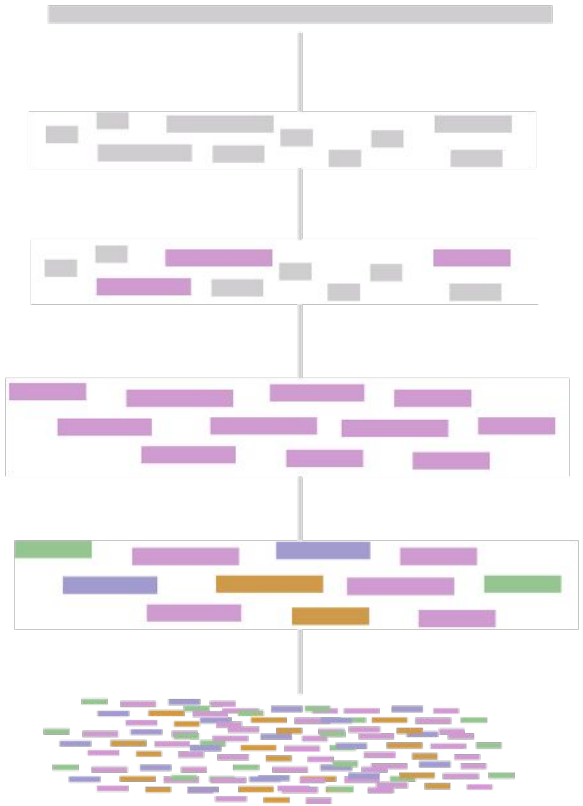
From [bp]	To [bp]	Corr. Area	% of Total	Average Size [bp]	Size distribution in CV [%]	Conc. [pg/µl]	Molarity [pmol/l]	Color
200	250	10.2	5	217	5.5	18.61	130.2	
225	275	4.5	2	244	6.7	7.99	49.8	
250	300	3.2	2	275	4.6	5.54	30.6	



For RADseq libraries:

$$\frac{\text{2Gb genome} \times \text{2\%}}{\text{275bp avg fragment length}} \times \text{10X coverage} = 1{,}454{,}545 \text{ reads}$$

WGS and reduced-representation sequencing: **library preparation**



Genomic DNA

Fragment genome
& ligate adaptors

Size selection

Unique identifiers
based on sample

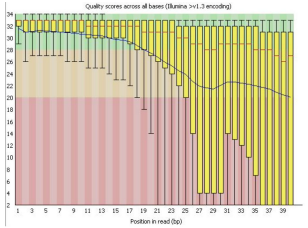
PCR to add seq primers
and enrich copies

WGS and reduced-representation sequencing: **bioinformatics**

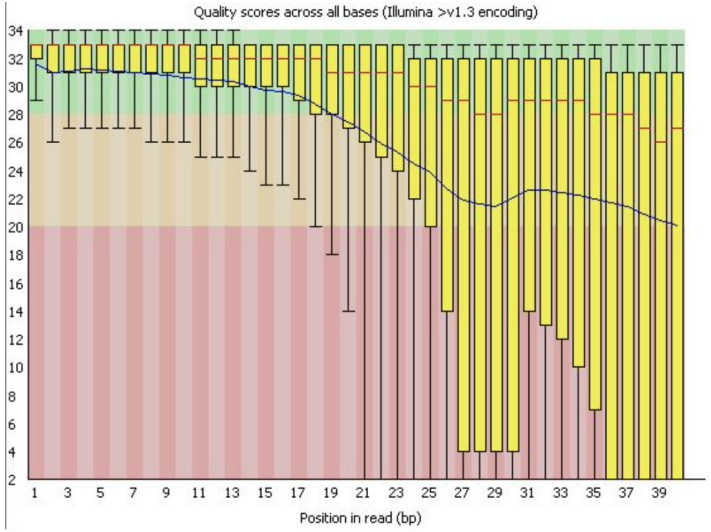
Raw reads

WGS and reduced-representation sequencing: **bioinformatics**

Raw reads

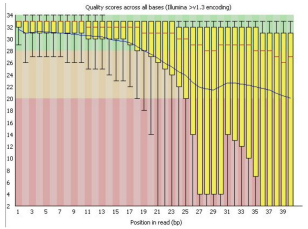


Quality check
(fastqc)

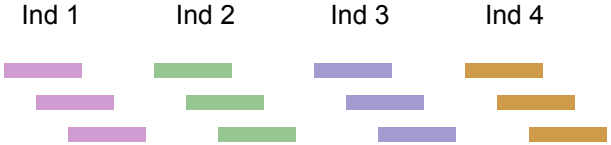


WGS and reduced-representation sequencing: **bioinformatics**

Raw reads



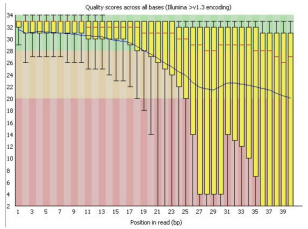
Quality check
(fastqc)



Demultiplexing
& trimming

WGS and reduced-representation sequencing: **bioinformatics**

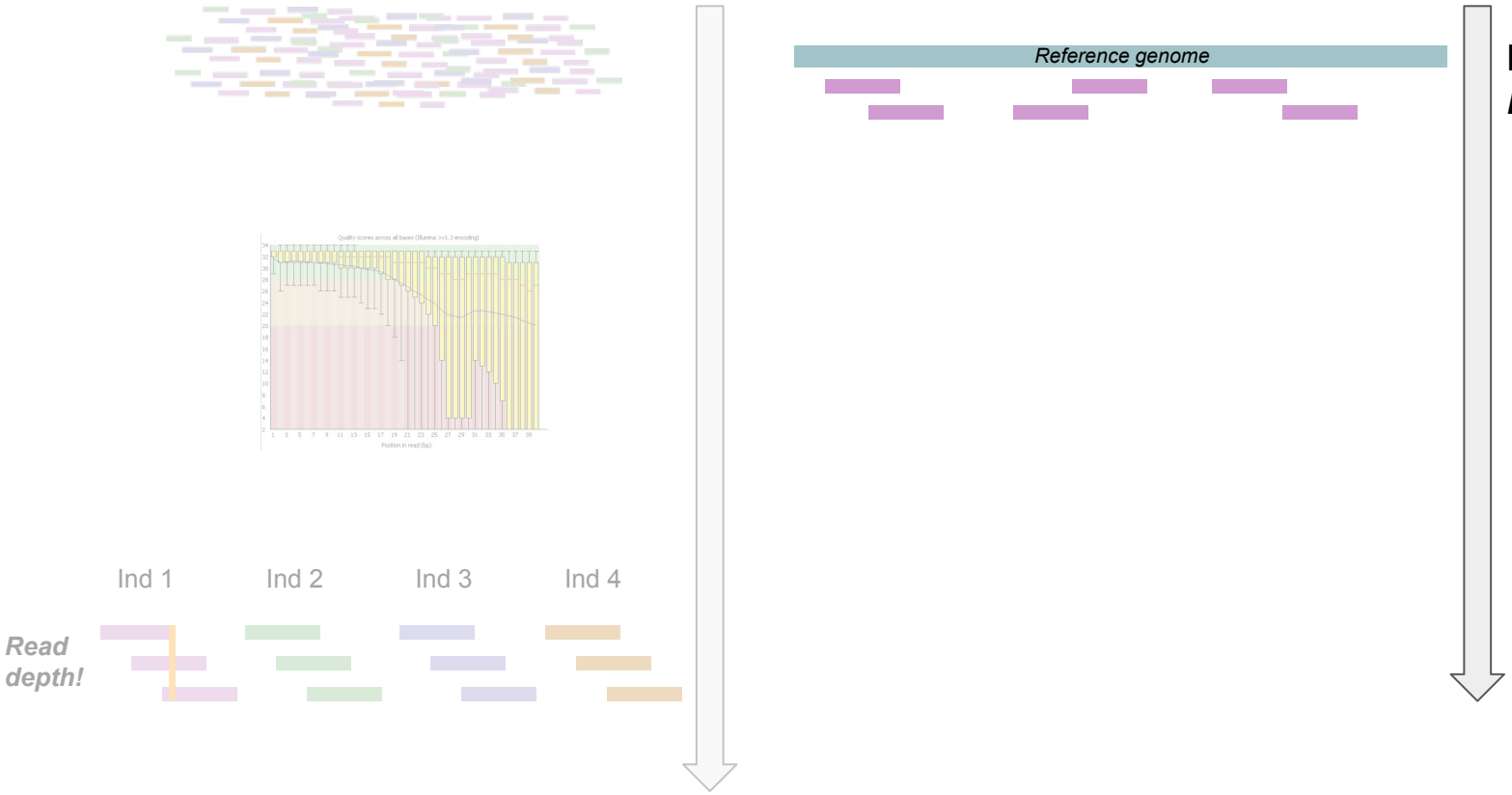
Raw reads



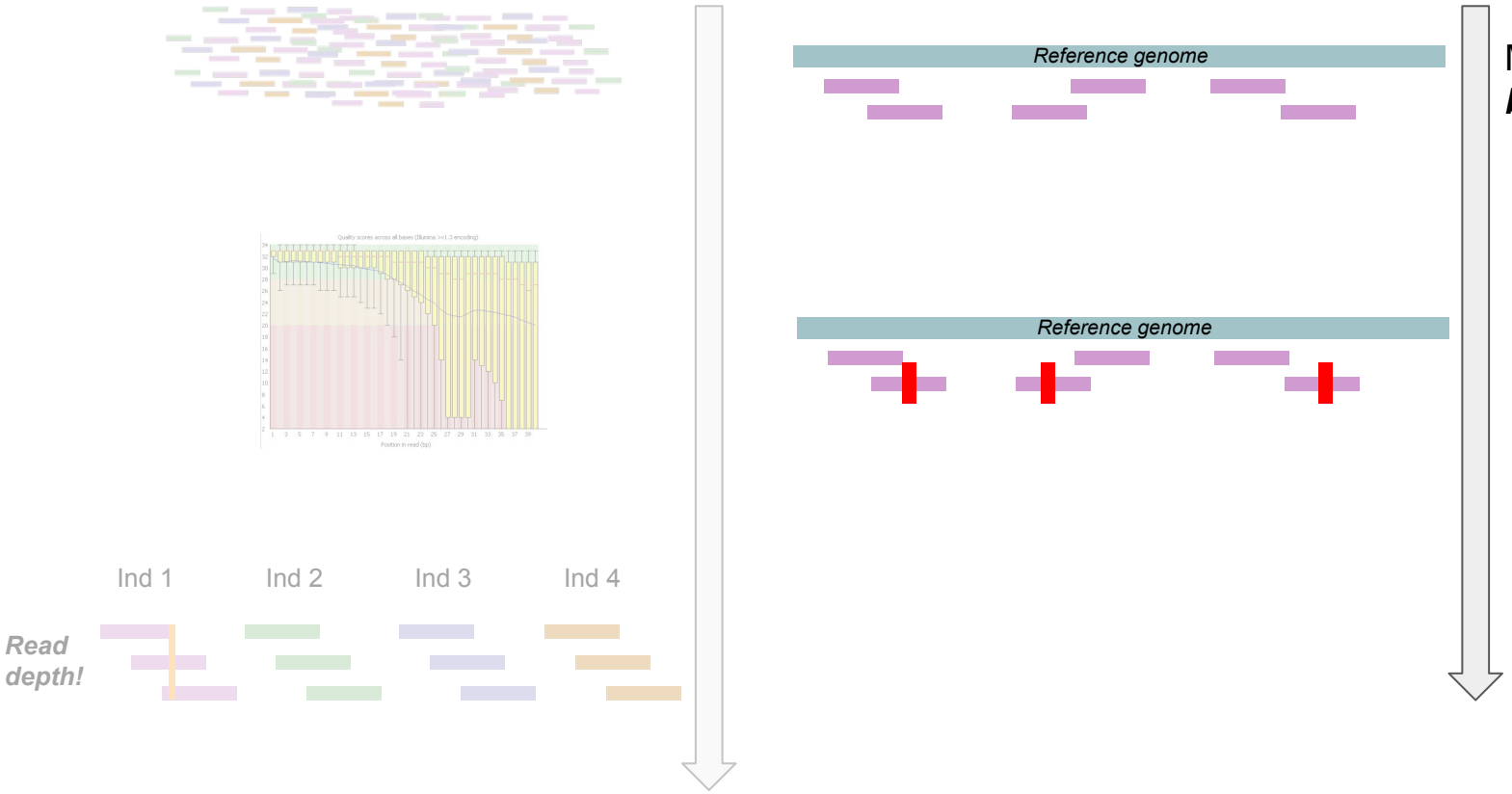
Quality check
(fastqc)

Demultiplexing
& trimming

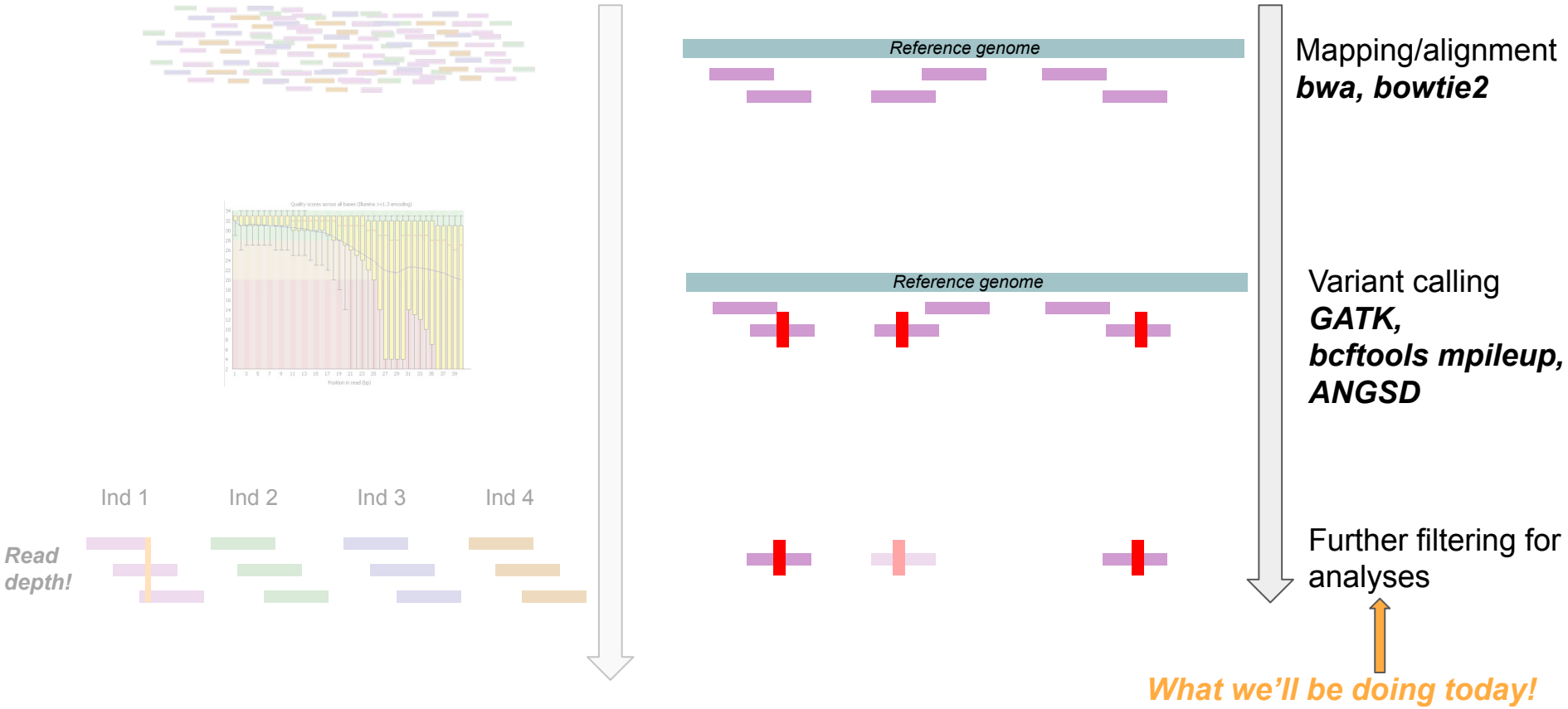
WGS and reduced-representation sequencing: **bioinformatics**



WGS and reduced-representation sequencing: **bioinformatics**



WGS and reduced-representation sequencing: **bioinformatics**



Biases in genomic data

Where do biases arise from? How can we go about identifying them?

De novo assembly errors are going to differ from those produced with a reference genome

Errors from sequencing:
PCR duplicates, genotyping and sequencing errors, read mapping errors

Other types of errors/biases:
Contamination, wrong species (misidentifications), close relatives sequenced, low coverage, differences in library preparation

Think about these biases all the way through your pipeline:

RADseq

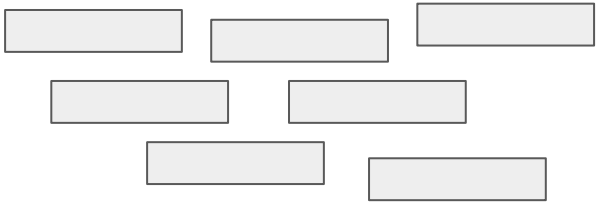
Paralogs are a problem

Think about these biases all the way through your pipeline:

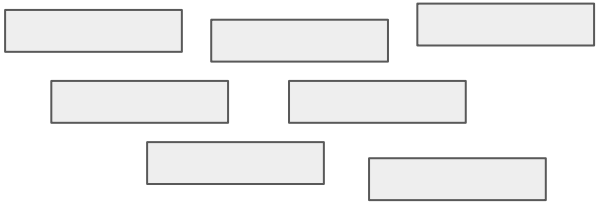
RADseq

Paralogs are a problem

Individual 1: *cluster 1*



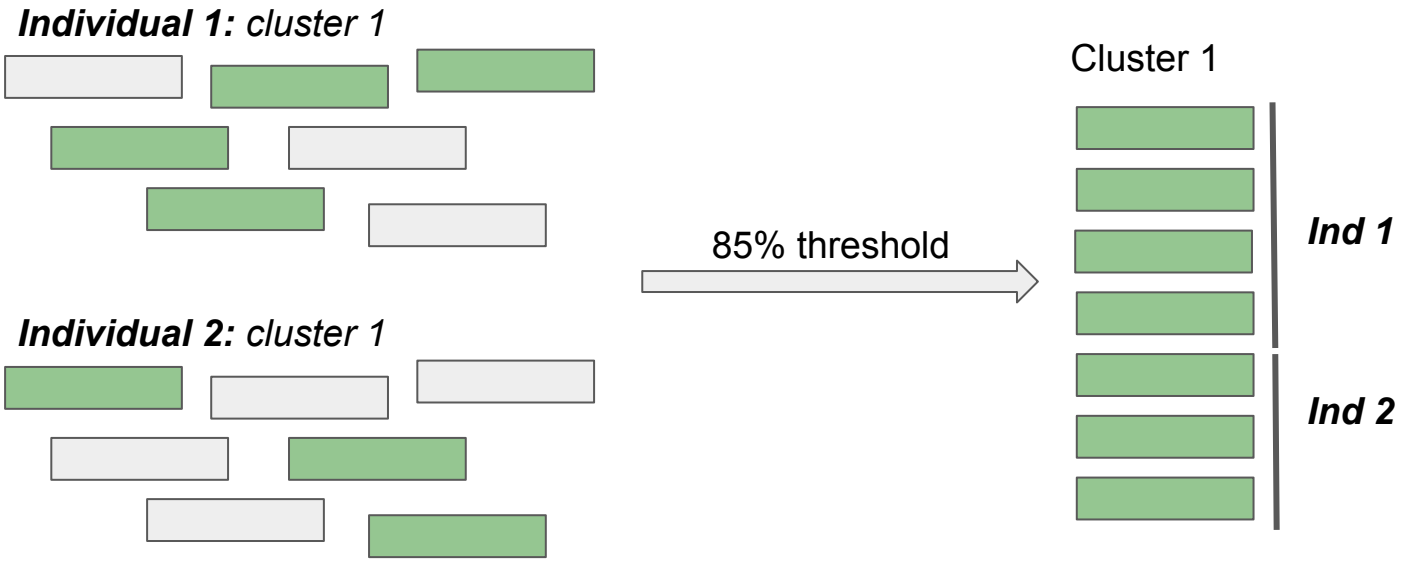
Individual 2: *cluster 1*



Think about these biases all the way through your pipeline:

RADseq

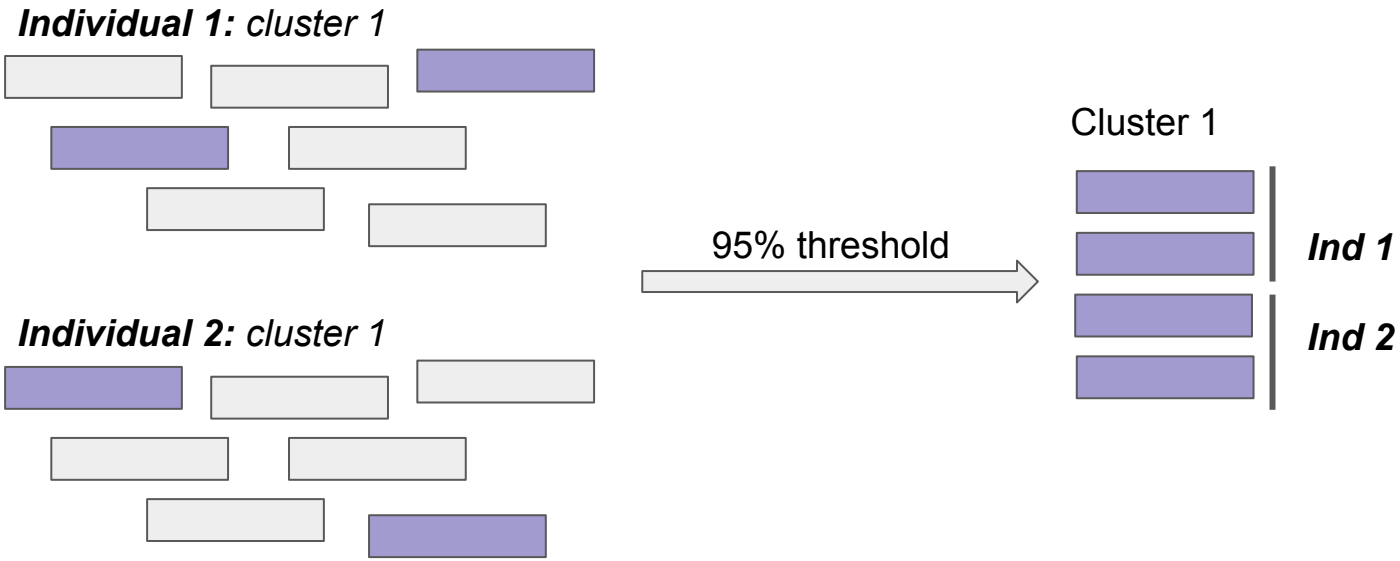
Paralogs are a problem



Think about these biases all the way through your pipeline:

RADseq

Paralogs are a problem



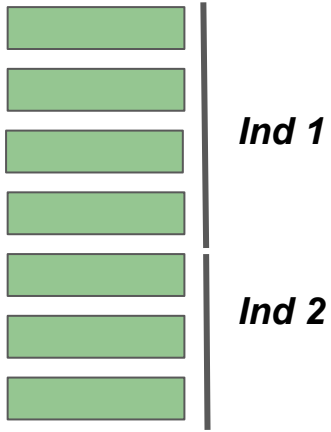
Think about these biases all the way through your pipeline:

RADseq

Paralogs are a problem

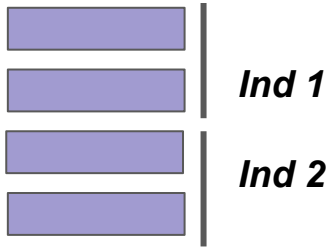
Too relaxed:

Higher likelihood of clustering paralogs



Too stringent:

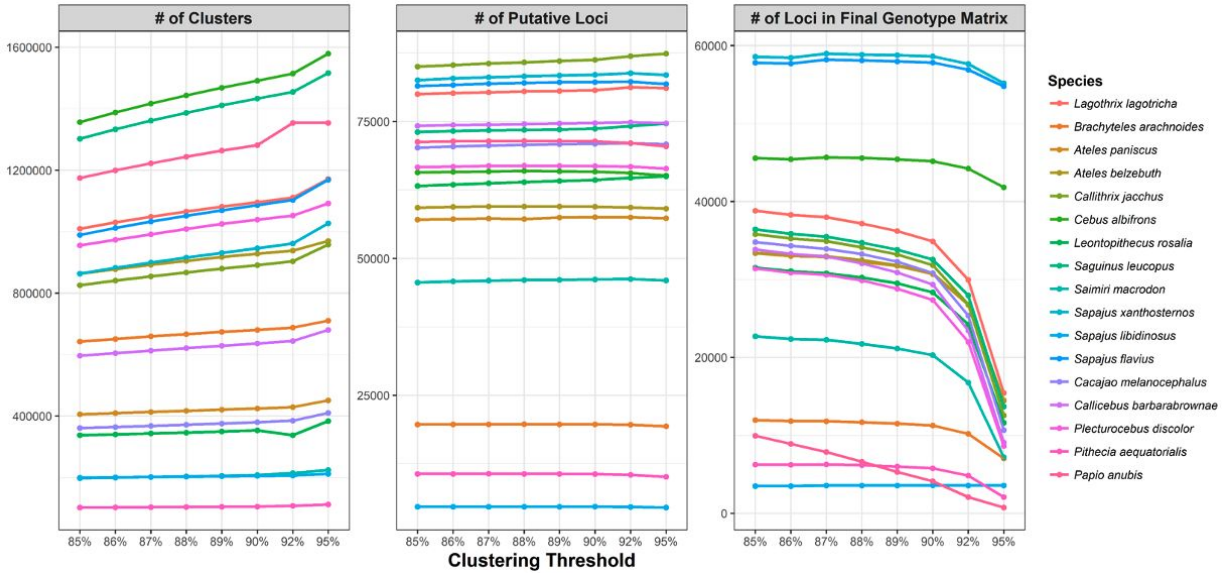
Higher likelihood of losing homologs



Think about these biases all the way through your pipeline:

RADseq

Paralogs are a problem



Valencia et al. (2018) PLOS ONE 13(8):e0201254

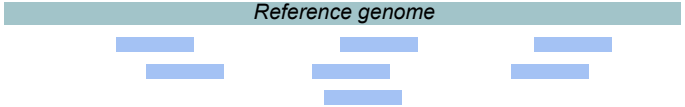
Think about these biases all the way through your pipeline:
RADseq

Paralogs are a problem

Biases across **different libraries**, making recovery of shared sites (fragments) between individuals difficult

Library 1

Library 2



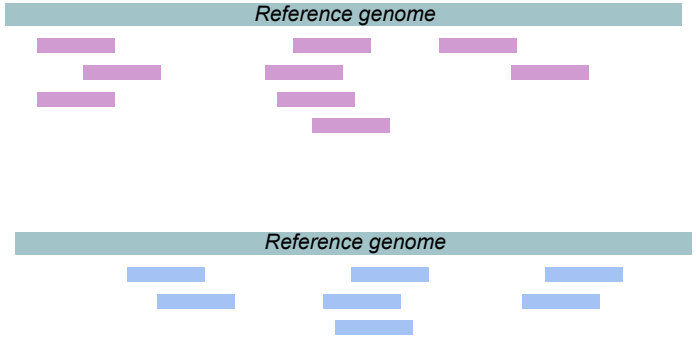
Think about these biases all the way through your pipeline:
RADseq

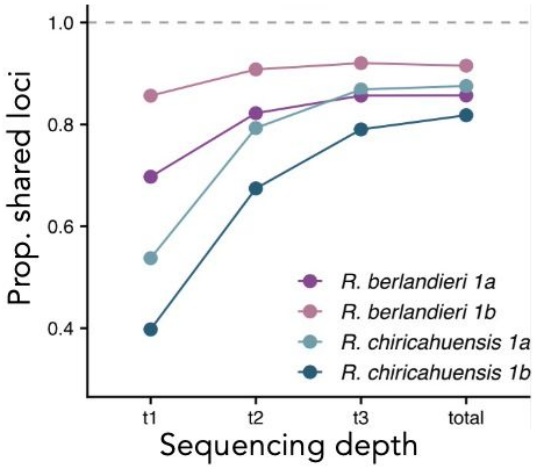
Paralogs are a problem

Biases across **different libraries**, making recovery of shared sites (fragments) between individuals difficult

Library 1

Library 2





Think about these biases all the way through your pipeline:

RADseq

Paralogs are a problem

Biases across **different libraries**, making recovery of shared sites (fragments) between individuals difficult

Missing data (especially in big genomes!)

- May need to retain sites with lots of MD simply because you can’t lose outgroups
- Helpful to think about the source of missing data

Think about these biases all the way through your pipeline:

WGS

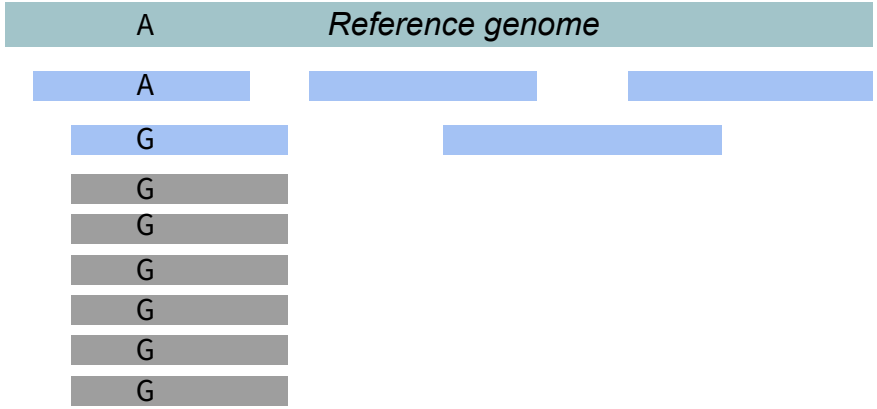
PCR duplicates cause some variants to be amplified more than others; **read mapping errors** produce erroneous variant calls

Adapted (with permission) from J. McLaughlin

Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls

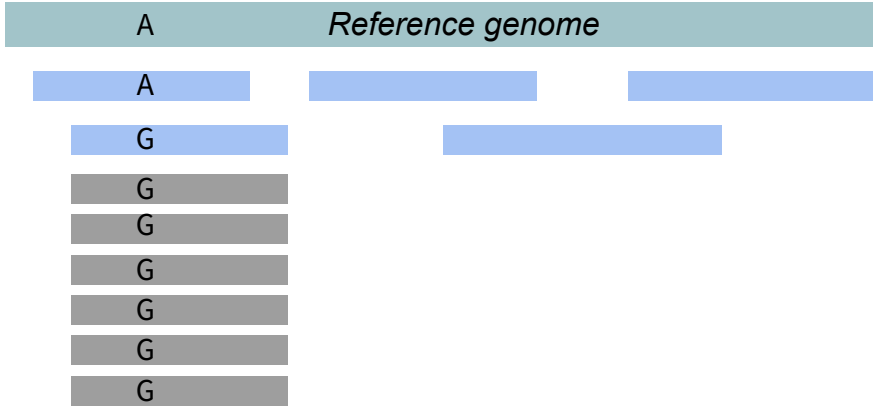


Adapted (with permission) from J. McLaughlin

Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls



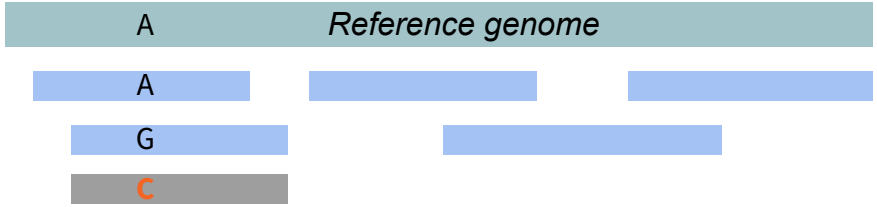
Calculate F-statistics that compare observed vs expected heterozygosity in your data

Adapted (with permission) from J. McLaughlin

Think about these biases all the way through your pipeline:

WGS

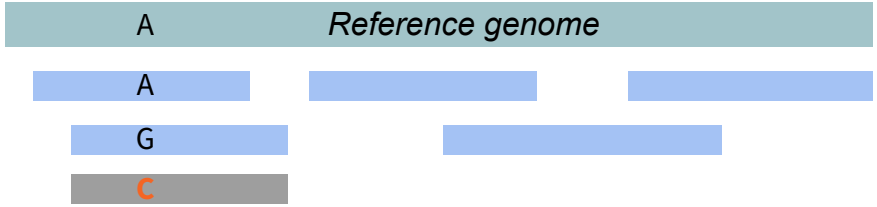
PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls



Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls



Remove any sites with minor allele frequencies lower than a given threshold

But see Linck & Battery (2019): https://doi.org/10.1111/1755-0998.12995

Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls

Low coverage assemblies

Can run a pipeline like ANGSD that generates genotype likelihoods which reduces the number of lost sites

Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls

Low coverage assemblies

Linkage among variants to be used for downstream analyses

Think about these biases all the way through your pipeline:

WGS

PCR duplicates cause some variants to be amplified more than others; **sequencing and read mapping errors** produce erroneous variant calls

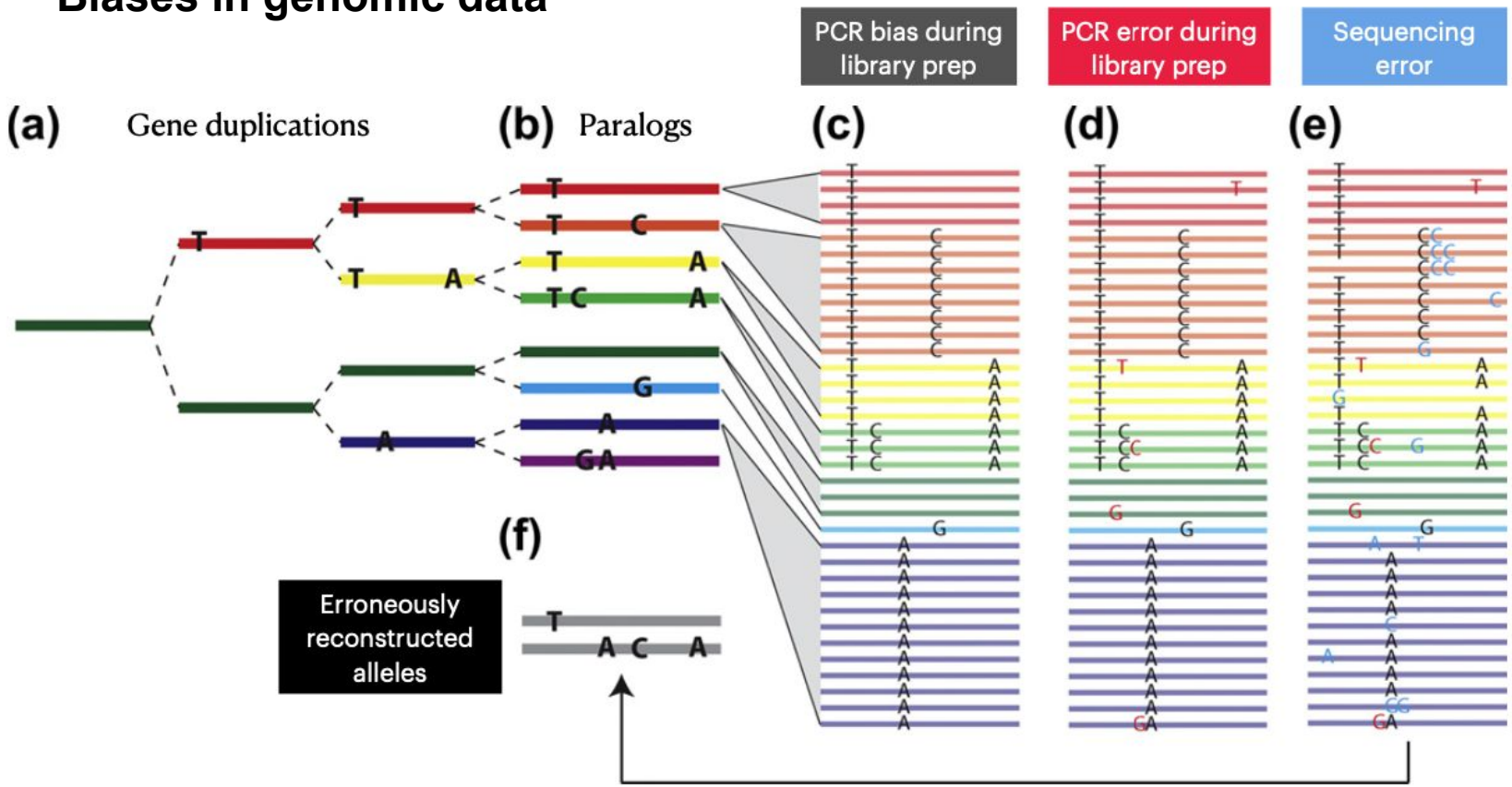
Low coverage assemblies

Linkage among variants to be used for downstream analyses

Should perform linkage disequilibrium (LD)-pruning on any datasets intended for things like population structure

Takes in window size, how much to move window, and correlation

Biases in genomic data



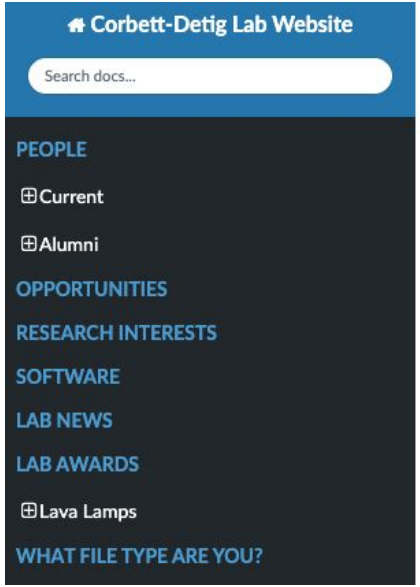
McCormack et al. (2013) https://doi.org/10.1016/j.ympev.2011.12.007

A field guide to common file types

Illustration (with permission): J.F. McLaughlin

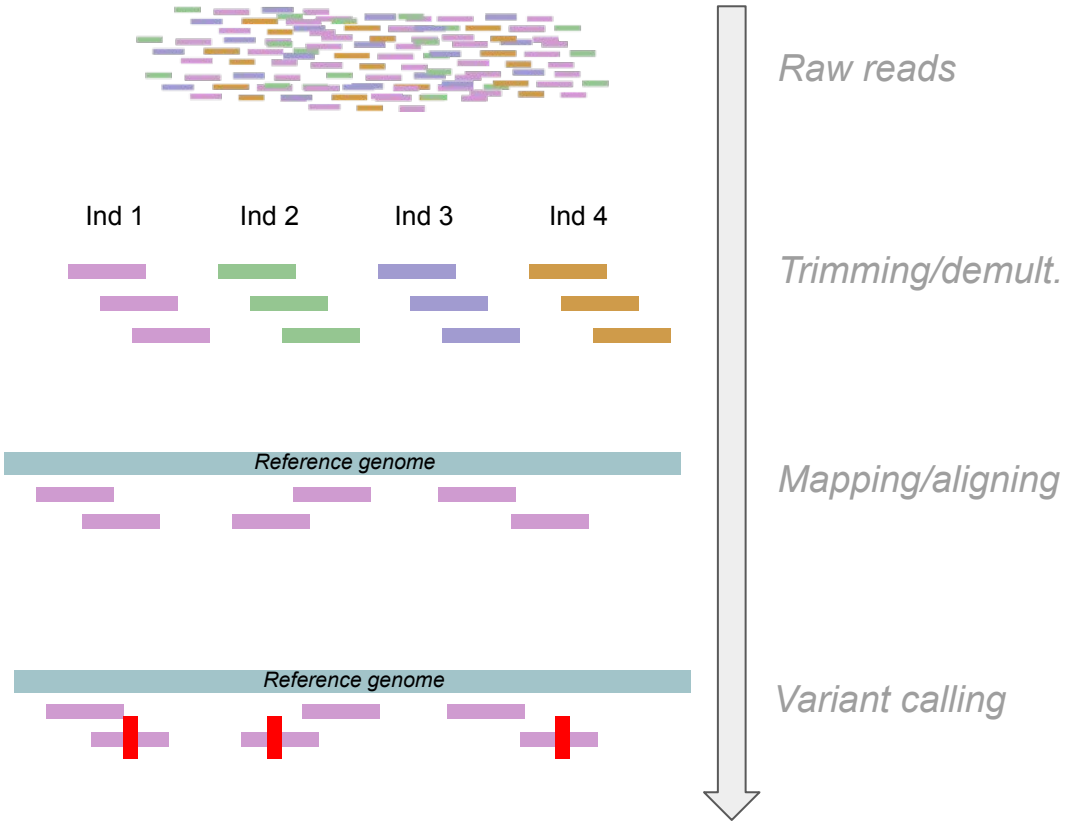
A field guide to common file types

Illustration (with permission): J.F. McLaughlin

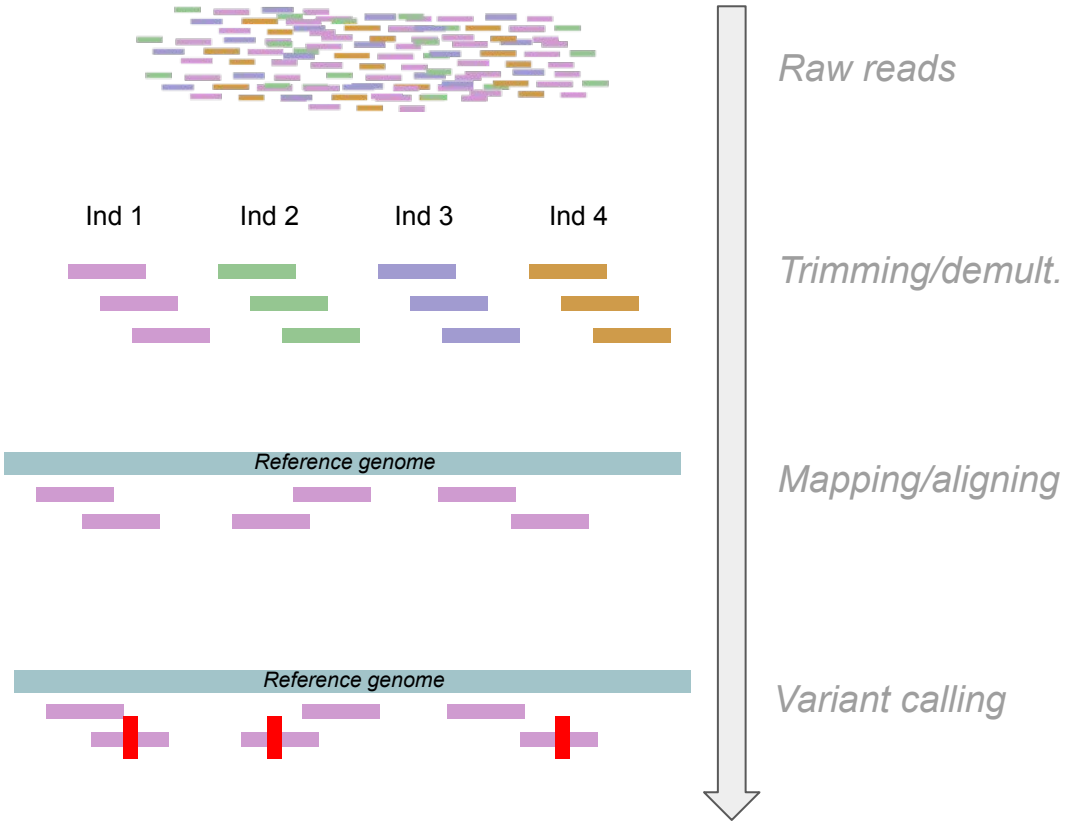


https://corbett-lab.github.io/what_file_type/

A field guide to common file types

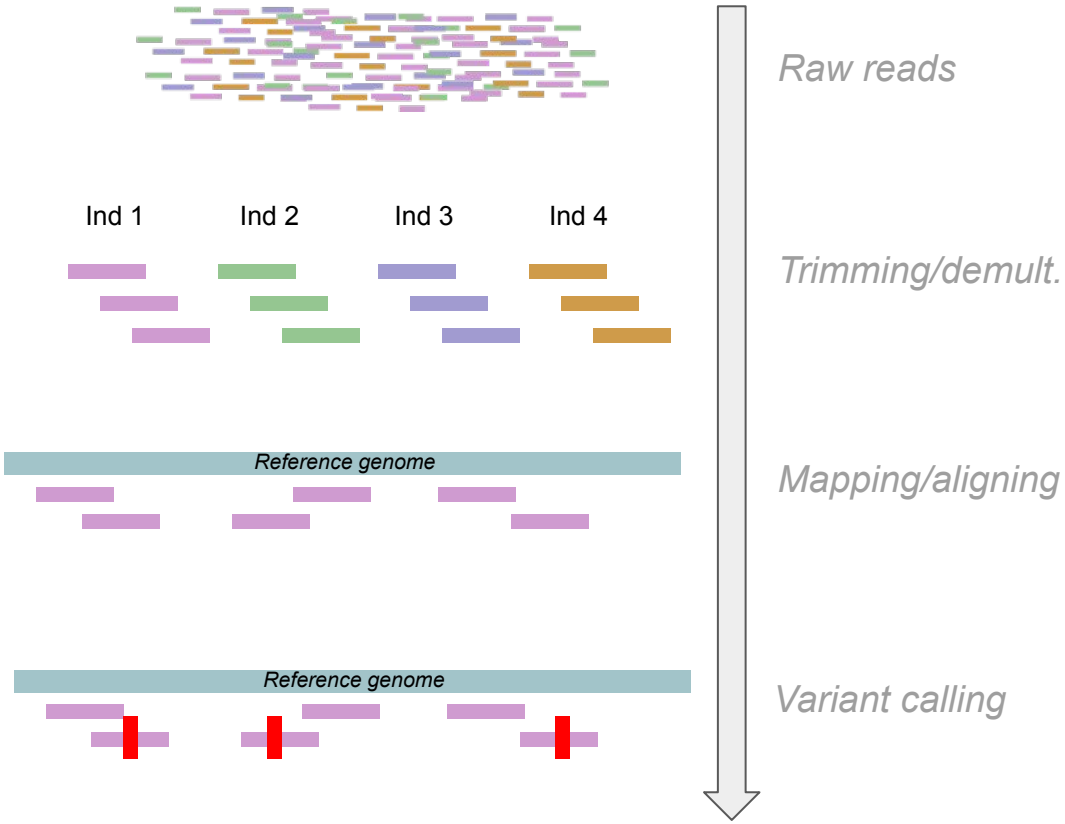


A field guide to common file types



Raw reads and quality scores
(may be paired):
.fastq (fastq.gz) or **.fq**

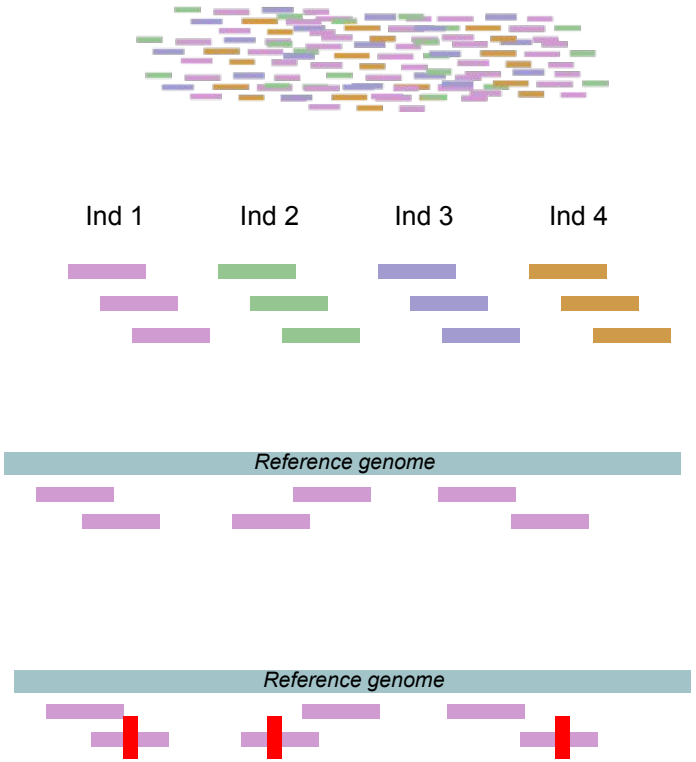
A field guide to common file types



Raw reads and quality scores
(may be paired):
.fastq (fastq.gz) or **.fq**

Alignments and sequences:
.sam, .bam, or **.fasta**

A field guide to common file types



Raw reads

Raw reads and quality scores
(may be paired):
.fastq (fastq.gz) or **.fq**

Trimming/demult.

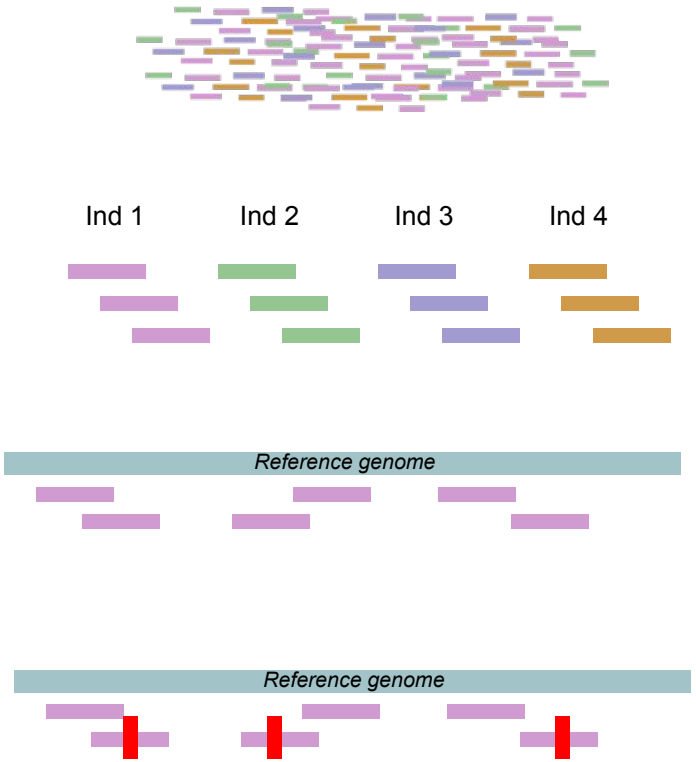
Mapping/aligning

Alignments and sequences:
.sam, .bam, or **.fasta**

Variant calling

Variant calls:
.vcf (vcf.gz)

A field guide to common file types



Raw reads

Raw reads and quality scores
(may be paired):
.fastq (fastq.gz) or **.fq**

Trimming/demult.

Mapping/aligning

Alignments and sequences:
.sam, .bam, or **.fasta**

Variant calling

Variant calls:
.vcf (vcf.gz)

These contain both SNPs and indels!

Variant call format (VCF) file

```
##fileformat=VCFv4.2
##fileDate=20220812
##source=PLINKv1.90
##contig=<ID=0,length=2147483645>
##INFO=<ID=PR,Number=0,Type=Flag,Description="Provisional reference allele, may
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
#CHROM  POS ID  REF ALT QUAL  FILTER  INFO  FORMAT  ALT3_ALT3 ANG8_ANG8 BAR360_B
0 10  Locus_10  G C . . PR  GT  0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0
0 15  Locus_15  T C . . PR  GT  0/0 0/0 0/0 0/0 ./. 0/0 0/0 ./. 0/1 0/0 0/0 0/0
0 22  Locus_22  T C . . PR  GT  0/0 0/0 0/0 0/0 0/0 0/0 ./. ./. 0/0 0/0 ./. 0/0
0 28  Locus_28  A C . . PR  GT  0/0 0/0 ./. ./. 0/0 ./. ./. ./. ./. 0/0 ./. 0/0
0 32  Locus_32  C A . . PR  GT  0/0 0/0 0/0 0/0 ./. 0/0 0/0 ./. 0/0 0/0 0/0 0/0
0 35  Locus_35  G A . . PR  GT  ./. 0/0 0/0 0/0 0/0 0/0 0/0 ./. 0/0 0/0 0/0 0/0
0 37  Locus_37  G A . . PR  GT  ./. ./. 0/0 ./. 0/0 ./. ./. 0/0 0/0 0/0 0/0 ./.
0 61  Locus_61  C T . . PR  GT  1/1 ./. 1/1 0/0 0/1 0/0 0/0 0/0 ./. 0/0 0/0 0/0
0 71  Locus_71  G A . . PR  GT  0/0 0/0 0/0 ./. ./. 0/0 0/0 ./. ./. 0/0 0/0 ./.
0 72  Locus_72  A T . . PR  GT  0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 0/0 ./. 0/0 0/0
0 82  Locus_82  C T . . PR  GT  0/0 0/1 0/0 0/0 0/0 0/0 0/0 0/0 ./. 0/0 0/0 ./.
0 84  Locus_84  A G . . PR  GT  0/0 0/0 0/0 0/0 ./. 0/0 0/0 0/0 0/0 0/0 0/0 0/0
0 90  Locus_90  C T . . PR  GT  0/0 0/0 ./. 0/0 0/0 0/0 0/0 0/1 ./. 0/0 0/0 0/0
0 91  Locus_91  T A . . PR  GT  0/0 0/0 0/0 0/0 ./. 0/0 0/0 1/1 ./. 0/0 0/0 0/0
0 96  Locus_96  C T . . PR  GT  0/0 ./. 0/0 0/0 0/0 ./. 0/0 0/0 0/0 0/0 0/0 ./.
0 104 Locus_104 T G . . PR  GT  0/0 0/0 0/0 ./. 0/0 ./. 0/0 0/0 ./. 0/0 ./. ./.
0 122 Locus_122 C G . . PR  GT  ./. ./. ./. ./. ./. ./. 0/0 ./. ./. ./. ./. ./.
0 125 Locus_125 T G . . PR  GT  0/0 0/0 ./. ./. ./. 0/0 ./. ./. 0/0 0/0 ./. ./.
0 131 Locus_131 G T . . PR  GT  0/1 ./. ./. ./. 0/0 ./. ./. ./. ./. ./. ./. ./.
0 133 Locus_133 C T . . PR  GT  0/0 0/0 ./. ./. ./. 0/0 ./. ./. 0/0 ./. ./. 0/0
0 138 Locus_138 G A . . PR  GT  0/0 ./. 0/0 ./. ./. ./. ./. ./. ./. ./. ./. ./.
0 154 Locus_154 A G . . PR  GT  0/0 0/0 ./. 0/0 ./. 0/0 ./. ./. ./. ./. ./. ./.
0 162 Locus_162 C T . . PR  GT  0/1 0/1 0/0 ./. ./. 0/0 0/0 ./. 0/0 0/0 1/1 0/0
0 166 Locus_166 C T . . PR  GT  0/0 ./. 0/0 ./. 0/0 ./. ./. 0/1 ./. ./. 0/0 ./.
0 177 Locus_177 G C . . PR  GT  0/0 ./. 0/0 ./. ./. 0/0 0/0 ./. ./. ./. ./. ./.
```

Variant call format (VCF) file

```
##fileformat=VCFv4.2
##FILTER=<ID=PASS,Description="All filters passed">
##FILTER=<ID=FS_SOR_filter,Description="(vc.isSNP() && ((vc.hasAttribute('FS') && F
##FILTER=<ID=MQ_filter,Description="vc.isSNP() && ((vc.hasAttribute('MQ') && MQ < 4
##FILTER=<ID=QUAL_filter,Description="QUAL < 30.0">
##FILTER=<ID=RPRS_filter,Description="(vc.isSNP() && (vc.hasAttribute('ReadPosRankS
##FORMAT=<ID=AD,Number=R,Type=Integer,Description="Allelic depths for the ref
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read depth">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=PGT,Number=1,Type=String,Description="Physical phasing haplotype
##FORMAT=<ID=PID,Number=1,Type=String,Description="Physical phasing ID inform
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="The phred-scaled genotype
##GATKCommandLine=<ID=VariantFiltration,CommandLine="VariantFiltration --outp
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes, f
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele frequency, for each ALT
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in c
##INFO=<ID=BaseQRankSum,Number=1,Type=Float,Description="Z-score from Wilcoxo
##INFO=<ID=ClippingRankSum,Number=1,Type=Float,Description="Z-score From Wilc
##INFO=<ID=DP,Number=1,Type=Integer,Description="Combined depth across sample
##INFO=<ID=ExcessHet,Number=1,Type=Float,Description="Phred-scaled p-value fo
##INFO=<ID=FS,Number=1,Type=Float,Description="Phred-scaled p-value using Fis
##INFO=<ID=InbreedingCoeff,Number=1,Type=Float,Description="Inbreeding coeffi
##INFO=<ID=MLEAC,Number=A,Type=Integer,Description="Maximum likelihood expect
##INFO=<ID=MLEAF,Number=A,Type=Float,Description="Maximum likelihood expectat
##INFO=<ID=MQ,Number=1,Type=Float,Description="RMS mapping quality">
##INFO=<ID=MQRankSum,Number=1,Type=Float,Description="Z-score From Wilcoxon r
##INFO=<ID=QD,Number=1,Type=Float,Description="Variant Confidence/Quality by Depth
##INFO=<ID=ReadPosRankSum,Number=1,Type=Float,Description="Z-score from Wilcoxon ra
##INFO=<ID=SOR,Number=1,Type=Float,Description="Symmetric Odds Ratio of 2x2 contin
##SentieonCommandLine.GVCFtyper=<ID=GVCFtyper,Version="sentieon-genomics-202112.04"
##SentieonCommandLine.Haplotyper=<ID=Haplotyper,Version="sentieon-genomics-202112.0
##contig=<ID=SCAF_1,length=88620470,assembly=unknown>
##contig=<ID=SCAF_2,length=80884353,assembly=unknown>
##contig=<ID=SCAF_3,length=68994874,assembly=unknown>
##contig=<ID=SCAF_4,length=60156485,assembly=unknown>
```

```
##reference=file://ccgp-workflow-results/41-Cyanocitta/data/genome/bCyaSte1.NCBI.p_
##source=VariantFiltration
##bcftools_viewVersion=1.12+htslib-1.12
##bcftools_viewCommand=view -S ccgp-workflow-results/41-Cyanocitta/results/41-Cyano
#CHROM  POS ID  REF ALT QUAL  FILTER  INFO  FORMAT  MVZCCGP-Cst1_I-A01  MVZCCGP-Cst
SCAF_1  8234  . C T 9034.03 . AC=48;AF=0.16;AN=300;BaseQRankSum=-0.174;ClippingRank
SCAF_1  14861 . C T 665.39  . AC=6;AF=0.02;AN=300;BaseQRankSum=0.431;ClippingRankSu
SCAF_1  21459 . T C 13570 . AC=66;AF=0.221;AN=298;BaseQRankSum=0.21;ClippingRankSum
SCAF_1  36143 . T C 3685.22 . AC=27;AF=0.0906;AN=298;BaseQRankSum=-0;ClippingRankSu
SCAF_1  47228 . T C 3681.84 . AC=17;AF=0.057;AN=298;BaseQRankSum=-0;ClippingRankSum
SCAF_1  48964 . C A 751.02  . AC=6;AF=0.0201;AN=298;BaseQRankSum=-0;ClippingRankSum
SCAF_1  58848 . A T 6379.99 . AC=31;AF=0.103;AN=300;BaseQRankSum=-0;ClippingRankSum
SCAF_1  65432 . T C 2476.73 . AC=19;AF=0.0633;AN=300;BaseQRankSum=0.253;ClippingRan
SCAF_1  76112 . C T 7906.82 . AC=40;AF=0.133;AN=300;BaseQRankSum=0;ClippingRankSum=
SCAF_1  87858 . T C 465.78  . AC=3;AF=0.01;AN=300;BaseQRankSum=0.711;ClippingRankSu
SCAF_1  93071 . G C 774.35  . AC=4;AF=0.0133;AN=300;BaseQRankSum=-0;ClippingRankSum
SCAF_1  102600  . G A 3434.05 . AC=19;AF=0.0638;AN=298;BaseQRankSum=-0;ClippingRank
```

Programs we're working with today

bcftools: helpful for data processing (can also do variant calling)

https://samtools.github.io/bcftools/bcftools.html

Different commands we'll use in exercises: ***query*** and ***view***

```
bcftools COMMAND [OPTIONS] file.vcf
```

Programs we're working with today

bcftools: helpful for data processing (can also do variant calling)

https://samtools.github.io/bcftools/bcftools.html

Different commands we'll use in exercises: ***query*** and ***view***

```
bcftools COMMAND [OPTIONS] file.vcf

bcftools query -l file.vcf
```

Programs we're working with today

bcftools: helpful for data processing (can also do variant calling)

https://samtools.github.io/bcftools/bcftools.html

Different commands we'll use in exercises: ***query*** and ***view***

```
bcftools COMMAND [OPTIONS] file.vcf

bcftools query -l file.vcf

bcftools query -f '%CHROM' file.vcf
```

Programs we're working with today

bcftools: helpful for data processing (can also do variant calling)

https://samtools.github.io/bcftools/bcftools.html

Different commands we'll use in exercises: ***query*** and ***view***

```
bcftools COMMAND [OPTIONS] file.vcf

bcftools query -l file.vcf

bcftools query -f '%CHROM' file.vcf

bcftools query -f '%CHROM' file.vcf | head -3
```

Programs we're working with today

bcftools: helpful for data processing (can also do variant calling)

https://samtools.github.io/bcftools/bcftools.html

Different commands we'll use in exercises: ***query*** and ***view***

```
bcftools COMMAND [OPTIONS] file.vcf

bcftools query -l file.vcf

bcftools query -f '%CHROM' file.vcf

bcftools query -f '%CHROM' file.vcf | head -3
```

-q, --min-af *FLOAT*[*:nref*|*:alt1*|*:minor*|*:major*|*:nonmajor*]

vcftools and Plink

vcftools and **plink**: helpful for summary statistics

Plink works a lot with .ped and .bed files

vcftools and Plink

vcftools and **plink**: helpful for summary statistics

Plink works a lot with .ped and .bed files

```
vcftools --vcf file.vcf --out outfile_name --OPTION
```

vcftools and Plink

vcftools and **plink**: helpful for summary statistics

Plink works a lot with .ped and .bed files

```
vcftools --vcf file.vcf --out outfile_name --OPTION

vcftools --vcf file.vcf --out file_name --missing-indv
```

vcftools and Plink

vcftools and **plink**: helpful for summary statistics

Plink works a lot with .ped and .bed files

```
vcftools --vcf file.vcf --out outfile_name --OPTION

vcftools --vcf file.vcf --out file_name --missing-indv

plink --vcf file.vcf --out prefix_name --distance square --make-bed --recode vcf
```

Dataset we're working with today

Lampropeltis triangulum
ddRAD data (~50K SNPs)

EXERCISE 1: gathering basic statistics

Download the vcf file on GitHub here:
https://github.com/eachambers/EvoGeno-Methods-Workshop/blob/main/Workshop1/Data/lampro.vcf

Download the worksheet here:
https://github.com/eachambers/EvoGeno-Methods-Workshop/blob/main/Workshop1/Exercises/EvoGenomics_Ws1_Ex1.txt

```
# ==========================================================================================
#                                        EXERCISE 1
# ==========================================================================================

# bcftools is a great program for processing data (and is also actively maintained), whereas
# vcftools and plink are great for generating summary statistics. In this exercise, we'll
# use all three of these programs in the Terminal.

# ======================= 1. BASIC EXAMINATION OF THE VCF DATA FILE =======================

# Answer the following questions using bash.

#       (1a) Take a look at the lampro.vcf file. How was this vcf generated?
#               *** YOUR ANSWER HERE ***

#       (1b) What info is contained (per site) within the vcf?
#               *** YOUR ANSWER HERE ***
```