SciLifeLab

Analysing re-sequencing samples

Anna Johansson

Anna.johansson@scilifelab.se

WABI / SciLifeLab

Re-sequencing

Reference genome assembly

...GTGCGTAGACTGCTAGATCGAAGA...

Re-sequencing

IND 1
GTAGACT
AGATCGG
GCGTAGT

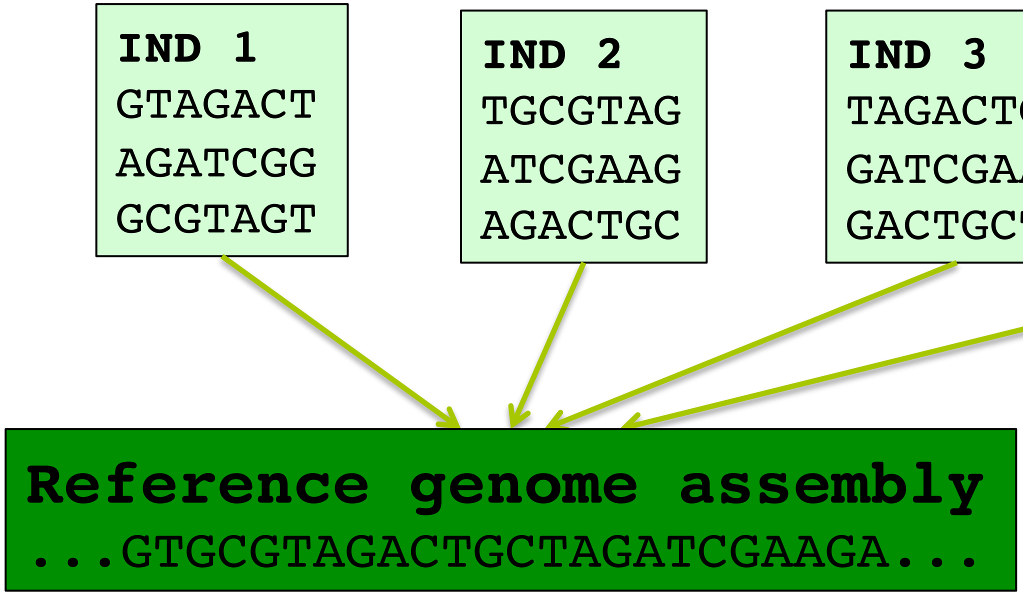
IND 2
TGCGTAG
ATCGAAG
AGACTGC

IND 3
TAGACTG
GATCGAA
GACTGCT

IND 4
AGATCGA
GTAGACT
GATCGAA

Reference genome assembly
...GTGCGTAGACTGCTAGATCGAAGA...

Re-sequencing



IND 1
GTAGACT
AGATCGG
GCGTAGT

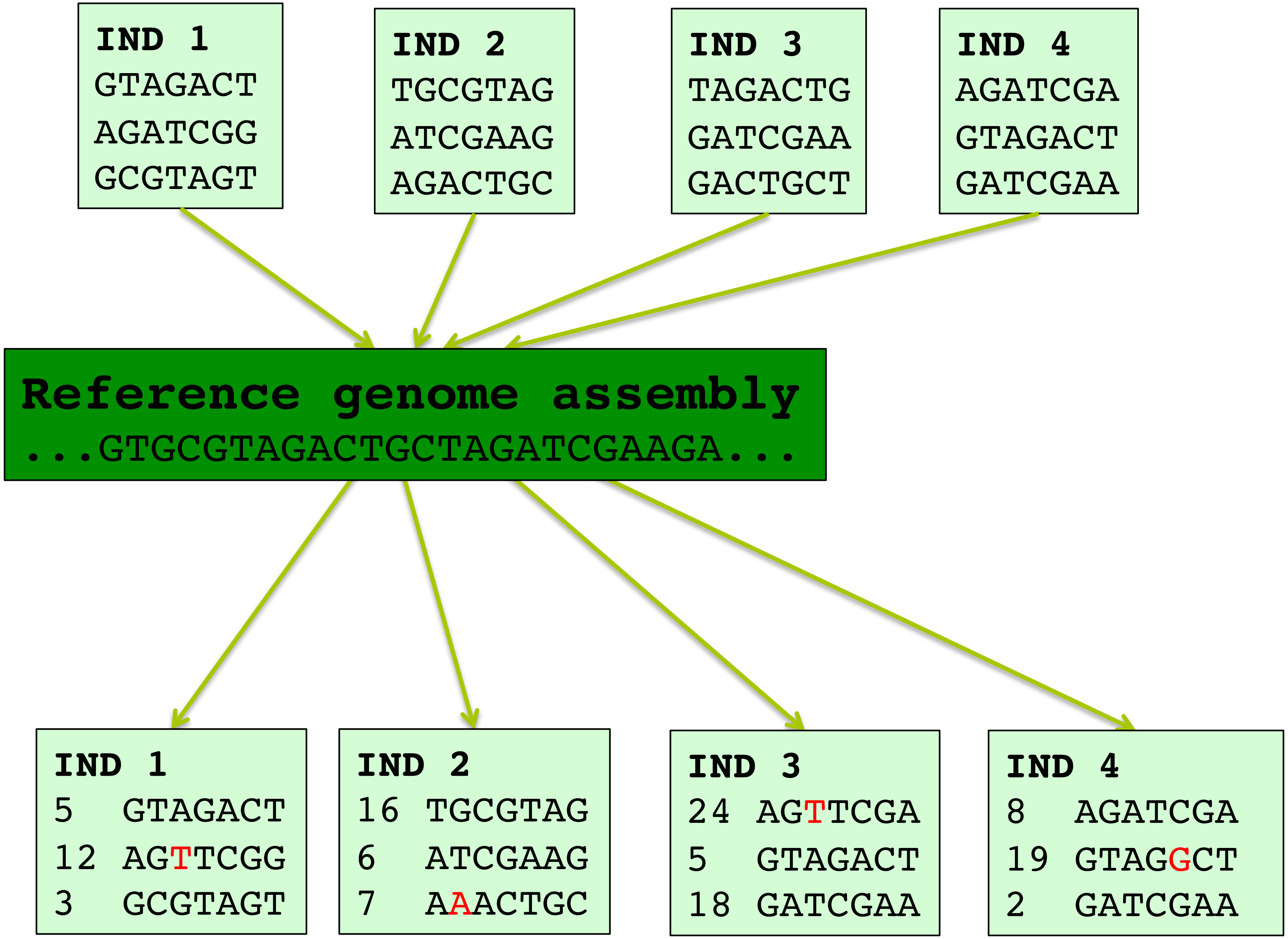
IND 2
TGCGTAG
ATCGAAG
AGACTGC

IND 3
TAGACTG
GATCGAA
GACTGCT

IND 4
AGATCGA
GTAGACT
GATCGAA

Reference genome assembly
...GTGCGTAGACTGCTAGATCGAAGA...

Re-sequencing



IND 1
GTAGACT
AGATCGG
GCGTAGT

IND 2
TGCGTAG
ATCGAAG
AGACTGC

IND 3
TAGACTG
GATCGAA
GACTGCT

IND 4
AGATCGA
GTAGACT
GATCGAA

Reference genome assembly
...GTGCGTAGACTGCTAGATCGAAGA...

IND 1
5 GTAGACT
12 AGTTCGG
3 GCGTAGT

IND 2
16 TGCGTAG
6 ATCGAAG
7 AAACTGC

IND 3
24 AGTTCGA
5 GTAGACT
18 GATCGAA

IND 4
8 AGATCGA
19 GTAGGCT
2 GATCGAA

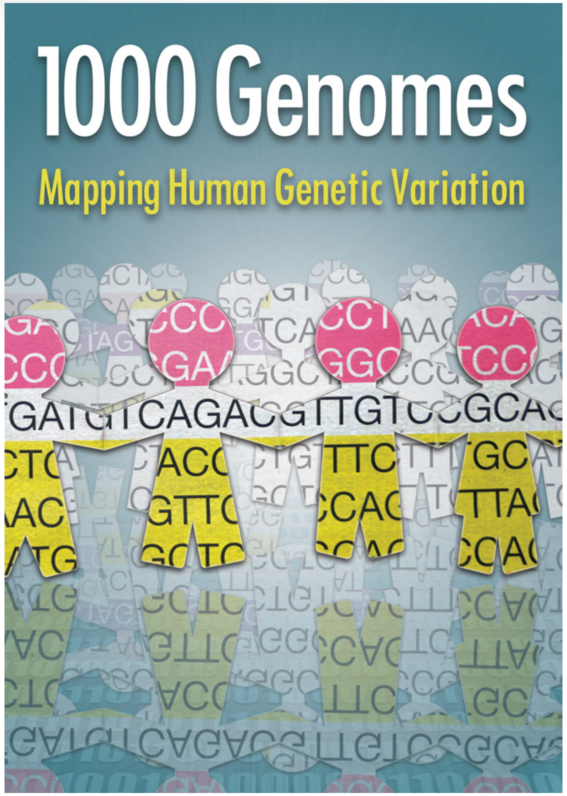
Rare variants in human

SciLifeLab

deCODE genetics

UK 10K

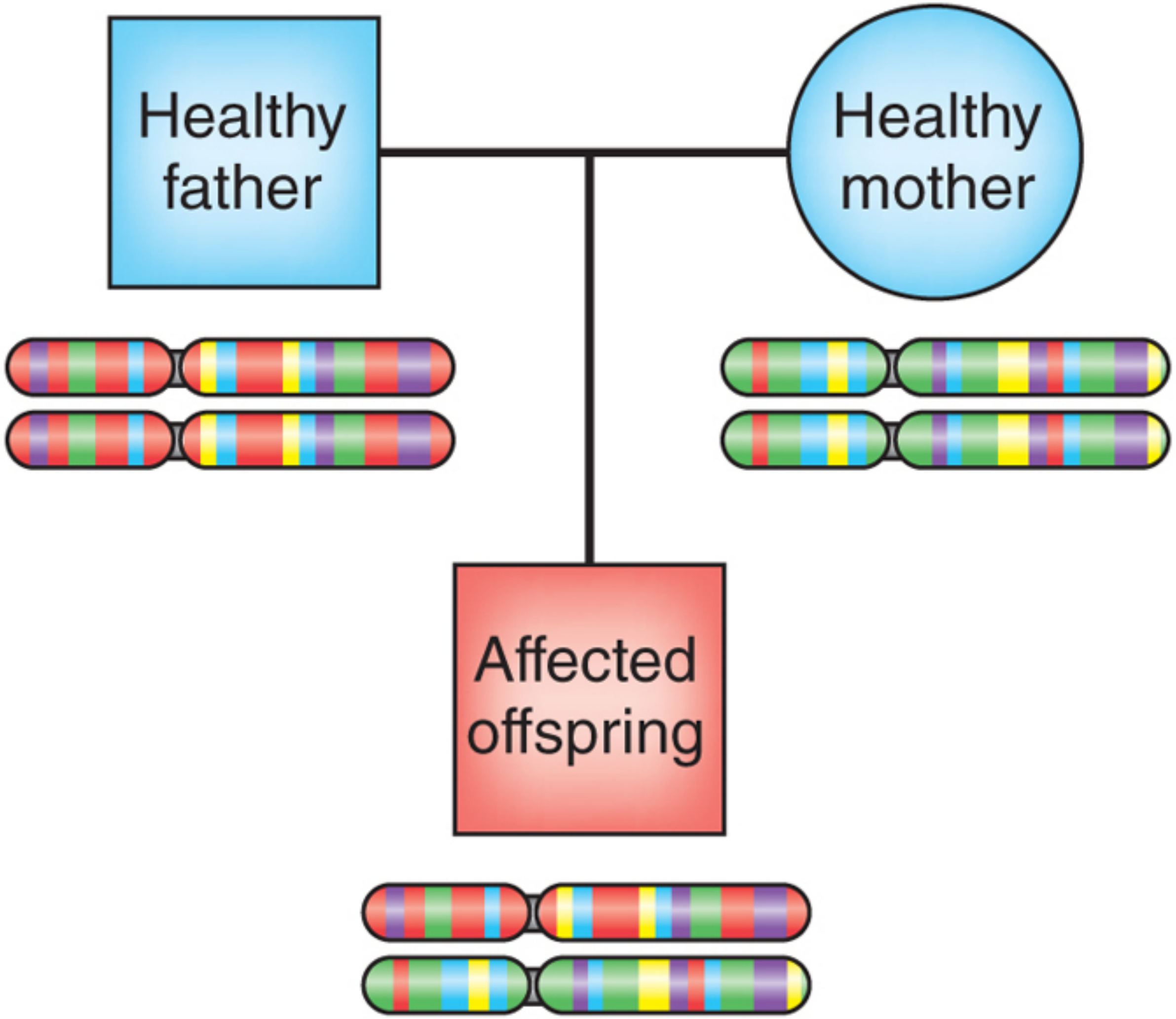
RARE GENETIC VARIANTS IN HEALTH AND DISEASE



1000 Genomes

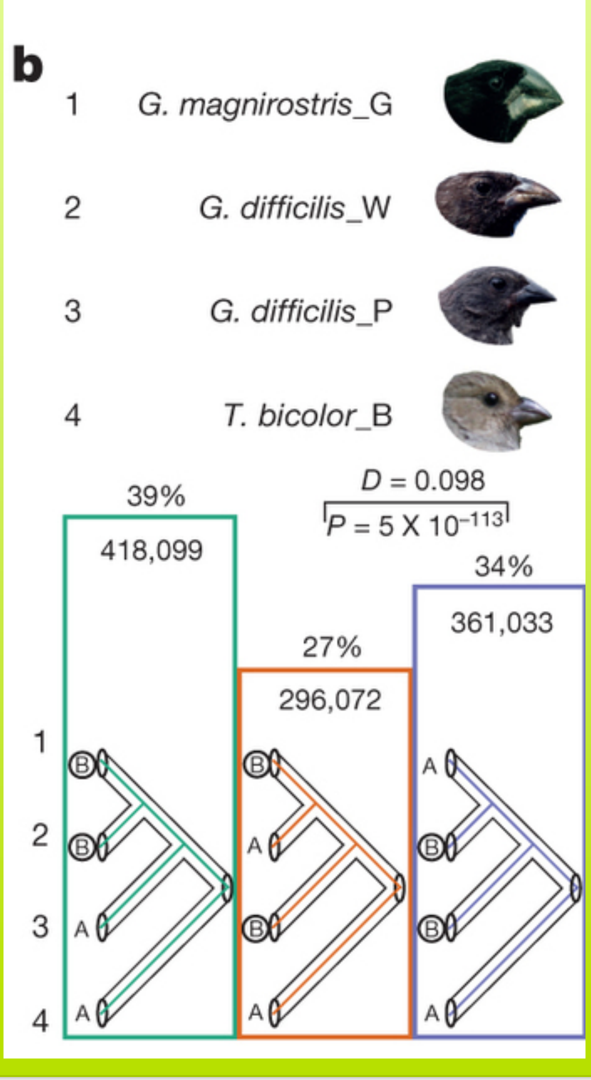
Mapping Human Genetic Variation

Exome sequencing in trios to detect *de novo* coding variants

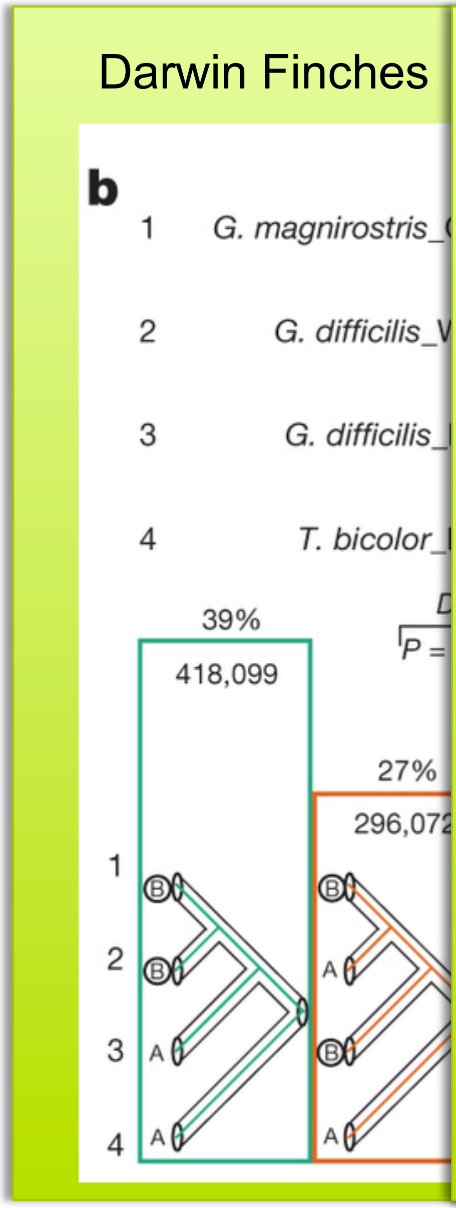


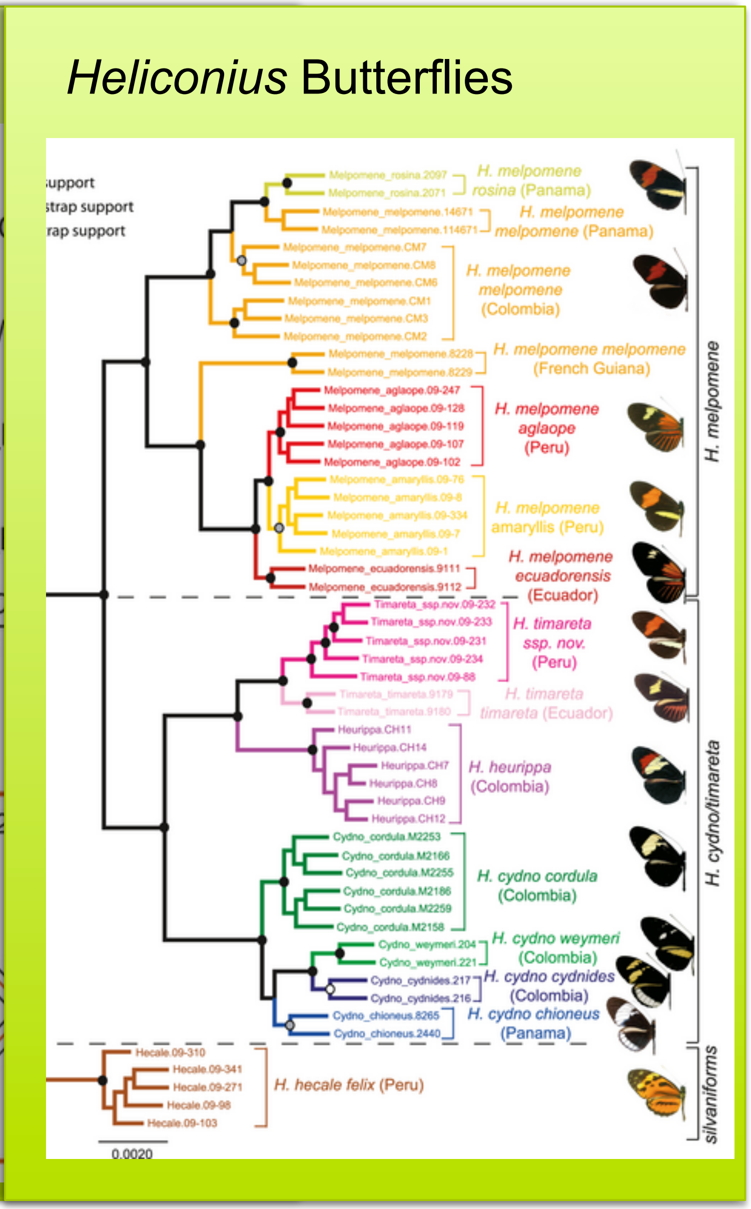
Population genetics – speciation, adaptive evolution

Darwin Finches



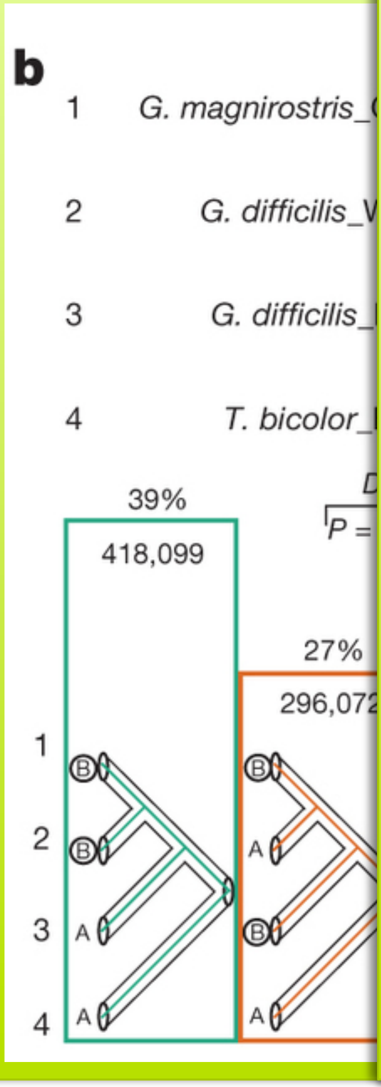
Population genetics – speciation, adaptive evolution



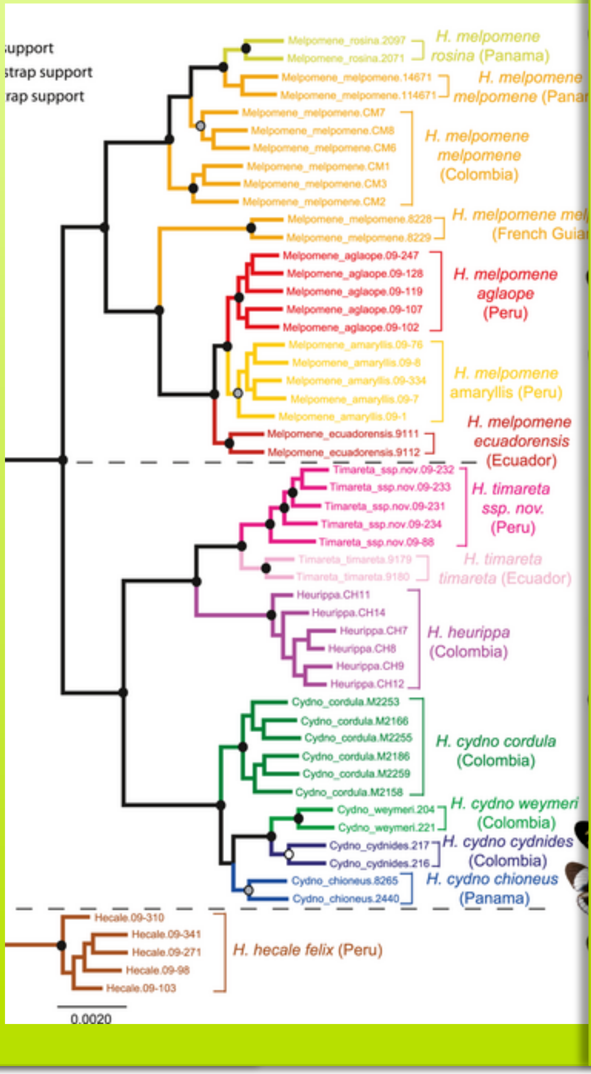

Heliconius Butterflies
support
strap support
rap support
H. melpomene rosina (Panama)
H. melpomene melpomene (Panama)
H. melpomene melpomene (Colombia)
H. melpomene melpomene (French Guiana)
H. melpomene aglaope (Peru)
H. melpomene amaryllis (Peru)
H. melpomene ecuadorensis (Ecuador)
H. timareta ssp. nov. (Peru)
H. timareta timareta (Ecuador)
H. heurippa (Colombia)
H. cydno cordula (Colombia)
H. cydno weymeri (Colombia)
H. cydno cydnides (Colombia)
H. cydno chioneus (Panama)
H. hecale felix (Peru)
H. melpomene
H. cydno/timareta
silvaniforms
0.0020

Population genetics – speciation, adaptive evolution

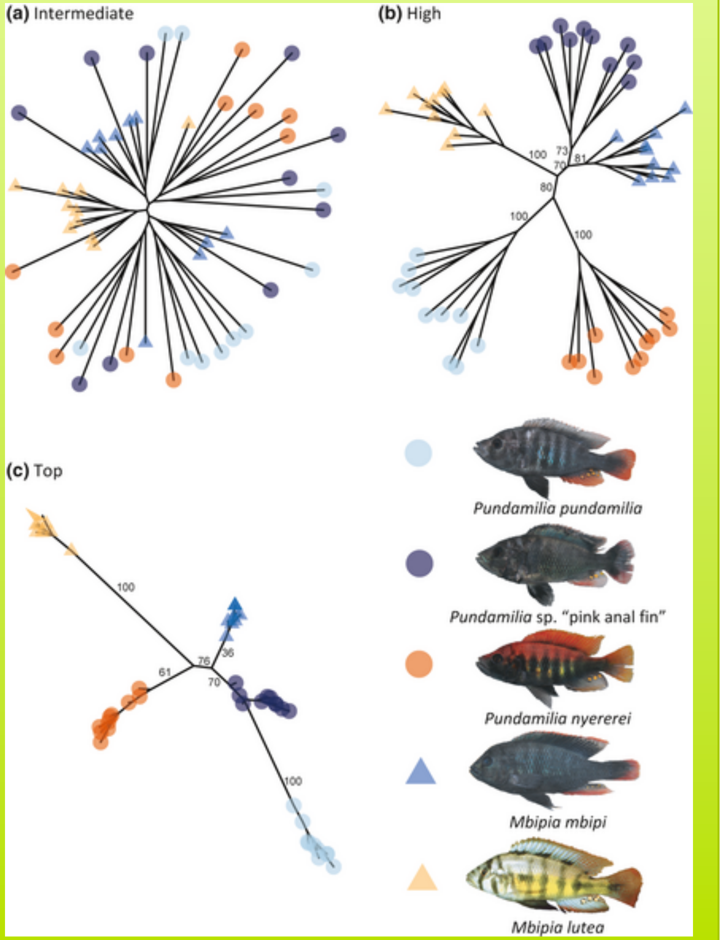
Darwin Finches



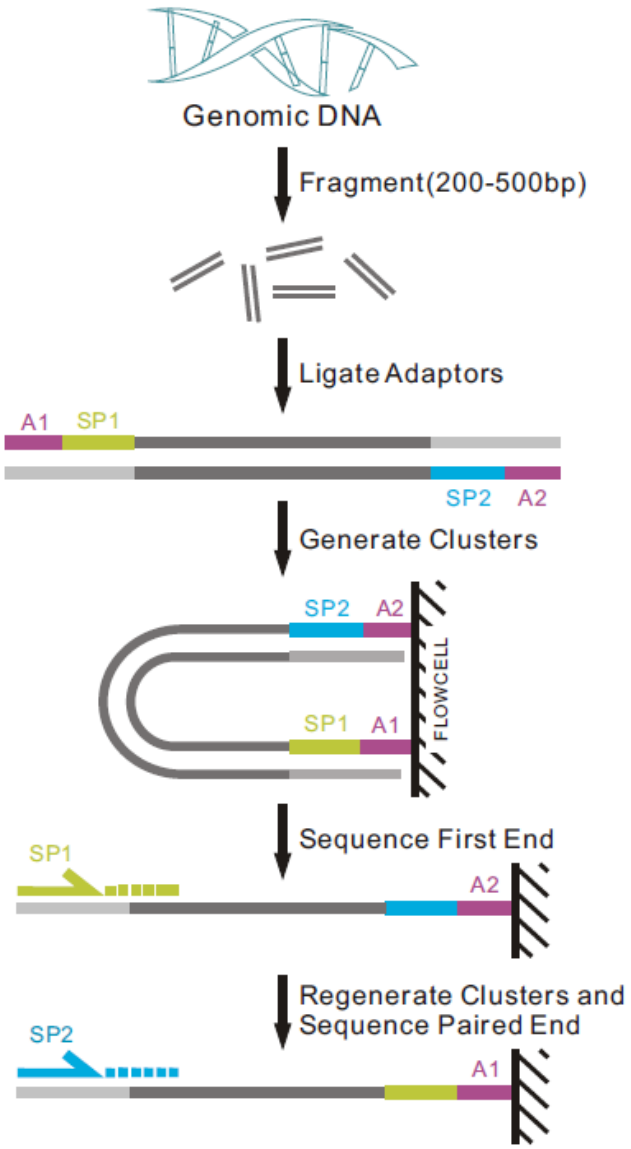
Heliconius Butterflies


support
strap support
rap support
H. melpomene rosina (Panama)
H. melpomene melpomene (Pana
H. melpomene melpomene (Colombia)
H. melpomene mel (French Guia
H. melpomene aglaope (Peru)
H. melpomene amaryllis (Peru)
H. melpomene ecuadorensis (Ecuador)
H. timareta ssp. nov. (Peru)
H. timareta timareta (Ecuador)
H. heurippa (Colombia)
H. cydno cordula (Colombia)
H. cydno weymeri (Colombia)
H. cydno cydnides (Colombia)
H. cydno chioneus (Panama)
H. hecale felix (Peru)
0.0020

Lake Victoria cechlid fishes



Paired end sequencing



Pair-end reads

- Two .fastq files containing the reads are created
- The order in the files are identical and naming of reads are the same with the exception of the end
- The naming of reads is changing and depends on software version

ID_R1_001.fastq

```
@HISEQ:100:C3MG8ACXX:
5:1101:1160:2197 1:N:0:ATCACG
CAGTTGCGATGAGAGCGTTGAGAAGTATAATAGG
AGTTAAACTGAGTAACAGGATAAGAAATAGTGAG
ATATGGAAACGTTGTGGTCTGAAAGAAGATGT
+
B@CFFFFFHHHHHGJJJJJJJJJJJFHHIIIIJJ
JIHGIIJJJJIJIJIJJJJIIJJJJJIIEIHHIJ
HGHHHHHDFFFEDDDDDCDDDCDDDDDDDCDC
```

ID_R2_001.fastq

```
@HISEQ:100:C3MG8ACXX:
5:1101:1160:2197 2:N:0:ATCACG
CTTCGTCCACTTTCATTATTCCTTTCATACATG
CTCTCCGGTTTAGGGTACTCTTGACCTGGCCTT
TTTTCAAGACGTCCCTGACTTGATCTTGAAACG
+
CCCFFFFFHHHHHJJJJIJJJJJJJJJJJJJJJ
JJJJJJJIJIJGIJHBGHHIIIJIJJJJJJJJI
JJJHFFFFFFDDDDDDDDDDDDDDDEDCCDDDD
```

Pair-end reads

SciLifeLab

- Two .fastq files containing the reads are created
- The order in the files are identical and naming of reads are the same with the exception of the end
- The naming of reads is changing and depends on software version

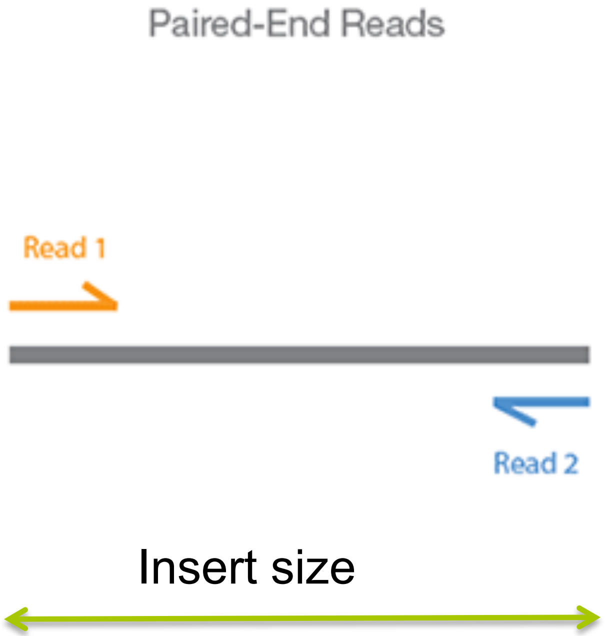
ID_R1_001.fastq

```
@HISEQ:100:C3MG8ACXX:
5:1101:1160:2197 1:N:0:ATCACG
CAGTTGCGATGAGAGCGTTGAGAAGTATAATAGG
AGTTAAACTGAGTAACAGGATAAGAAATAGTGAG
ATATGGAAACGTTGTGGTCTGAAAGAAGATGT
+
B@CFFFFFHHHHHGJJJJJJJJJJJFHHIIIIJJ
JIHGIIJJJJIJIJIJJJJIIJJJJJIIEIHHIJ
HGHHHHHDFFFEDDDDDCDDDCDDDDDDDCDC
```

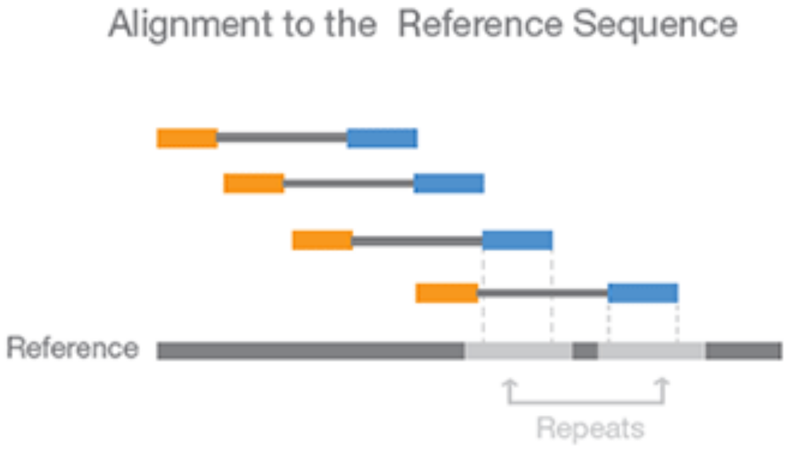
ID_R2_001.fastq

```
@HISEQ:100:C3MG8ACXX:
5:1101:1160:2197 2:N:0:ATCACG
CTTCGTCCACTTTCATTATTCCTTTCATACATG
CTCTCCGGTTTAGGGTACTCTTGACCTGGCCTT
TTTTCAAGACGTCCCTGACTTGATCTTGAAACG
+
CCCFFFFFHHHHHJJJJIJJJJJJJJJJJJJJJ
JJJJJJJIJIJGIJHBGHHIIIJIJJJJJJJJI
JJJHFFFFFFDDDDDDDDDDDDDDDEDCCDDDD
```

Mapping of pair-end reads

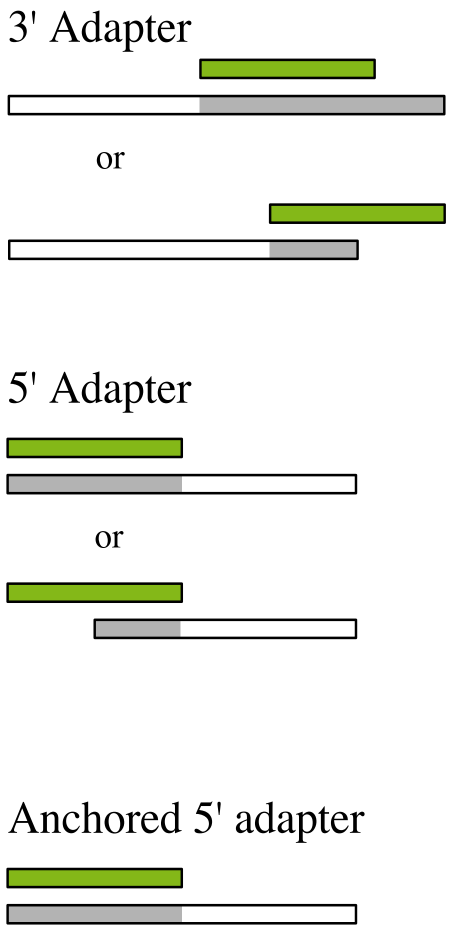


Insert size



Adaptor trimming

```
module add cutadapt
```



When the adaptor has been read in sequencing it is present in reads and needs to be removed prior to mapping

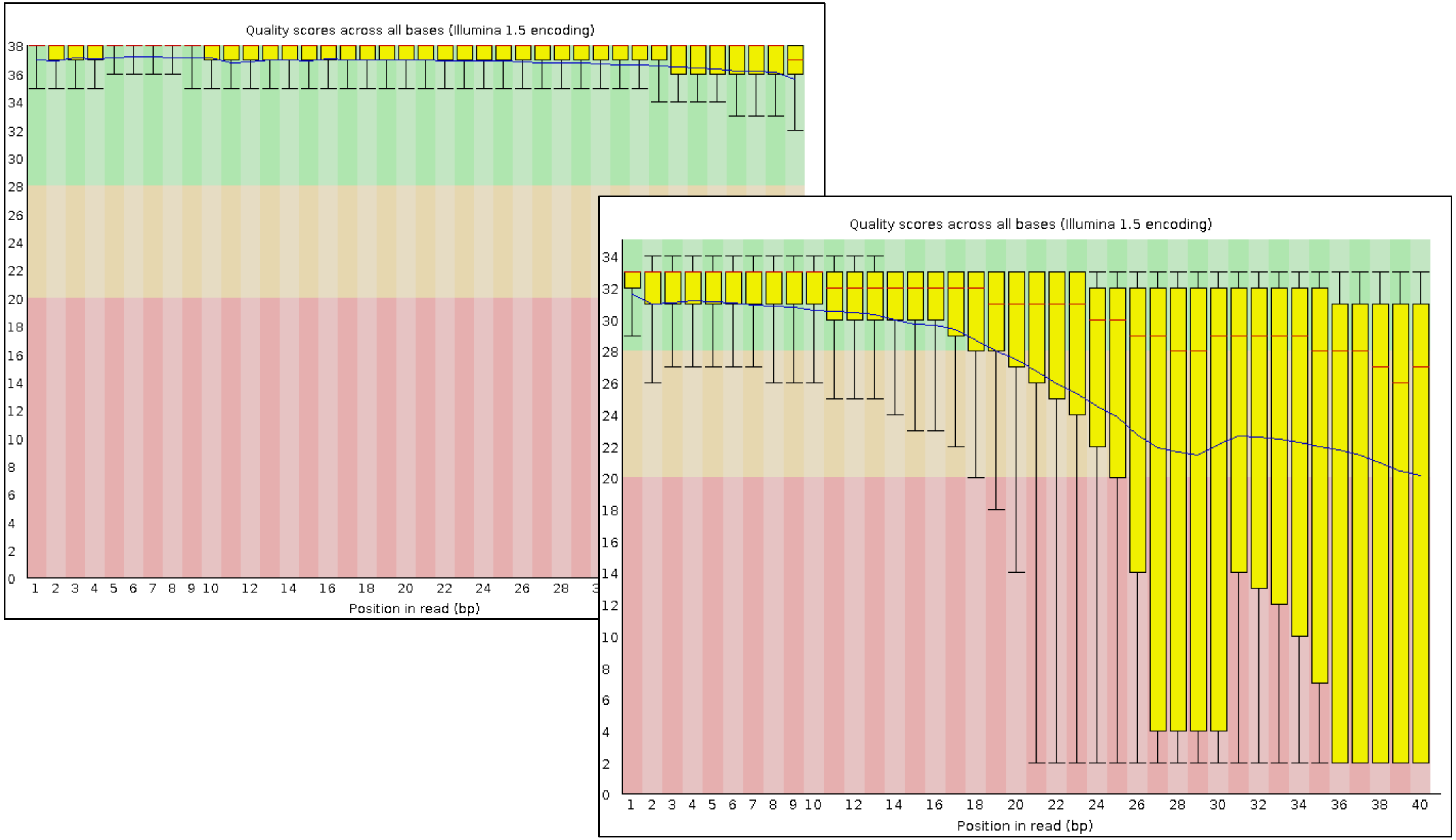
□ Read

🟩 Adapter

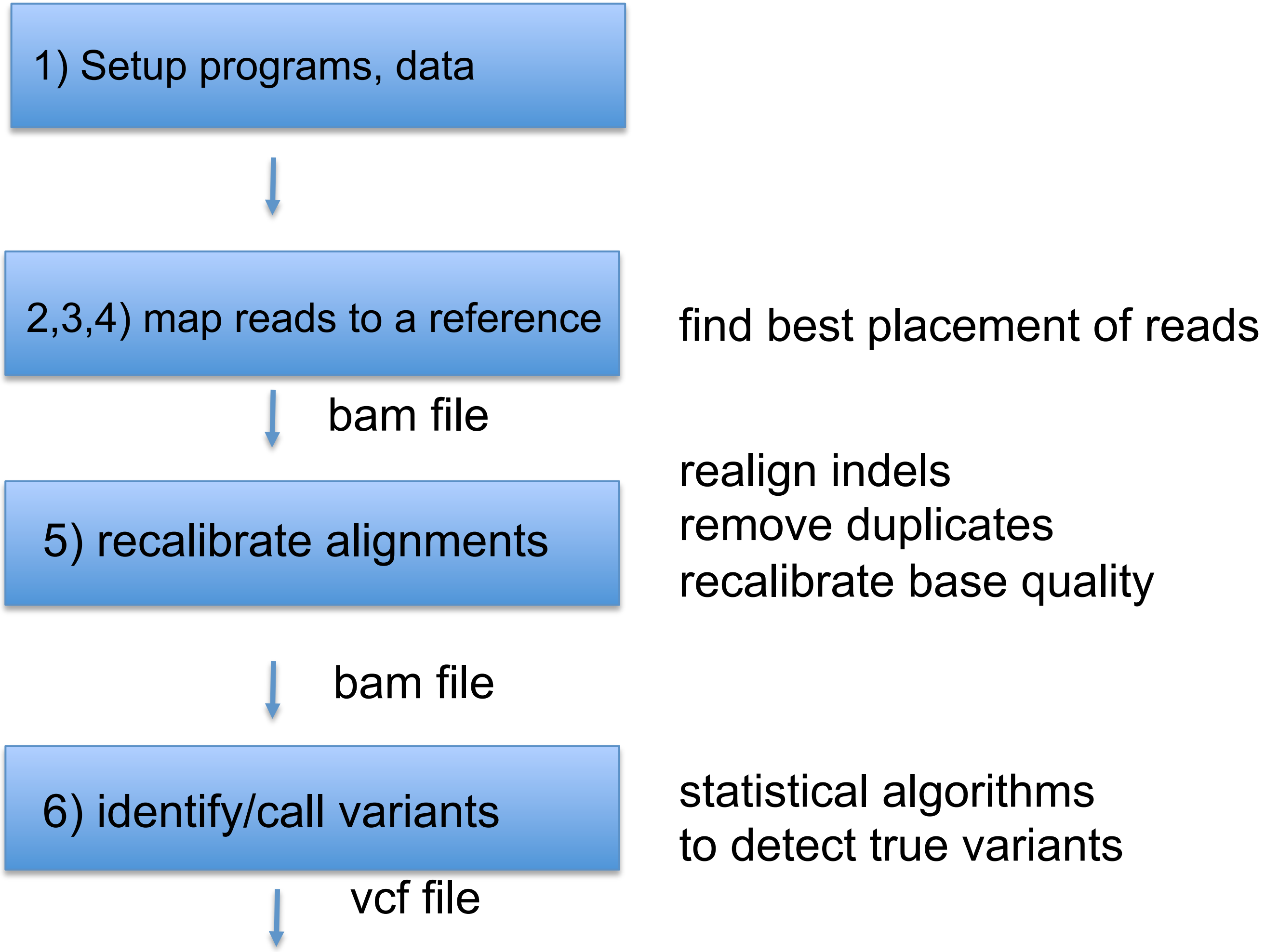
▨ Removed sequence

Basic quality control - FASTQC

```
module add FastQC
```



Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

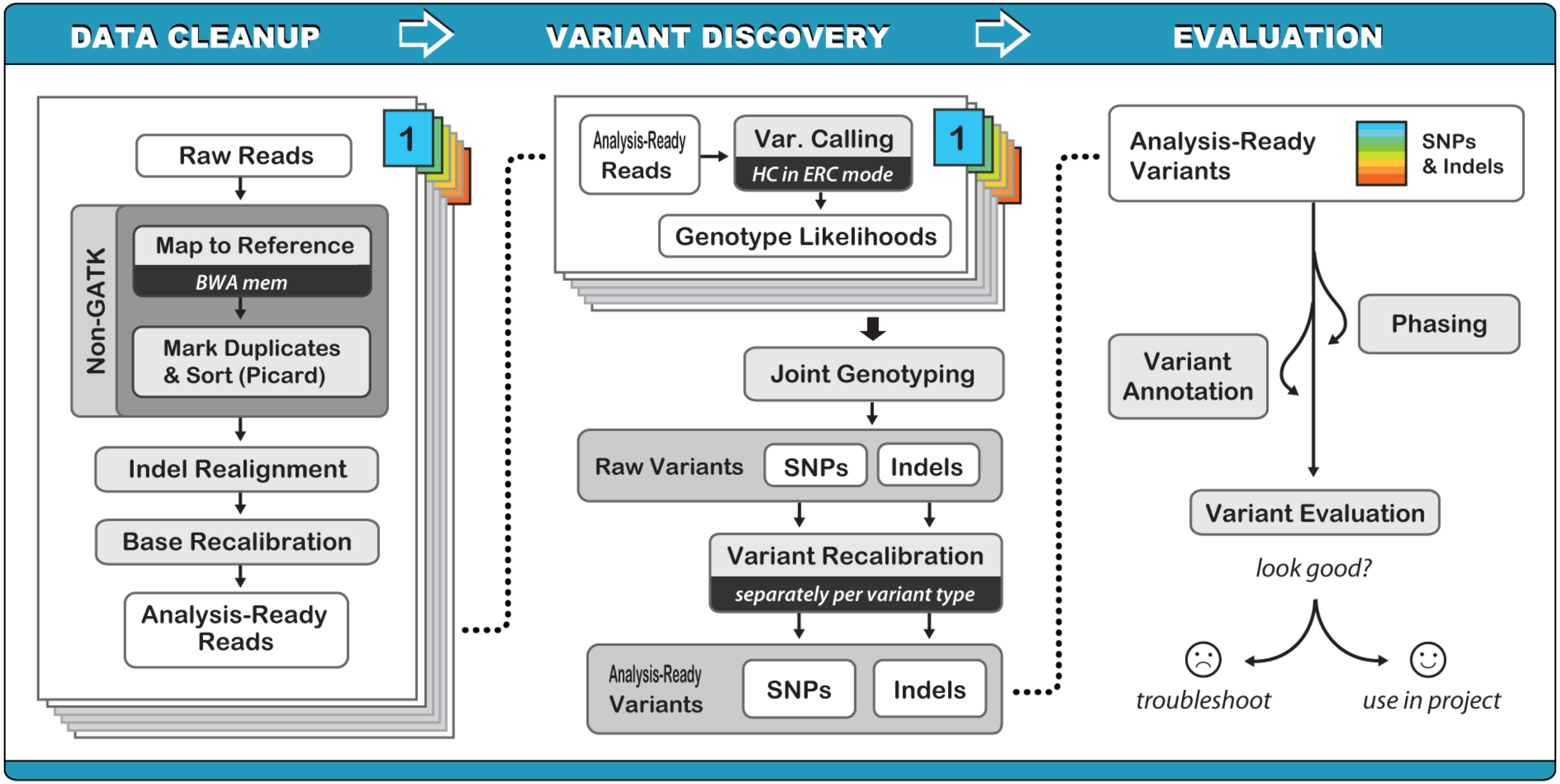
bam file

6) identify/call variants

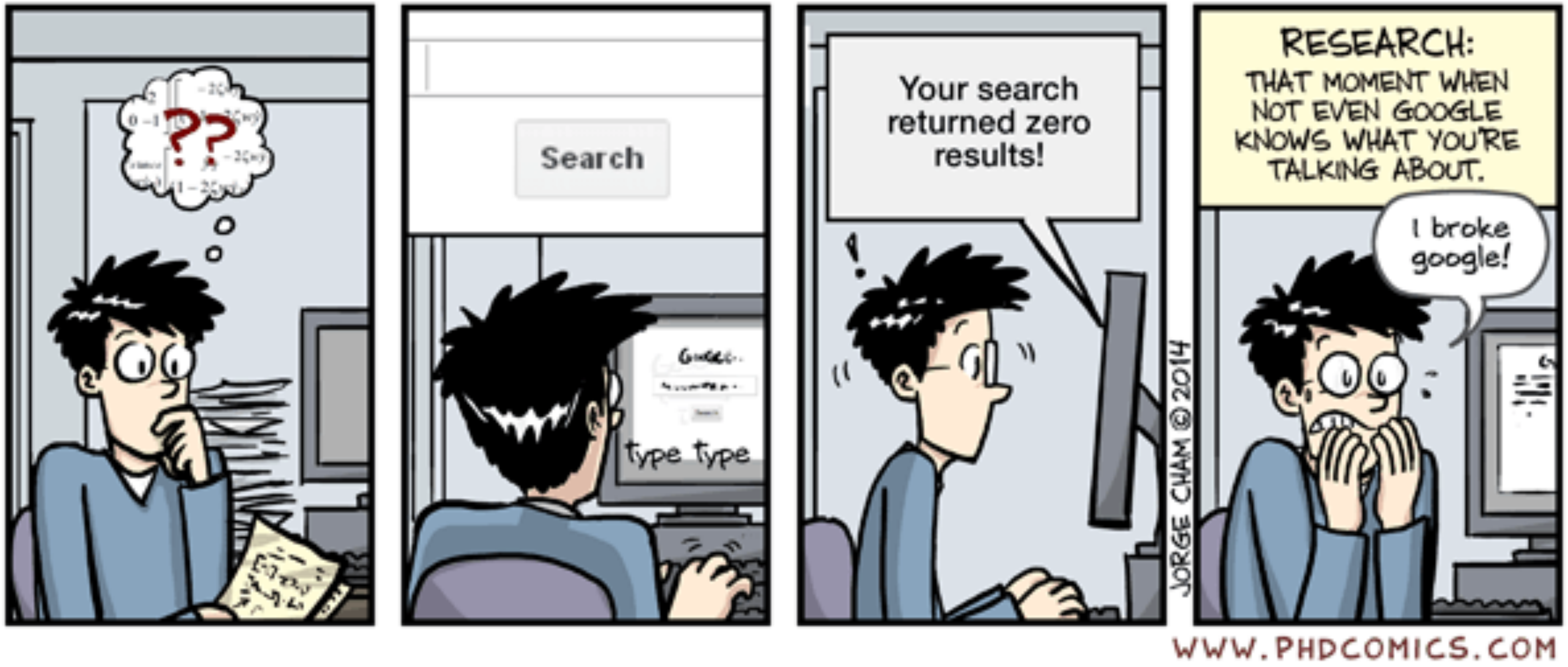
statistical algorithms
to detect true variants

vcf file

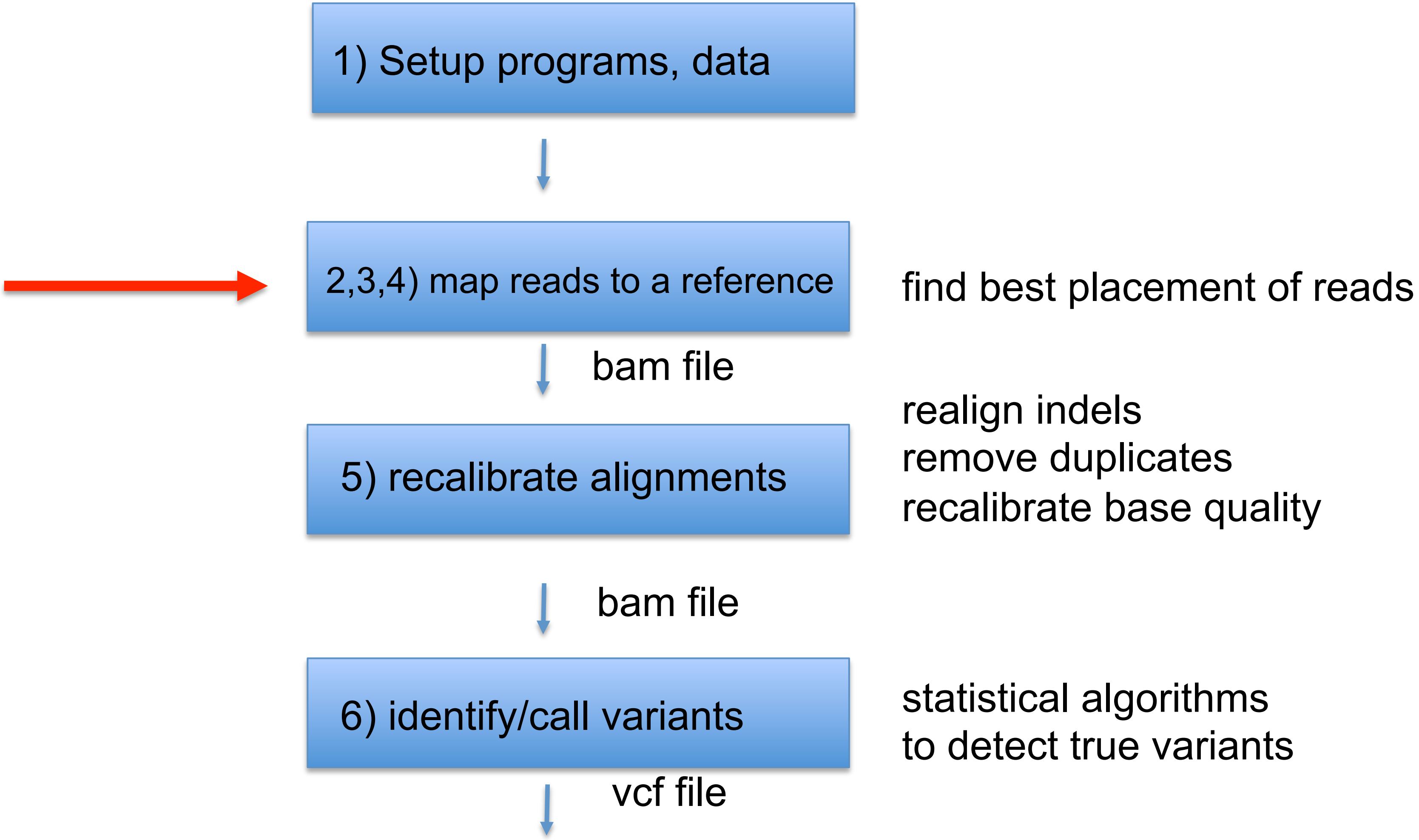
GATK version



When in doubt, google it!


??
Search
type type
Your search returned zero results!
RESEARCH: THAT MOMENT WHEN NOT EVEN GOOGLE KNOWS WHAT YOU'RE TALKING ABOUT.
I broke google!
JORGE CHAM © 2014
WWW.PHDCOMICS.COM

Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

bam file

6) identify/call variants

statistical algorithms
to detect true variants

vcf file

brute force

```
TCGATCC
x
GACCTCATCGATCCCACTG
```

brute force

```
 TCGATCC
 x
GACCTCATCGATCCCACTG
```

brute force

```
  TCGATCC
  x
GACCTCATCGATCCCACTG
```

brute force

```
   TCGATCC
   x
GACCTCATCGATCCCACTG
```

brute force

```
    TCGATCC
    ||x
GACCTCATCGATCCCACTG
```

brute force

```
     TCGATCC
     x
GACCTCATCGATCCCACTG
```

brute force

```
      TCGATCC
      x
GACCTCATCGATCCCACTG
```

brute force

```
       TCGATCC
       |||||||
GACCTCATCGATCCCACTG
```

hash tables

build an index of the reference sequence for fast access

seed length 7

```
0    5    10   15

GACCTCATCGATCCCACTG
GACCTCA              →   chromosome 1, pos 0
 ACCTCAT             →   chromosome 1, pos 1
  CCTCATC            →   chromosome 1, pos 2
   CTCATCG           →   chromosome 1, pos 3
    TCATCGA          →   chromosome 1, pos 4
     CATCGAT         →   chromosome 1, pos 5
      ATCGATC        →   chromosome 1, pos 6
       TCGATCC       →   chromosome 1, pos 7
        CGATCCC      →   chromosome 1, pos 8
         GATCCCA     →   chromosome 1, pos 9
```

hash tables

build an index of the reference sequence for fast access

```
TCGATCC ?

                0    5    10   15

                GACCTCATCGATCCCACTG
                GACCTCA              →   chromosome 1, pos 0
                 ACCTCAT             →   chromosome 1, pos 1
                  CCTCATC            →   chromosome 1, pos 2
                   CTCATCG           →   chromosome 1, pos 3
                    TCATCGA          →   chromosome 1, pos 4
                     CATCGAT         →   chromosome 1, pos 5
                      ATCGATC        →   chromosome 1, pos 6
                       TCGATCC       →   chromosome 1, pos 7
                        CGATCCC      →   chromosome 1, pos 8
                         GATCCCA     →   chromosome 1, pos 9
```

hash tables

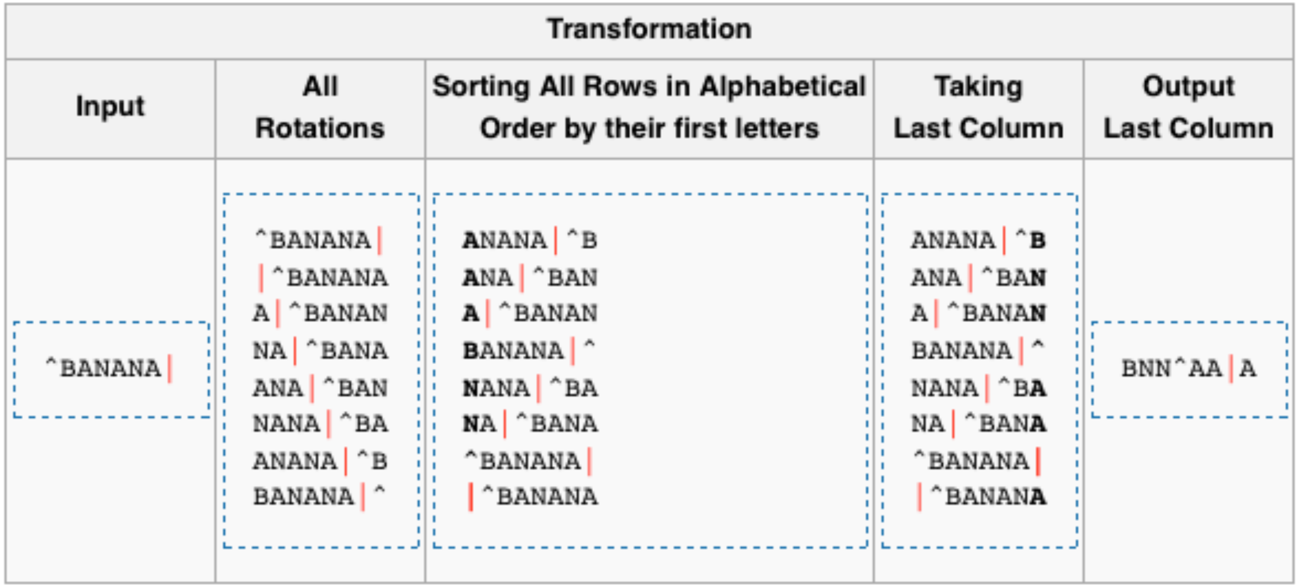
build an index of the reference sequence for fast access

```
TCGATCC = chromosome 1, pos 7

                0    5    10   15

                GACCTCATCGATCCCACTG
                GACCTCA                →   chromosome 1, pos 0
                 ACCTCAT               →   chromosome 1, pos 1
                  CCTCATC              →   chromosome 1, pos 2
                   CTCATCG             →   chromosome 1, pos 3
                    TCATCGA            →   chromosome 1, pos 4
                     CATCGAT           →   chromosome 1, pos 5
                      ATCGATC          →   chromosome 1, pos 6
                       TCGATCC         →   chromosome 1, pos 7
                        CGATCCC        →   chromosome 1, pos 8
                         GATCCCA       →   chromosome 1, pos 9
```

Burroughs-Wheeler Aligner

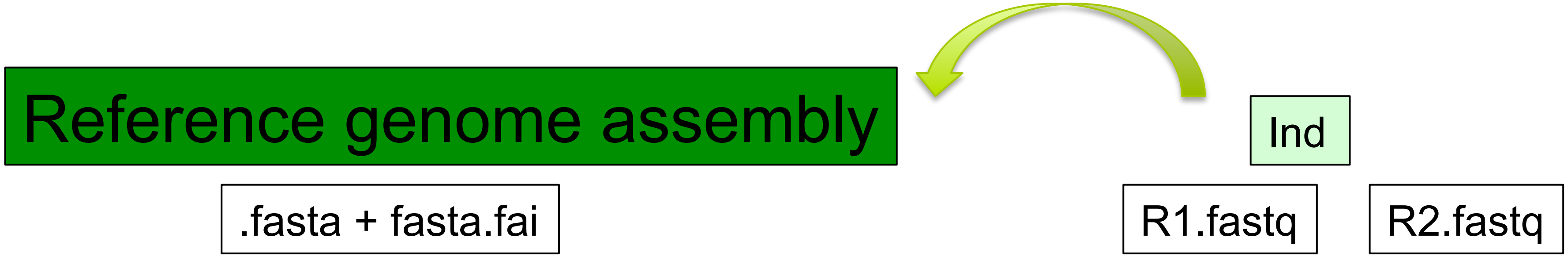


Transformation				
Input	All Rotations	Sorting All Rows in Alphabetical Order by their first letters	Taking Last Column	Output Last Column
^BANANA|	^BANANA|	ANANA|^B	ANANA|^B	BNN^AA|A
	|^BANANA	ANA|^BAN	ANA|^BAN	
	A|^BANAN	A|^BANAN	A|^BANAN	
	NA|^BANA	BANANA|^	BANANA|^	
	ANA|^BAN	NANA|^BA	NANA|^BA	
	NANA|^BA	NA|^BANA	NA|^BANA	
	ANANA|^B	^BANANA|	^BANANA|	
	BANANA|^	|^BANANA	|^BANANA	

algorithm used in computer science for file compression
original sequence can be reconstructed

BWA (`module add bwa`) Burroughs-Wheeler Aligner

Input to mapping – reference + raw reads



Reference genome assembly

.fasta + fasta.fai

Ind

R1.fastq

R2.fastq

```
>Potra000002
CACGAGGTTTCATCATGGACTTGGCACCAT
AAAAGTTCTCTTTCATTATATTCCCTTTAG
GTAAAATGATTCTCGTTCATTTGATAATTT
TGTAATAACCGGCCTCATTCAACCCATGAT
CCGACTTGATGGTGAATACTTGTGTAATAA
CTGATAATTTACTGTGATTTATATAACTAT
CTCATAATGGTTCGTCAAAATCTTTTAAAA
GATAAAAAAAACCTTTATCAATTATCTATA
TAAATTCAAATTTGTACACATTTACTAGAA
ATTACAACTCAGCAATAAAATTGACAAAAT
ATAAAACAGAACCGTTAAATAAGCTATTAT
TTATTTCATCACAAAACATCTAAGTCAAAA
```

```
@HISEQ:100:C3MG8ACXX:5:1101:116
CAGTTGCGATGAGAGCGTTGAGAAGTATAAT
ATAAGAAATAGTGAGATATGGAAACGTTGTG
+
B@CFFFFFHHHHHGJJJJJJJJJJFHHIII
IIJJJJJIIEIHHIJHGHHHHHDFFFEDDDD
@HISEQ:100:C3MG8ACXX:5:1101:144
NAGATTGTTTGTGTGCCTAAATAAATAAATA
AAGGAATTTGAAATTAAGATTGAGATATTGA
+
#1=DDFFEHHDFHHJGGIJJJJGIHIGIJJJ
FHHHIIJJIIJJIGIIJJJIJIGHGHIIJJI
@HISEQ:100:C3MG8ACXX:5:1101:156
```

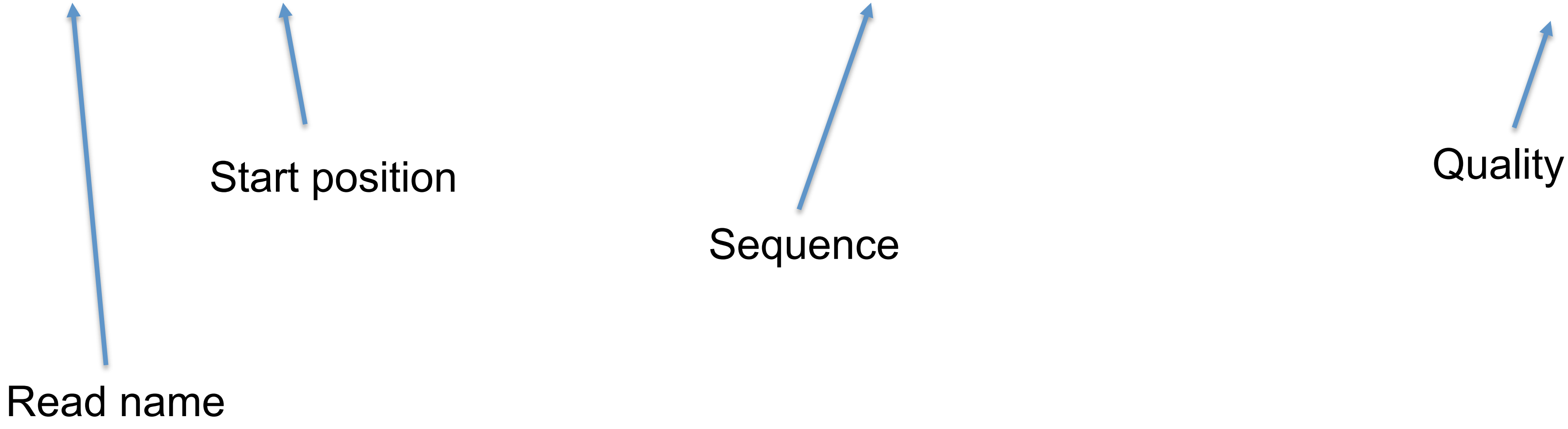
Output from mapping - SAM format

HEADER SECTION

```
@HD VN:1.0 SO:coordinate
@SQ SN:1 LN:249250621 AS:NCBI37 UR:file:/data/local/ref/GATK/human_g1k_v37.fasta M5:1b22b98cdeb4a9304cb5d48026a85128
@SQ SN:2 LN:243199373 AS:NCBI37 UR:file:/data/local/ref/GATK/human_g1k_v37.fasta M5:a0d9851da00400dec1098a9255ac712e
@SQ SN:3 LN:198022430 AS:NCBI37 UR:file:/data/local/ref/GATK/human_g1k_v37.fasta M5:fdfd811849cc2fadebc929bb925902e5
@RG ID:UM0098:1 PL:ILLUMINA PU:HWUSI-EAS1707-615LHAAXX-L001 LB:80 DT:2010-05-05T20:00:00-0400 SM:SD37743 CN:UMCORE
@RG ID:UM0098:2 PL:ILLUMINA PU:HWUSI-EAS1707-615LHAAXX-L002 LB:80 DT:2010-05-05T20:00:00-0400 SM:SD37743 CN:UMCORE
@PG ID:bwa VN:0.5.4
```

ALIGNMENT SECTION

```
8_96_444_1622 73 scaffold00005 155754 255 54M * 0 0 ATGTAAAGTATTTCCATGGTACACAGCTTGGTCGTAATGTGATTGCTGAGCCAG C@B5)5CBBCCBCCCBC@@7C
8_80_1315_464 81 scaffold00005 155760 255 54M = 154948 0 AGTACCTCCCTGGTACACAGCTTGGTAAAAATGTGATTGCTGAGCCAGACCTTC B?@?BA=>@>>7;ABA
8_17_1222_1577 73 scaffold00005 155783 255 40M1116N10M * 0 0 GGTAAAAATGTGATTGCTGAGCCAGACCTTCATCATGCAGTGAGAGACGC BB@BA??>CCBA2AAA
8_43_1211_347 73 scaffold00005 155800 255 23M1116N27M * 0 0 TGAGCCAGACCTTCATCATGCAGTGAGAGACGCAAACATGCTGGTATTTG #>8<=<@6/:@9';@7A
8_32_1091_284 161 scaffold00005 156946 255 54M = 157071 0 CGCAAACATGCTGGTAGCTGTGACACCACATCAACAGCTTGACTATGTTTGTAA BBBBB@AABACBCA8
```



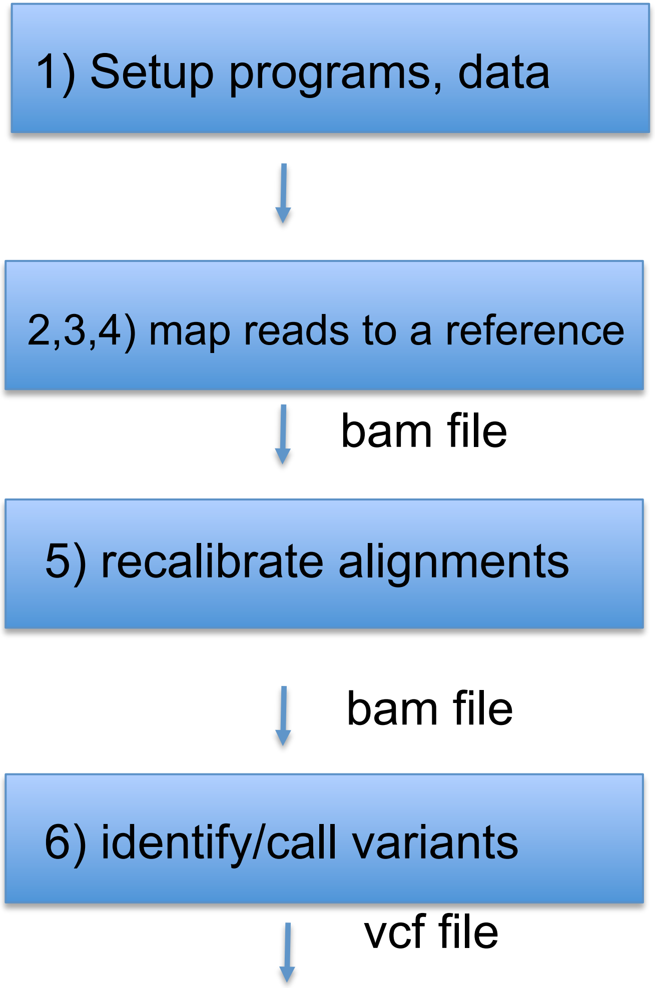
Start position

Quality

Sequence

Read name

Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

bam file

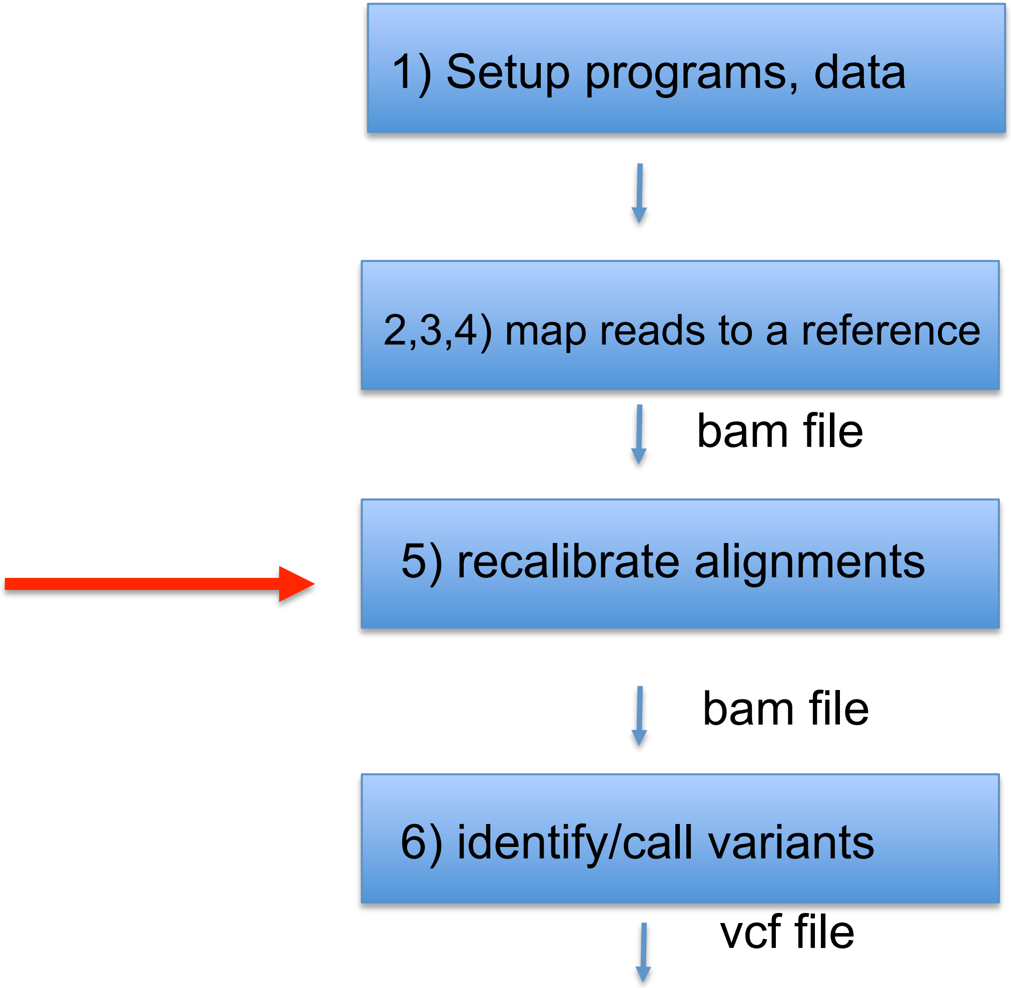
6) identify/call variants

statistical algorithms
to detect true variants

vcf file

Steps in resequencing analysis

SciLifeLab



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

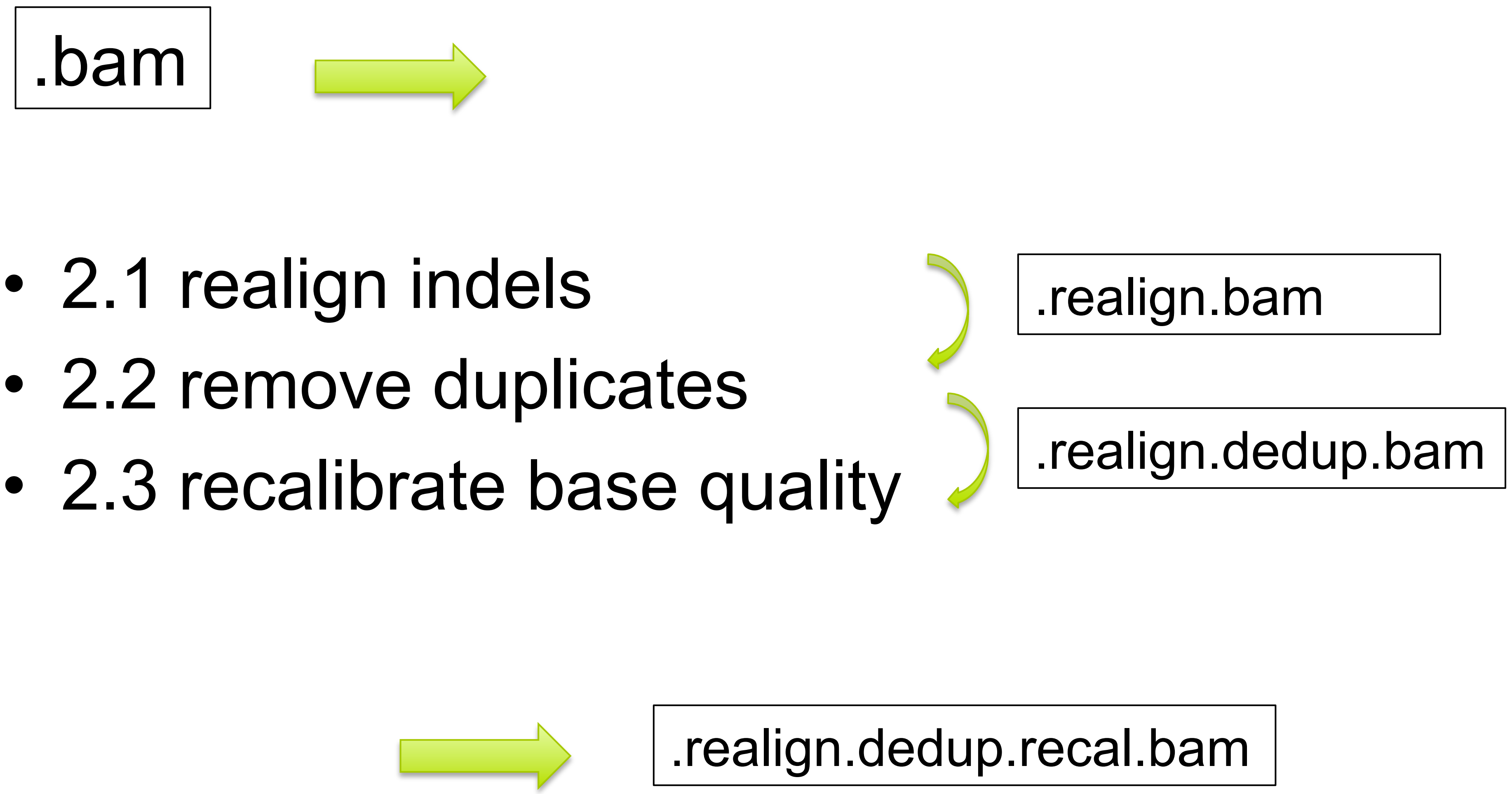
bam file

6) identify/call variants

statistical algorithms
to detect true variants

vcf file

step 2: recalibration



.bam

- 2.1 realign indels
- 2.2 remove duplicates
- 2.3 recalibrate base quality

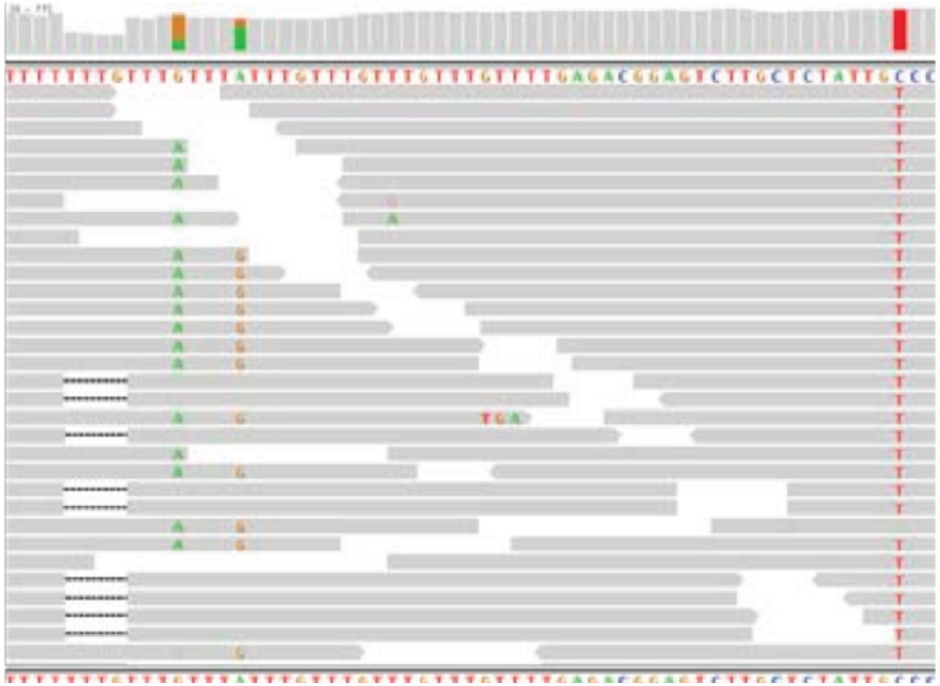
.realign.bam

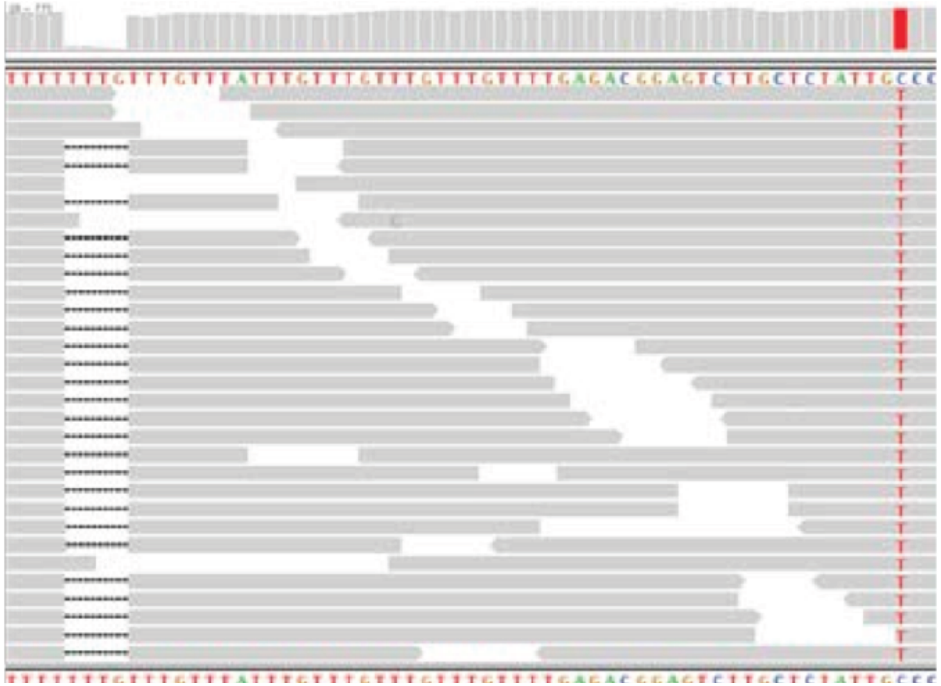
.realign.dedup.bam

.realign.dedup.recal.bam

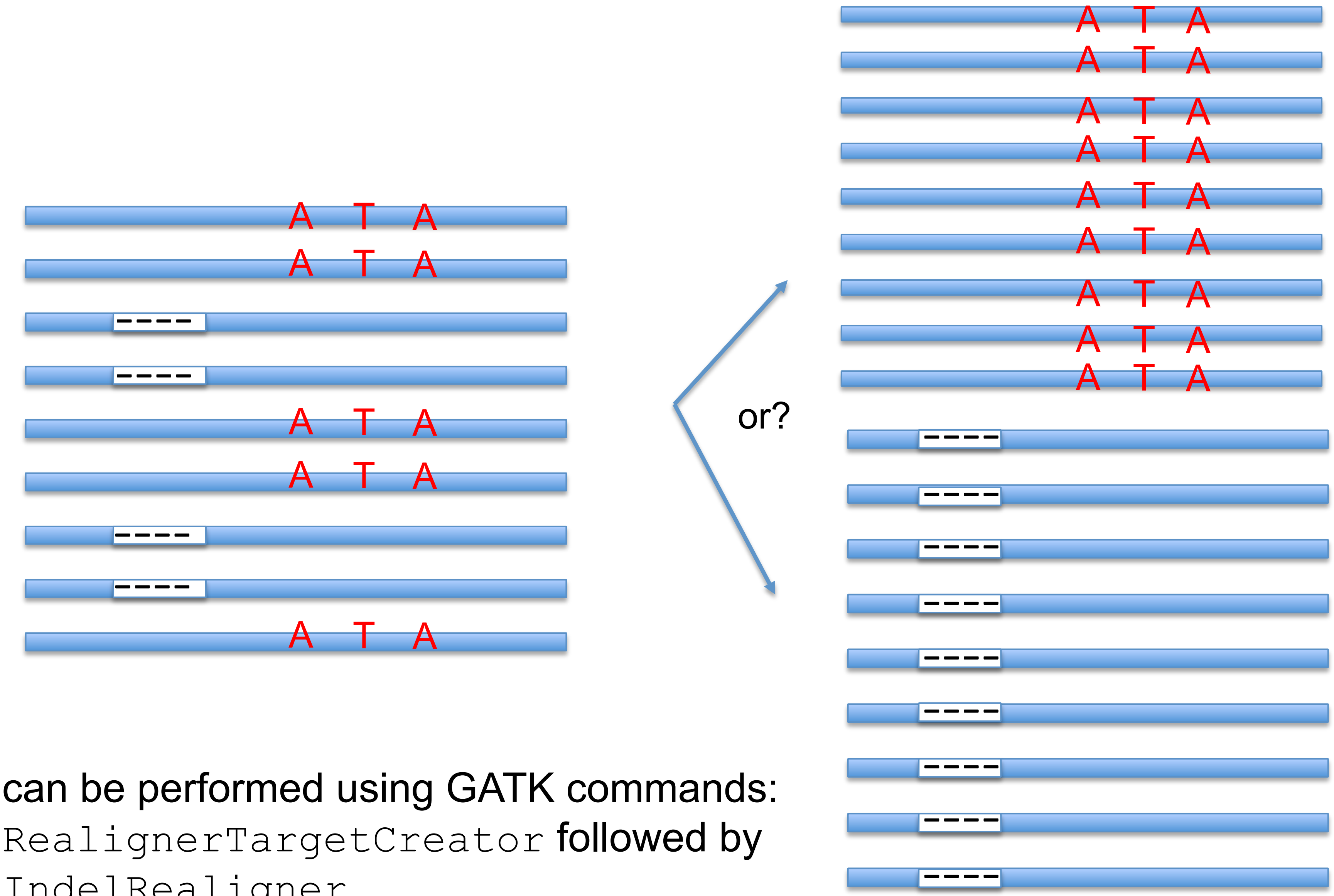
2.1 local realignment

- mapping is done one read at a time
- single variants may be split into multiple variants
- solution: realign these regions taking all reads into account

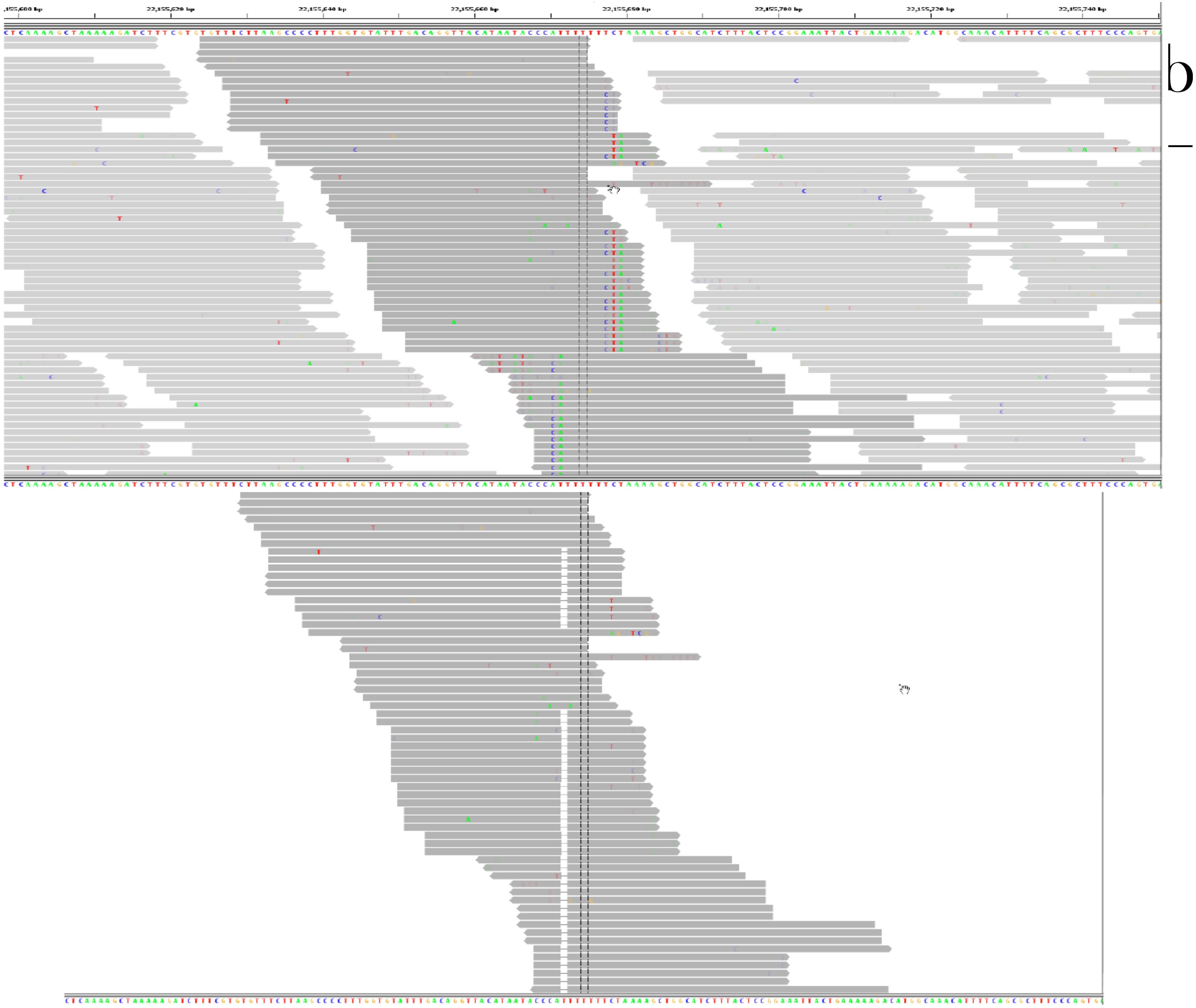




2.1 local realignment



can be performed using GATK commands: `RealignerTargetCreator` followed by `IndelRealigner`



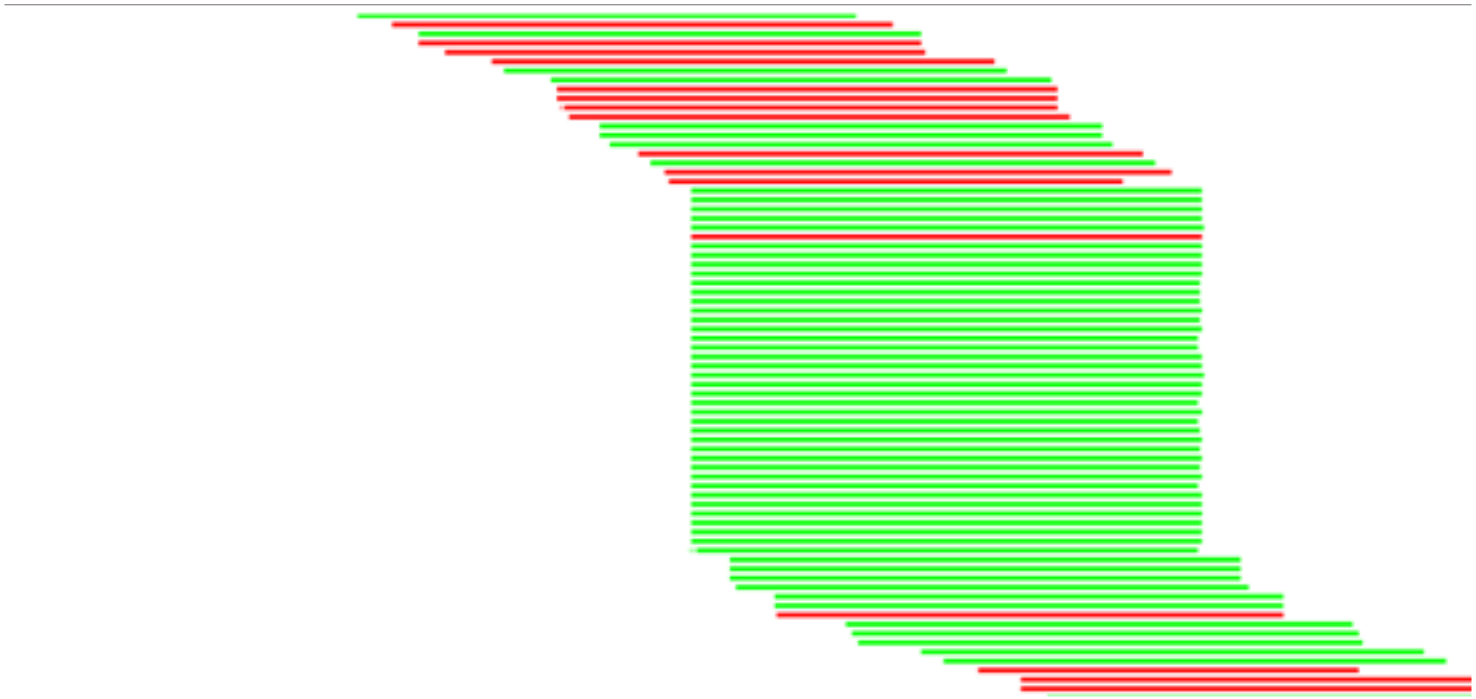
b
,155,600 bp
22,155,620 bp
22,155,640 bp
22,155,660 bp
22,155,680 bp
22,155,700 bp
22,155,720 bp
22,155,740 bp

2.2 PCR duplicates

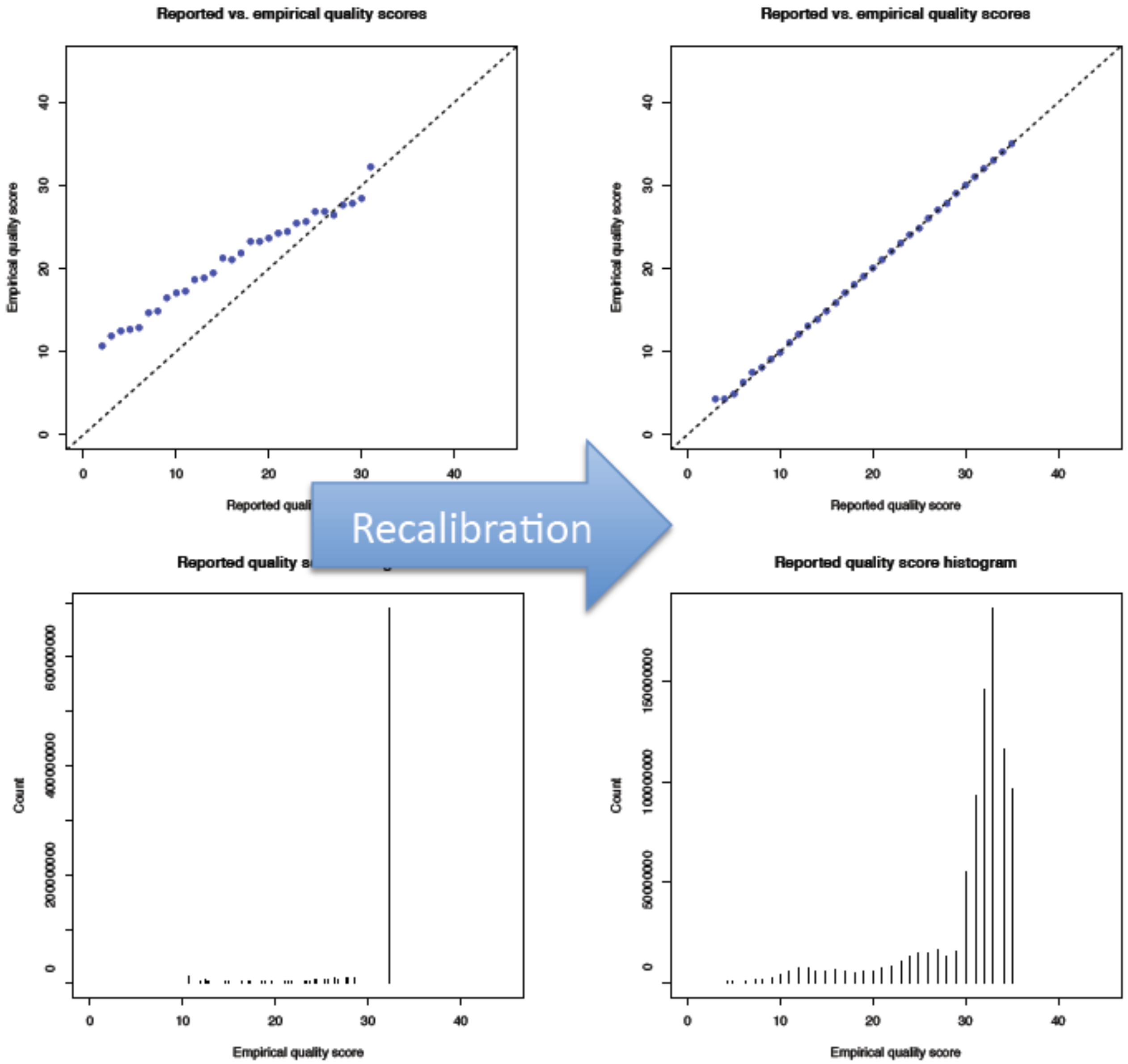
- When two or more reads originate from same molecule (artificial duplicates)
  - not independent observations
  - skew allele frequency and read depth
  - errors double counted
- PCR duplicates occur
  - during library prep, or
  - optical duplicates (one cluster read as two)
- mark or remove

Identify PCR duplicates

- Single or paired reads that map to identical positions
- Picard `MarkDuplicates`



2.3 base quality recalibration

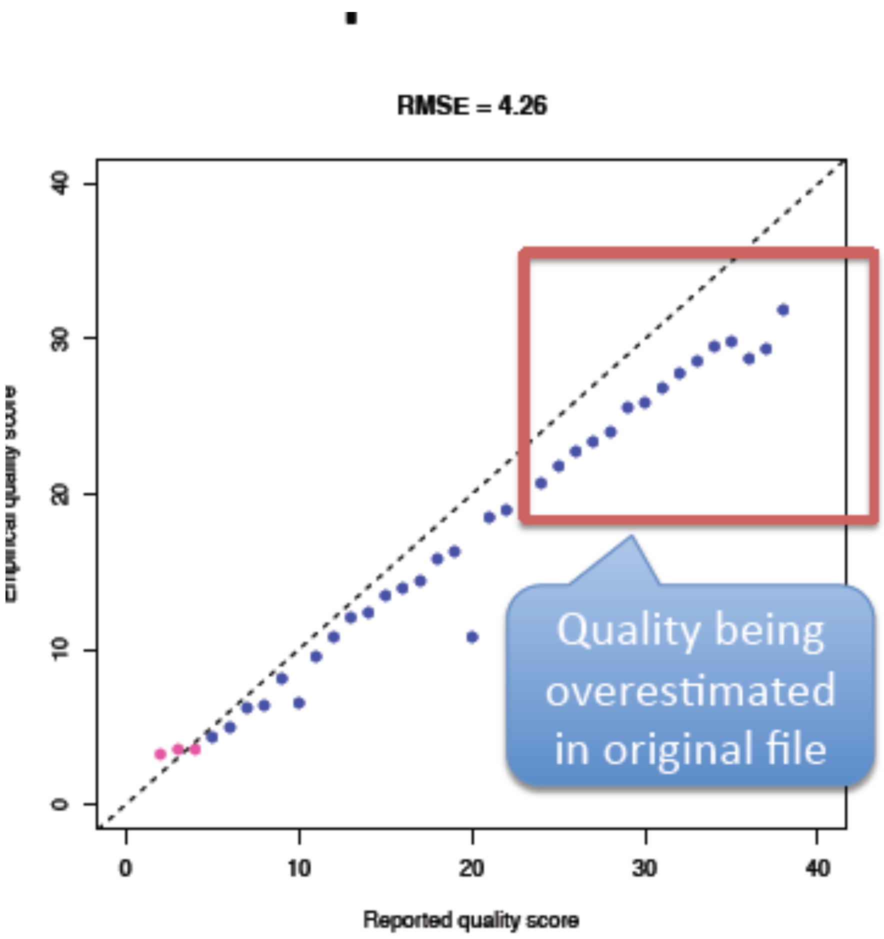


Recalibration

Recalibration Method

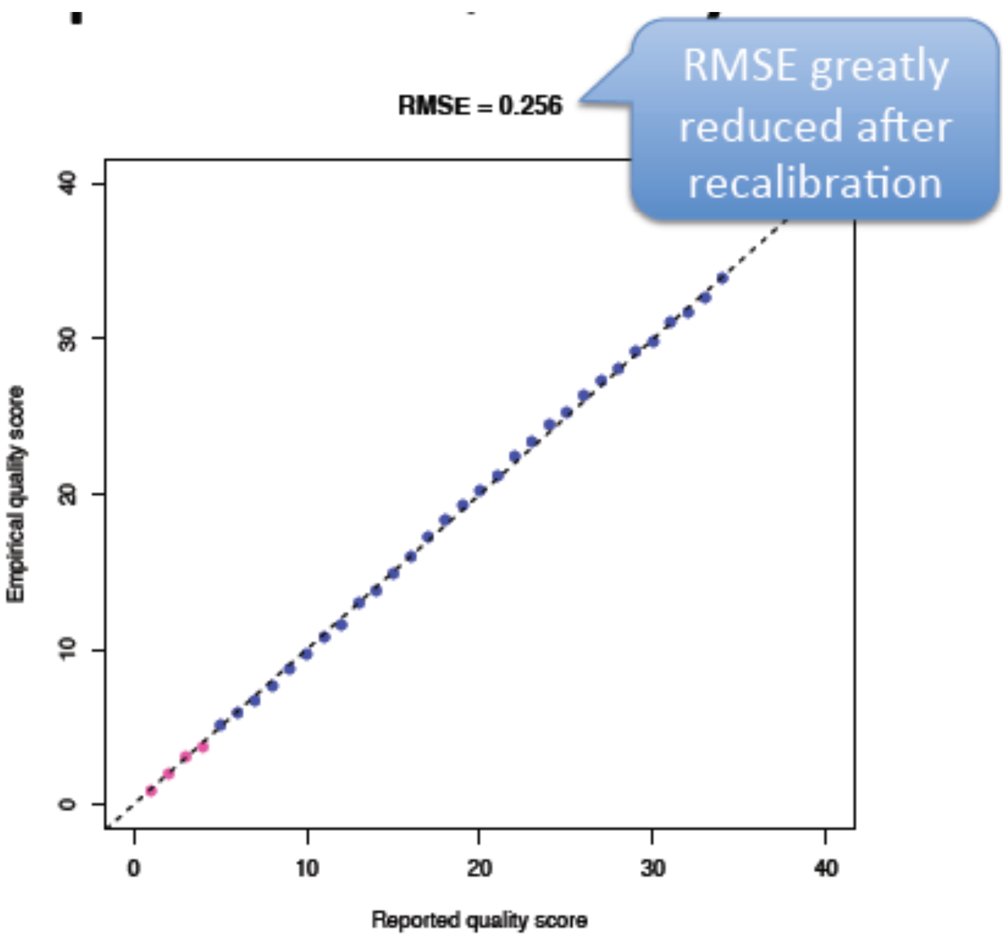
- Bin each base by
  - read group
  - called quality
  - position in read
  - local dinucleotide context
- score observed quality per bin
  - # of mismatches +1 / # of observed bases
- scale compared to reported quality

Reported vs empiral quality scores



Quality being overestimated in original file

Before Recalibration

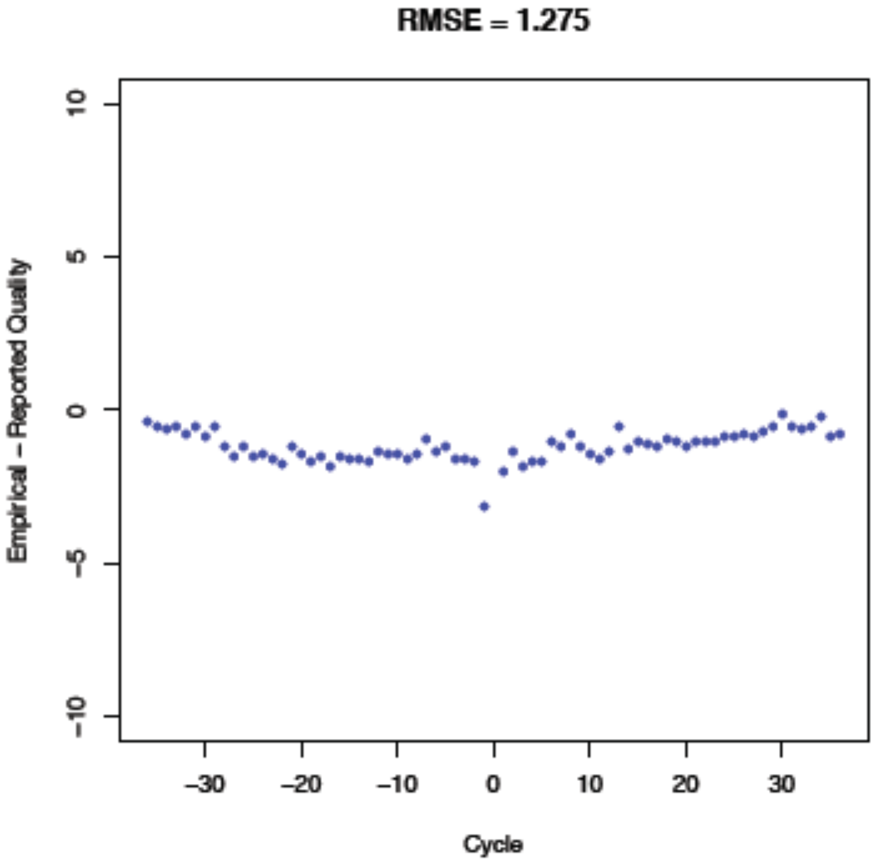


RMSE greatly reduced after recalibration

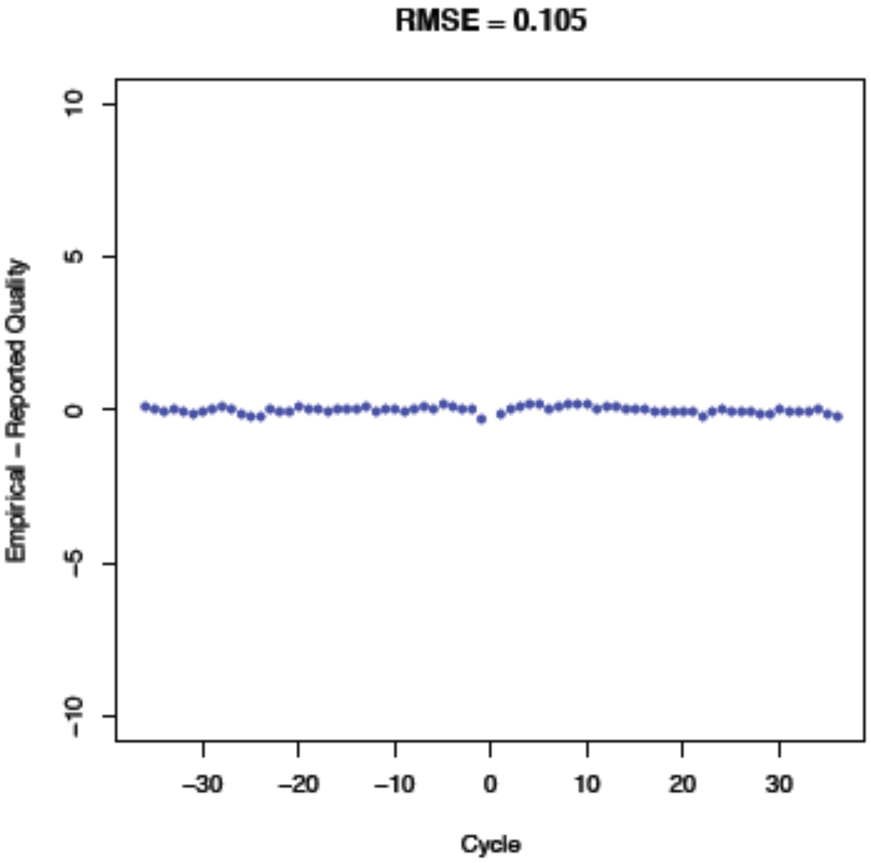
After Recalibration

Residual error by machine cycle

SciLifeLab

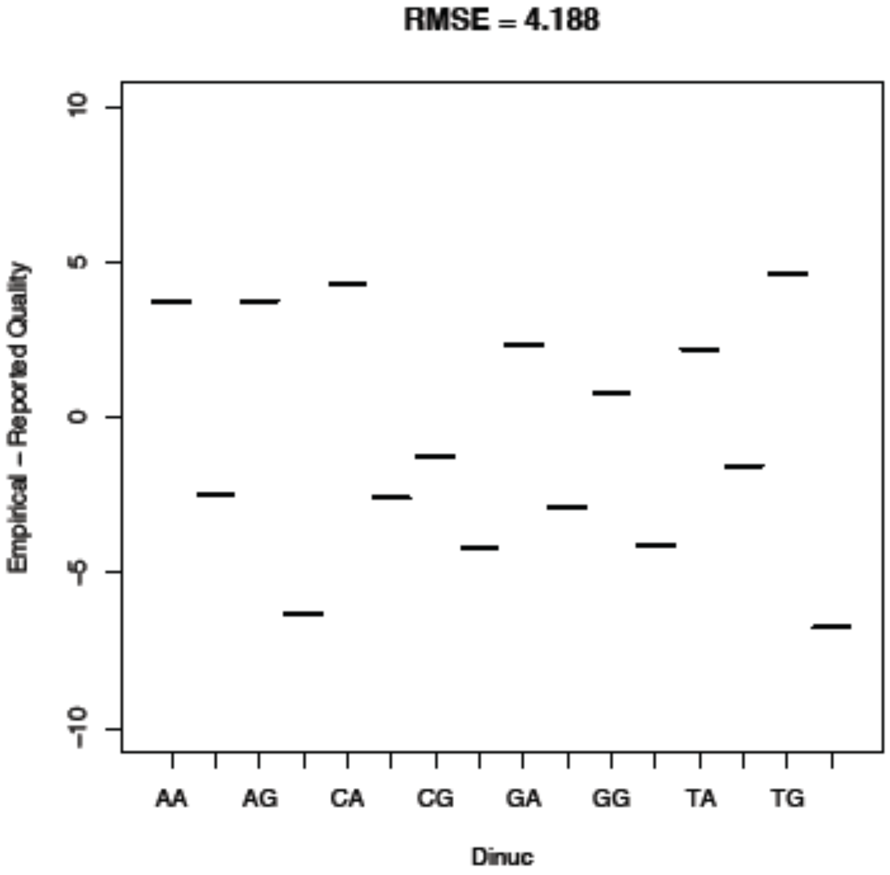


Before Recalibration

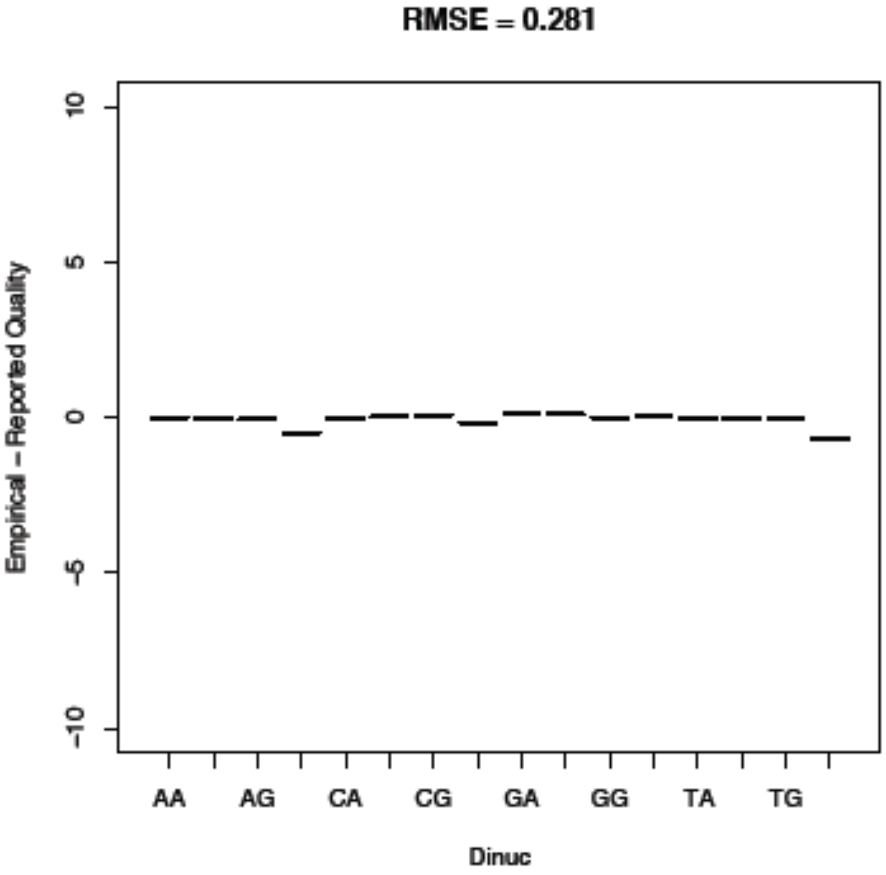


After Recalibration

Residual error by dinucleotide

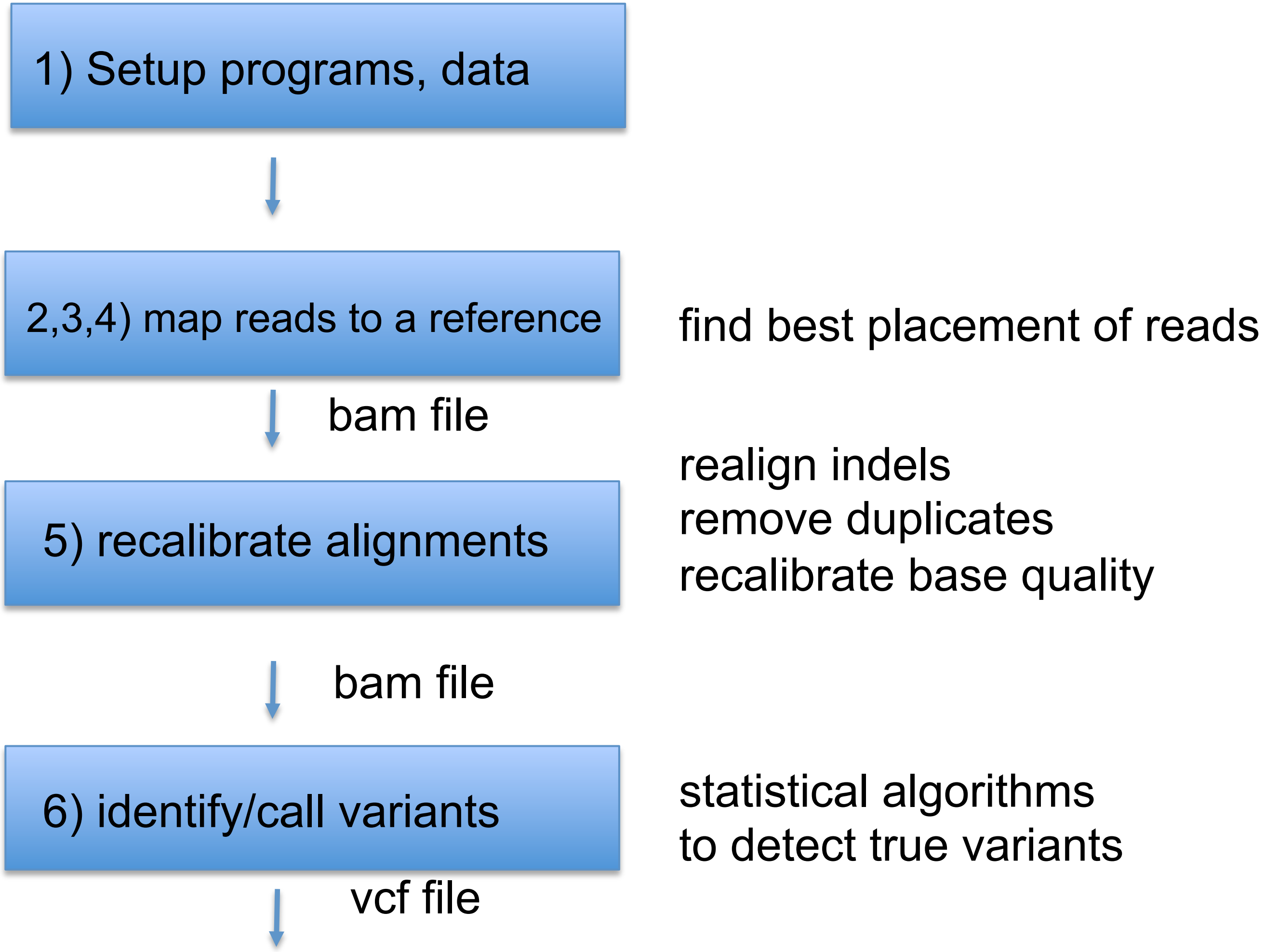


Before Recalibration



After Recalibration

Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

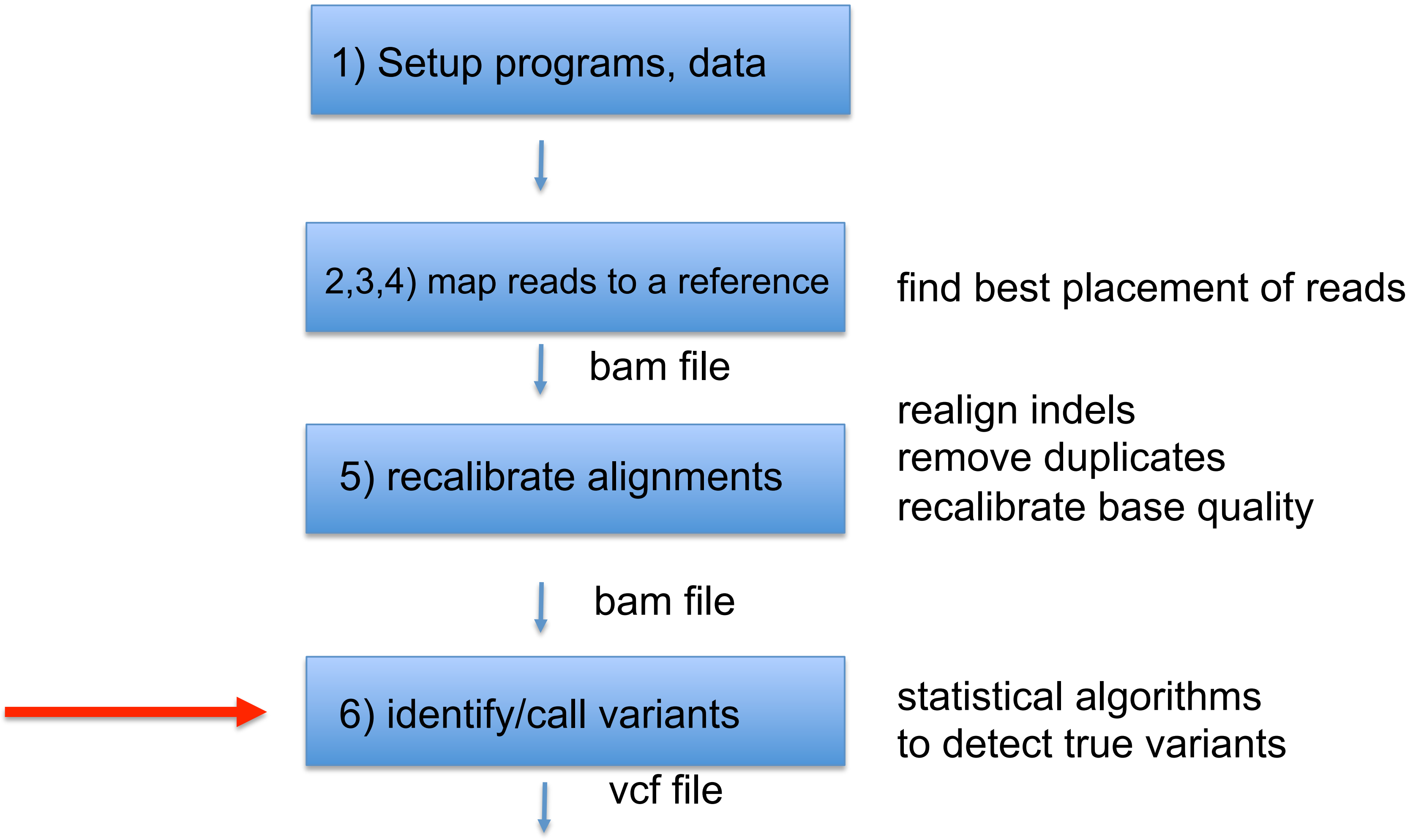
bam file

6) identify/call variants

statistical algorithms
to detect true variants

vcf file

Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

bam file

6) identify/call variants

statistical algorithms
to detect true variants

vcf file

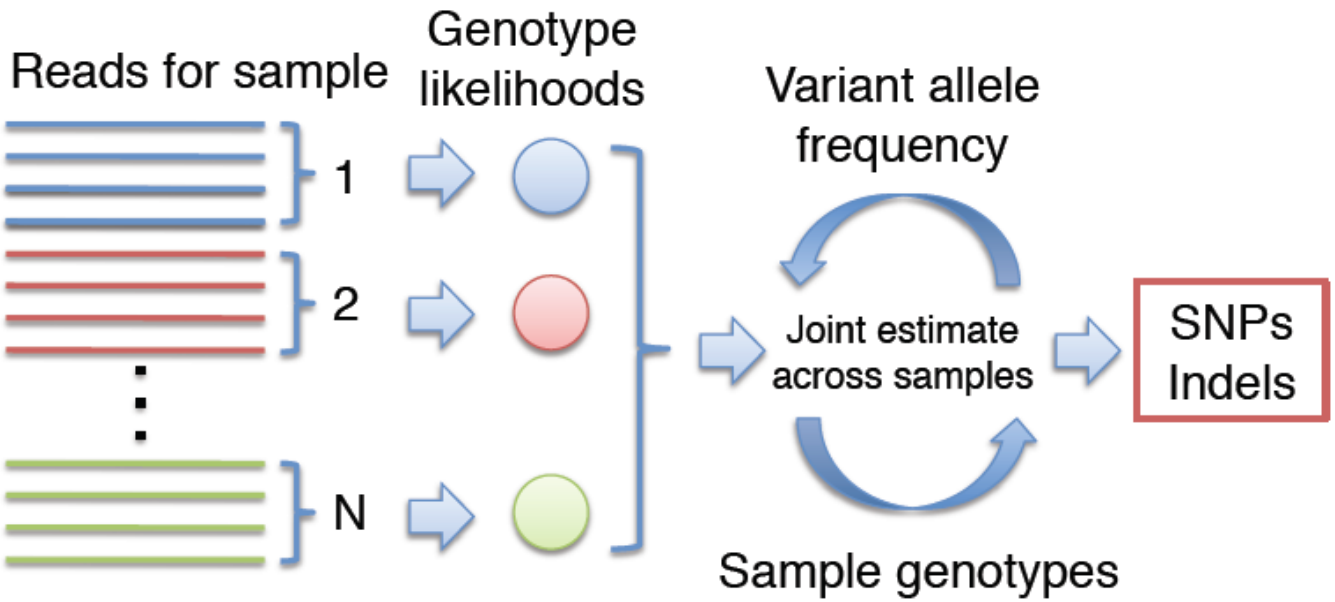
simple pileup methods

SciLifeLab

```
acacagatagacatagacatagacagatgag
acacagatagacatagacatagacagatgag
acacacatagacatagacatagacagatgag
acacagatagacatagacatagacagatgag
acacagatagacatatacatagacagatgag
acacagatagacatatacatagacagatgag
acacagatagacatatacatagacagttgag
acacagatagacatagacatagacagatgag
acacagatagacatatacatagacagatgag
acacagatagacatagacatagacagatgag
```

Reference genome assembly

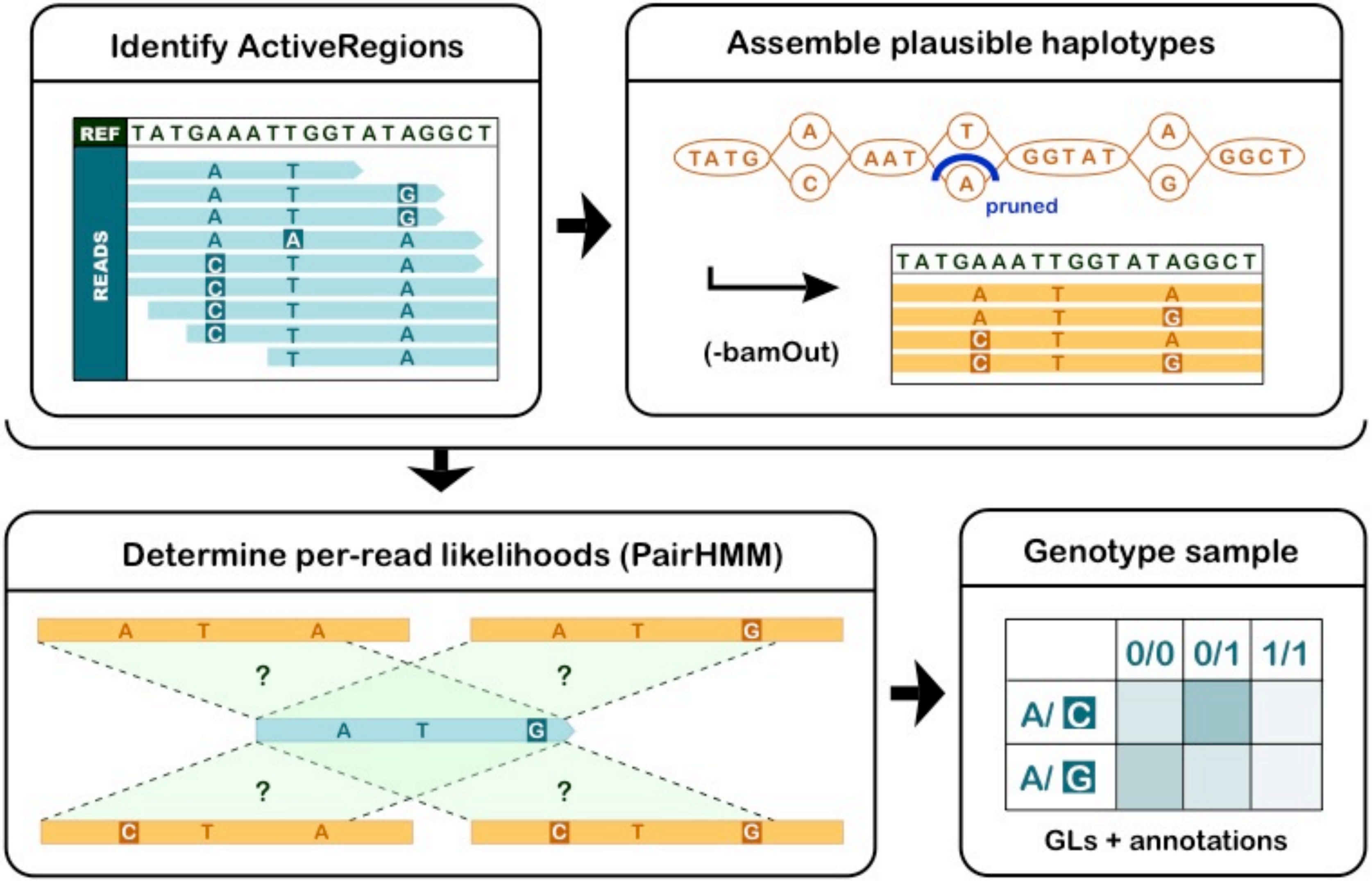
Baysian population-based calling



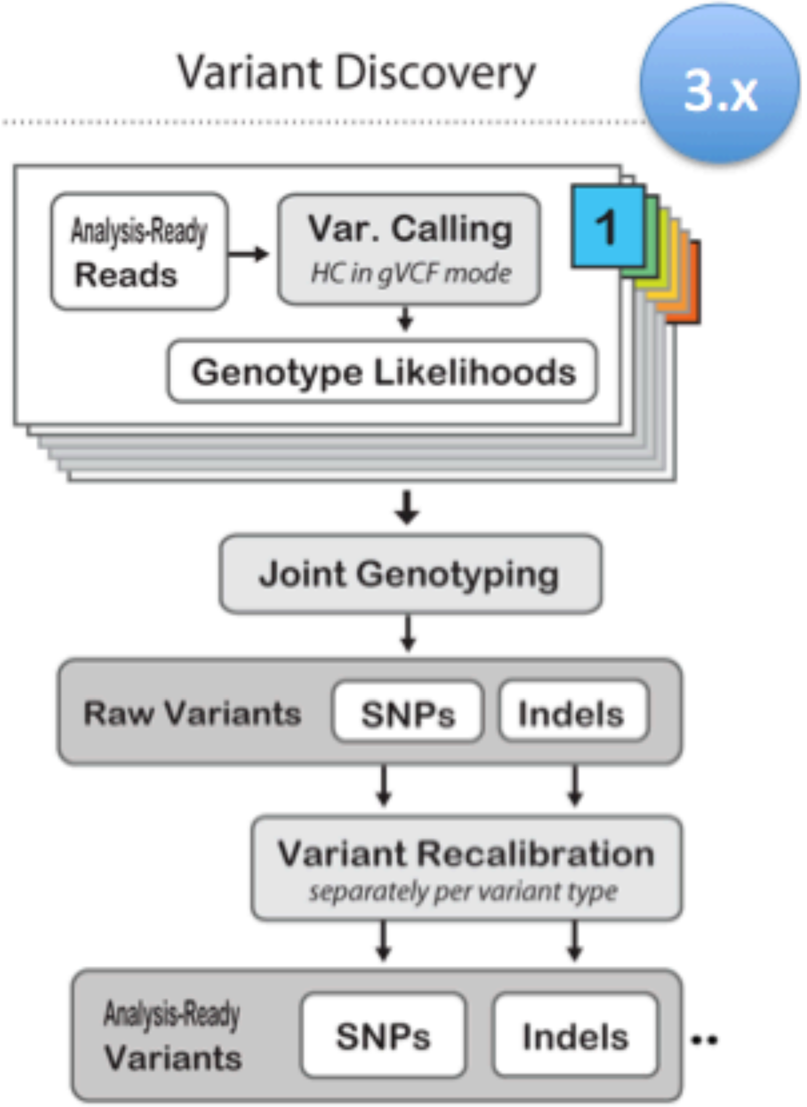
Simultaneous estimation of:

- Allele frequency (AF) spectrum: Pr{AF = i I D}
- The prob. that a variant exists: Pr{AF > 0 I D}
- Assignment of genotypes to each sample

GATK haplotype caller



GATK best practice for cohorts



VCF format

```
##fileformat=VCFv4.0 ##fileDate=20090805
##source=myImputationProgramV3.1
##reference=1000GenomesPilot-NCBI36
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51
1/1:43:5:.,.
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|
1:2:0:18,2 2/2:35:4
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20 1234567 microsat1 GTCT G,GTACT 50 PASS NS=3;DP=9;AA=G GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3
```

VCF format

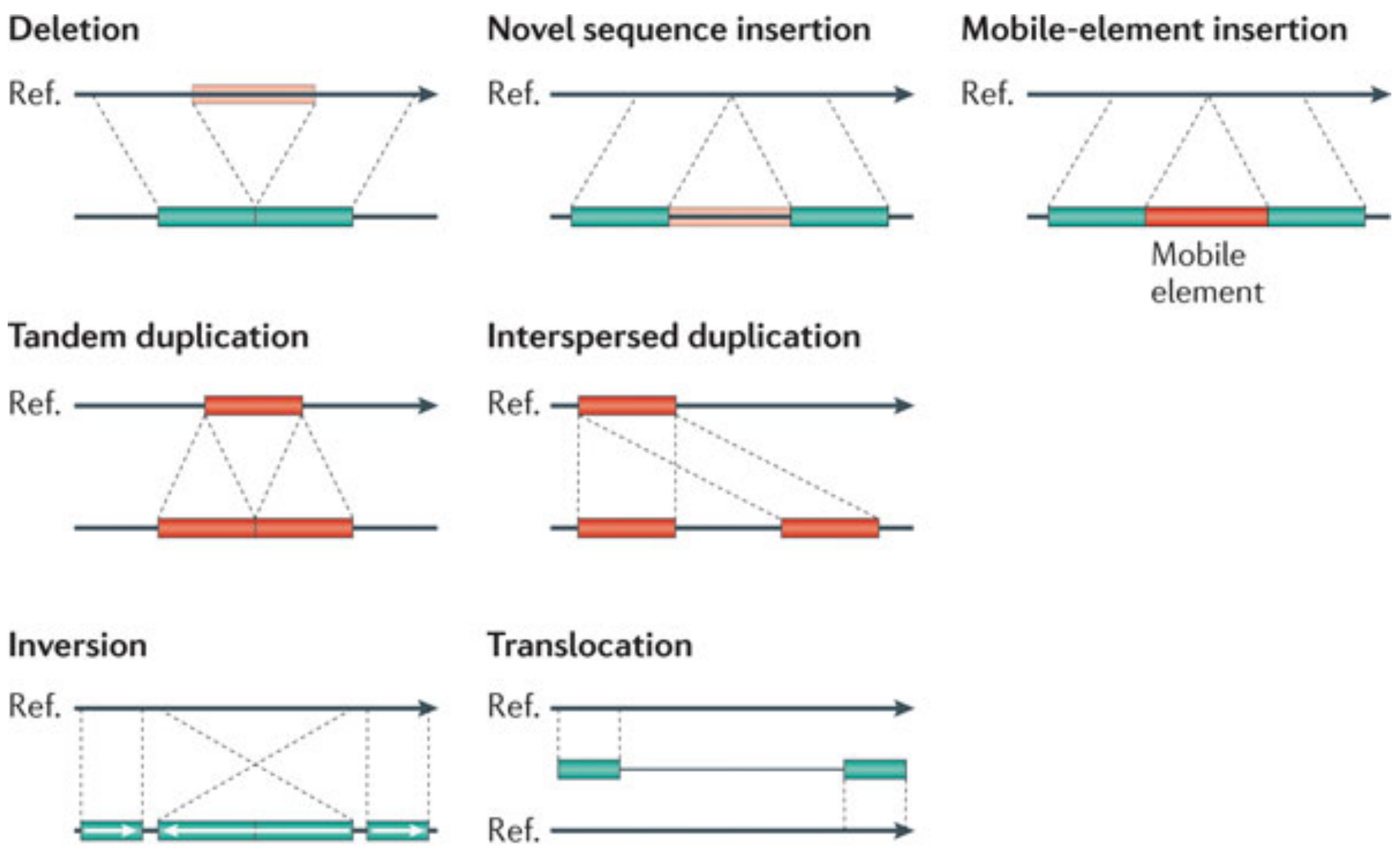
```
##fileformat=VCFv4.0 ##fileDate=20090805
##source=myImputationProgramV3.1
##reference=1000GenomesPilot-NCBI36
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth"> ##INFO=<ID=AF,Number=.,Type=Float,De
Frequency"> ##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership"> ##FILTER=<ID=q10,Description="Qu
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype"> ##FORMAT=<ID=GQ,Number=1,Type=Integer,
Quality"> ##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51 1/
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20 1234567 microsat1 GTCT G,GTACT 50 PASS NS=3;DP=9;AA=G GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3
```

gVCF format

```
##fileformat=VCFv4.0 ##fileDate=20090805
##source=myImputationProgramV3.1
##reference=1000GenomesPilot-NCBI36
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership"> ##FILTER=<ID=q10,Description="Quality
below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
##GVCFBlock=minGQ=0(inclusive),maxGQ=5(exclusive)
##GVCFBlock=minGQ=20(inclusive),maxGQ=60(exclusive)
##GVCFBlock=minGQ=5(inclusive),maxGQ=20(exclusive)
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA00001 NA00002 NA00003
20 14370 rs6054257 G A 29 PASS NS=3;DP=14;AF=0.5;DB;H2 GT:GQ:DP:HQ 0|0:48:1:51,51 1|0:48:8:51,51
1/1:43:5:.,.
20 17330 . T A 3 q10 NS=3;DP=11;AF=0.017 GT:GQ:DP:HQ 0|0:49:3:58,50 0|1:3:5:65,3 0/0:41:3
20 1110696 rs6040355 A G,T 67 PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ 1|2:21:6:23,27 2|1:2:0:18,2
2/2:35:4
20 1230237 . T . 47 PASS NS=3;DP=13;AA=T GT:GQ:DP:HQ 0|0:54:7:56,60 0|0:48:4:51,51 0/0:61:2
20 1234567 microsat1 GTCT G,GTACT 50 PASS NS=3;DP=9;AA=G GT:GQ:DP 0/1:35:4 0/2:17:2 1/1:40:3
```

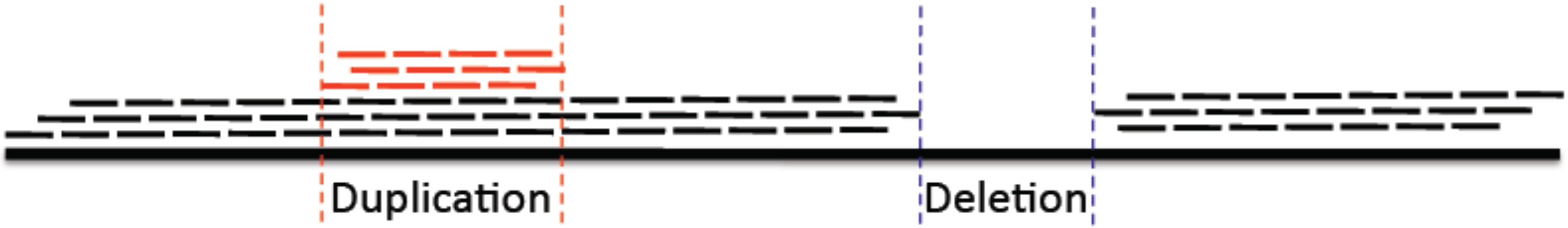
Discovery of structural variants

SciLifeLab



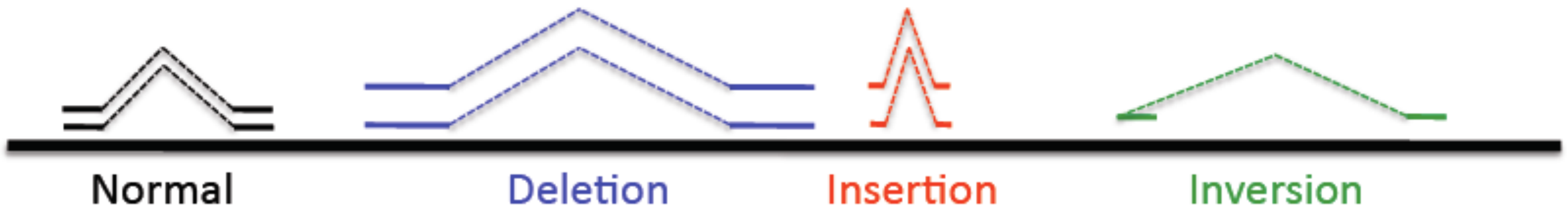
1) Read depth analysis

- Depth of coverage can be used to estimate copy number
- Samples may exhibit variation in depth indicative of polymorphic copy number variants
- How many copies of a duplication in the reference?
- How similar are the copies?
- Difficult to distinguish homozygotes and heterozygotes.



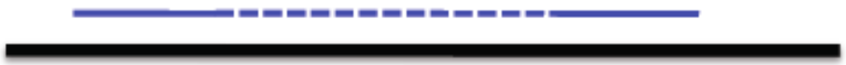
2) Paired end analysis

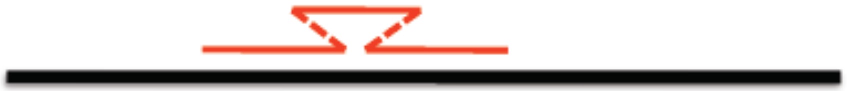
- Paired ends have a fixed length between them
- Genomic rearrangements cause them to vary
  - Deletion: reads will map too far apart
  - Insertion: reads will map too close
  - Inversion: reads in wrong orientation
- more reliable with long pairs

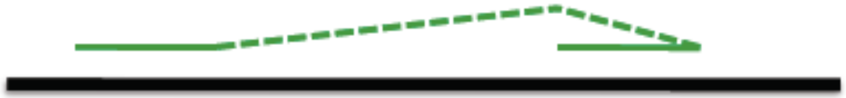


3) Split-read alignments

- Base-level breakpoint resolution
- Only works with long reads
  - – short reads have many spurious splits
- Caveat: breakpoints may be duplicated
  - – reads won't split if single alignment is good

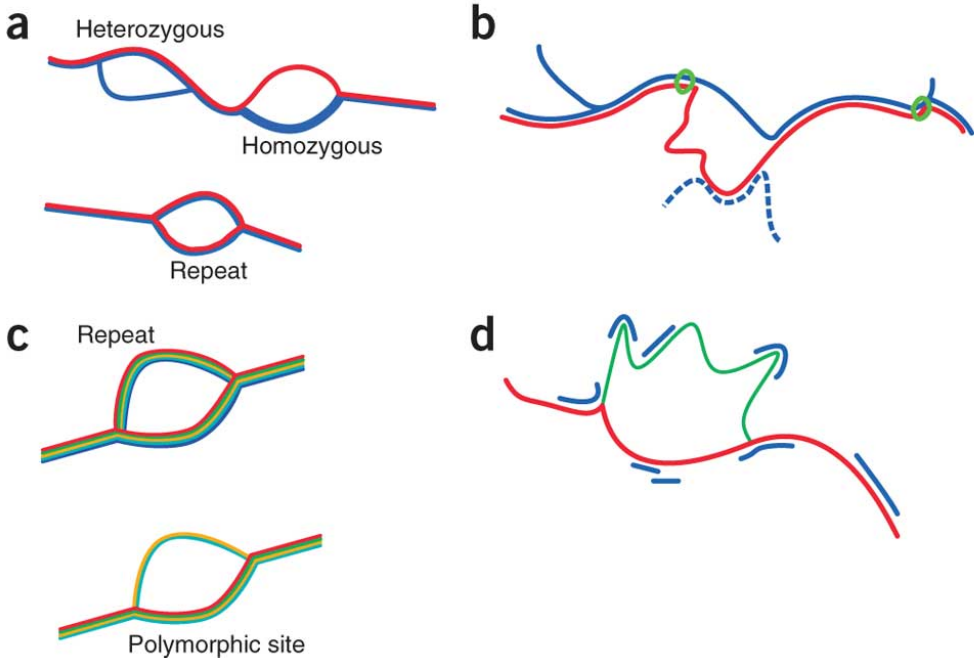

Deletion


Insertion


Inversion

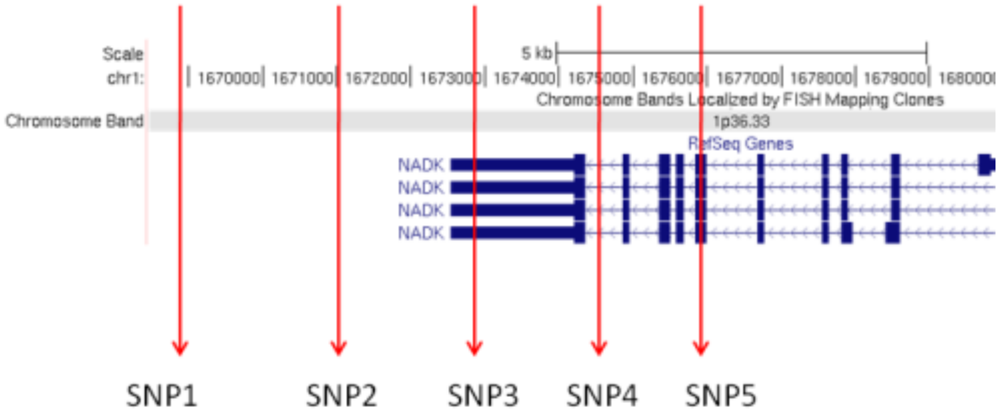
4) *De novo* assembly to identify structural variants

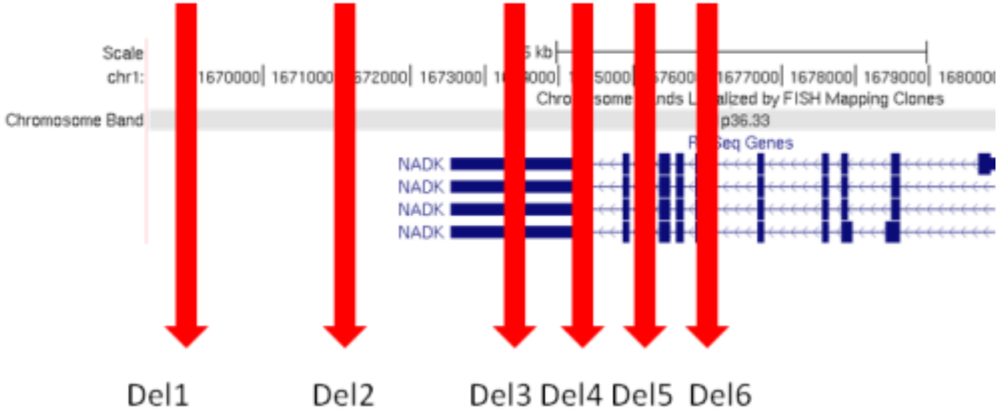
- Assemble contigs
- Align to reference
- Look for insertions, deletions, rearrangements



Annotation of variants

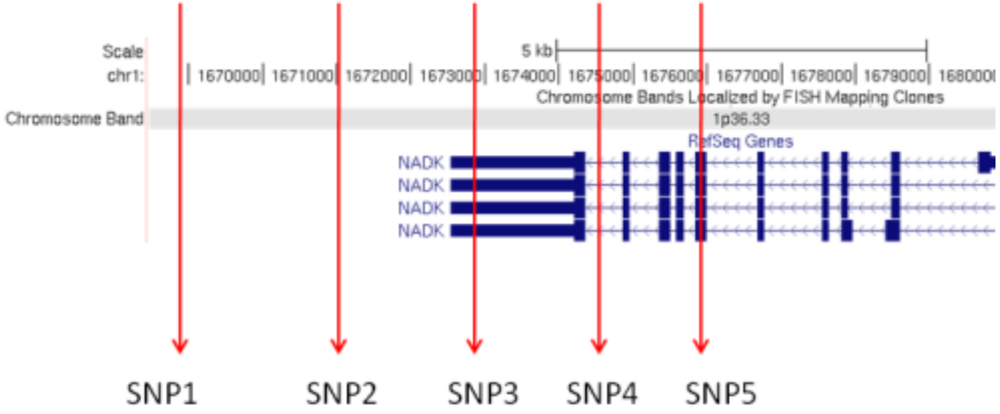
By comparing with existing annotation for the reference genome it is possible to gain information about localization and expected effect

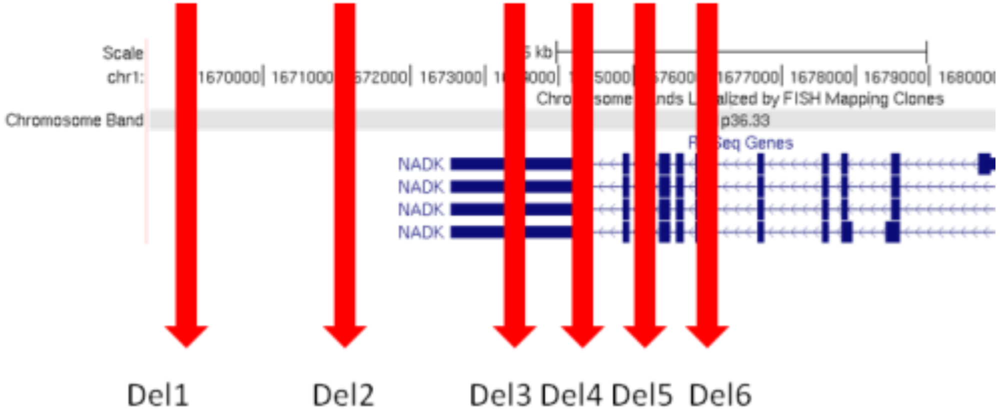




Annotation of variants

By comparing with existing annotation for the reference genome it is possible to gain information about localization and expected effect



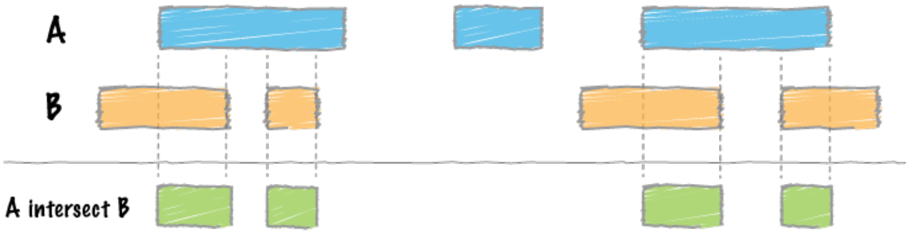


Most commonly used tools are Annovar and SNPEff

Downstream analysis

Software for file handling

- BEDTools – enables genome arithmetics – (`module add BEDTools`)



- Vcftools – for manipulations of vcf-files - (`module add vcftools`)
- bcftools – for manipulations of bcf-files - (`module add bcftools`)
- bamtools – for manipulations of bam-files - (`module add bamtools`)

Annotations to compare with can be extracted from e.g the UCSC browser, ensemble database, etc

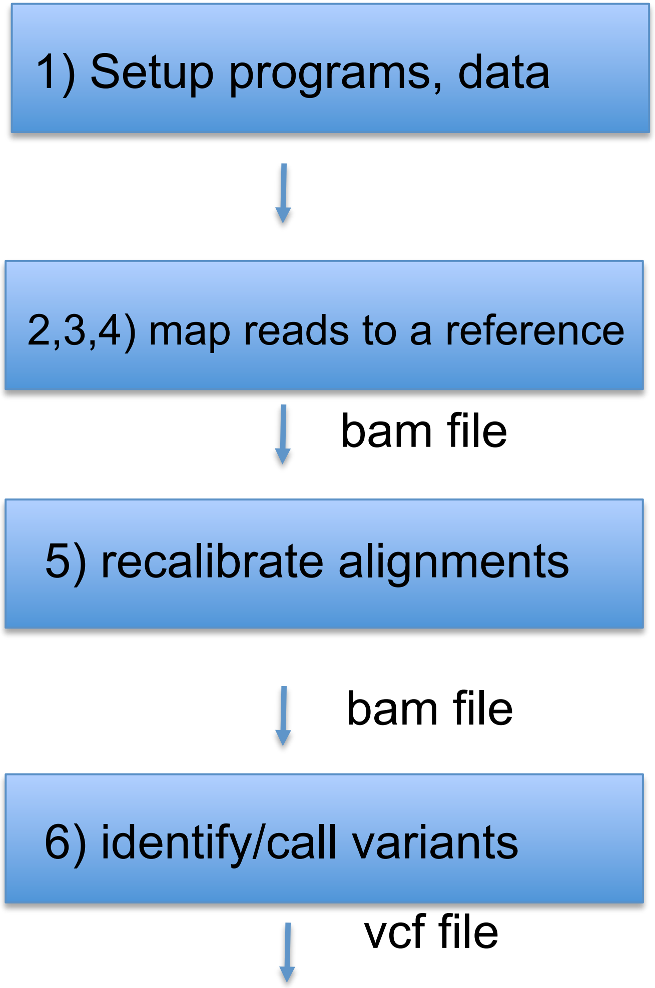
Scripting yourself with .. Perl / python / bash / awk

Overview of excercise

1. Access to data and programs
2. Mapping (BWA)
3. Merging alignments (BWA)
4. Creating BAM files (Picard)
5. Processing files (GATK)
6. Variant calling and filtering (GATK)
7. Viewing data (IGV)

X. Optional extras

Steps in resequencing analysis



1) Setup programs, data

2,3,4) map reads to a reference

find best placement of reads

bam file

5) recalibrate alignments

realign indels
remove duplicates
recalibrate base quality

bam file

6) identify/call variants

statistical algorithms
to detect true variants

vcf file

1) Access to data and programs

- Data comes from 1000 genomes pilot project
  - 81 low coverage (2-4 x) Illumina WGS samples
  - 63 Illumina exomes
  - 15 low coverage 454
- ~ 1 Mb from chromosome 17

- Tasks: align a couple of samples to reference, process, reacalibration, call and filter variants

1) Access to data and programs

- BWA and samtools modules can be loaded:

```
module load bioinfo-tools
module load bwa
module load samtools
```

- picard and GATK are are set of java programs:

```
/bubo/sw/apps/bioinfo/GATK/3.4-46/
/bubo/sw/apps/bioinfo/picard/1.69/kalkyl/
```

2) Align each paired end separately

```
bwa aln <ref> <fq1> > <sai1>
bwa aln <ref> <fq2> > <sai2>
```

<ref>	= reference sequence
<fq1>	= fastq reads seq 1 of pair
<fq2>	= fastq reads seq 2 of pair
<sai1>	= alignment of seq 1 of pair
<sai2>	= alignment of seq 2 of pair

3) Merging alignments

Combine alignments from paired ends into a SAM file

```
bwa sampe <ref> <sai1> <sai2> <fq1> <fq2> > align.sam
```

<ref> = reference sequence

<sai1> = alignment of seq 1 of pair

<sai2> = alignment of seq 2 of pair

<fq1> = fastq reads seq 1 of pair

<fq2> = fastq reads seq 2 of pair

4) Creating and editing BAM files

- Create .bam and add read groups (picard)

```
java -Xmx2g –jar /path/AddOrReplaceReadGroups.jar
INPUT=<sam file>
OUTPUT=<bam file>
... more options
```

- index new BAM file (picard)

```
java -Xmx2g –jar /path/BuildBamIndex.jar
INPUT=<bam file>
... more options
```

5) Processing files

- mark problematic indels (GATK)

```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-I <bam file>
-R <ref file>
-T RealignerTargetCreator
-o <intervals file>
```

- realign around indels (GATK)

```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-I <bam file>
-R <ref file>
-T IndelRealigner
-o <realigned bam>
-targetIntervals <intervals file>
```

5) Processing files

- mark duplicates (picard)

```
java -Xmx2g -jar /path/MarkDuplicates.jar
INPUT=<input bam>
OUTPUT=<marked bam>
METRICS_FILE=<metrics file>
```

- quality recalibration - compute covariation (GATK)

```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-T BaseRecalibrator
-I <input bam>
-R <ref file>
-knownSites <vcf file>
-recalFile <calibration table>
```

- Second step quality recalibration - compute covariation (GATK)

```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-T PrintReads -BQSR <calibration table>
-I <input bam>
-R <ref file>
-o <recalibrated bam>
```

6) Variant calling

- HaplotypeCaller (GATK)

```
java -Xmx2g
-jar /path/GenomeAnalysisTK.jar
-T HaplotypeCaller
-R <ref file>
-I <bam>
-o <filename.g.vcf>
-emitRefConfidence GVCF
-variant_index_type LINEAR
-variant_index_parameter 128000
```

Processing files

NEXT:

repeat steps 2-5 for at least another sample!

6) Genotyping gvcf

- Assigning genotypes based on joint analysis of multiple samples

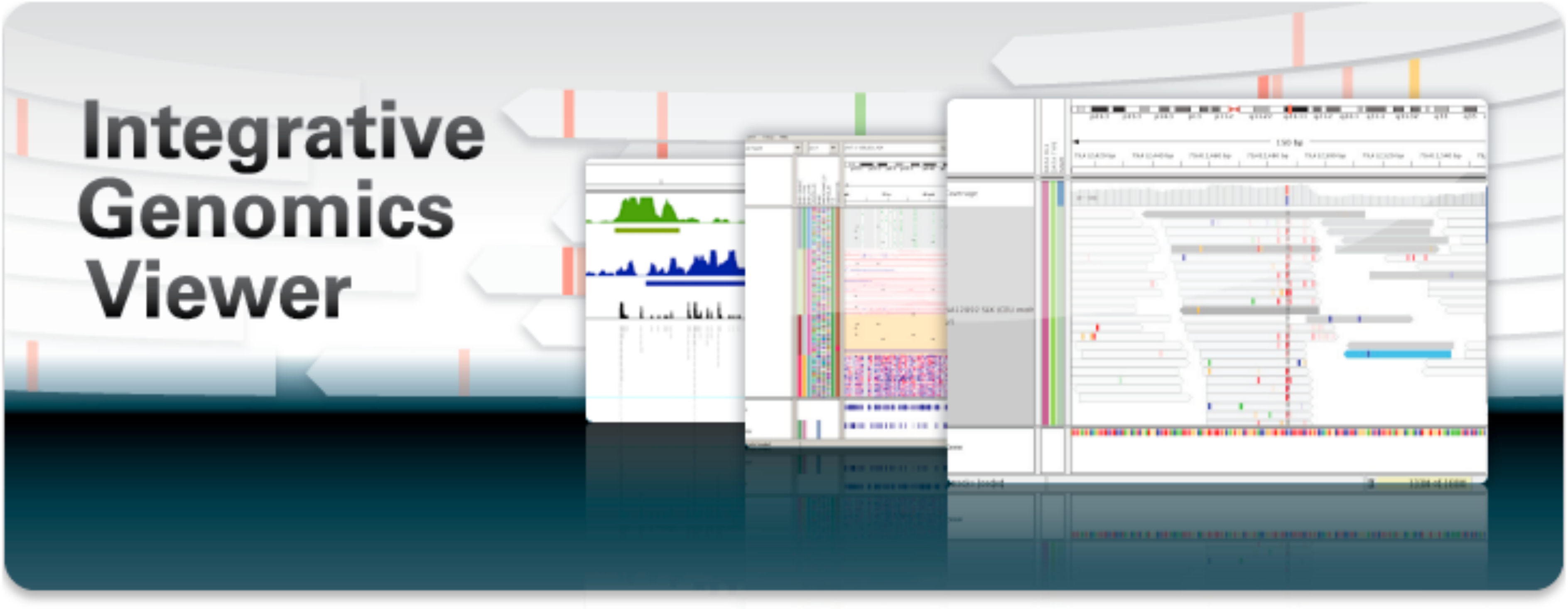
```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-T GenotypeGVCFs
-R <ref file>
--variant <sample1>.g.vcf
--variant <sample2>.g.vcf
...
-o <output vcf>
```

6) Filtering variants

- variant filtering

```
java -Xmx2g -jar /path/GenomeAnalysisTK.jar
-T VariantFiltration
-R <reference>
-V <input vcf>
-o <output vcf>
--filterExpression "QD<2.0" --filterName QDfilter
--filterExpression "MQ<40.0" --filterName MQfilter
--filterExpression "FS>60.0" --filterName FSfilter
--filterExpression "HaplotypeScore>13.0" --filterName HSfilter
--filterExpression "MQRankSum<-12.5" --filterName MQRSfilter
--filterExpression "ReadPosRankSum<-8.0" --filterName RPRSfilter
```

7) Viewing data with IGV



http://www.broadinstitute.org/igv/

X) Extra

Extra 1: View data in UCSC-browser

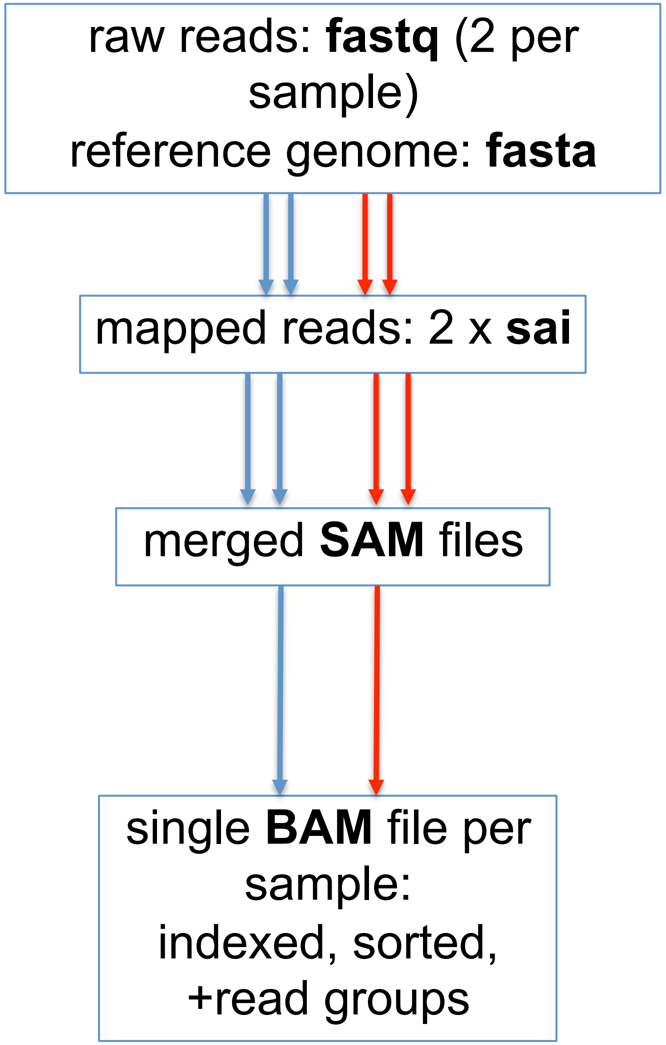
Extra 2: Select subset with BEDTools

Extra 3: Annotate variants with annovar

Extra 4: Make a script to run pipeline

pipeline (1)

2. Mapping
   - `bwa index`
   - `samtools faidx`
   - `bwa aln`
3. Merging alignments
   - `bwa sampe`
4. Creating BAM files
   - `picard AddOrReplaceReadGroups`
   - `picard BuildBamIndex`



raw reads: **fastq** (2 per sample)
reference genome: **fasta**

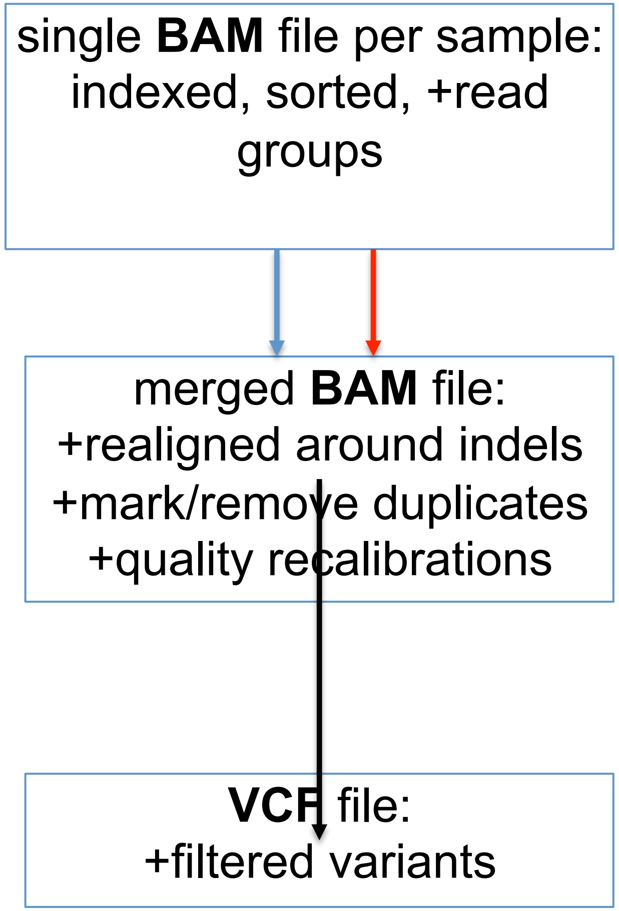
mapped reads: 2 x **sai**

merged **SAM** files

single **BAM** file per sample:
indexed, sorted,
+read groups

pipeline (2)

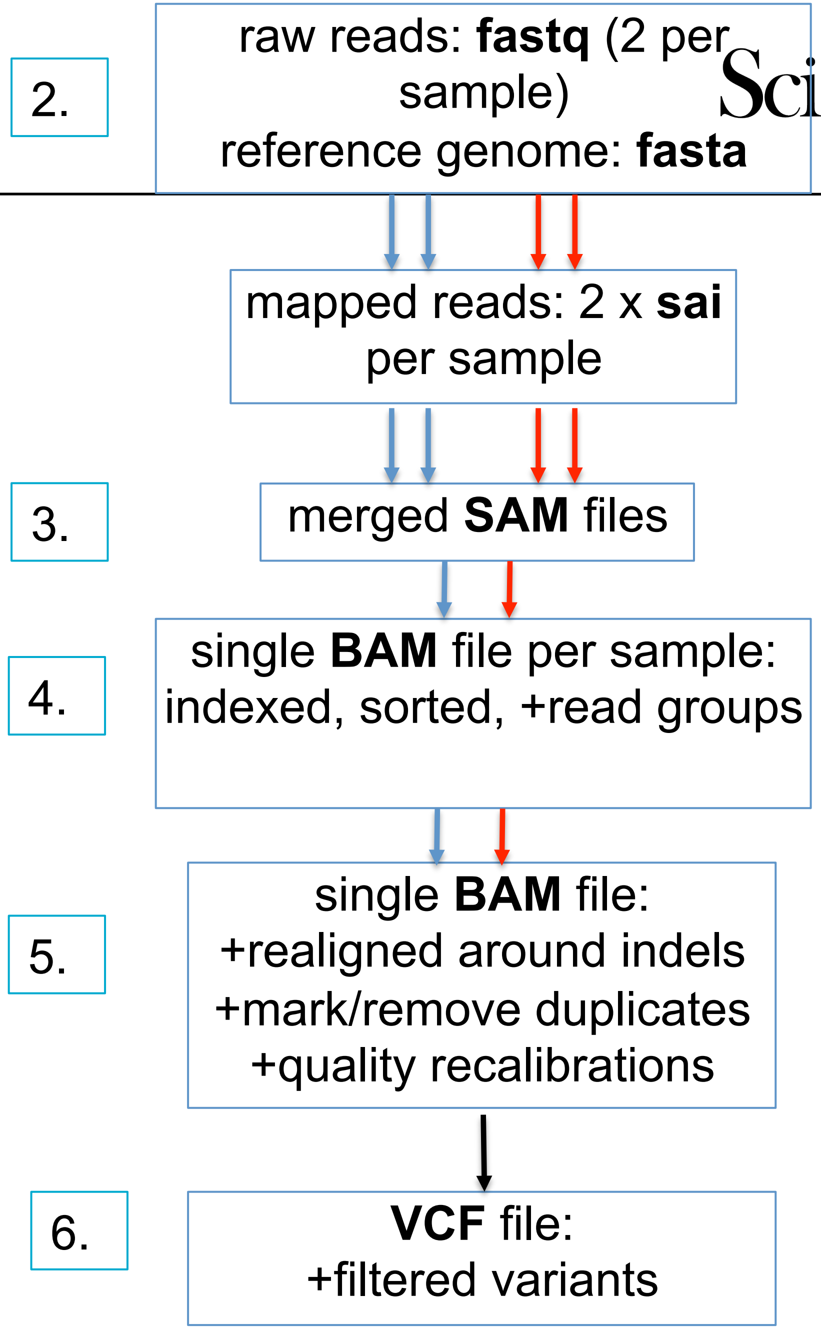
5. Processing files (GATK)
   - GATK RealignerTargetCreator
   - GATK IndelRealigner
   - picard MarkDuplicates
   - GATK CountCovariates
   - picard MergeSamFiles
6. Variant calling and filtering (GATK)
   - GATK UnifiedGenotyper
   - GATK VariantFiltration
7. Viewing data (IGV)



single **BAM** file per sample: indexed, sorted, +read groups

merged **BAM** file: +realigned around indels +mark/remove duplicates +quality recalibrations

VCF file: +filtered variants



2.

raw reads: **fastq** (2 per sample)
reference genome: **fasta**

mapping

mapped reads: 2 x **sai** per sample

3.

merged **SAM** files

4.

single **BAM** file per sample:
indexed, sorted, +read groups

processing

5.

single **BAM** file:
+realigned around indels
+mark/remove duplicates
+quality recalibrations

variant calling

6.

VCF file:
+filtered variants

Naming conventions

Initial file name according to information about the content

NA06984.ILLUMINA.low_coverage.17q

For each step of the pipeline, create a new file

NA06984.ILLUMINA.low_coverage.17q.merge.bam

NA06984.ILLUMINA.low_coverage.17q.merge.realign.bam

NA06984.ILLUMINA.low_coverage.17q.merge.realign.dedup.bam

NA06984.ILLUMINA.low_coverage.17q.merge.realign.dedup.recal.bam

…