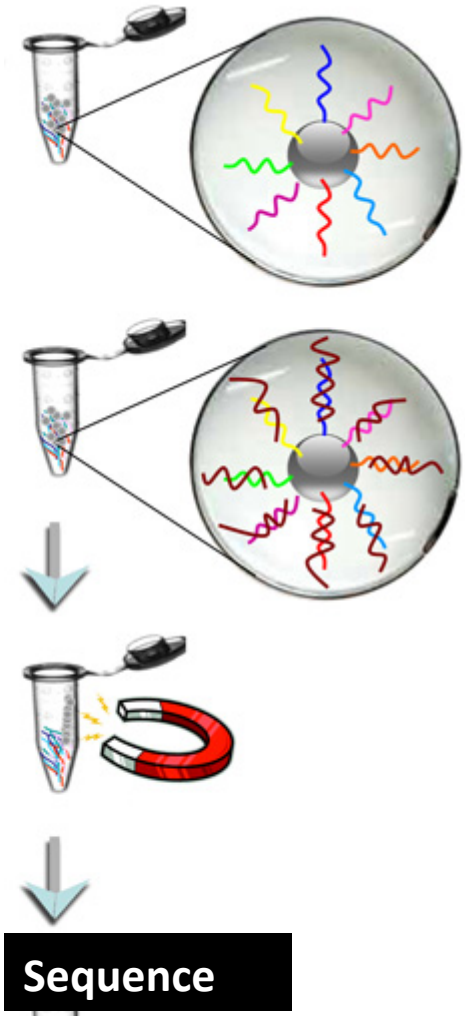
SNP and Genotype Calling

Genomics Tools Workshop #3

4 May 2015 (Happy May Fourth!)

HC Lim

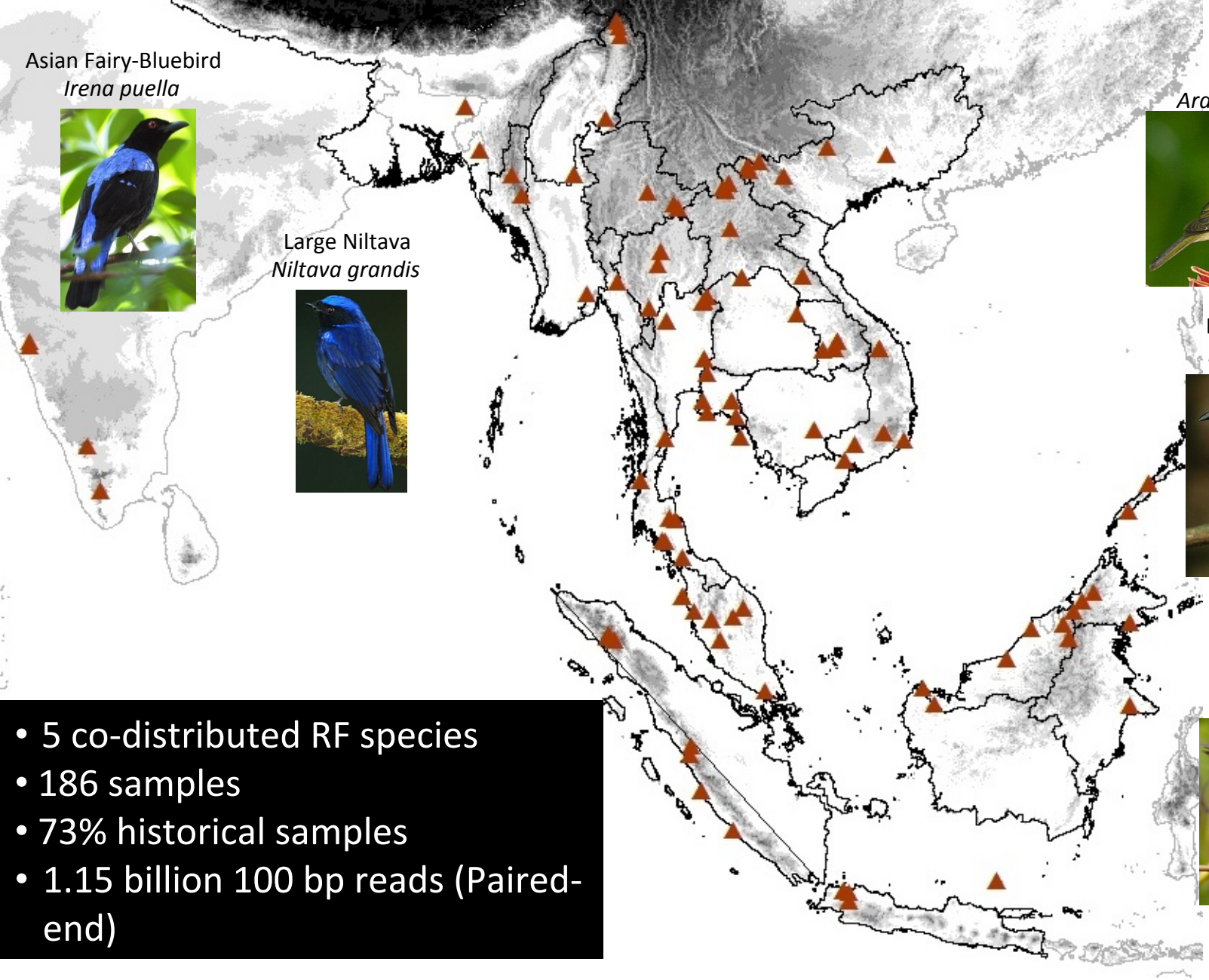
Sequence Capture



Probe set: 5060 tetrapod ultraconserved elements (UCE's) Faircloth, McCormack, Crawford et al. (2012)

DNA from toe pads

Ancient DNA Lab
Smithsonian Conservation Biology Institute



Asian Fairy-Bluebird
Irena puella

Large Niltava
Niltava grandis

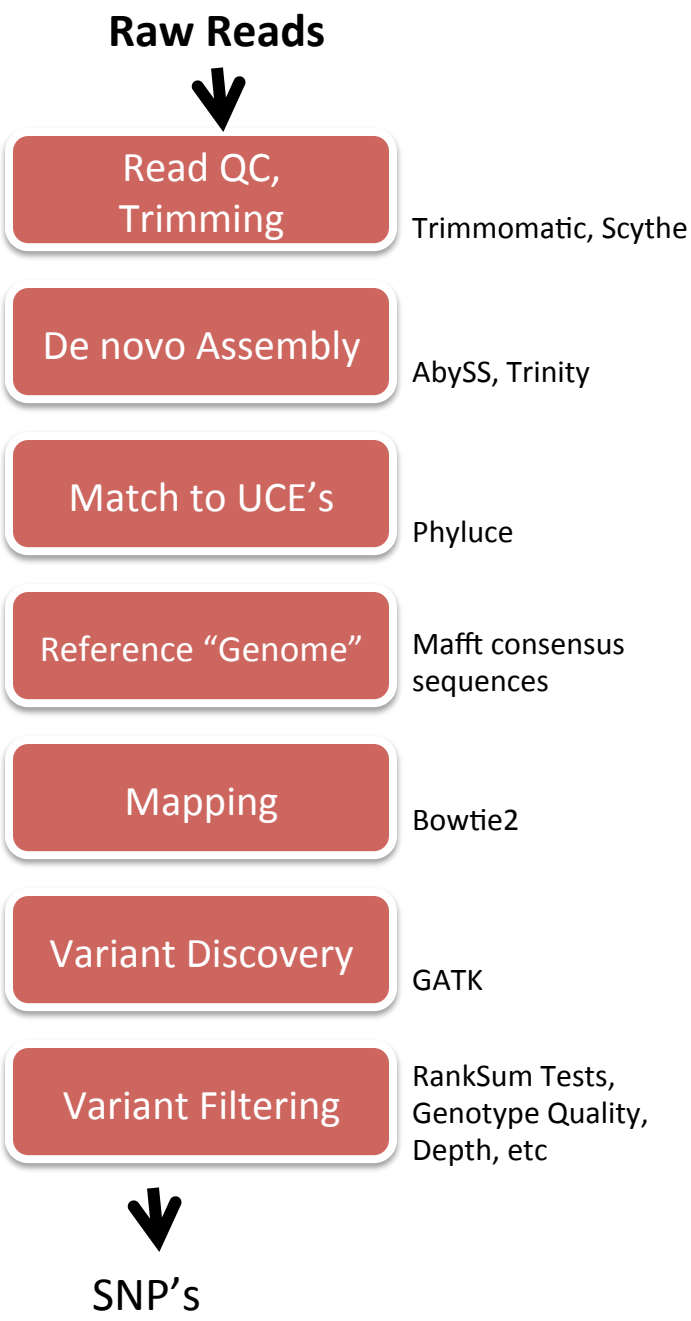
Little Spiderhunter
Arachnothera longirostra

Black-headed Bulbul
Pycnonotus atriceps

Grey-throated Babbler
Stachyris nigriceps

- 5 co-distributed RF species
- 186 samples
- 73% historical samples
- 1.15 billion 100 bp reads (Paired-end)

Bird images: orientalbirdimages, uniprot


Raw Reads
Read QC, Trimming
Trimmomatic, Scythe
De novo Assembly
AbySS, Trinity
Match to UCE's
Phyluce
Reference "Genome"
Mafft consensus sequences
Mapping
Bowtie2
Variant Discovery
GATK
Variant Filtering
RankSum Tests, Genotype Quality, Depth, etc
SNP's

'Automation Addiction': Are Pilots Forgetting How to Fly?

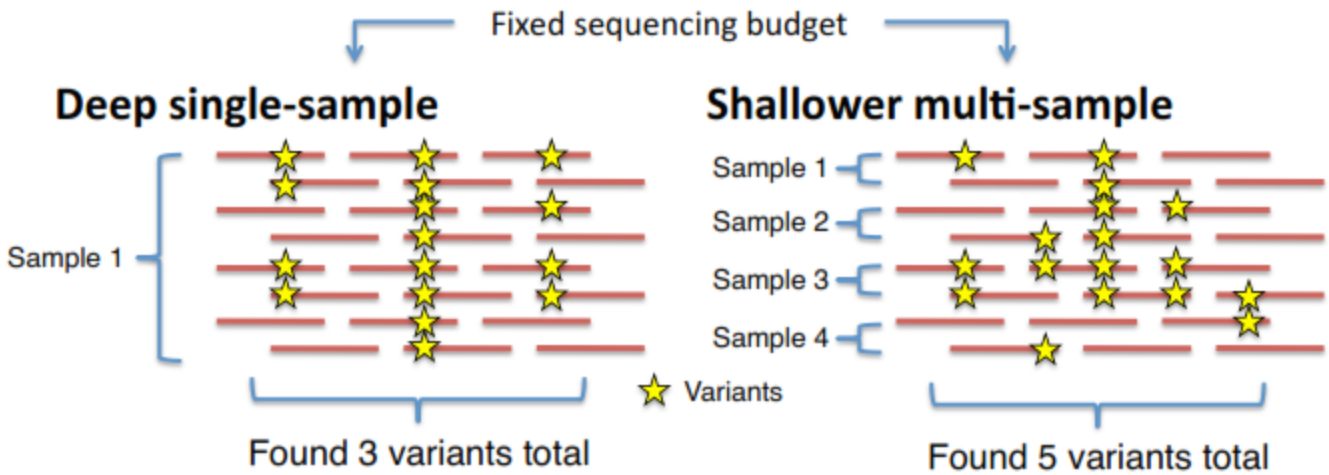
Aug. 31, 2011

By KEVIN DOLAK via **WORLD NEWS**

AUTO START: ON | OFF

"There's no such thing as a data analysis pipeline" John McCutheon (U Montana)

Sampling scheme: Deep vs Wide



Shallow, single sample (Functional, architecture questions)

Deep Single-Sample Sequencing

- High sensitivity, most or all variants discovered in a sample
- More accurate genotyping (fewer false homozygotes)
- Fewer total variants discovered

Important for individual-based analyses (e.g. Structure), not so much for population-based analyses (e.g., theta, Tajima's D, Site Frequency Spectrum)

OPEN ACCESS Freely available online PLOS | ONE

SNP Calling, Genotype Calling, and Sample Allele Frequency Estimation from New-Generation Sequencing Data

Rasmus Nielsen[1,2,3*], Thorfinn Korneliussen[3], Anders Albrechtsen[3], Yingrui Li[1], Jun Wang[1,3*]

1 BGI-Shenzhen, Shenzhen, China, 2 Departments of Integrative Biology and Statistics, University of California, Berkeley, California, United States of America, 3 Department of Biology, University of Copenhagen, Copenhagen, Denmark

Abstract

We present a statistical framework for estimation and application of sample allele frequency spectra from New-Generation Sequencing (NGS) data. In this method, we first estimate the allele frequency spectrum using maximum likelihood. In contrast to previous methods, the likelihood function is calculated using a dynamic programming algorithm and numerically optimized using analytical derivatives. We then use a Bayesian method for estimating the sample allele frequency in a single site, and show how the method can be used for genotype calling and SNP calling. We also show how the method can be extended to various other cases including cases with deviations from Hardy-Weinberg equilibrium. We evaluate the statistical properties of the methods using simulations and by application to a real data set.

Citation: Nielsen R, Korneliussen T, Albrechtsen A, Li Y, Wang J (2012) SNP Calling, Genotype Calling, and Sample Allele Frequency Estimation from New-Generation Sequencing Data. PLoS ONE 7(7): e37558. doi:10.1371/journal.pone.0037558

Editor: Philip Awadalla, University of Montreal, Canada

http://popgen.dk/wiki/index.php/ANGSD

Log in

Page | Discussion | Read | View source | View history | Search | Go | Search

Pages

- ANGSD overview
- Installation
- Quick Start/Testdata
- Input data
- Filters
- snpFilters

Population genetics

- SFS Estimation
- Thetas,Tajima,Neutrality test
- HWE and inbreeding
- 2d SFS Estimation

ANGSD: Analysis of next generation Sequencing Data

Can read CRAM now (using htslib)!!

http://www.biomedcentral.com/1471-2105/15/356/abstract

Latest version is 0.700, see Change_log for changes, and download it here.

ANGSD

ANGSD is a software for analyzing next generation sequencing data. The software can handle a number of different input types from mapped reads to imputed genotype probabilities. Most methods take genotype uncertainty into account instead of basing the analysis on called genotypes. This is especially useful for low and medium depth data. The software is written in C++ and has been used on large sample sizes.

This program is not for manipulating '.bam' files, but solely a tool to perform various kinds of analysis. We recommend the excellent program SAMtools for outputting and modifying bamfiles.

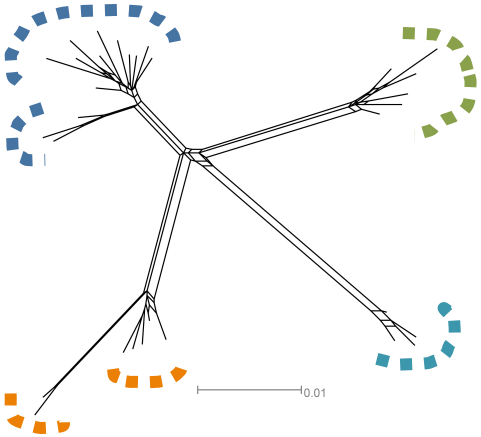
ANGSD is also on github now: https://github.com/ANGSD/angsd

We recognize that CRAM is an emerging format for storing NGS data. We have therefore begun including CRAM support by using the htslib in the latest github version. This of course requires that users also install htslib.

Good genotyping is still important

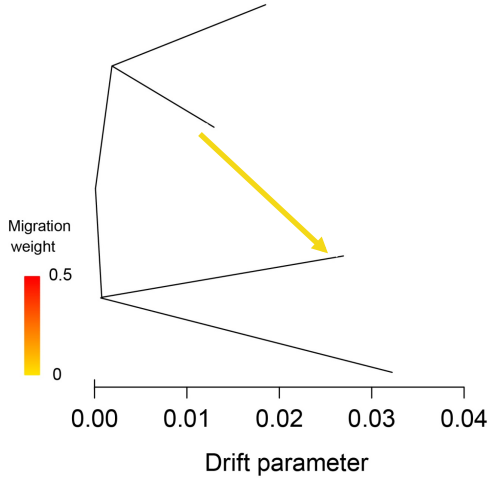
White-rumped Shama
Copsychus malabaricus

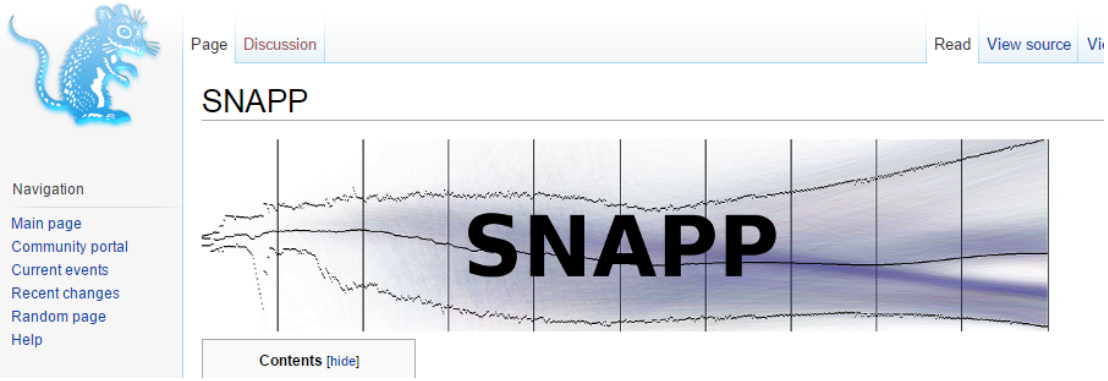
K = 2 3290 SNPs



Splits network
9154 SNP's

TreeMix





A high-level overview of NGS data processing

Base calling
(vendor tool)

Illumina 1.5/Phred 64
Illumina 1.8+/Phred33/Sanger

FASTQ file:
raw NGS reads

Alignment or
assembly

SAM/BAM file:
aligned NGS reads

Variant calling

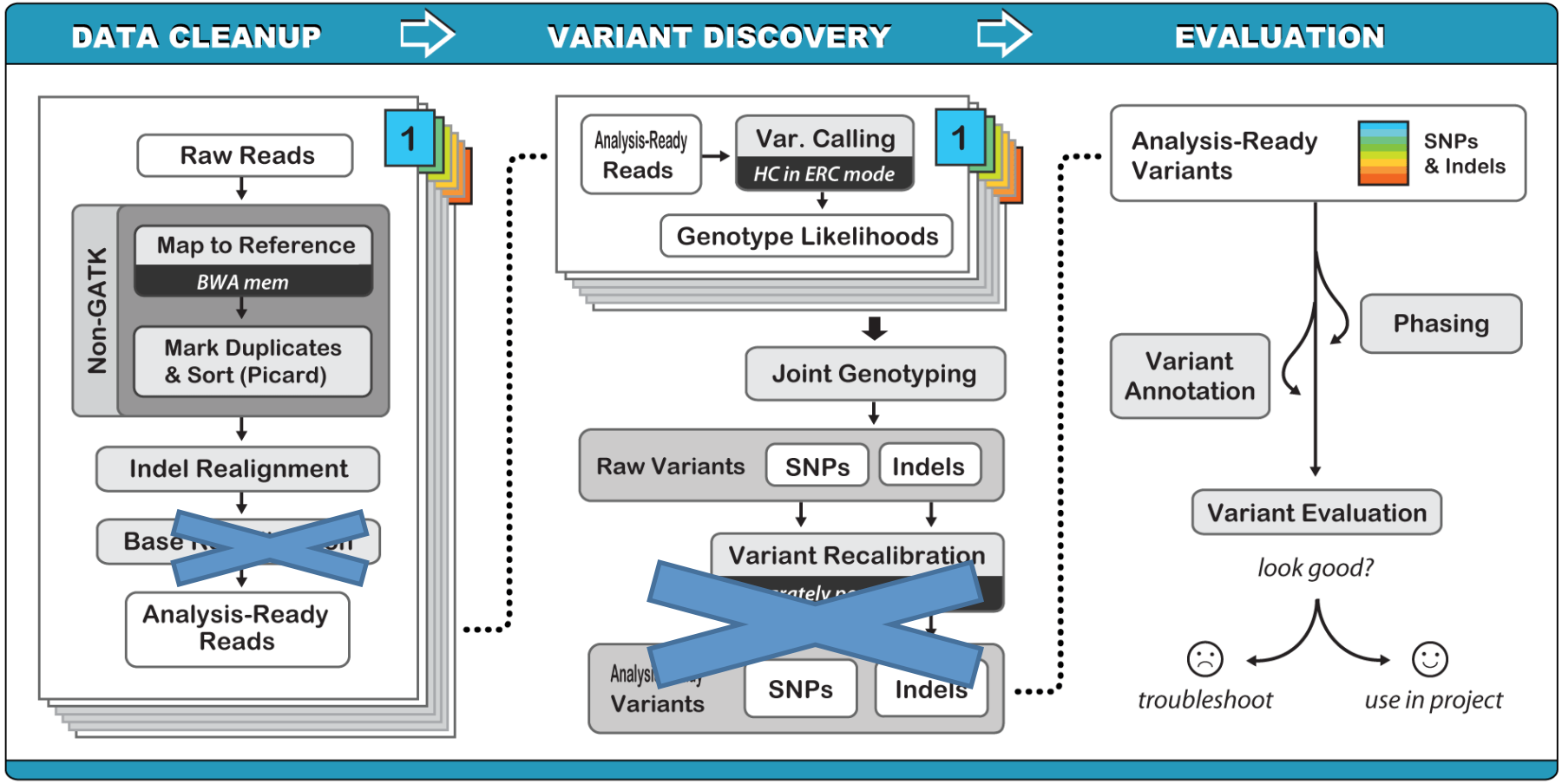
VCF file:
genomic variation

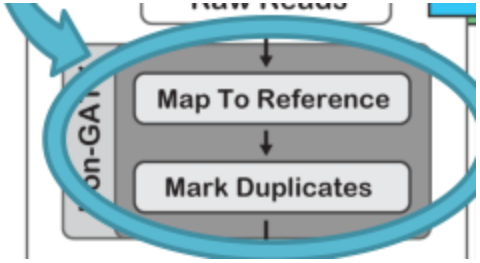
.sam file: Uncompressed text file
.bam file: Compressed and indexed file

VCF Files can be block-compressed and indexed.

GATK (Broad Institute)
Samtools (mpileup -> bcf)
FreeBayes (Erik Garrison)

GATK best practices

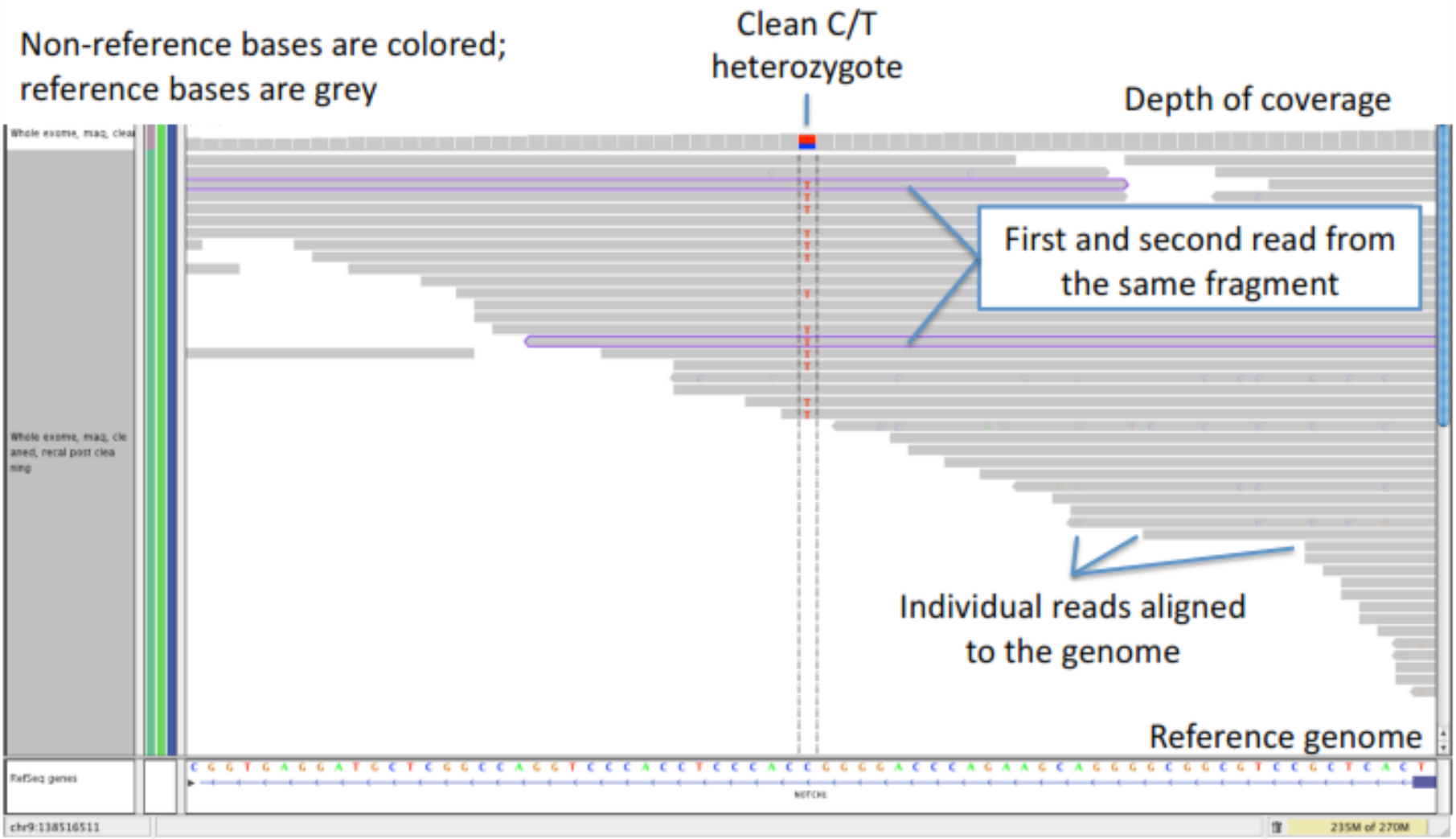




Map to Reference

- Short-read mapping: bwa or bowtie/bowtie2
  - bowtie2-build –f ref.fasta built.ref
  - bowtie2 --local --very-sensitive-local -N 1 -I 100 -X 700 -x built.ref -p 12 --phred64 --rg-id "\1" --rg SM:"\1" --rg PL:"ILLUMINA" --rg LB:"hiseq.phred64" -1 READ1.fastq.gz -2 READ2.fastq.gz –S sam 2> bow.stat

Visualization in IGV



SAM/BAM Headers & Sequences

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

```
@HD    VN:1.0  GO:none SO:coordinate
@SQ    SN:chrM        LN:16571
@SQ    SN:chr1        LN:247249719
@SQ    SN:chr2        LN:242951149
[cut for clarity]
@SQ    SN:chr9        LN:140273252
@SQ    SN:chr10       LN:135374737
@SQ    SN:chr11       LN:134452384
[cut for clarity]
@SQ    SN:chr22    LN:49691432
@SQ    SN:chrX        LN:154913754
@SQ    SN:chrY        LN:57772954
@RG    ID:20FUK.1     PL:illumina    PU:20FUKAAXX100202.1    LB:Solexa-18483 SM:NA12878    CN:BI
@RG    ID:20FUK.2     PL:illumina    PU:20FUKAAXX100202.2    LB:Solexa-18484 SM:NA12878    CN:BI
@RG    ID:20FUK.3     PL:illumina    PU:20FUKAAXX100202.3    LB:Solexa-18483 SM:NA12878    CN:BI
@RG    ID:20FUK.4     PL:illumina    PU:20FUKAAXX100202.4    LB:Solexa-18484 SM:NA12878    CN:BI
@RG    ID:20FUK.5     PL:illumina    PU:20FUKAAXX100202.5    LB:Solexa-18483 SM:NA12878    CN:BI
@RG    ID:20FUK.6     PL:illumina    PU:20FUKAAXX100202.6    LB:Solexa-18484 SM:NA12878    CN:BI
@RG    ID:20FUK.7     PL:illumina    PU:20FUKAAXX100202.7    LB:Solexa-18483 SM:NA12878    CN:BI
@RG    ID:20FUK.8     PL:illumina    PU:20FUKAAXX100202.8    LB:Solexa-18484 SM:NA12878    CN:BI
@PG    ID:BWA  VN:0.5.7      CL:tk
@PG    ID:GATK TableRecalibration     VN:1.0.2864
20FUKAAXX100202:1:1:12730:189900     163    chrM   1     60     101M   =     282    381
        GATCACAGGTCTATCACCCTATTAACCACTCACGGGAGCTCTCCATGCATTTGGTA...[more bases]
        ?BA@A>BBBBACBBAC@BBCBBCBC@BC@CAC@:BBCBBCACAACBABCBCCAB...[more quals]
        RG:Z:20FUK.1    NM:i:1  SM:i:37 AM:i:37 MD:Z:72G28     MQ:i:60 PG:Z:BWA      UQ:i:33
```

Required: Standard header

Essential: contigs of aligned reference sequence. Should be in karyotypic order.

Essential: read groups. Carries platform (PL), library (LB), and sample (SM) information. Each read is associated with a read group

Useful: Data processing tools applied to the reads

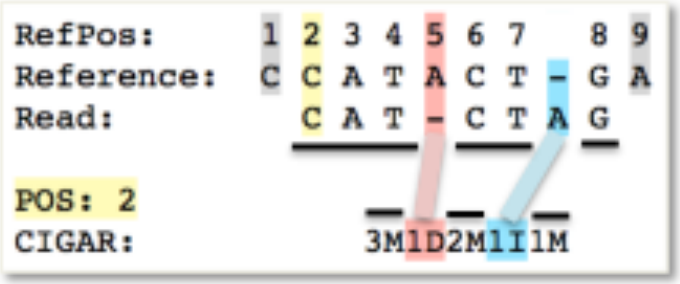
```
20FUKAAXX100202:1:1:12730:189900    163    chrM    1    60    101M    =    282    381
GATCACAGGTCTATCACCCTATTAACCACTCACGGGAGCTCTCCATGCATTTGGTA...[more bases]
?BA@A>BBBBACBBAC@BBCBBCBC@BC@CAC@:BBCBBCACAACBABCBCCAB...[more quals]
RG:Z:20FUK.1    NM:i:1  SM:i:37 AM:i:37 MD:Z:72G28    MQ:i:60 PG:Z:BWA    UQ:i:33
```

Col	Field	Type	Regexp/Range	Brief description
1	QNAME	String	`[!-?A-~]{1,255}`	Query template NAME
2	FLAG	Int	$[0,2^{16}-1]$	bitwise FLAG
3	RNAME	String	`\*|[!-()+-<>-~][!-~]*`	Reference sequence NAME
4	POS	Int	$[0,2^{31}-1]$	1-based leftmost mapping POSition
5	MAPQ	Int	$[0,2^{8}-1]$	MAPping Quality
6	CIGAR	String	`\*|([0-9]+[MIDNSHPX=])+`	CIGAR string
7	RNEXT	String	`\*|=|[!-()+-<>-~][!-~]*`	Ref. name of the mate/next read
8	PNEXT	Int	$[0,2^{31}-1]$	Position of the mate/next read
9	TLEN	Int	$[-2^{31}+1,2^{31}-1]$	observed Template LENgth
10	SEQ	String	`\*|[A-Za-z=.]+`	segment SEQuence
11	QUAL	String	`[!-~]+`	ASCII of Phred-scaled base QUALity+33

Bit	Description
0x1	template having multiple segments in sequencing
0x2	each segment properly aligned according to the aligner
0x4	segment unmapped
0x8	next segment in the template unmapped
0x10	SEQ being reverse complemented
0x20	SEQ of the next segment in the template being reversed
0x40	the first segment in the template
0x80	the last segment in the template
0x100	secondary alignment
0x200	not passing quality controls
0x400	PCR or optical duplicate
0x800	supplementary alignment

Explain SAM flags

CIGAR String



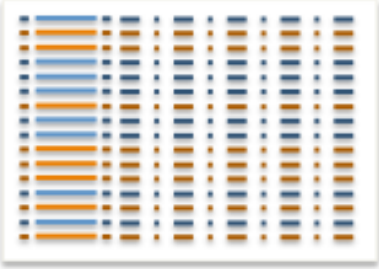
- *Phred value* = −10 * *log*10(ε)
- Examples:
  - 90% confidence (10% error rate) = Q10
  - 99% confidence (1% error rate) = Q20
  - 99.9% confidence (.1% error rate) = Q30

Sam into bam; sorting

- samtools view –uS; take sam and turn into bam
- samtools sort in.bam; sort by leftmost coordinates

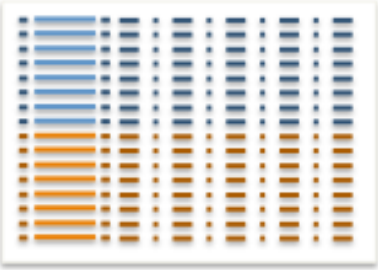
The information for this:

... is actually stored as a text file with one line per read which from far away looks like this:



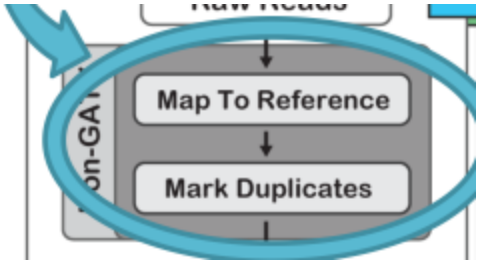
The reads are in no particular order...

... but the GATK wants reads to be sorted by starting position like this:



So we need to explicitly sort the SAM file...

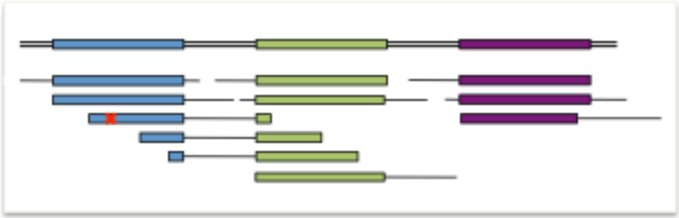
The importance of de-duplicating



✖ = sequencing error propagated in duplicates

FP variant call (bad)

After marking duplicates, the GATK will only see :



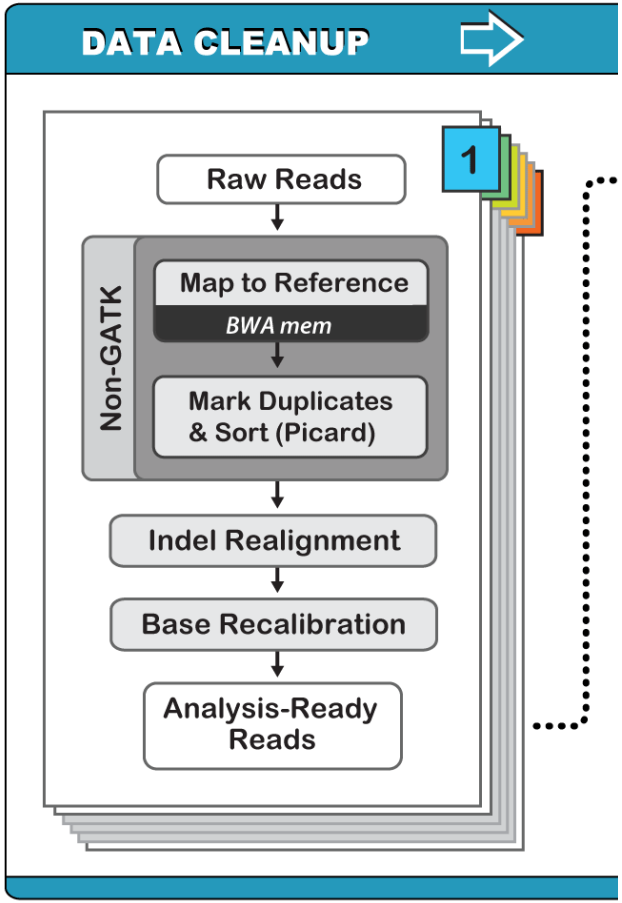
... and thus be more likely to make the right call

Same CIGAR string

Picard tools (http://broadinstitute.github.io/picard/)

java -Xmx4G –jar MarkDuplicates.jar
I=sorted.bam
M=metric
O=mrk.dup.bam

Sorted bam files; deduplicated


DATA CLEANUP
1
Raw Reads
Non-GATK
Map to Reference
BWA mem
Mark Duplicates
& Sort (Picard)
Indel Realignment
Base Recalibration
Analysis-Ready
Reads

Realignment of bam files

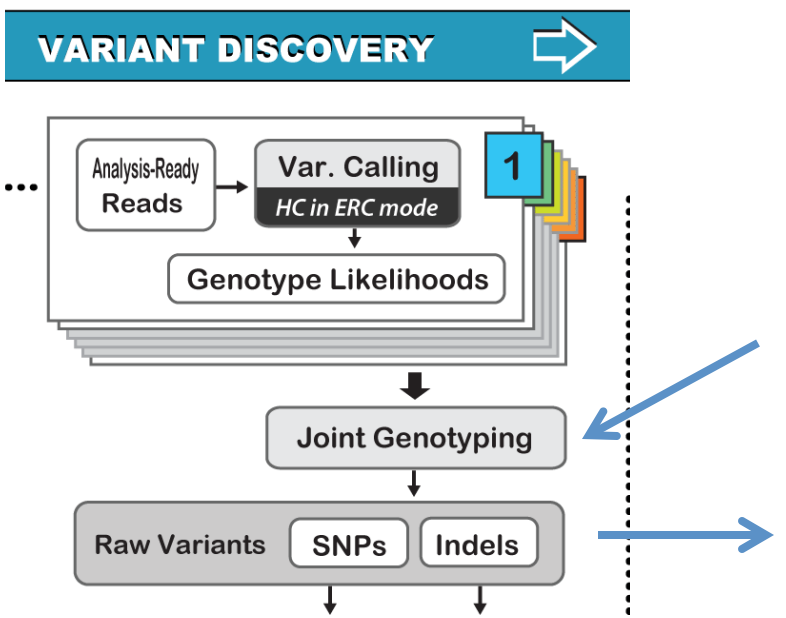
- Read mapping algorithms operate on each read independently
- Locally realign reads to minimize mismatching bases across all the reads

- 1) Determining (small) suspicious intervals which are likely in need of realignment
  - java -Xmx4g -jarGenomeAnalysisTK.jar -T RealignerTargetCreator -R ref.fasta –I mrk.dup.input.bam -o list
- 2) Running the realigner over those intervals
  - java -Xmx4g –jar GenomeAnalysisTK.jar -T IndelRealigner -R ref.fasta -I mrk.dup.input.bam -o mrk.dup.indel.bam –targetIntervals list

Haplotype Caller & gVCF

- java -Xmx4g –jar GenomeAnalysisTK.jar -T HaplotypeCaller --emitRefConfidence GVCF --variant_index_type LINEAR --variant_index_parameter 128000 -R ref.fasta –I mrk.dup.indel.bam –o gVCF

indv



-T GenotypeGVCFs

unfiltered VCF files

All samples

VCF format (headers + variants)

```
##fileformat=VCFv4.0
##FILTER=<ID=LowQual,Description="QUAL < 50.0">
##FORMAT=<ID=AD,Number=.,Type=Integer,Description="Allelic depths for the ref and alt alleles in the order listed">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth (only filtered reads used for calling)">
##FORMAT=<ID=GQ,Number=1,Type=Float,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=PL,Number=3,Type=Float,Description="Normalized, Phred-scaled likelihoods for AA,AB,BB genotypes where A=ref and B=alt; not applicable if site is not biallelic">
##INFO=<ID=AC,Number=.,Type=Integer,Description="Allele count in genotypes, for each ALT allele, in the same order as listed">
##INFO=<ID=AF,Number=.,Type=Float,Description="Allele Frequency, for each ALT allele, in the same order as listed">
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotypes">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP Membership">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=DS,Number=0,Type=Flag,Description="Were any of the samples downsampled?">
##INFO=<ID=Dels,Number=1,Type=Float,Description="Fraction of Reads Containing Spanning Deletions">
##INFO=<ID=HRun,Number=1,Type=Integer,Description="Largest Contiguous Homopolymer Run of Variant Allele In Either Direction">
##INFO=<ID=HaplotypeScore,Number=1,Type=Float,Description="Consistency of the site with two (and only two) segregating haplotypes">
##INFO=<ID=MQ,Number=1,Type=Float,Description="RMS Mapping Quality">
##INFO=<ID=MQ0,Number=1,Type=Integer,Description="Total Mapping Quality Zero Reads">
##INFO=<ID=QD,Number=1,Type=Float,Description="Variant Confidence/Quality by Depth">
##INFO=<ID=SB,Number=1,Type=Float,Description="Strand Bias">
##INFO=<ID=VQSLOD,Number=1,Type=Float,Description="log10-scaled probability of variant being true under the trained gaussian mixture model">
##UnifiedGenotyperV2="analysis_type=UnifiedGenotyperV2 input_file=[TEXT CLIPPED FOR CLARITY]"
#CHROM  POS ID      REF ALT QUAL    FILTER  INFO    FORMAT  NA12878
chr1    873762  .       T   G   5231.78 PASS
AC=1;AF=0.50;AN=2;DP=315;Dels=0.00;HRun=2;HaplotypeScore=15.11;MQ=91.05;MQ0=15;QD=16.61;SB=-1533.02;VQSLOD=-1.5473
GT:AD:DP:GQ:PL   0/1:173,141:282:99:255,0,255
```

INFO

FORMAT

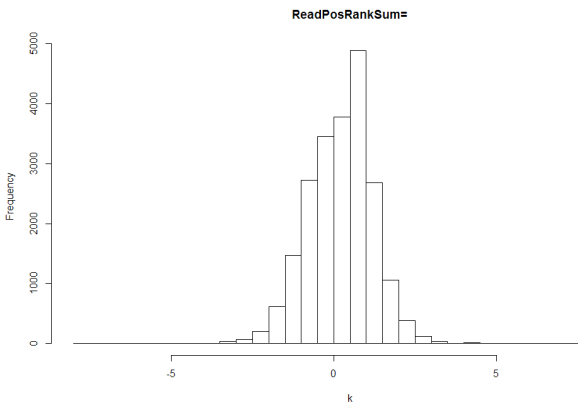
One line = One variant

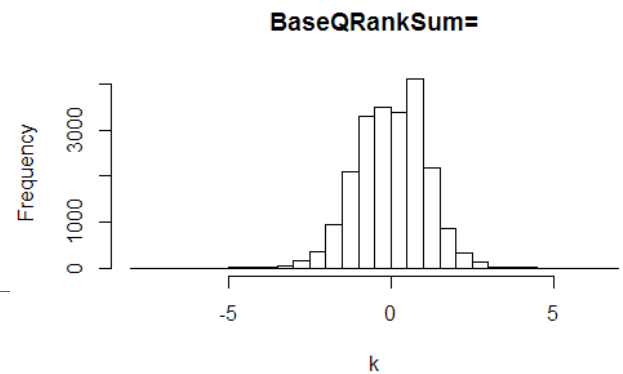
More samples = more columns to the right

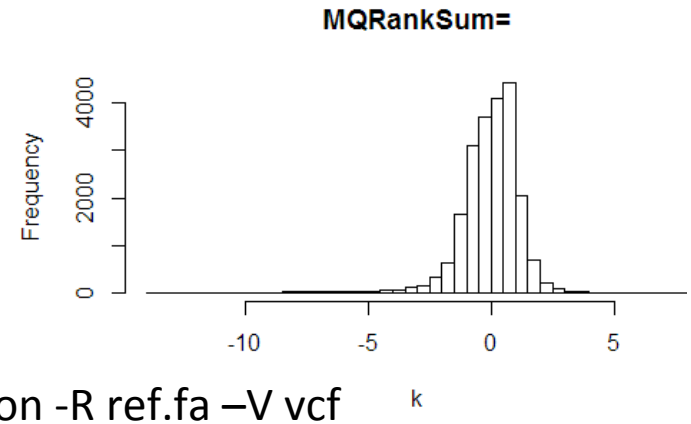
```
[HEADER LINES]
#CHROM  POS ID      REF ALT QUAL    FILTER  INFO        FORMAT        NA12878
chr1    873762  .       T   G   5231.78 PASS    [ANNOTATIONS] GT:AD:DP:GQ:PL  0/1:173,141:282:99:255,0,255
chr1    877664  rs3828047   A   G   3931.66 PASS    [ANNOTATIONS] GT:AD:DP:GQ:PL  1/1:0,105:94:99:255,255,0
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
```

- ANNOTATIONS (under INFO) show quality of the variant calls
- Filtering can be done at both the variant level and/or individual level

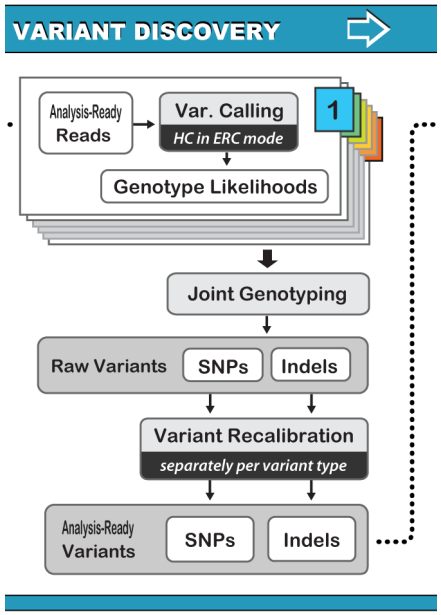
Variant-level filtering








```
java -Xmx4g –jar GenomeAnalysisTK.jar -T VariantFiltration -R ref.fa –V vcf
--filterName "filter0"
--filterExpression
"ReadPosRankSum < -1.96 || ReadPosRankSum > 1.96
|| BaseQRankSum < -1.96 || BaseQRankSum > 1.96
|| MQRankSum < -1.96 || MQRankSum > 1.96"
```



Strand Bias & Mapping Quality

```
--filterExpression "FS > 20.0 || MQ < 30.0"
```

Individual-level filtering (DP ≥ 8 & GQ ≥ 20)

Carson *et al. BMC Bioinformatics* 2014, **15**:125
http://www.biomedcentral.com/1471-2105/15/125

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

Effective filtering strategies to improve data quality from population-based whole exome sequencing studies

Andrew R Carson[1†], Erin N Smith[1†], Hiroko Matsui[1], Sigrid K Brækkan[2,3], Kristen Jepsen[1], John-Bjarne Hansen[2,3] and Kelly A Frazer[1,4,5,6*]

```
[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
```