GATK Best Practices for Variant Discovery

Variant Filtering by Variant Quality Score Recalibration

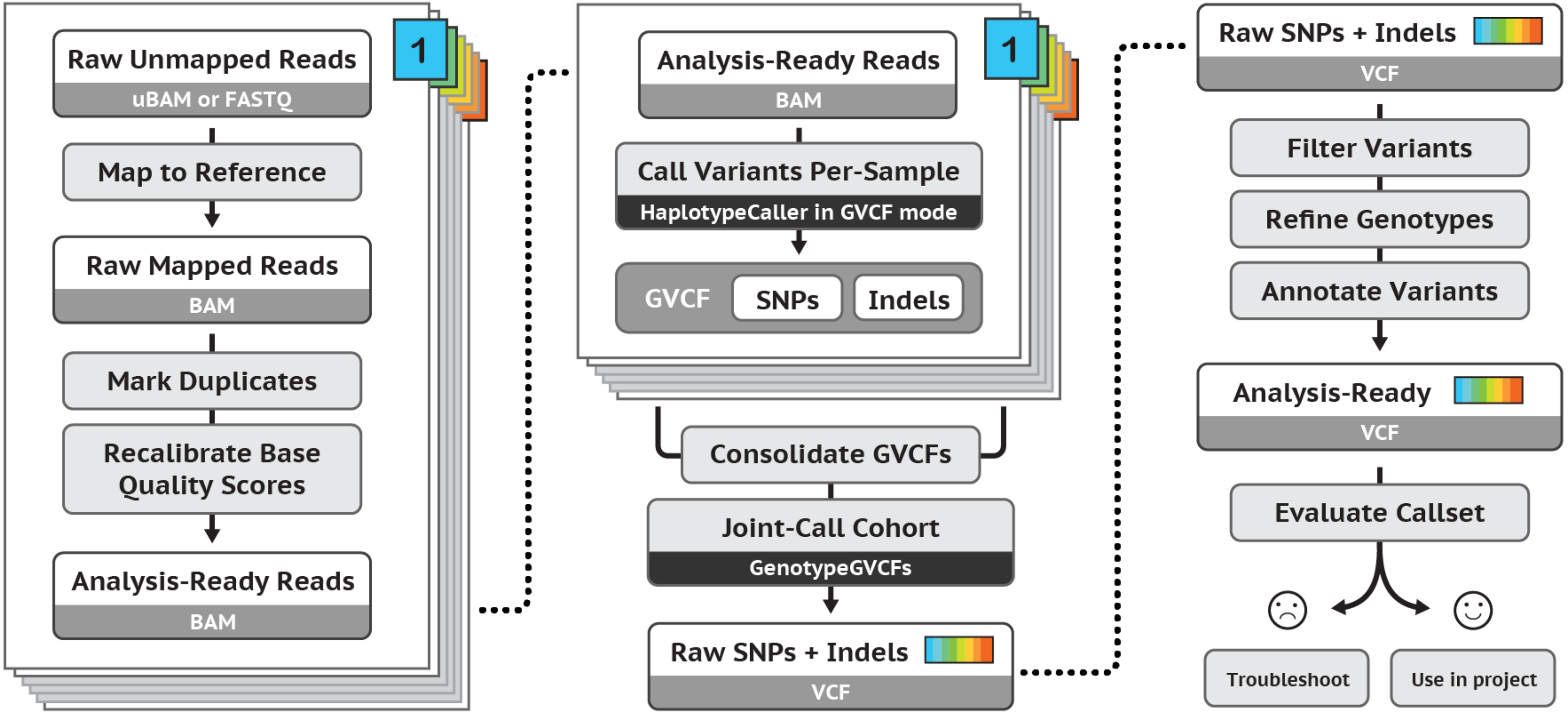
Assigning accurate confidence scores to each putative mutation call

BROAD INSTITUTE

http://software.broadinstitute.org/gatk/

gatk

Best Practices for Germline SNP & INDEL Discovery



Raw callsets must be filtered to balance sensitivity and specificity

- Mutation calling algorithms are very permissive by design
- Raw, high-sensitivity callsets contain many false positives
- Two filtering approaches
  - Hard-filters using binary thresholds
    - Applicable to all BUT requires expertise to define appropriately
  - Variant “recalibration” using machine learning
    - More powerful BUT requires well-curated known resources
- Both entail trade-off between sensitivity and specificity
- AND use variant context annotations

Variant context annotations describe the observed data

Each variant has a diverse set of statistics associated with it:

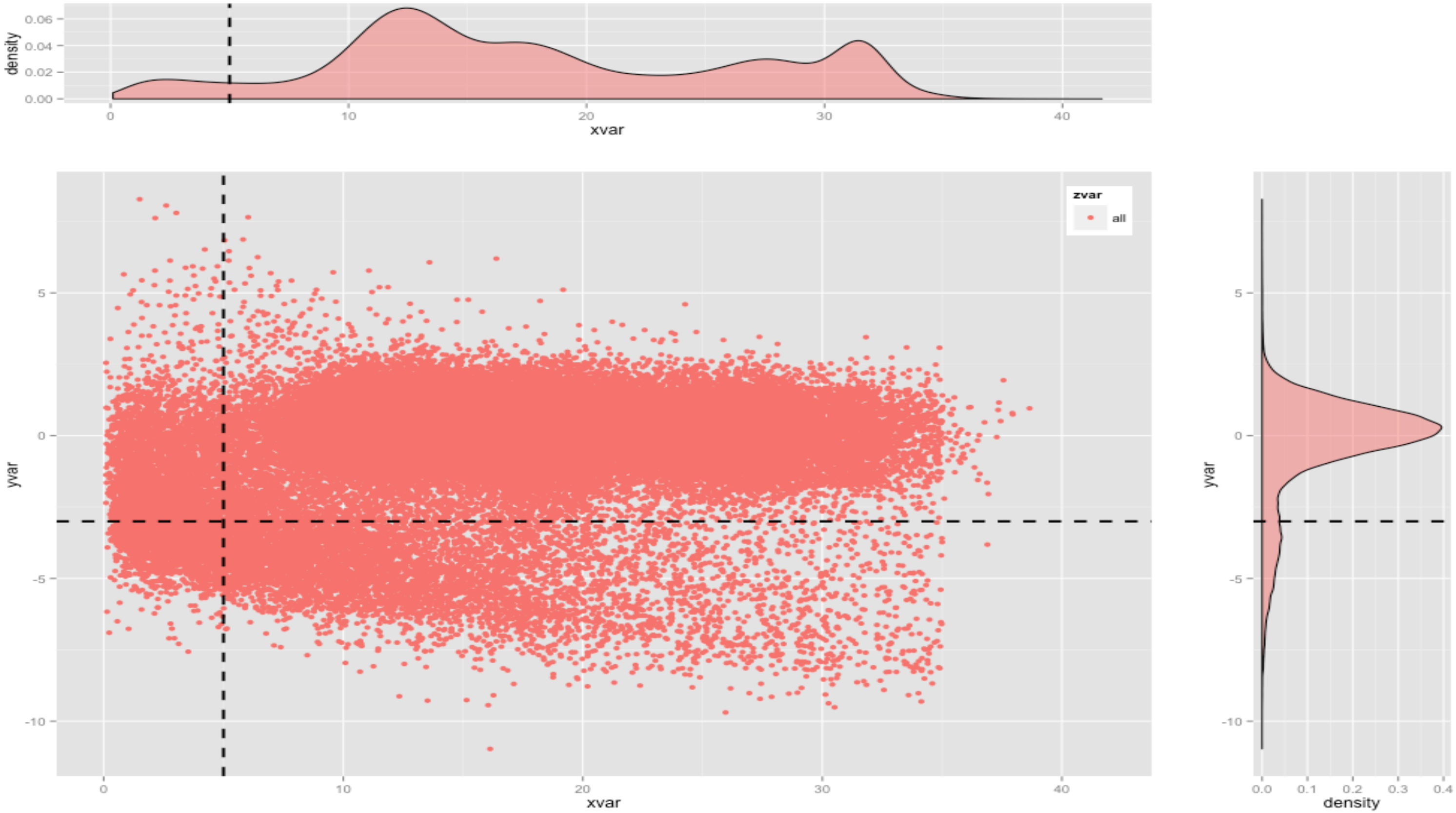
VCF record for an A/G SNP at 22:49582364

```
22 49582364            .         A        G        198.96   .
   AC=3;
   AF=0.50;
   AN=6;
   DP=87;
   MLEAC=3;
   MLEAF=0.50;
   MQ=51.31;
   MQ0=22;
   QD=2.29;
   SB=-31.76
   GT:DP:GQ    0/1:12:99    0/1:11:89    0/1:28:37
```

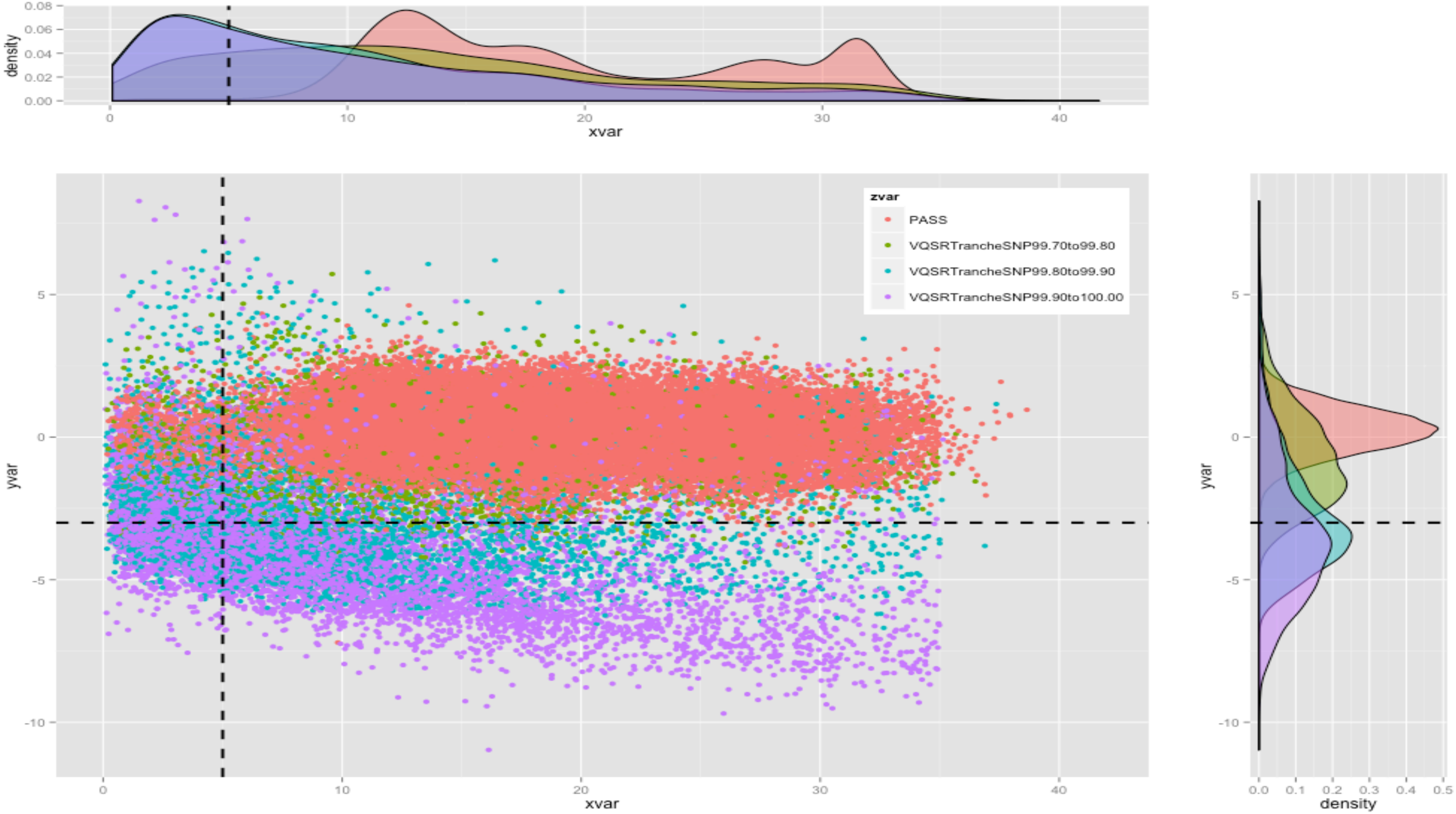
INFO field

AC	No. chromosomes carrying alt allele	MLEAF	Max likelihood AF
AN	Total no. of chromosomes	MQ	RMS MAPQ of all reads
AF	Allele frequency	MQ0	No. of MAPQ 0 reads at locus
DP	Depth of coverage	QD	QUAL score over depth
MLEAC	Max likelihood AC		

Hard-filtering is a very blunt instrument



Variant recalibration is far more discriminating

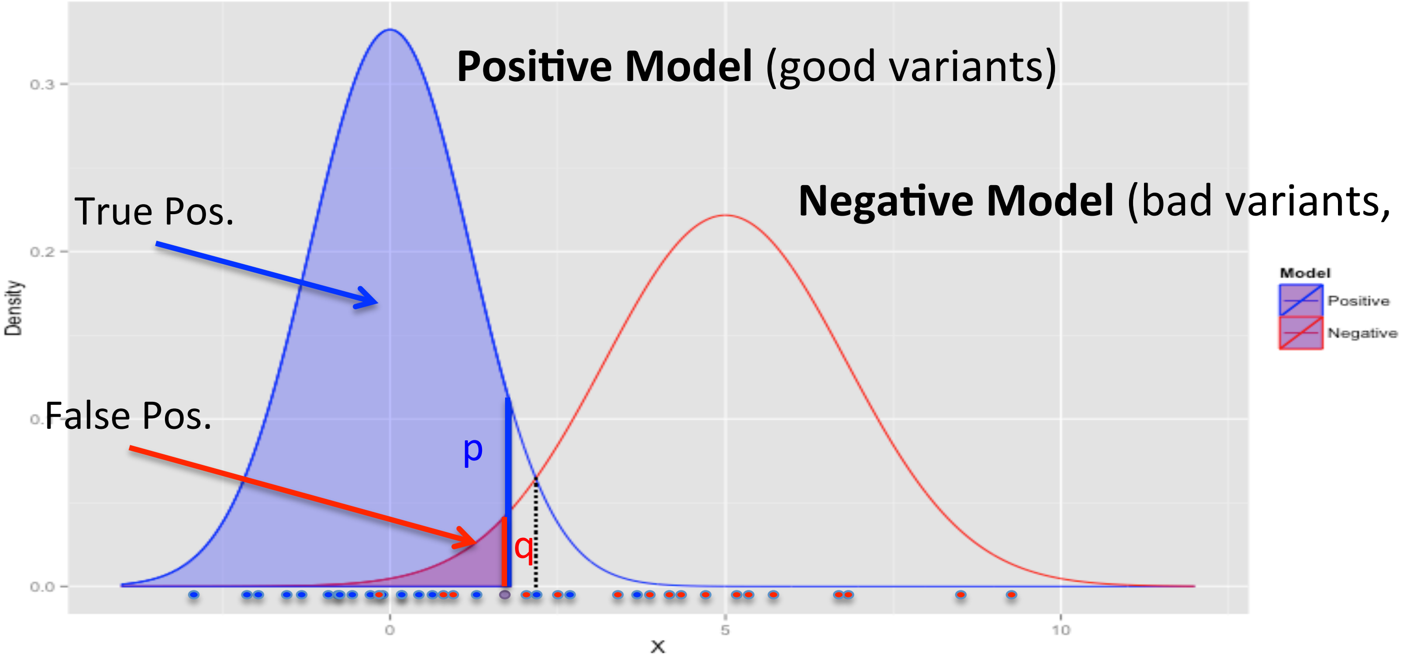


How variant recalibration works

Train on high-confidence known sites to determine the probability that other sites are true or false

- Assume annotations tend to form **Gaussian clusters**
- Build a "Gaussian mixture model" from annotations of **known variants** in our dataset
- Score **all variants** by where their annotations lie relative to these clusters
- Filter base on **sensitivity to truth set**

Actually two models: positive and negative



Positive Model (good variants)

Negative Model (bad variants, no biscuit)

True Pos.

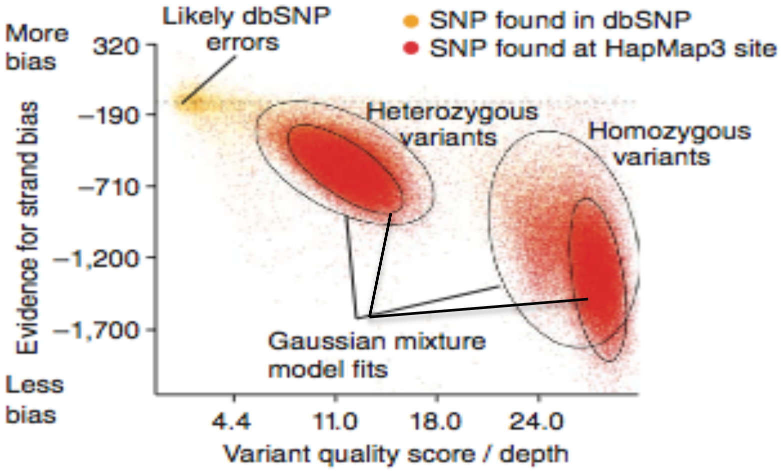
False Pos.

VQSLOD(x) = Log(p(x)/q(x))

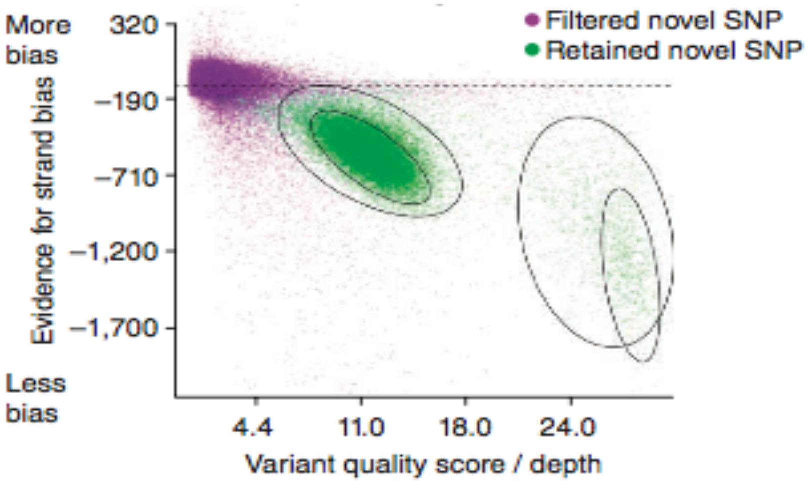
Done for each annotation X
then integrated into single overall VQSLOD

Based on the model, calculate VQSLOD for variants in callset

Model trained on HapMap



Model applied to new SNPs

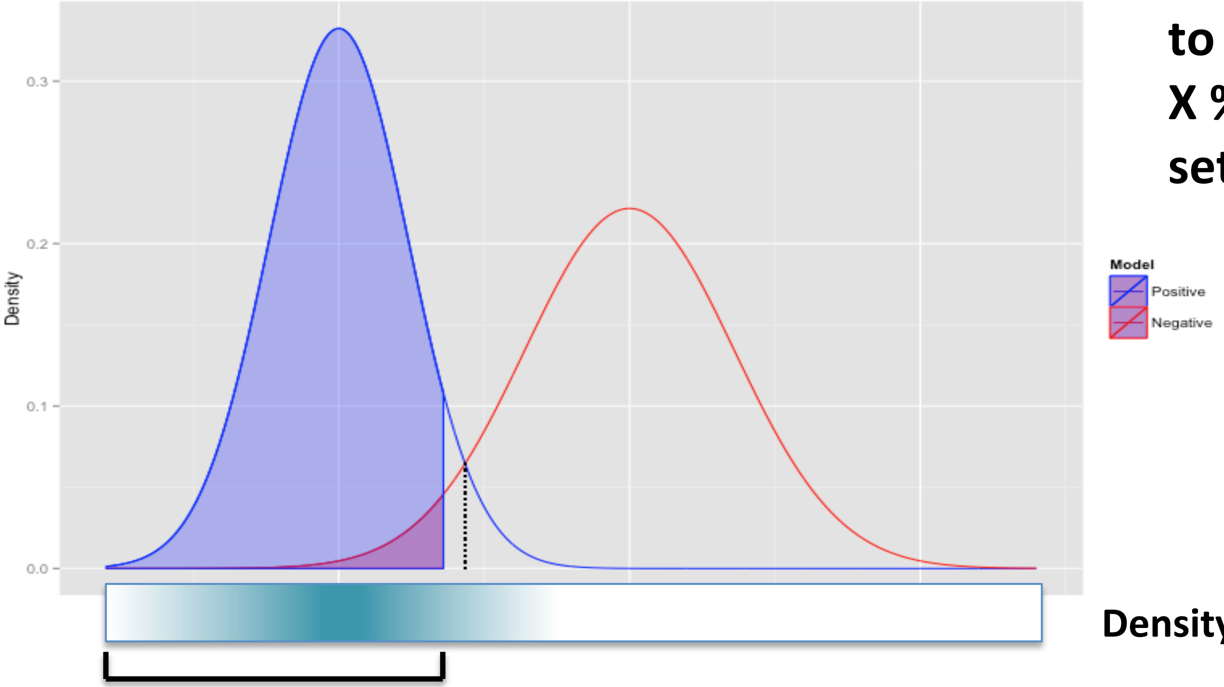


Modified from DePristo et al. Nature Genetics. 2011

Applying filtering is now a matter of setting a VQSLOD threshold

VQSLOD threshold is set by **sensitivity to truth data**

What threshold do we need to set to capture X % of the sites in the truth set?

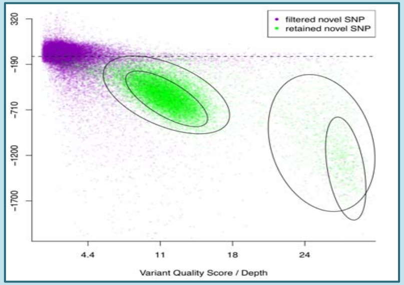


Density of sites in truth set

Variant Recalibration steps & tools

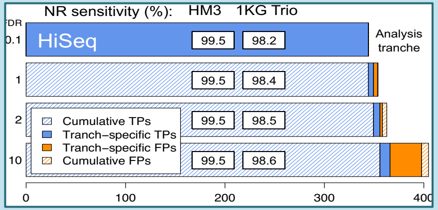
- Build and Apply the models (from resources and callset)

  → **VariantRecalibrator**

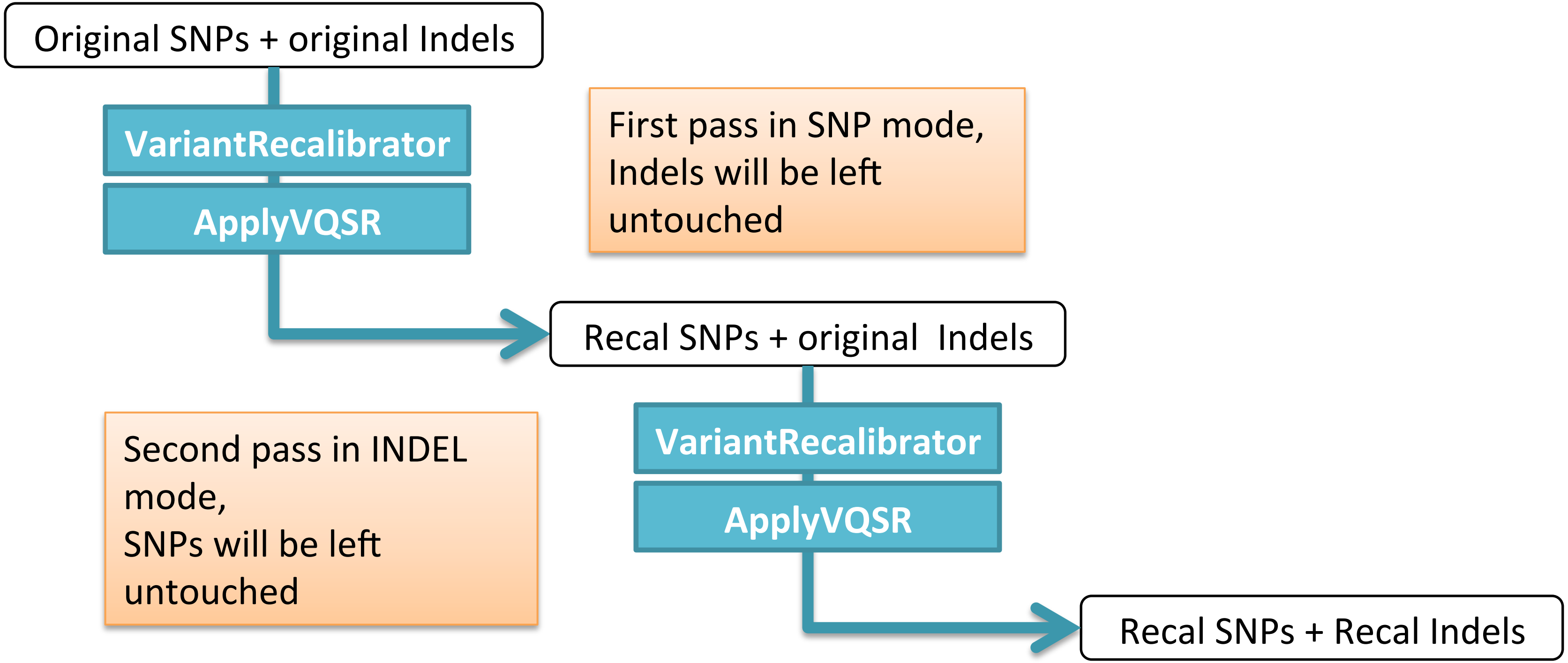


- Use VQSLOD to filter variants and write a new annotated VCF

  → **ApplyVQSR**



NOTE: SNPs and Indels must be recalibrated separately!



Original SNPs + original Indels

VariantRecalibrator

ApplyVQSR

First pass in SNP mode, Indels will be left untouched

Recal SNPs + original Indels

Second pass in INDEL mode,
SNPs will be left untouched

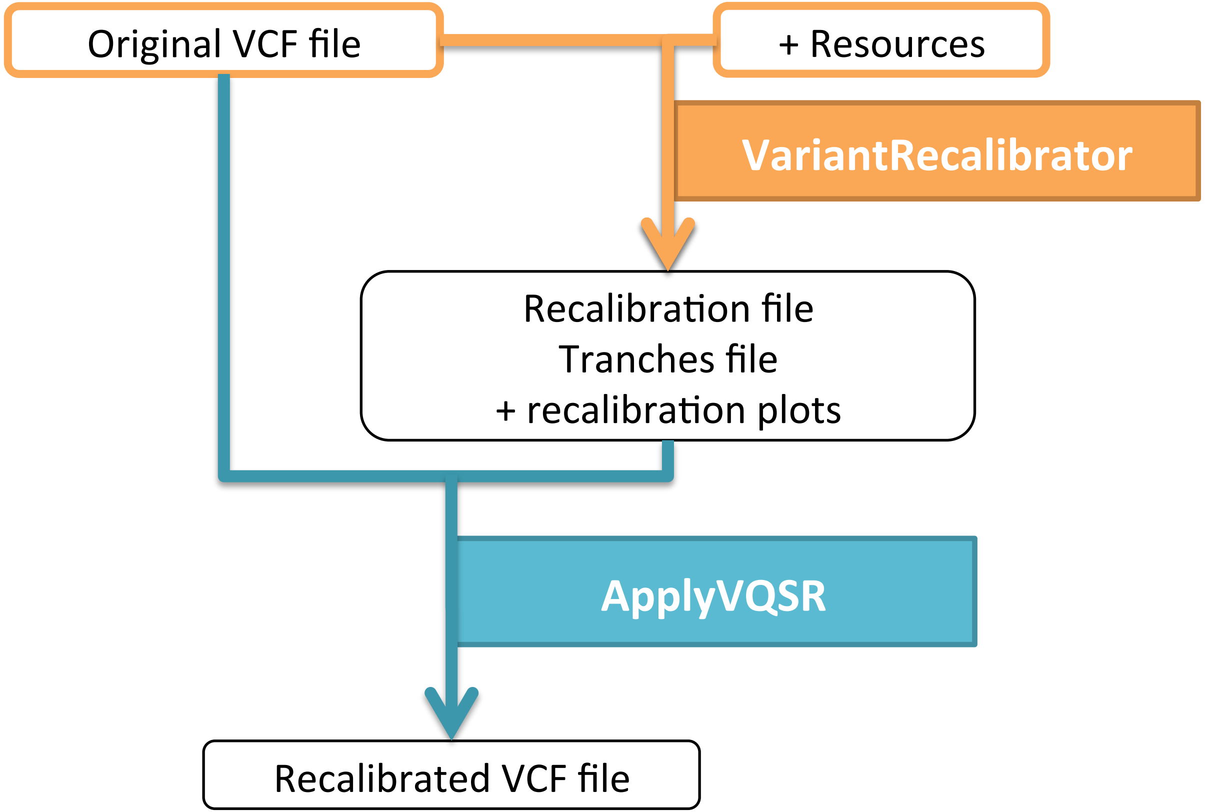
VariantRecalibrator

ApplyVQSR

Recal SNPs + Recal Indels

Pro-tip: Run VQSR twice in succession according to this workflow.
That way you avoid having to split them, recalibrate and combine them again.

Step 1: VariantRecalibrator



Original VCF file

+ Resources

VariantRecalibrator

Recalibration file
Tranches file
+ recalibration plots

ApplyVQSR

Recalibrated VCF file

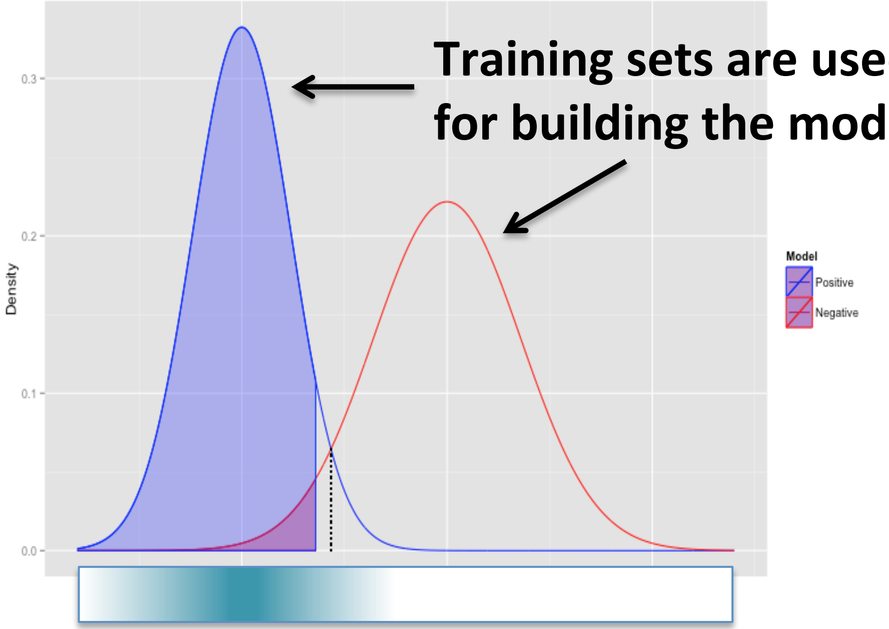
VariantRecalibrator builds the Gaussian mixture model

- Uses the variants in the input callset that overlap the training data
- Uses the annotations *in our callset*, not the resource callset

```
    gatk VariantRecalibrator \
        -R human.fasta \
        -V raw.SNPs.vcf \
        -resource [tags]:filename.vcf \
        -an DP -an QD -an FS -an MQRankSum {…} \
        -mode SNP \
        -recal-file raw.SNPs.recal \
        -tranches-file raw.SNPs.tranches \
        -rscript-file recal.plots.R
```

SNP example – see documentation for indel recommendations

Role of training and truth resources



Training sets are used for building the model

Truth set is used for translating VQSLOD values into sensitivity tranches

- **Training –** input variants that overlap with these training sites to build the model
- **Truth –**determine where to set the cutoff in VQSLOD sensitivity
- **Known –** only for reporting purposes, not used in any calculations
- **Prior –** Phred-scaled estimate of data accuracy

Specifying VQSR resources

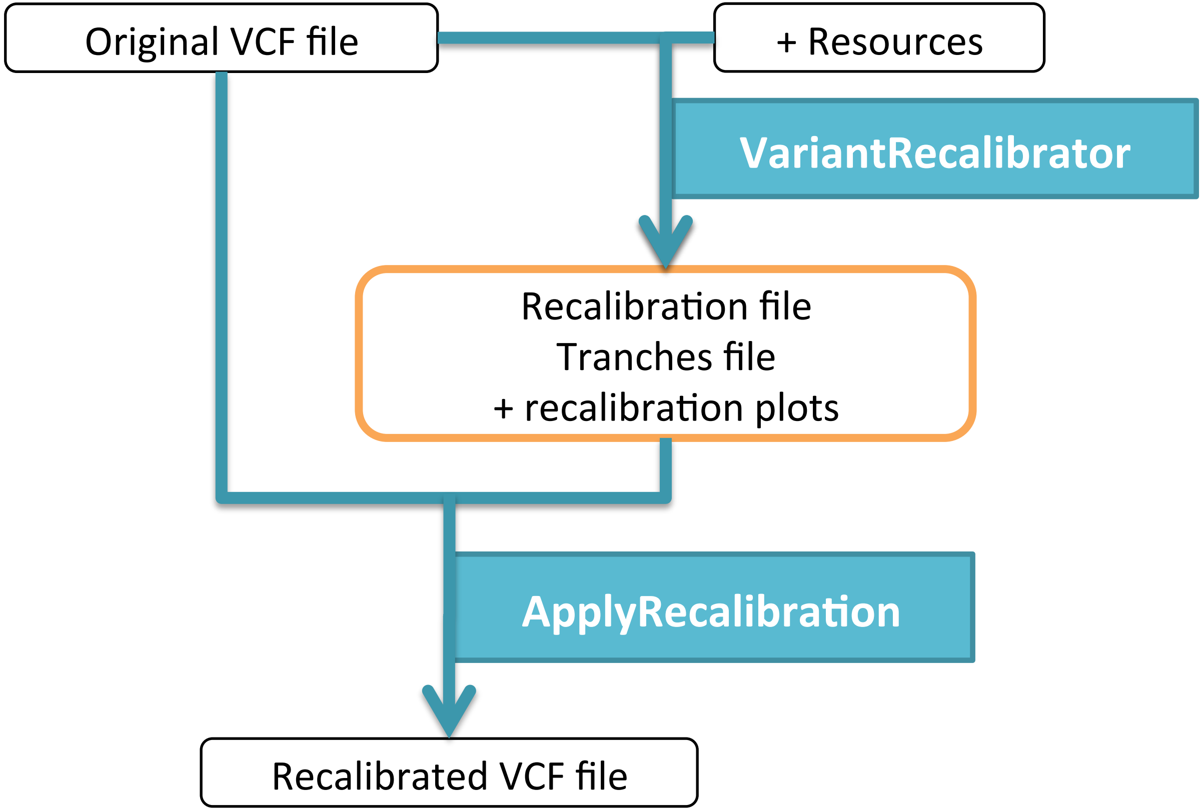
```
-resource [tags]:filename.vcf
```

Tags for SNP resources:

```
hapmap,known=false,training=true,truth=true,prior=15.0
omni,known=false,training=true,truth=false,prior=12.0
1000G,known=false,training=true,truth=false,prior=10.0
dbsnp,known=true,training=false,truth=false,prior=2.0
```

SNP example – see documentation for indel recommendations

Outputs of VR: recal file, tranches, plots



Original VCF file

+ Resources

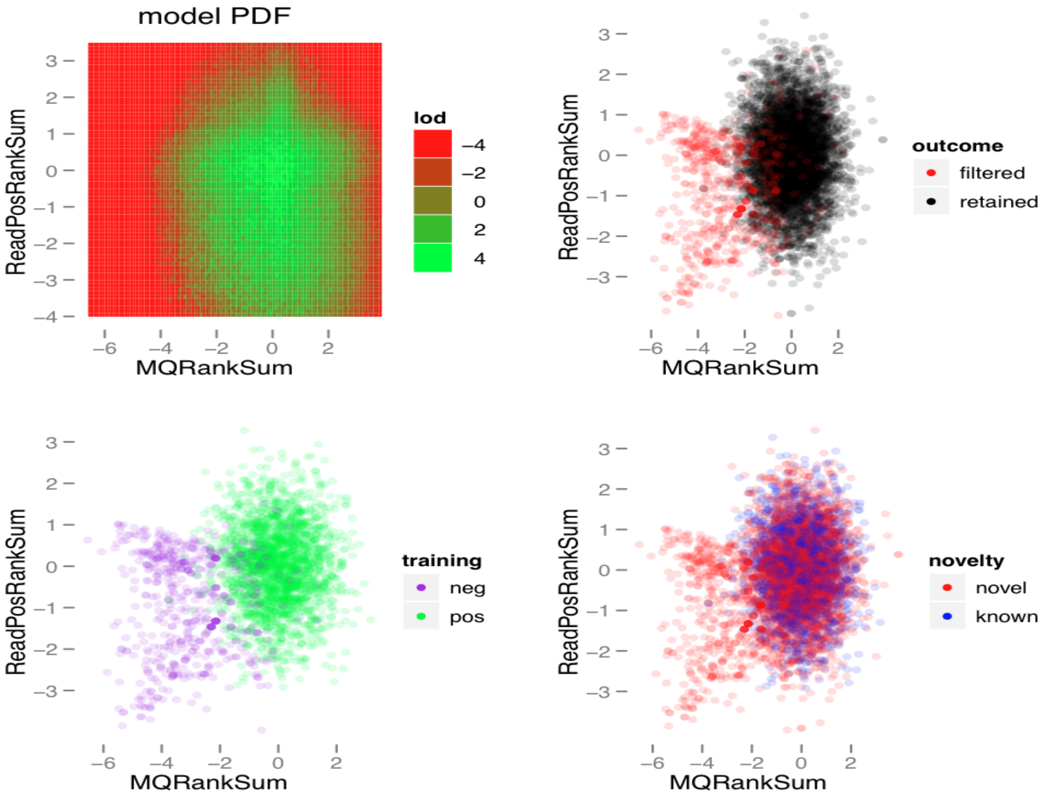
VariantRecalibrator

Recalibration file
Tranches file
+ recalibration plots

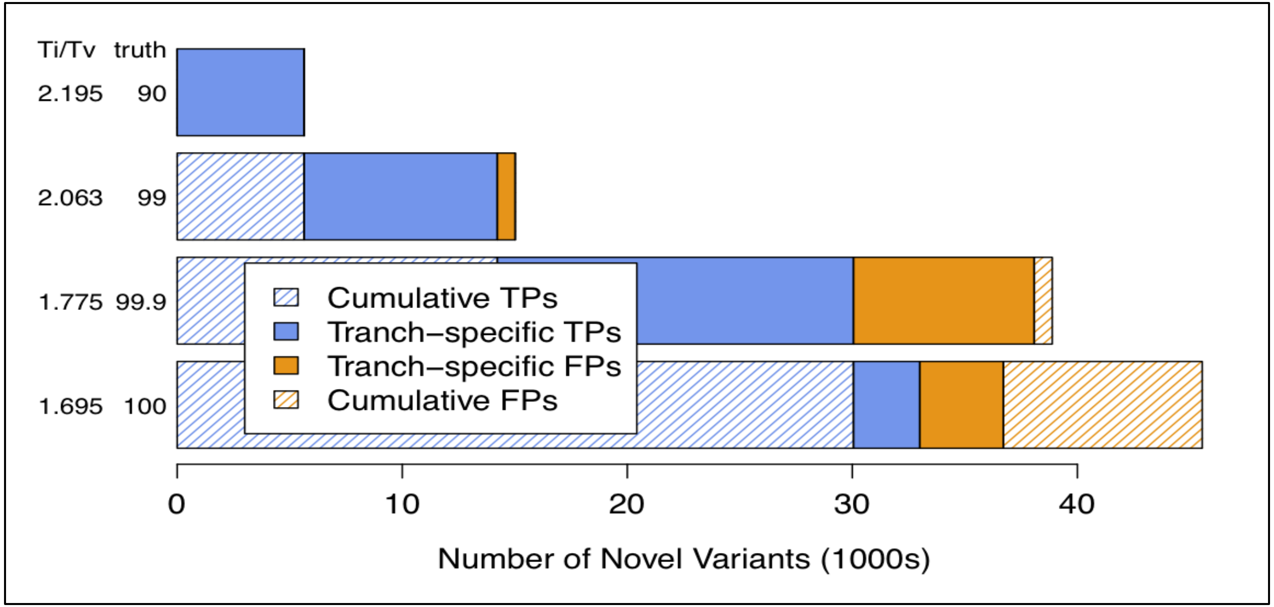
ApplyRecalibration

Recalibrated VCF file

Recalibration plots show aspects of the model



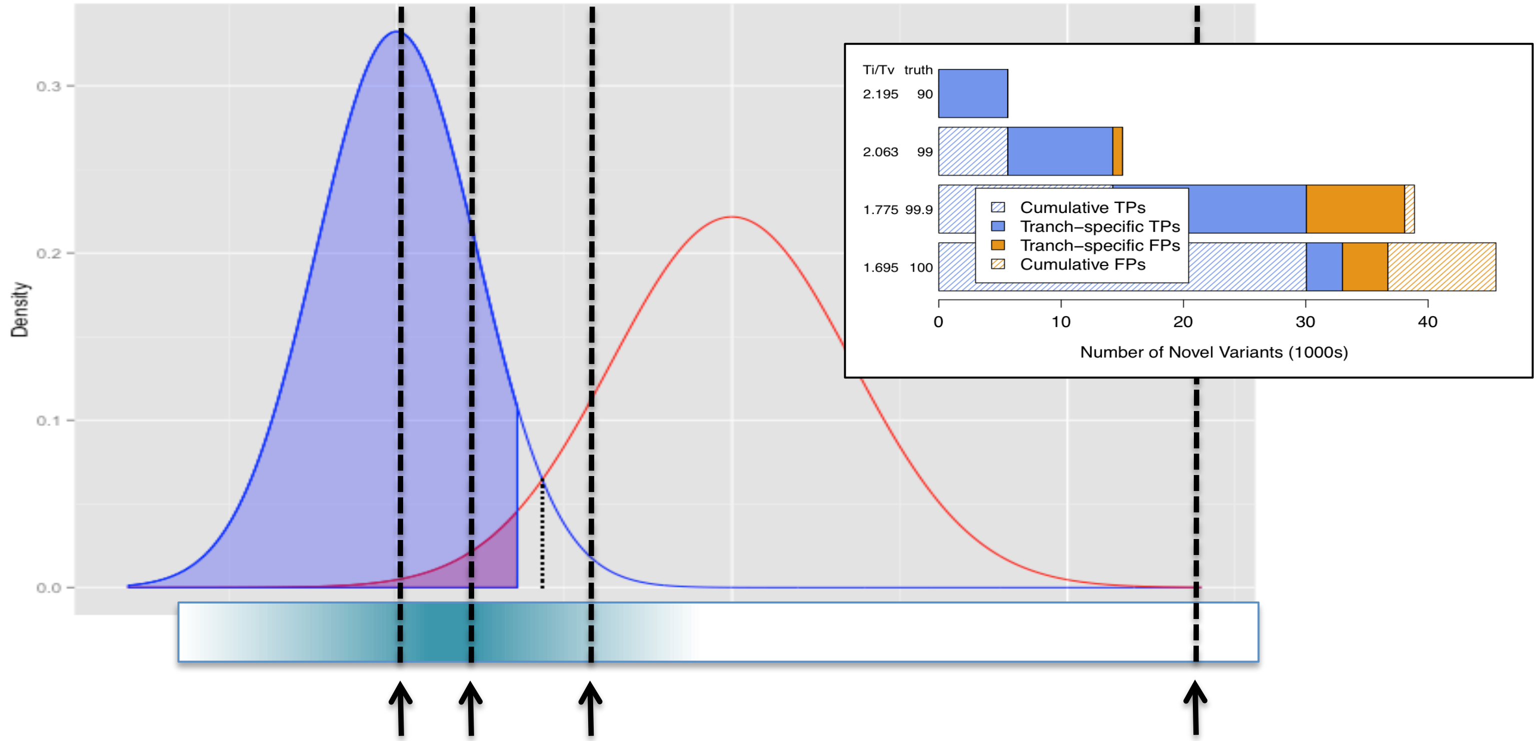
Tranche plots show estimated TP vs FP tradeoff



Estimation is based on Ti/Tv ratio of novel variants

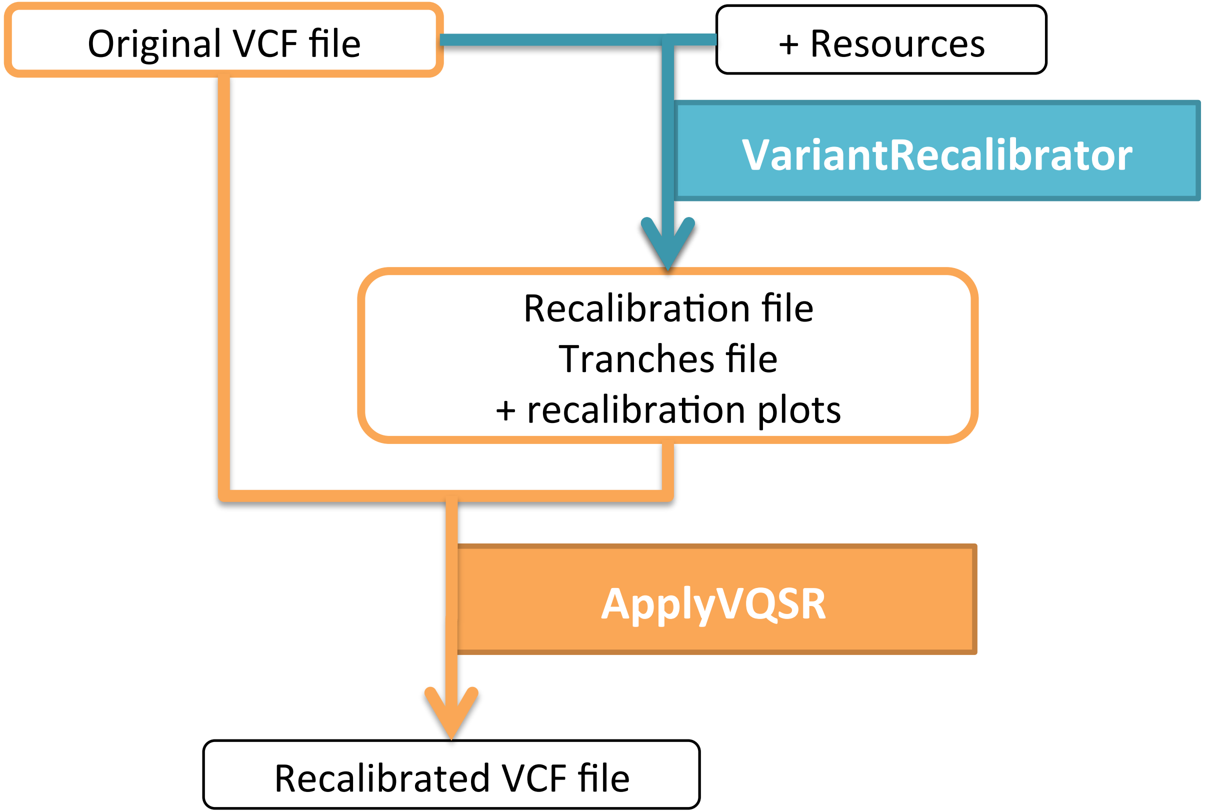
Default target Ti/Tv is for WGS and must be adapted for exomes

Tranches : slices of sensitivity threshold values



Truth sensitivity (%) 90 99 99.9 100

Step 2: ApplyVQSR



Original VCF file

+ Resources

VariantRecalibrator

Recalibration file
Tranches file
+ recalibration plots

ApplyVQSR

Recalibrated VCF file

ApplyVQSR applies the filtering threshold

- Executes the desired sensitivity / specificity tradeoff
  by applying filters to the input callset (no new calculations)

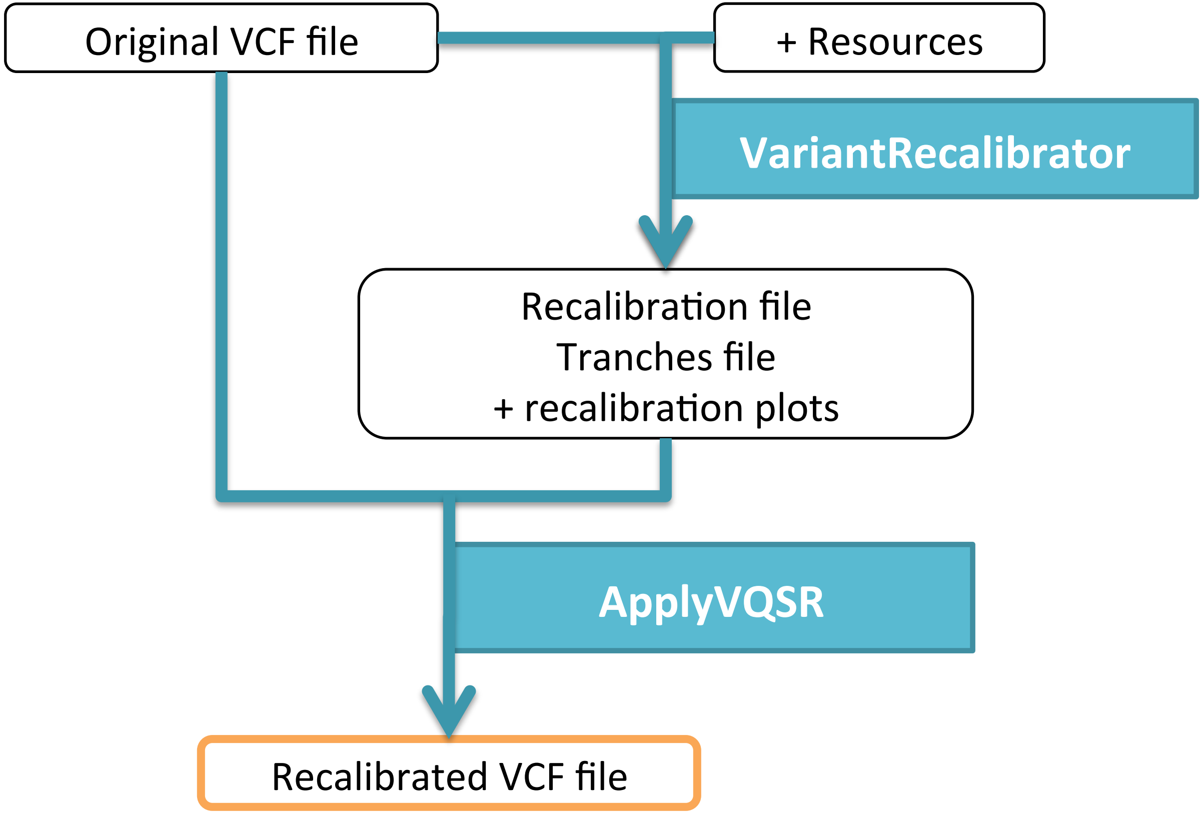
- Creates a new, filtered, analysis-worthy VCF file.

```
gatk ApplyVQSR \
    -R human.fasta \
    -V raw.vcf \
    -mode SNP \
    -recal-file raw.SNPs.recal \
    -tranches-file raw.SNPs.tranches \
    -O recal.SNPs.vcf \
    -ts-filter-level 99.0
```

- Additionally every variant is now annotated with its VQSLOD score.

SNP example – see documentation for indel recommendations

Output: the final recalibrated VCF file



Original VCF file

+ Resources

VariantRecalibrator

Recalibration file
Tranches file
+ recalibration plots

ApplyVQSR

Recalibrated VCF file

VQSR output VCF (vs. Hard Filter)

- Before VQSR (input vcf):

```
#CHROM   POS      FILTER                          INFO
1        10146    .                               AC=1;DP=32;FS=9.208; MQ=31.96;MQRankSum=0.085;...
1        10403    .                               AC=1;DP=64;FS=1.645;MQ=41.86;MQRankSum=1.87;...
1        234313   .                               AC=1;DP=239;FS=12.675;MQ=38.19;MQRankSum=-0.122;...
```

- After VQSR (output vcf):

```
#CHROM   POS      FILTER                          INFO
1        10146    VQSRTrancheINDEL99.30to99.50    AC=1...;NEGATIVE_TRAIN_SITE;VQSLOD=-1.328;culprit=SOR
1        10403    PASS                            AC=1;...;QD=0.60; VQSLOD=0.794;culprit=QD
1        234313   VQSRTrancheSNP99.90to100.00     AC=1;...;POSITIVE_TRAIN_SITE;VQSLOD=-5.356;culprit=MQ
```

- Hard filtered vcf:

```
#CHROM   POS      FILTER                          INFO
1        10146    PASS                            AC=1;DP=32;FS=9.208; MQ=31.96;MQRankSum=0.085;...
1        10403    INDEL_Filter                    AC=1;DP=64;FS=1.645;MQ=41.86;MQRankSum=1.87;...
1        234313   SNP_Filter                      AC=1;DP=239;FS=12.675;MQ=38.19;MQRankSum=-0.122;...
```

Tips for running VQSR on exome data

- Smaller number of variants per sample compared to WGS

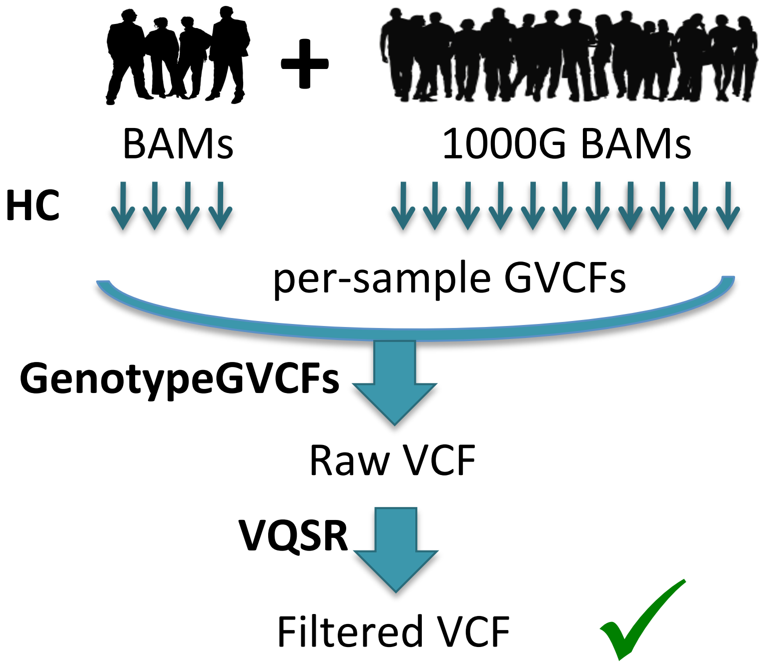
  -> typically insufficient to build a robust recalibration model if running on only a few samples

- ➢ Analyze samples jointly in cohorts of at least 30 samples
- ➢ If necessary, add exomes from 1000G Project or comparable

- What to look for in samples for padding a cohort:
  - **Similar technical generation** is paramount (technology, capture, read length, depth)

How to add exomes from 1000G to your analysis

ALWAYS do this:



BAMs + 1000G BAMs

HC

per-sample GVCFs

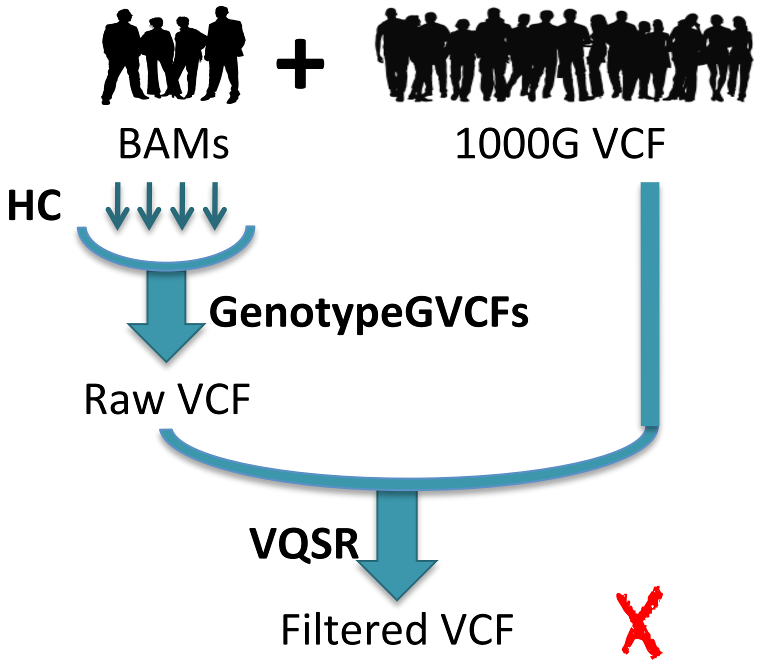
GenotypeGVCFs

Raw VCF

VQSR

Filtered VCF ✓

NEVER do this :



BAMs + 1000G VCF

HC

GenotypeGVCFs

Raw VCF

VQSR

Filtered VCF ✗

When should you NOT run VQSR?

- Non-human organisms where known resources are unavailable or insufficiently curated

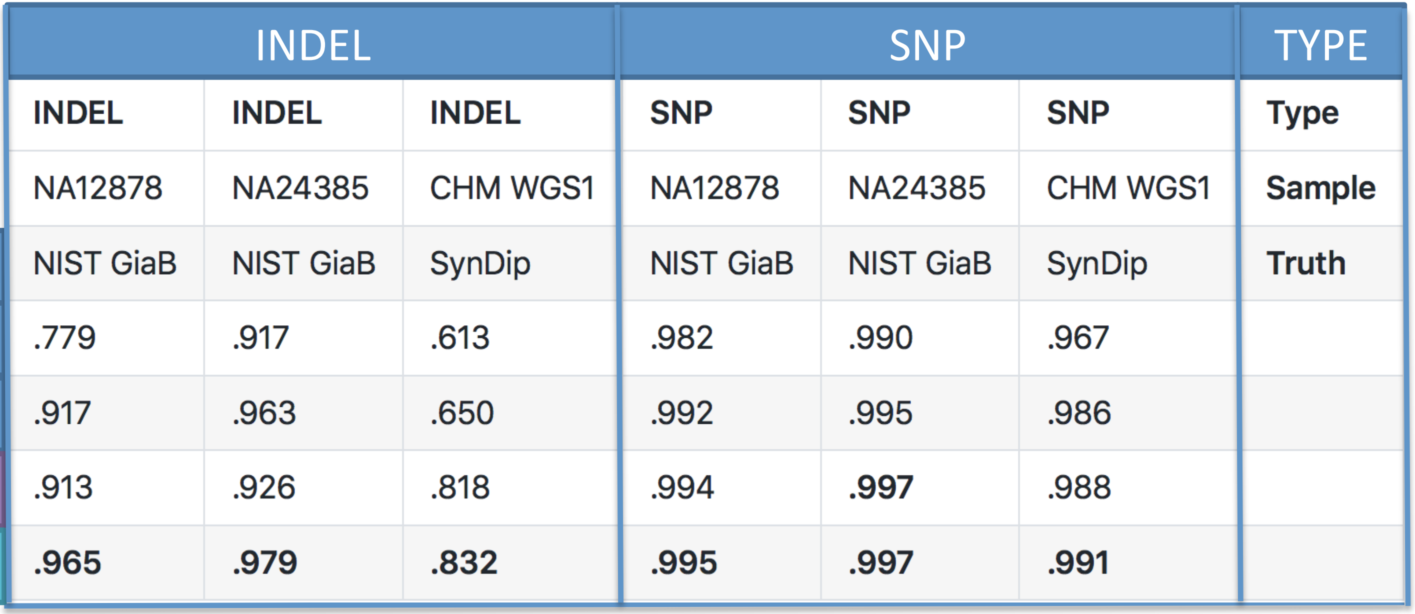
- RNAseq data → see RNAseq-specific filtering

- Cohort is too small and no other samples are available for “padding” the cohort

→ Use manual filtering recommendations instead

Deep learning : a new frontier in genomics

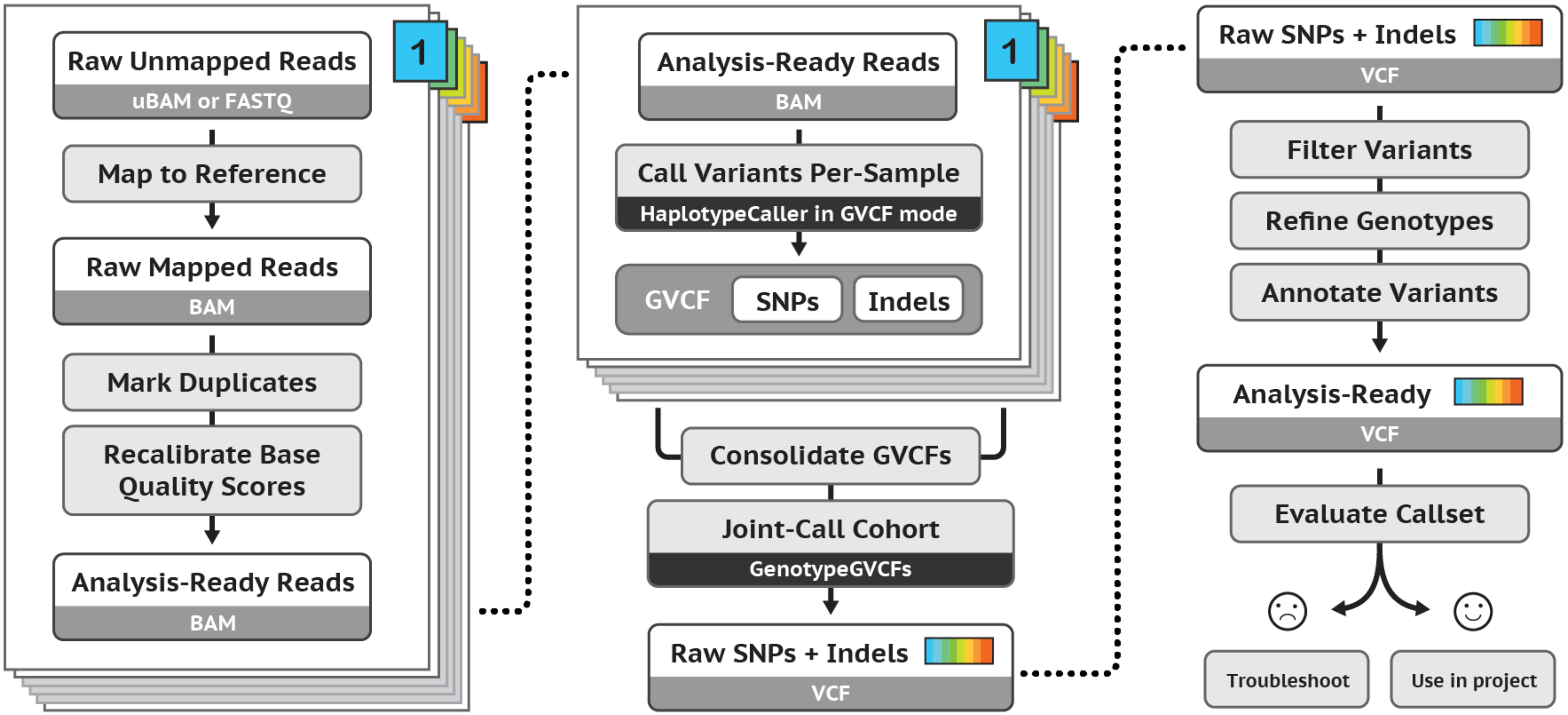
Area under Precision-Recall Curve



	INDEL			SNP			TYPE
	INDEL	**INDEL**	**INDEL**	**SNP**	**SNP**	**SNP**	**Type**
	NA12878	NA24385	CHM WGS1	NA12878	NA24385	CHM WGS1	**Sample**
Architecture	NIST GiaB	NIST GiaB	SynDip	NIST GiaB	NIST GiaB	SynDip	**Truth**
VQSR 1-sample	.779	.917	.613	.982	.990	.967	
VQSR gnomAD	.917	.963	.650	.992	.995	.986	
Deep Variant	.913	.926	.818	.994	**.997**	.988	
GATK4 CNN	**.965**	**.979**	**.832**	**.995**	**.997**	**.991**	

Convolutional Neural Networks

Best Practices for Germline SNP & INDEL Discovery



Raw Unmapped Reads
uBAM or FASTQ
Map to Reference
Raw Mapped Reads
BAM
Mark Duplicates
Recalibrate Base Quality Scores
Analysis-Ready Reads
BAM

Analysis-Ready Reads
BAM
Call Variants Per-Sample
HaplotypeCaller in GVCF mode
GVCF
SNPs
Indels
Consolidate GVCFs
Joint-Call Cohort
GenotypeGVCFs
Raw SNPs + Indels
VCF

Raw SNPs + Indels
VCF
Filter Variants
Refine Genotypes
Annotate Variants
Analysis-Ready
VCF
Evaluate Callset
Troubleshoot
Use in project