Exome Analysis

Metrics and Quality Control

Raony Guimarães
raonyguimaraes@gmail.com
UFMG - Belo Horizonte - Brazil

Summary

- Picard Metrics
- GATK Metrics
- Annovar e Vaast

Picard Metrics

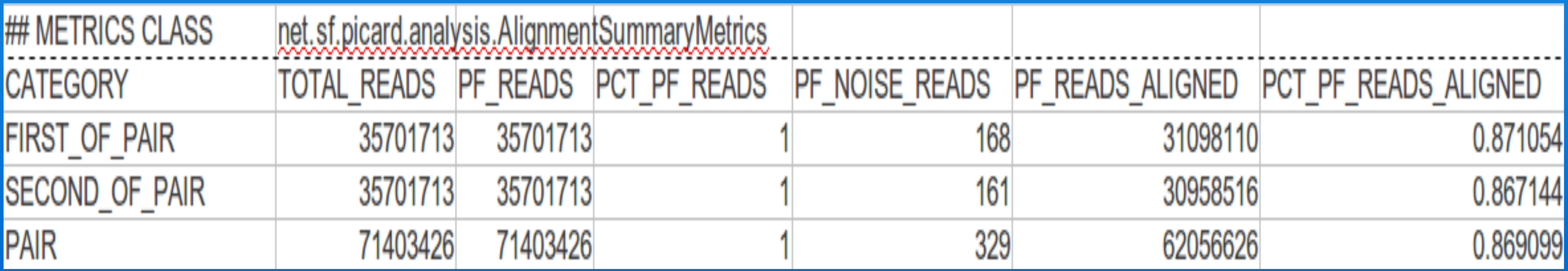
Picard Metrics

- CollectAlignmentSummaryMetrics
- CollectGcBiasMetrics
- CollectInsertSizeMetrics
- MeanQualityByCycle
- QualityScoreDistribution

Picard Metrics - CollectAlignmentSummaryMetrics

Reads a SAM or BAM file and writes a file containing summary alignment metrics.

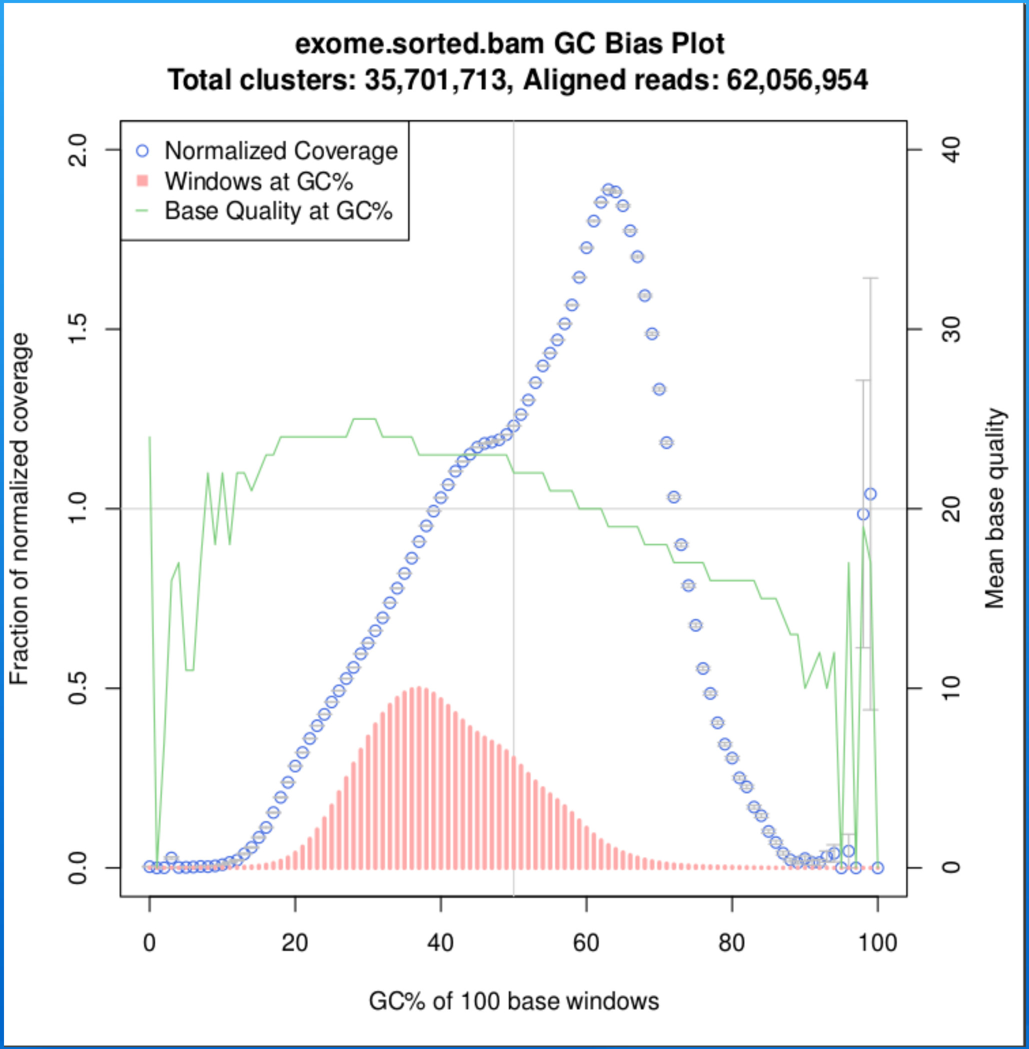
Planilha com os dados:
exome.b37_1kg.AlignmentSummaryMetrics.csv



## METRICS CLASS	net.sf.picard.analysis.AlignmentSummaryMetrics					
CATEGORY	TOTAL_READS	PF_READS	PCT_PF_READS	PF_NOISE_READS	PF_READS_ALIGNED	PCT_PF_READS_ALIGNED
FIRST_OF_PAIR	35701713	35701713	1	168	31098110	0.871054
SECOND_OF_PAIR	35701713	35701713	1	161	30958516	0.867144
PAIR	71403426	71403426	1	329	62056626	0.869099

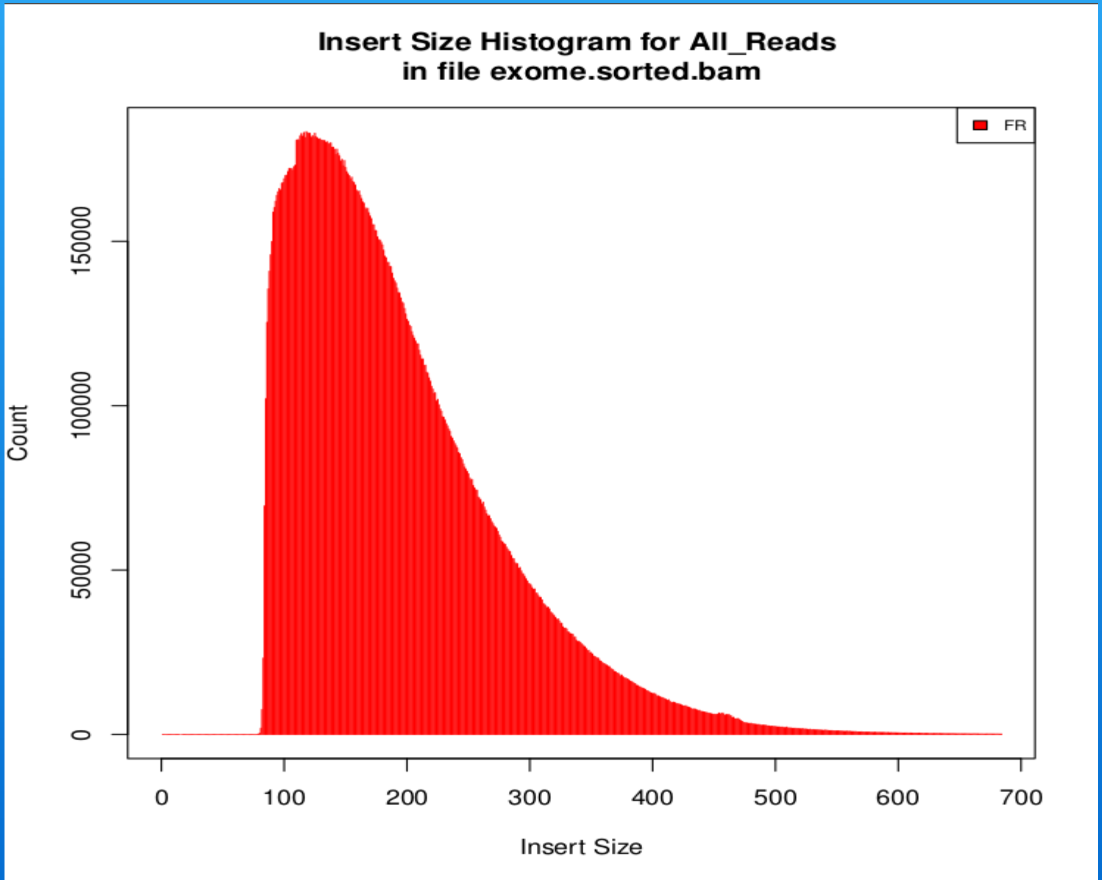
Picard Metrics - *CollectGcBiasMetrics*

Tool to collect information about GC bias in the reads in a given BAM file. Computes the number of windows (of size specified by WINDOW_SIZE) in the genome at each GC% and counts the number of read starts in each GC bin. What is output and plotted is the "normalized coverage" in each bin - i.e. the number of reads per window normalized to the average number of reads per window across the whole genome.



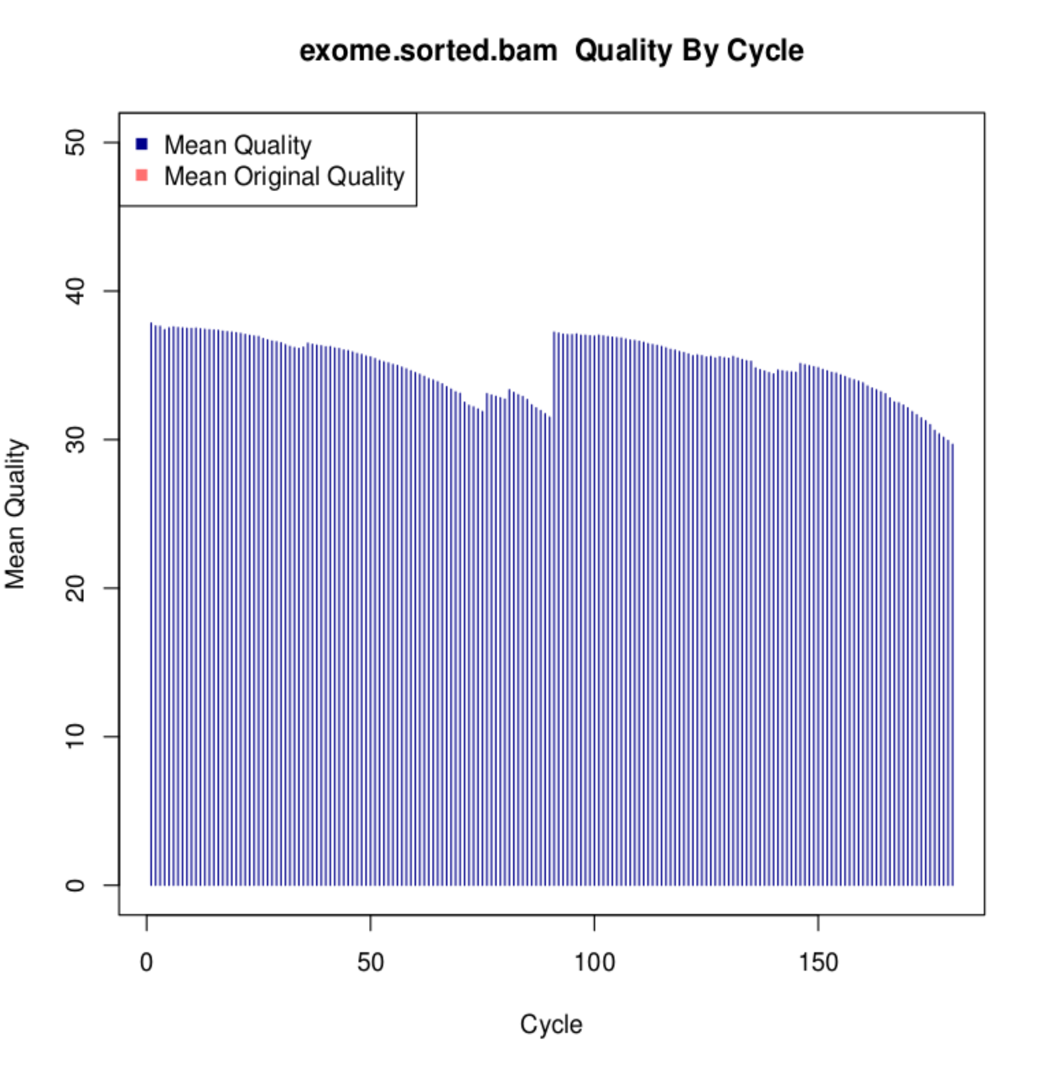
Picard Metrics - *CollectInsertSizeMetrics*

Command line program to read non-duplicate insert sizes, create a histogram and report distribution statistics.

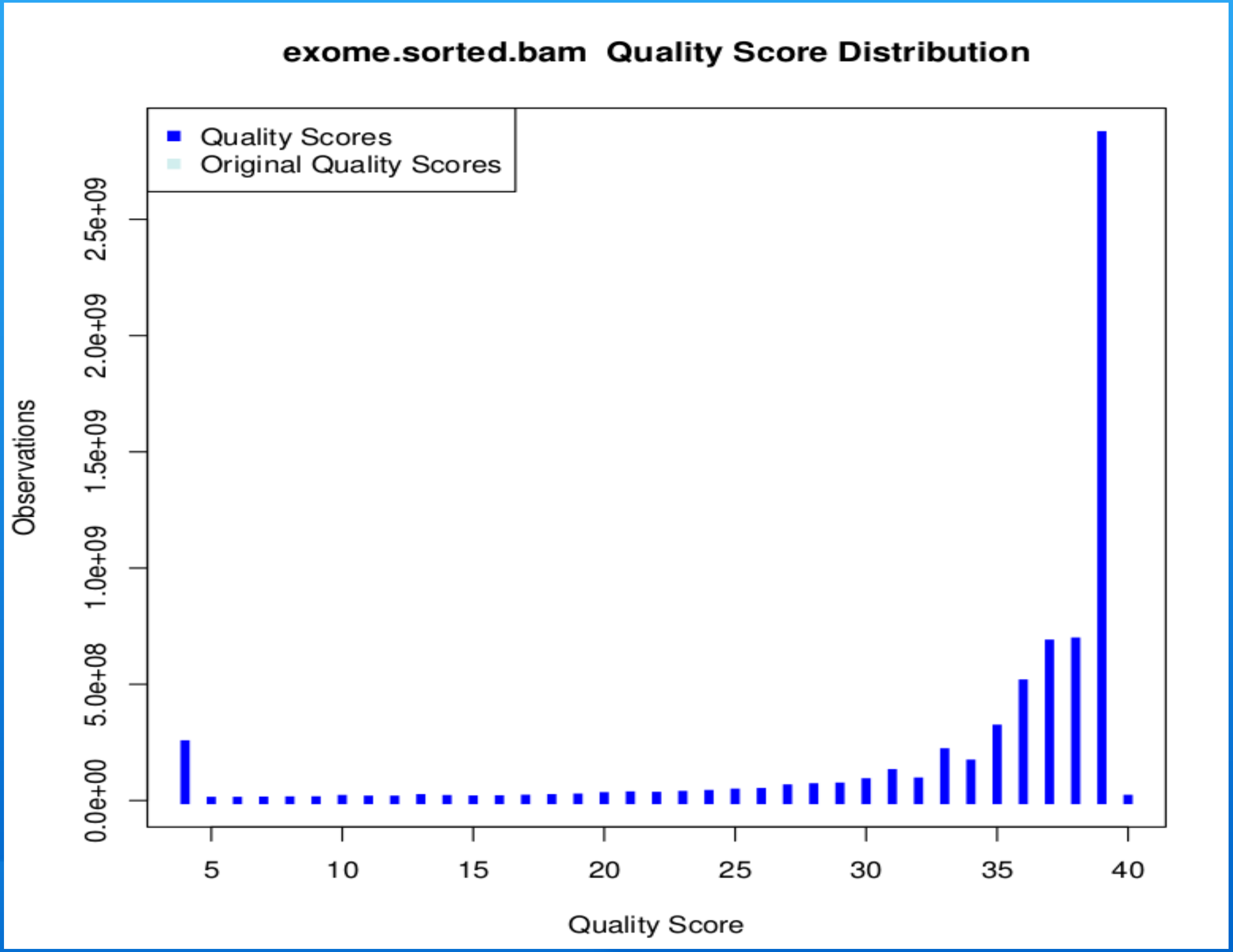


Picard Metrics - *MeanQualityByCycle*

Program to generate a data table and chart of mean quality by cycle from a BAM file. Works best on a single lane/run of data, but can be applied to merged BAMs - the output may just be a little confusing.



Picard Metrics - *QualityScoreDistribution*



GATK Metrics

Base Quality Score Recalibration:
Before/After

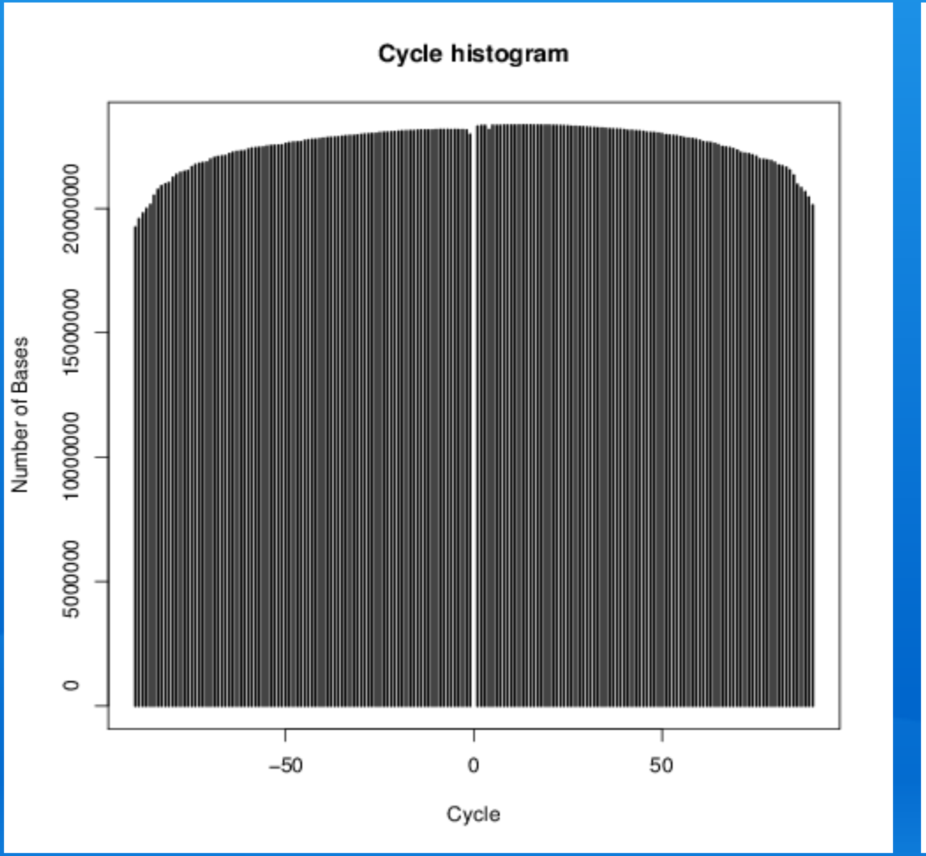
GATK - Base Quality Score Recalibration

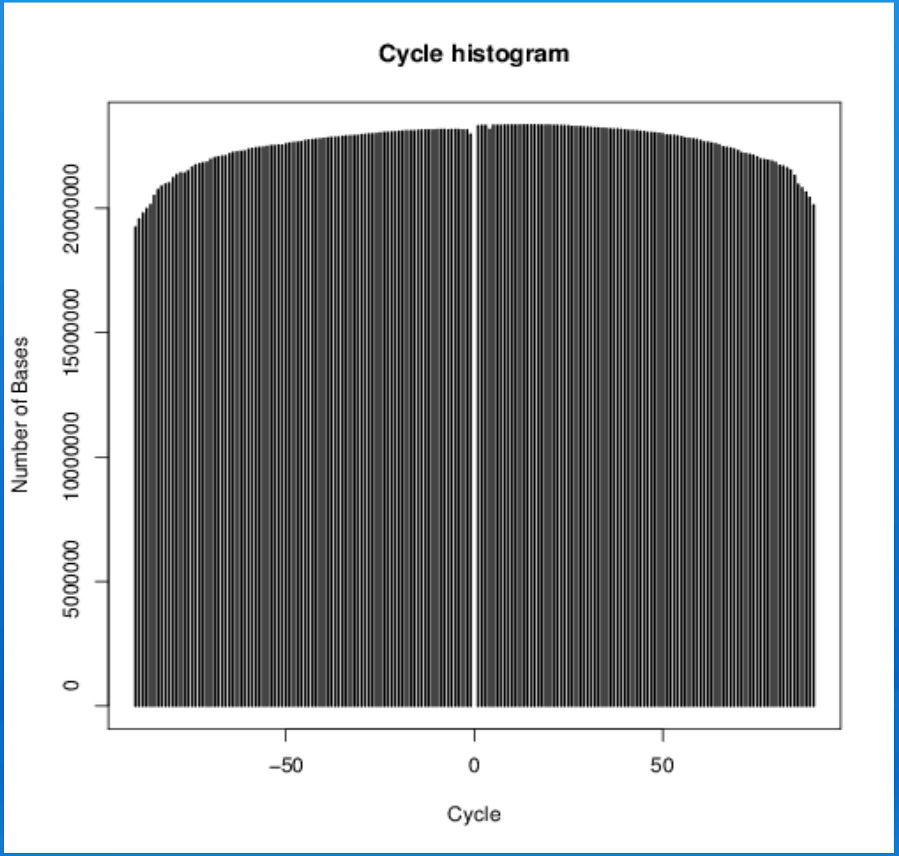
CycleCovariate: The machine cycle for this base (different definition for the various technologies and therefore platform [@PL tag] is pulled out of the read's read group).
DinucCovariate: The combination of this base and the previous base.
HomopolymerCovariate: The number of consecutive previous bases that match the current base.
MappingQualityCovariate: The mapping quality assigned to this read by the aligner.
MinimumNQSCovariate: The minimum base quality score in a small window in the read around this base.
PositionCovariate: The position along the length of the read. For Illumina this is the same as machine cycle but that is not the case for the other platforms.
PrimerRoundCovariate: The primer round for this base (only meaningful for SOLiD reads).
QualityScoreCovariate: The reported base quality score for this base.
ReadGroupCovariate: The read group this read is a member of.

GATK - Base Quality Score Recalibration

Cycle_hist

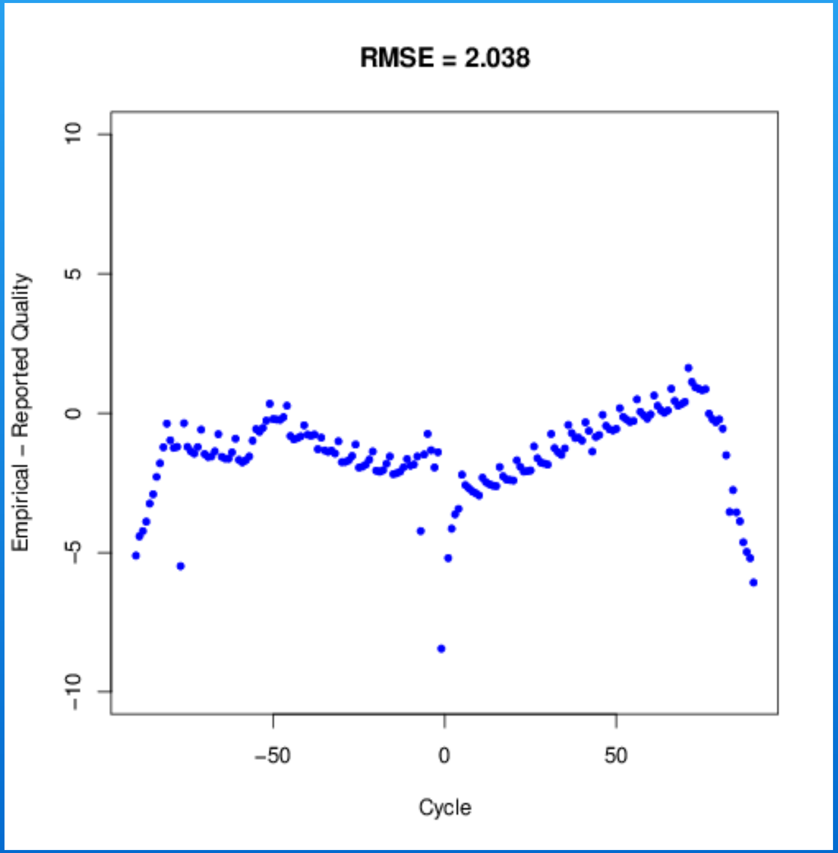
The machine cycle for this base (different definition for the various technologies and therefore platform [@PL tag] is pulled out of the read's read group).

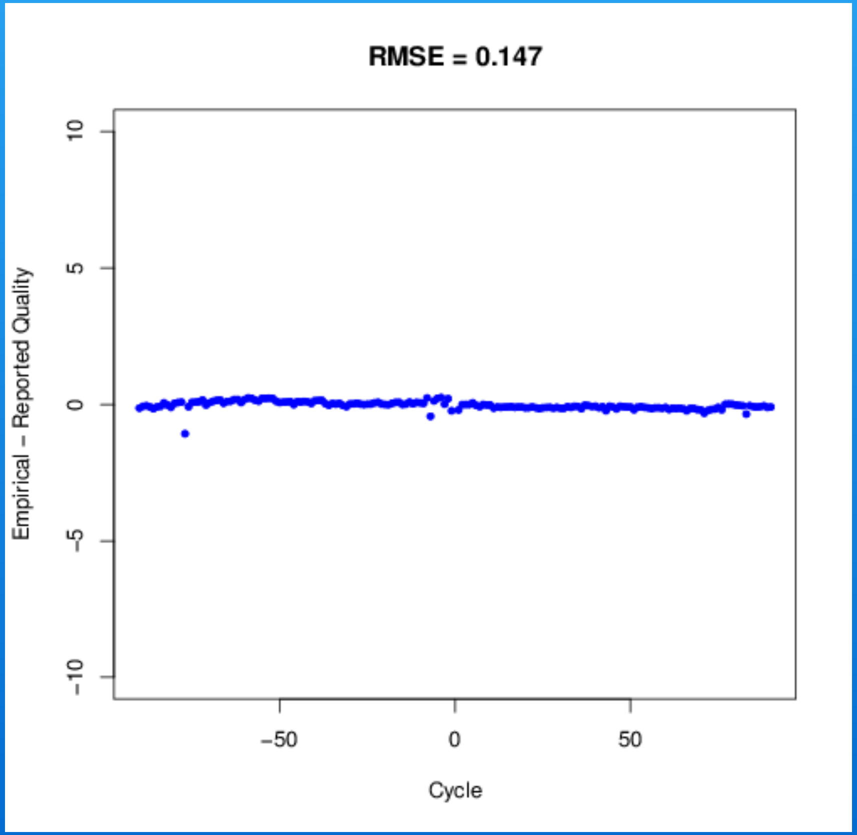




GATK - Base Quality Score Recalibration

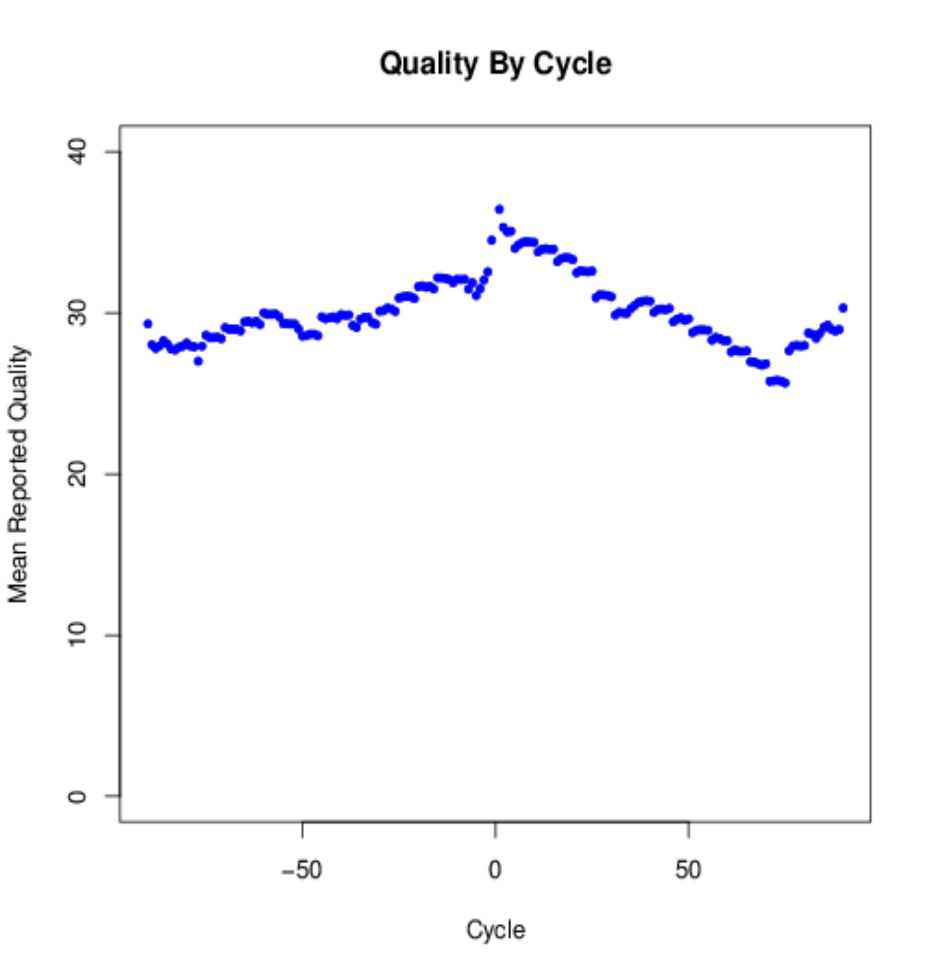
CycleCovariate - *qual_diff_v_Cycle*

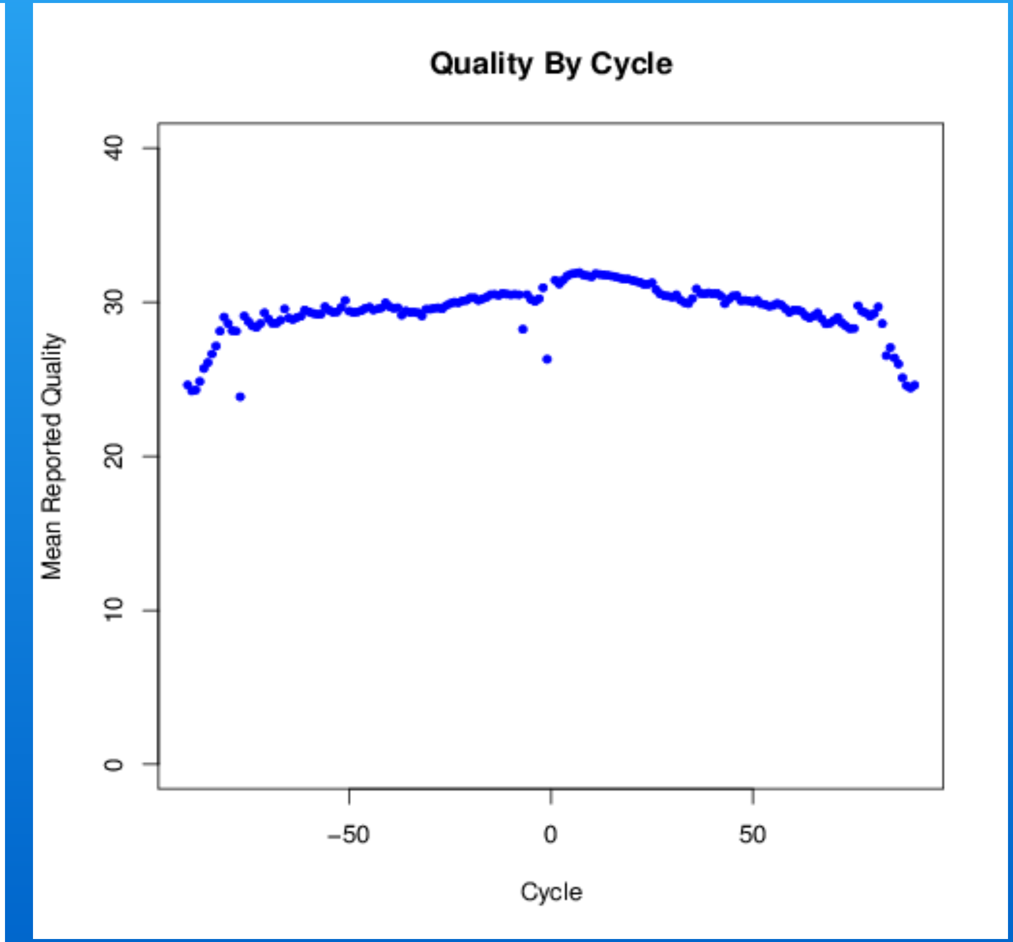




GATK - Base Quality Score Recalibration

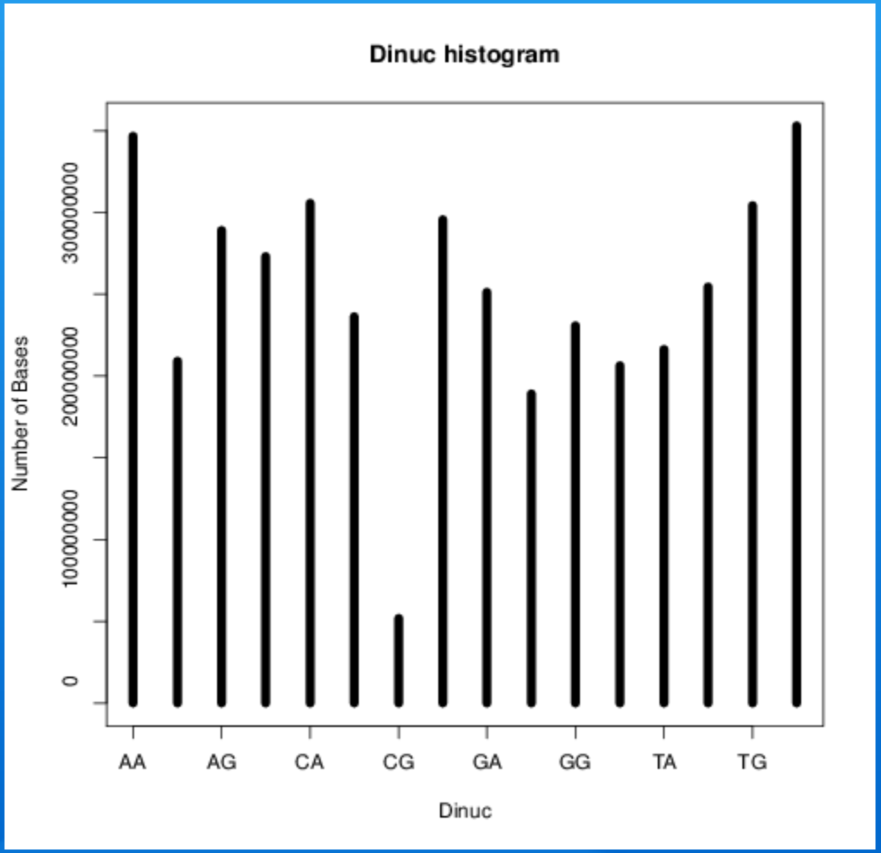
CycleCovariate - *reported_qual_v_Cycle*

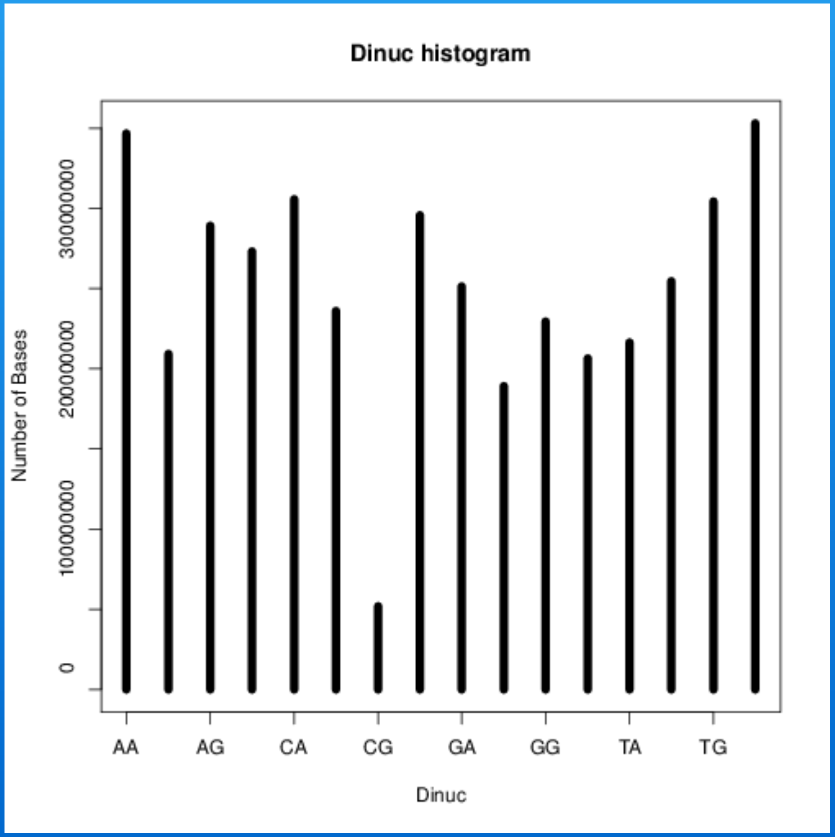




GATK - Base Quality Score Recalibration

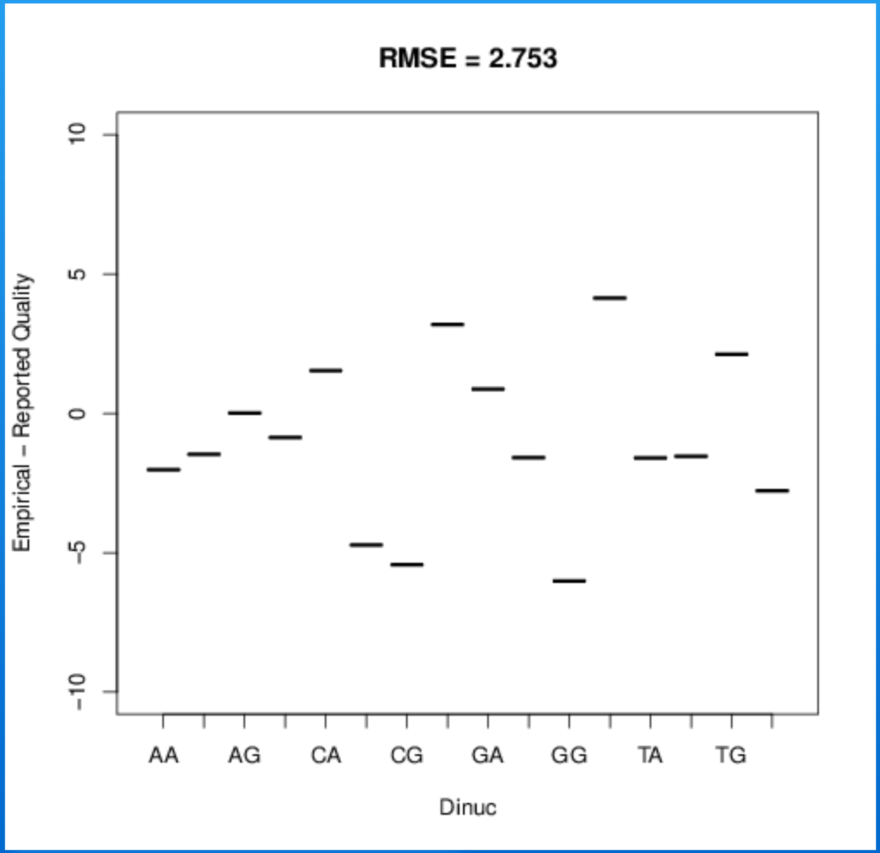
DinucCovariate - *Dinuc_hist*

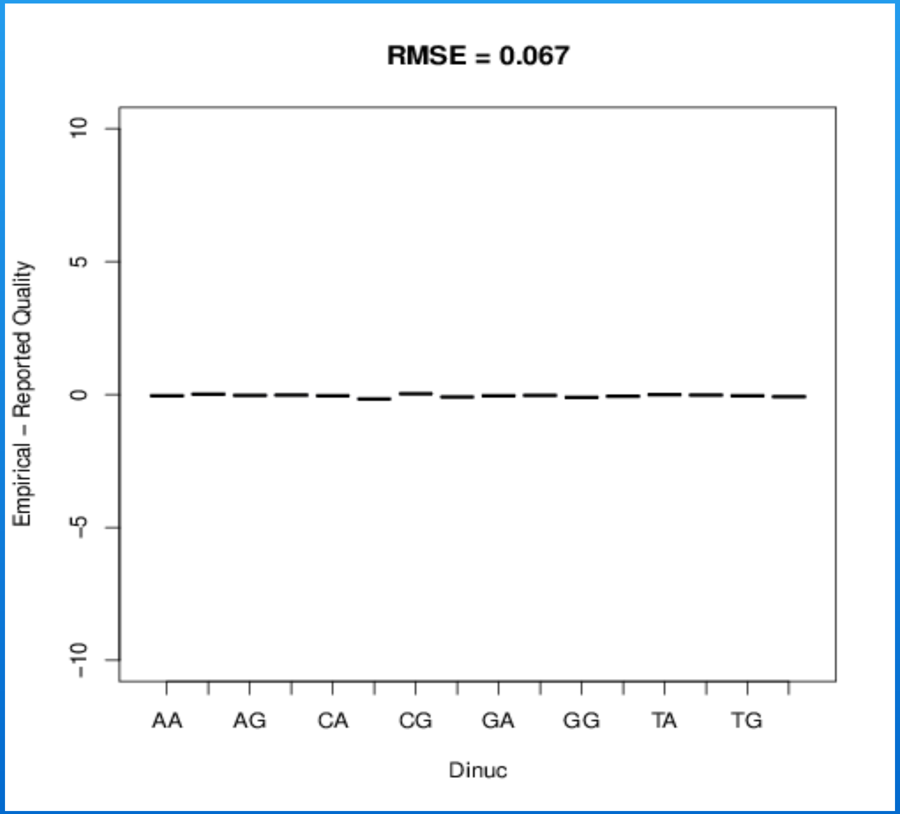




GATK - Base Quality Score Recalibration

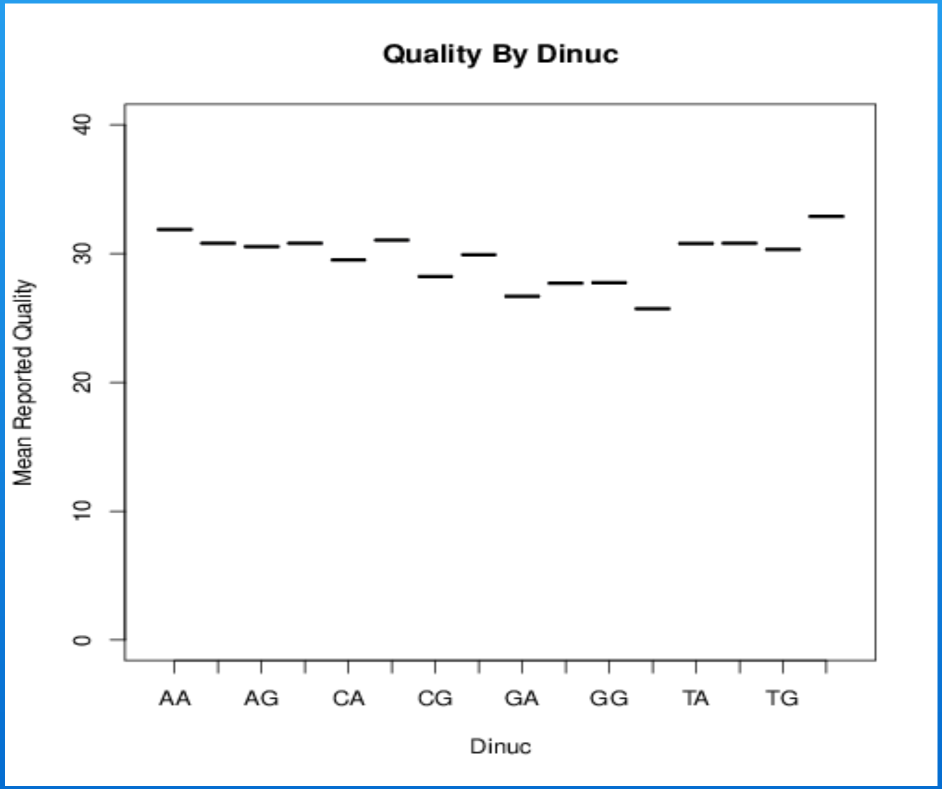
DinucCovariate - *qual_diff_v_Dinuc*

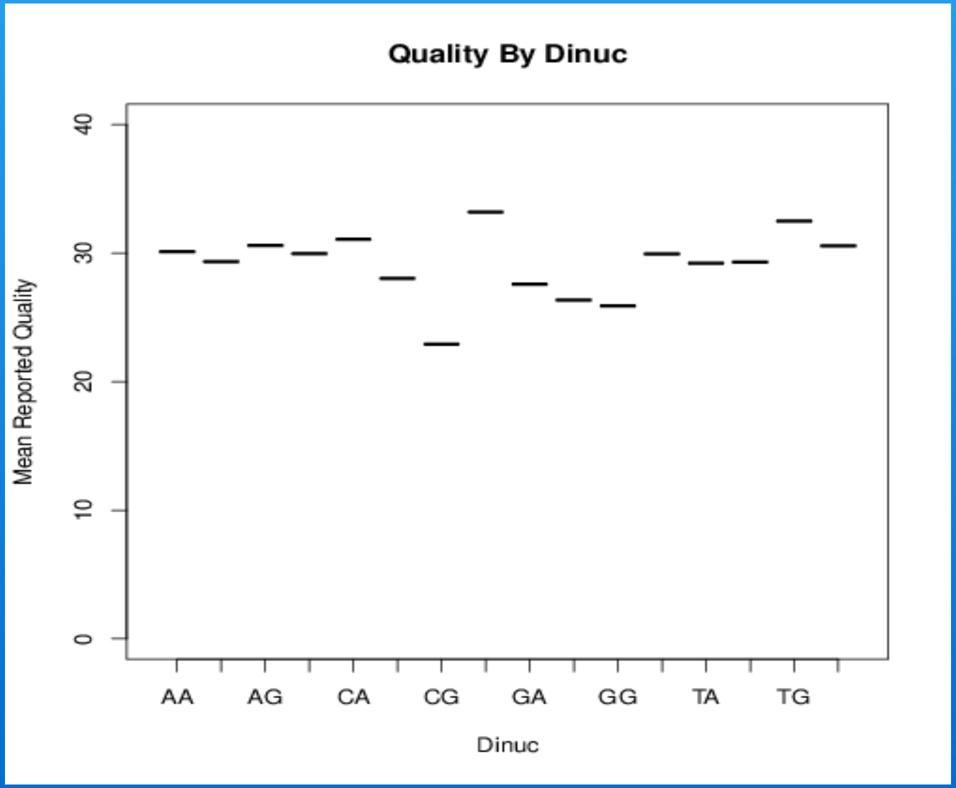




GATK - Base Quality Score Recalibration

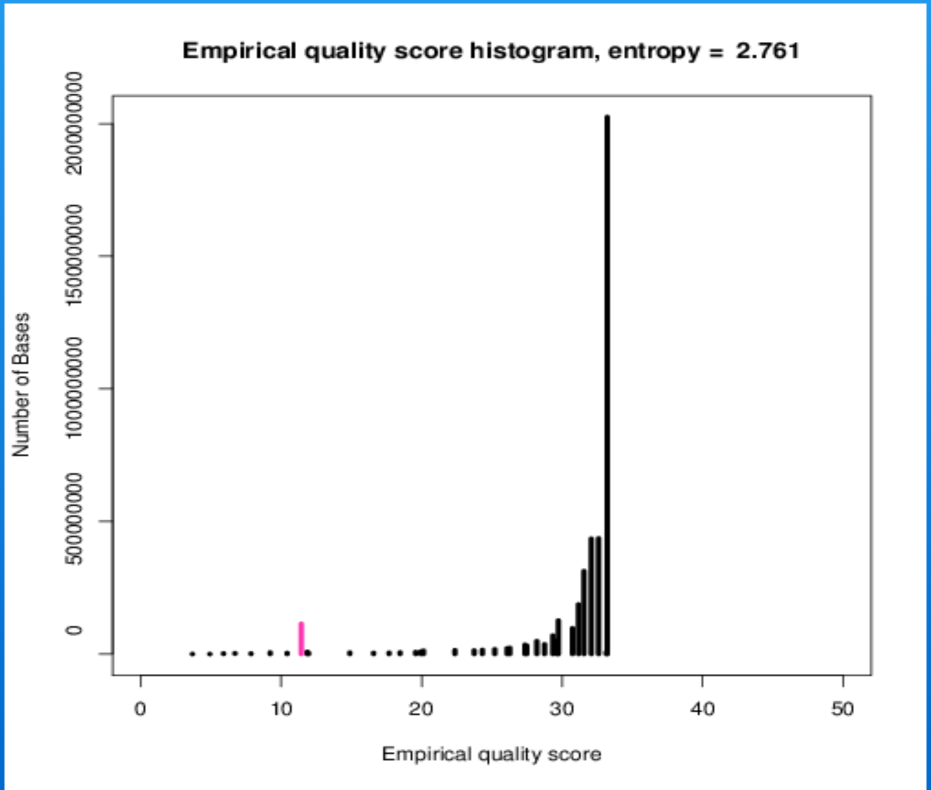
DinucCovariate - *reported_qual_v_Dinuc*

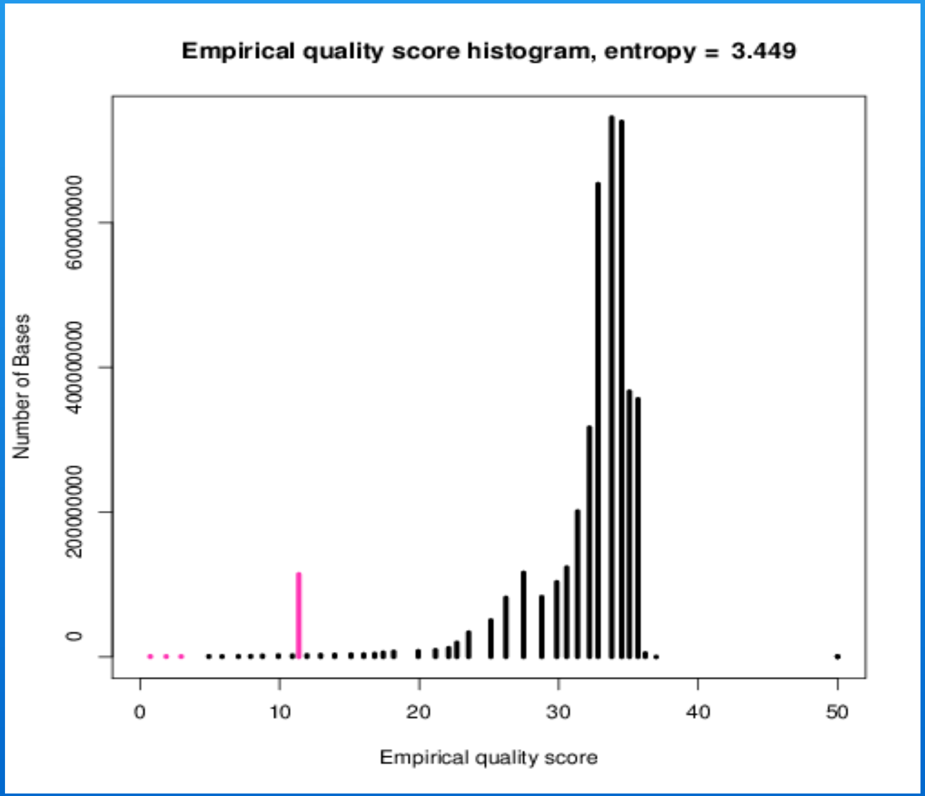




GATK - Base Quality Score Recalibration

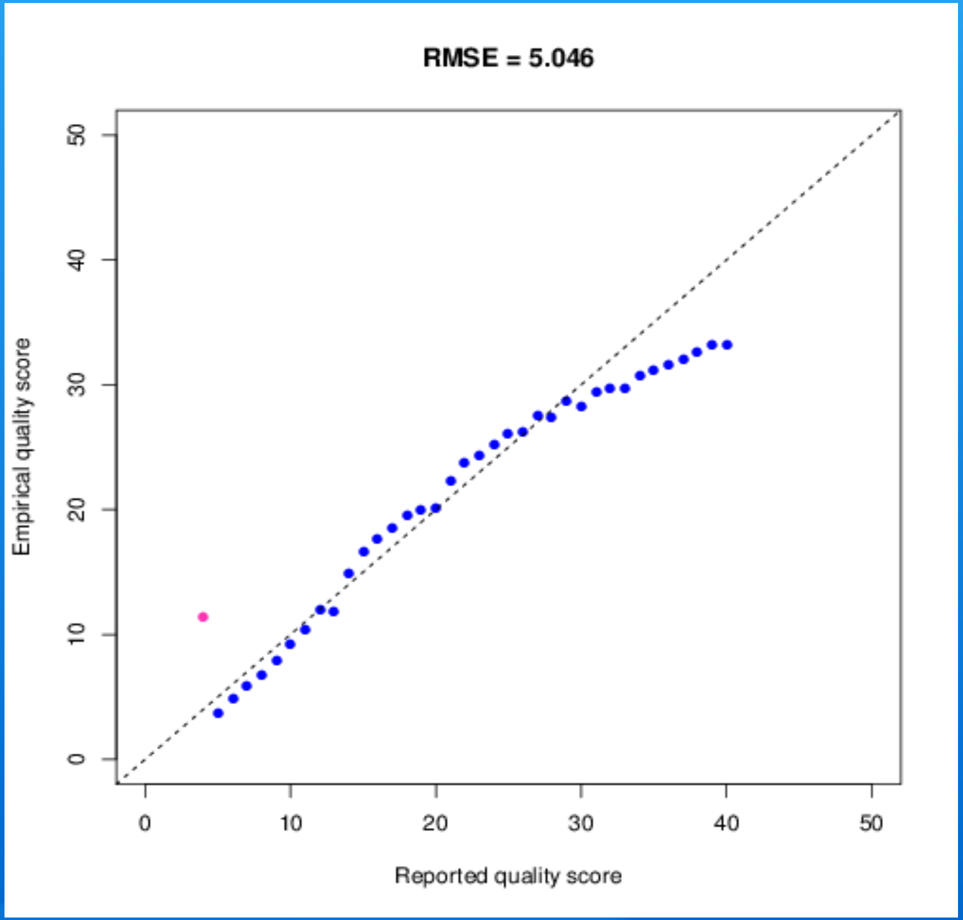
quality_emp_hist

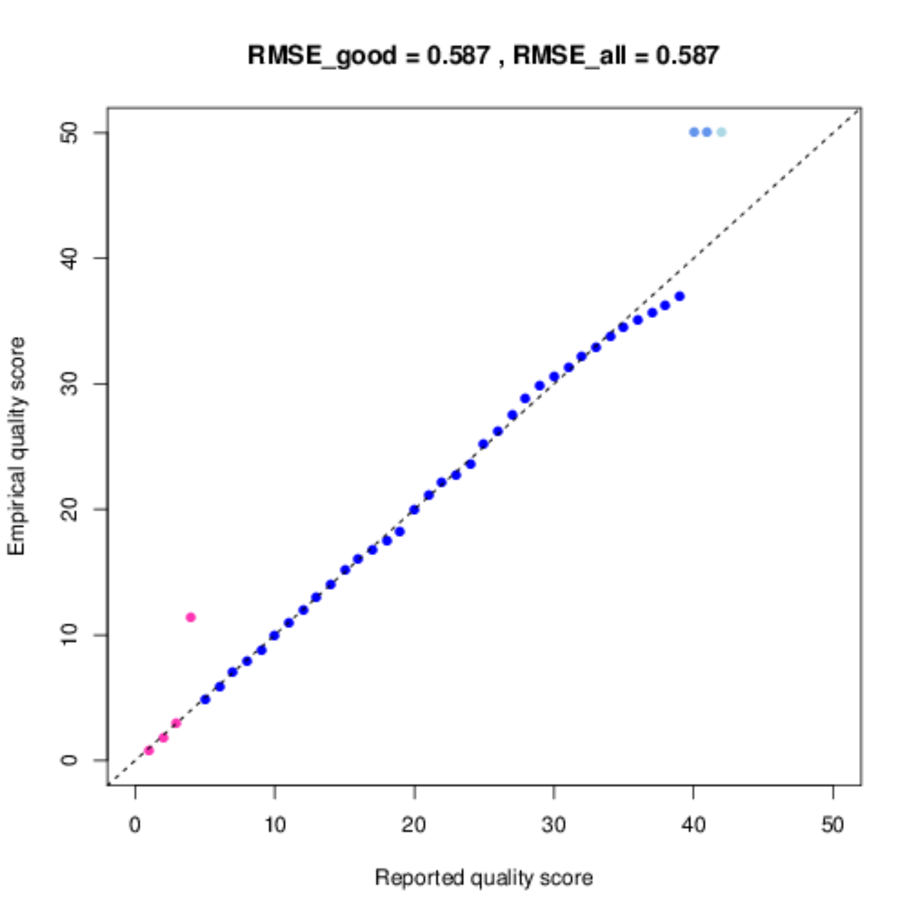




GATK - Base Quality Score Recalibration

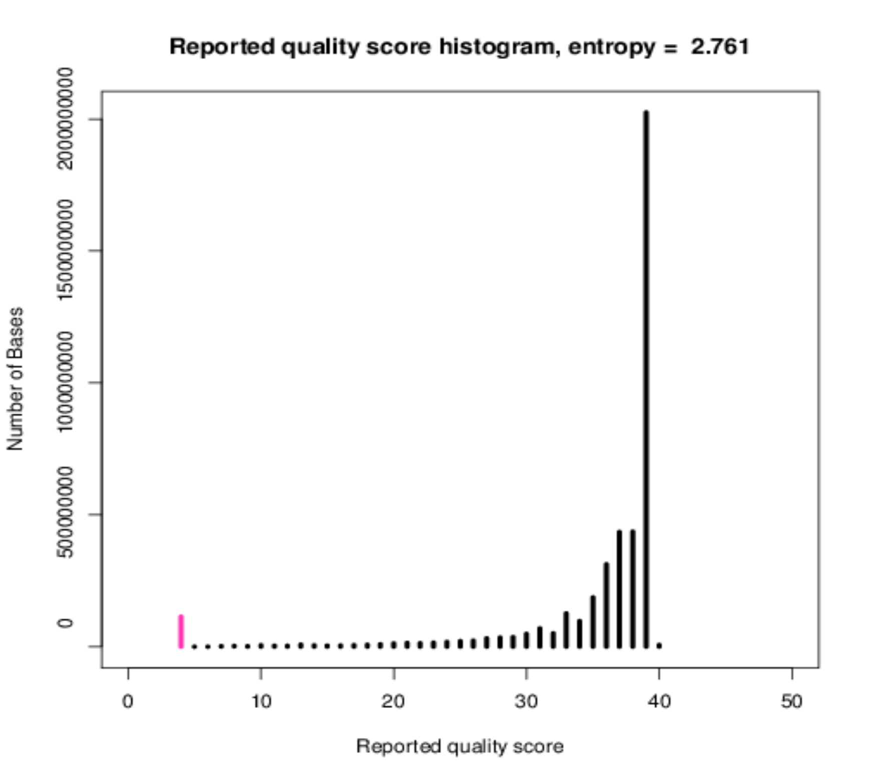
QualityScoreCovariate *quality_emp_v_stated*

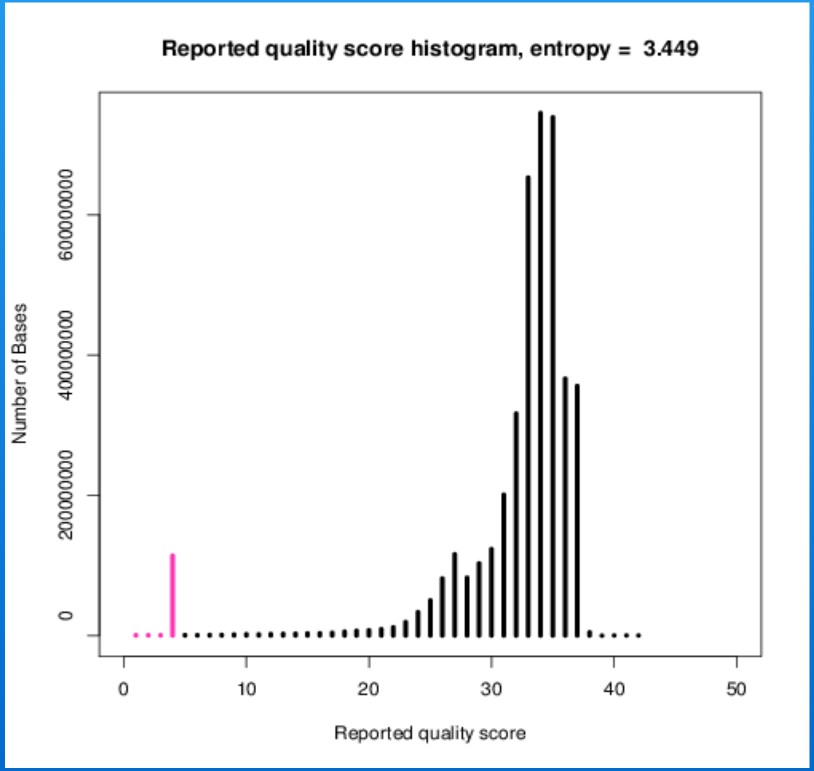




GATK - Base Quality Score Recalibration

QualityScoreCovariate *quality_rep_hist*





ToDo

Variant Quality Score Recalibration:

Training sets

HapMap 3.3: hapmap_3.3.b37.sites.vcf

These high quality sites are used both to train the Gaussian mixture model and then again when choosing a LOD threshold based on sensitivity to truth sites.

The parameters for these sites will be: known = false, training = true, truth = true, prior = Q15 (96.84%)

Omni 2.5M chip: 1000G_omni2.5.b37.sites.vcf

These polymorphic sites from the Omni genotyping array are used when training the model.

The parameters for these sites will be: known = false, training = true, truth = false, prior = Q12 (93.69%)

dbSNP build 132: dbsnp_132.b37.vcf

The dbsnp sites are generally considered to be not high quality enough to be used in training but here we stratify output metrics such as ti/tv ratio by presence in dbsnp (known sites) or not (novel sites).

The parameters for these sites will be: known = true, training = false, truth = false, prior = Q8 (84.15%)

Next Steps

Annovar e Vaast

Annovar

ANNOVAR is an efficient software tool to utilize update-to-date information to functionally annotate genetic variants detected from diverse genomes (including human genome hg18, hg19, as well as mouse, worm, fly, yeast and many others). Given a list of variants with chromosome, start position, end position, reference nucleotide and observed nucleotides, ANNOVAR can perform:

1. **Gene-based annotation**: identify whether SNPs or CNVs cause protein coding changes and the amino acids that are affected. Users can flexibly use RefSeq genes, UCSC genes, ENSEMBL genes, GENCODE genes, or many other gene definition systems.
2. **Region-based annotations:** identify variants in specific genomic regions, for example, conserved regions among 44 species, predicted transcription factor binding sites, segmental duplication regions, GWAS hits, database of genomic variants, DNAse I hypersensitivity sites, ENCODE H3K4Me1/H3K4Me3/H3K27Ac/CTCF sites, ChIP-Seq peaks, RNA-Seq peaks, or many other annotations on genomic intervals.
3. **Filter-based annotation:** identify variants that are reported in dbSNP, or identify the subset of common SNPs (MAF>1%) in the 1000 Genome Project, or identify subset of non-synonymous SNPs with SIFT score>0.05, or many other annotations on specific mutations.
4. **Other functionalities:** Retrieve the nucleotide sequence in any user-specific genomic positions in batch, identify a candidate gene list for Mendelian diseases from exome data, identify a list of SNPs from 1000 Genomes that are in strong LD with a GWAS hit, and many other creative utilities.

Vaast

VAAST (the Variant Annotation, Analysis and Search Tool) is a probabilistic search tool for identifying damaged genes and their disease-causing variants in personal genome sequences. VAAST builds upon existing amino acid substitution (AAS) and aggregative approaches to variant prioritization, combining elements of both into a single unified likelihood-framework that allows users to identify damaged genes and deleterious variants with greater accuracy, and in an easy-to-use fashion. VAAST can score both coding and non-coding variants, evaluating the cumulative impact of both types of variants simultaneously. VAAST can identify rare variants causing rare genetic diseases, and it can also use both rare and common variants to identify genes responsible for common diseases. VAAST thus has a much greater scope of use than any existing methodology.