KBBQ: A reference-free method for base quality score recalibration

1 Introduction

I. DNA sequencing and applications

   A. Next generation sequencing has a plethora of applications in biology.

   B. While the technology is widely applied, current technology is inherently erroneous and errors are common in sequencing data, with base substitution error rate estimates ranging between .1 and 1%, depending on the sequencing technology used.

   C. To help mitigate this, DNA molecules are usually copied and sequenced multiple times to ensure the sequence is correct, since multiple independent measurements are unlikely to all contain the same error at the same position. While this works well in general, it can be costly and additional sample manipulation can damage samples and insert mutations, further increasing the amount of technical error in the data.

1.1 Quality Scores

I. What are quality scores and why are they important? (Ewing, Hillier, *et al.* 1998) (Ewing & Green 1998)

   A. In addition to increased sequencing depth, quality scores are an important part of sequencing data that helps identify erroneous sequences. Sequencing data is usually presented along with quality scores in the Phred scale (Ewing, Hillier, *et al.* 1998; Ewing & Green 1998).

   B. These quality scores measure the confidence the instrument has in its determination that any particular base in the sequence is correct.

   C. Specifically, the quality score is equal to $-10\log_{10} P(e)$ (Ewing, Hillier, *et al.* 1998) (Ewing & Green 1998), where $P(e)$ is the probability the base is an error.

   D. For example, a base with a quality score of 40 has a .0001 probability of being incorrect while a base with a quality score of 10 has probability .1 of being incorrect. Generally, bases with scores lower than 10 are considered bad quality and bases with scores 30 and above are considered to be good quality. However, the score allows finer resolution than "good" and "bad", and is therefore more nuanced.

   E. The usefulness of quantitative scores over categorical can be illustrated by considering how variant calling algorithms utilize quality scores.

   F. Variant calling is a task to identify genetic variation in a sample from sequencing data.

   G. Variant calling models use quality scores to differentially weight the observed data. Generally, these models attempt to find the sample genotype most consistent with the observed data and recognize that low quality bases provide less reliable evidence for one genotype over another. The BCFTools multiallelic caller (H. Li, Handsaker, *et al.* 2009), GATK's HaplotypeCaller (Poplin *et al.* 2018), and FreeBayes (Garrison & Marth 2012)—three of the most popular variant callers—all take quality score into consideration when calling variants.

   H. Since the quality score is an exactly defined probability, it is straightforward to integrate these scores into a model.

   I. If so desired, quantitative scores can be collapsed into coarser categories, such as "good" and "bad". This is sometimes done as a heuristic; *ie.* scores less than 10 are untrustworthy and filtered out of a dataset and other scores are trusted and retained.

   J. The practice of binning quality scores together with neighboring scores is commonly performed to reduce file size.

   K. However, it's important to note that this process cannot be simply reversed as information is lost when the scores are binned.

1.2 Base Quality Score Recalibration

I. What is Base Quality Score Recalibration?

A. While base quality scores are important for identifying reliable data, base quality scores are often incorrect.

B. Base quality scores are exactly defined as a probability. They can be interpreted as a prediction giving the probability the reported base is an error. In general, probabilities are called *calibrated* if the reported probability accurately predicts the frequency of an event.

C. Base quality scores in Illumina sequencing reads are not well-calibrated (Callahan *et al.* 2016).

D. A few alternative base calling models have been developed for Illumina machines that improve base call accuracy and quality score calibration; however, these are difficult to use because they require the raw output from the sequencing machine, which is unavailable to most users of sequencing as they are usually disposed of after a sequencing run due to the large cost associated with storing that data.

E. Since base quality scores are used to some degree by most variant calling methods, it is probable that poorly calibrated reads reduce the quality of resulting variant calls.

F. Similarly, the reduced amount of information in binned quality scores may also impact the quality of variant calls made using the data.

G. Poor calibration combined with poor resolution resulting from quality score binning may have an important impact on algorithms that rely on these scores to function, but the exact affect of these phenomena are unknown.

H. Though increased sequencing depth can help mitigate the impact of random sequencing errors, sequencing is affected by non-random biases.

I. For these types of errors, increasing sequencing depth counterintuitively *increases* the effect of these errors, as they by definition occur preferentially at the same location. Thus, increased sequencing depth at that location adds more errors than are expected by chance, making those erroneous reads seem trustworthy.

J. These biases can be due to the nature of the DNA sequence itself, with errors induced during library preparation or the sequencing reaction likely due to secondary structure (Meacham *et al.* 2011; Nakamura *et al.* 2011).

K. At the same time, the sequencing reaction is also non-randomly biased. Bases at the end of a read are much more likely to be erroneous than bases at the beginning of the read, and the identity of the base and adjacent bases also affect the error rate.

L. Thus while random errors are troublesome, their impacts can be somewhat mitigated by more sequencing. Systematic errors cannot addressed in the same way, but they can be modeled.

II. GATK BQSR occurs in 3 phases.

A. Base quality score recalibration (BQSR) is the process of modeling errors in sequencing data and using the created model to update quality scores such that they reflect an accurate probability of error of any base.

B. GATK BQSR is the most popular method for BQSR and is recommended before variant calling by the GATK best practices.

C. The model integrates many covariates of error such that the output quality score reflects an accurate, independent measure of the probability of error.

D. The algorithm takes reads aligned to a reference and a database of potentially variable sites in the genome as input.

E. The algorithm proceeds in 3 phases.

F. In the first phase, the algorithm compares each read to the aligned reference. Potentially variable sites are ignored and mismatches from the reference sequence and non-mismatches are counted. The numbers of matching and mismatching bases are categorized according to the model covariates, which are: read group, assigned quality score, sequencing cycle, and the base identity along with the identity of the previous base (the base context).

G. In the second phase, a bayesian hierarchical model is trained with the count data. Using a normal distribution of the mean probability of error as a prior, the *maximum a posteriori* (MAP) quality score of the read group is calculated assuming the errors are binomially distributed. That is,

$$\begin{aligned}\hat{Q}_{rg} &= \text{argmax}_{Q_{rg}}\, P(Q_{rg}|Errors_{rg}) && (1)\\ &= \text{argmax}_{Q_{rg}}\, P(\mathcal{B}(Errors_{rg}|Observations_{rg}, Q_{rg}) * P(\mathcal{N}(Q_{rg}|\bar{Q})) && (2)\end{aligned}$$

H. This score is then used as the prior for calculating the *maximum a posteriori* quality score for each assigned quality score in that read group, $\hat{Q}_{assigned\,quality\,score}$, using a simular formula.

I. In turn, this score is used as the prior for calculating the *maximum a posteriori* scores for the sequencing cycle and context covariates of bases with that score. The difference between the MAP estimate and the prior is used to calculate the final score, which is the sum of the MAP estimate of the assigned quality score and these two differences. That is,

$$\begin{aligned}\Delta Q_{cycle} &= \hat{Q}_{cycle} - \hat{Q}_{assigned\,quality\,score} && (3)\\ \Delta Q_{context} &= \hat{Q}_{context} - \hat{Q}_{assigned\,quality\,score} && (4)\\ Q_{recalibrated} &= \hat{Q}_{assigned\,quality\,score} + \Delta Q_{cycle} + \Delta Q_{context} && (5)\end{aligned}$$

J. In the third phase, the quality score of each read is adjusted based on the four covariates for each base in the read according to values calculated in the previous phase.

III. How much does BQSR help?

A. BQSR improves quality score calibration in many cases, especially when there is moderate sequencing depth and the database of variable sites is nearly complete.

B. While BQSR is recommended in GATK's Best Practices, its affect on the resulting variant calls is not well-characterized, and there is ongoing debate about whether to continue recommending BQSR. (Van der Auwera 2020)

C. However, Ni & Stoneking 2016 find that improved quality score calibration aids detection of minor alleles in high coverage datasets by increasing sensitivity and reducing the number of false positive calls.

D. Thus, the need to perform BQSR and the performance of GATK's BQSR algorithm may vary from study to study.

E. One objective of this work is to help elucidate when GATK BQSR performs well and when it fails, and provide a method that works in situations that GATK BQSR struggles with.

F. As an example, the GATK developers recommend BQSR for use in cancer variant discovery (Cibulskis *et al.* 2013), but the tumor genome is likely much different from the human genome, and the database of variable sites will likely miss many truly variable sites due to the large mutation rate present in cancer genomes. Additionally, mismatches in reads misaligned due to chromosomal rearrangements may be mistakenly counted as evidence of sequencing error. The number of these errors required to significantly impact the performance of the algorithm is not clear.

1.3 Alternative Approaches for BQSR

I. What is the problem with current methods for BQSR?

    A. As illustrated above, the most problematic aspect of GATK BQSR occurs in the first phase of the algorithm: counting erroneous and non-erroneous bases. This is especially true when considering non-model organisms, where alignment errors may be common. Furthermore, in non-model organisms a database of variable sites is likely unavailable or largely incomplete. However, there are methods that attempt to overcome this deficiency. While there exist alternative approaches that implement different error models, such as Lacer (Chung & Chen 2017), the biggest problem for analyzing data without reliable reference information is the method of counting erroneous and non-erroneous bases.

II. Alternative Approaches

    A. Many alternative algorithms have been developed to avoid providing a database of variable sites. However, these approaches all still require a reference and alignment, and many require extra reagents and sequencing that increases the cost of analysis and cannot be used to reanalyze existing sequencing data that hasn't been specially prepared.

    B. ReQON (Cabanski *et al.* 2012), like GATK, considers bases that do not match the reference as errors but limits the number of acceptable errors at a position to minimize the effect of unknown variants. It then uses a logistic regression to recalibrate the quality scores.

    C. SOAP2 (R. Li *et al.* 2009) contains a model for consensus sequence construction that performs BQSR during construction.

    D. The methods of Zook *et al.* 2012 and Ni & Stoneking 2016 use synthetic spike-ins of known composition and GATK's model Zook *et al.* 2012 or piecewise regression (Ni & Stoneking 2016) to recalibrate quality scores. Since the sequence of the spike-in is known before hand, errors are easy to identify as there should be no biological variation in the spiked-in sample.

    E. Crucially, these methods require a reference and possibly other information to recalibrate reads that may not be available.

III. Here we present `kbbq`, a software package to recalibrate quality scores of whole genome sequencing data without a reference or database of variable sites. The only required input is the set of reads to be recalibrated. Rather than excluding variation and comparing to the reference like GATK does, `kbbq` uses k-mer subsampling to find likely errors. Once the number of errors and nonerrors are counted and categorized according to their covariates, it uses the same model GATK uses to recalibrate the reads. I show how simulated false negatives and false positives affect GATK's ability to recalibrate reads and compare GATK calibration to `kbbq`.

2 Materials and Methods

I. `kbbq` performs BQSR by adjusting how errors in the dataset are discovered.

    A. Instead of looking at the reference, `kbbq` implements the error-correction algorithm described in Song *et al.* 2014 and uses the errors detected by that procedure to train and apply the standard GATK model. Note that the sequenced bases are not actually changed; the detected errors are used only to train the model. If there is evidence according to the model that the base is erroneous, its base quality will be decreased according to the strength of that evidence.

    B. Briefly, the algorithm subsamples k-mers from the dataset. Since erroneous k-mers are expected to be unique, erroneous k-mers are less likely to be sampled than error-free k-mers. A binomial test is then conducted for every nucleotide in the dataset; if a sufficient number of k-mers contain the nucleotide, the base is likely not erroneous and called trusted. If k of these trusted bases appear next to each other, that k-mer is stored as a trusted k-mer. Once the trusted k-mers have all been stored, each read is iterated through once again and any bases on the edge of an island of trusted k-mers are changed such that the change produces the maximal number of trusted k-mers in the read. These changes are marked as errors and used to train the model.

    C. The model training and recalibration procedure are the same as those described above for GATK's BQSR method.

2.1 Program Input and Parameters

I. `kbbq` requires as input a set of reads in FASTQ (Cock *et al.* 2010) or BAM (H. Li, Handsaker, *et al.* 2009) format. If the input data is FASTQ formatted and consists of multiple read groups, the read group each read belongs to should be annotated in the name of each read. Alternatively, if the user has an original data file and a data file that has already been corrected using an error correction program, the user may supply the corrected file with the `--fixed` option to obtain errors from that correction rather than performing the error correction algorithm included in `kbbq`. The program's only required parameter is the approximate length of the sequenced region in base-pairs, and this information can be taken from the BAM header if it is present. It may be set with the `--genomelen` option. If BAM input is provided, the `--set-oq` and `--use-oq` flags can be used to set the OQ flag on the read before recalibrating or to use the quality scores encoded in the OQ flag rather than the quality scores in the primary quality score field.

II. Optionally, the approximate sequencing coverage may also be provided with the `--coverage` option. If it is not, it will be estimated by finding the length of the sequenced data divided by the provided genome length; $\text{Coverage} = \frac{\text{Sequence Length}}{\text{Genome Length}}$.

III. An α parameter may also be provided with the `--alpha` option; this is the same α parameter that Lighter uses, and is the fraction of reads sampled from the input data (Song *et al.* 2014). If not provided, the recommended value of $\alpha = \frac{7}{\text{Coverage}}$ is used.

IV. A `-k` parameter may also be provided, which changes the k-mer size used for the error detection algorithm. The maximum value of 32 is recommended and is the default value.

V. A summary of flags and options the program supports is listed in table 1.

VI. The genome length parameter, in addition to being used to estimate sequencing coverage, is also used to estimate the number of k-mers that will be sampled. This is used to parameterize the bloom filter that stores the sampled and trusted k-mers. This parameterization is different than that used in Lighter; there, the number of sampled k-mers is estimated to be $1.5 \times \text{Genome length}$. However, the expected number of sampled k-mers K_{sampled} is bounded by the expected value of the binomial distribution parameterized by $\text{Genome length} \times \text{Coverage}$ and α. Assuming *every* k-mer is unique, the number of possible k-mers in the dataset K_{total} is less than $\text{Coverage} \times \text{Genome length}$. Since the number of k-mers in each read is $\text{Read length} - k + 1$ and assuming equal read lengths, the total number of k-mers is:

$$\begin{aligned}
K_{\text{total}} &= \sum_{\text{Reads}} \text{Read length} - k + 1 && (6)\\
&= (\text{Read length} - k + 1) \times \text{Number of reads} && (7)\\
&< \text{Read length} \times \text{Number of reads} && (8)\\
&< \text{Read length} \times \frac{\text{Coverage} \times \text{Genome length}}{\text{Read length}} && (9)\\
&< \text{Coverage} \times \text{Genome length} && (10)
\end{aligned}$$

VII. So the expected number of sampled k-mers is

$$\begin{aligned}
E[K_{\text{sampled}}] &= E[\mathcal{B}(x; \alpha, K_{\text{total}})] && (11)\\
&= \alpha \times K_{\text{total}} && (12)\\
&< \alpha \times \text{Coverage} \times \text{Genome length} && (13)
\end{aligned}$$

VIII. Notably, if α is the recommended value of $\frac{7}{\text{Coverage}}$, the expected number of sampled k-mers is less than $7 \times \text{Genome length}$, a bound over 4 times larger than the estimate used by Lighter. For the data analyzed here, this provides a much better estimate of the true number of sampled k-mers. Ultimately,

Parameter	Short Option	Default Value	Summary
`--ksize`	`-k`	32	Size of k-mer to use for correction
`--use-oq`	`-u`	Off	Use BAM OQ tag values as quality scores
`--set-oq`	`-s`	Off	Set BAM OQ tag values before recalibration
`--genomelen`	`-g`	Estimated for BAM, required for FASTQ	The approximate size of the sequenced region in base-pairs.
`--coverage`	`-c`	Estimated from data	Approximate sequencing coverage
`--fixed`	`-f`	Off	Treat changes to reads in the given file as errors and recalibrate.
`--alpha`	`-a`	7 / coverage	Rate to sample k-mers

Table 1: `kbbq` parameters. Provides the short options for each long parameter name, the default value of the parameter and a summary of how each parameter changes the behavior of the program.

the larger estimate of elements inserted into the bloom filter causes an increase in size of the bloom filter but a smaller false positive rate.

2.2 Testing and Validation

I. Describe test dataset used for benchmarking.

   A. To test the performance of `kbbq`, I reanalyzed the synthetic diploid CHM1-CHM13 dataset from H. Li, Bloom, *et al.* 2018. Like other benchmarking datasets, this dataset includes a BED file describing confident regions in which the genotype of any site that differs from homozygous reference within those regions are included in a VCF file. For the purposes of this work, I assume these confident regions and associated VCF entries are correct and represent the true genotype of the sequenced sample.

   B. This dataset was specifically designed to study the impact of deep sequencing on variant calling and has a coverage of approximately 45x. Thus, the affect of sequence specific errors and other non-random biases in sequencing should be pronounced in this data.

   C. The dataset was constructed by adding equal concentrations of DNA of CHM1 and CHM13 human complete hydatidiform mole cell lines and sequencing the mixture. These moles are formed when a sperm combines with an egg containing no nucleus; the sperm then undergoes mitosis to generate a completely homozygous cell mass. They are effectively haploid, and it is significantly easier to genotype a haploid cell than a diploid one. Thus, the mixture simulates a diploid human cell but the genotype of each "haplotype" is known. This means there should be very few, if any, errors in the declared genotypes included with the data. This is what H. Li, Bloom, *et al.* 2018 find when validating their data as well.

   D. To measure the performance of `kbbq` and compare to GATK's BaseRecalibrator and ApplyBQSR tools, I subset the full dataset to only reads aligned on Chromosome 1 and overlapping the BED of confident regions using the samtools view command (H. Li, Handsaker, *et al.* 2009). I then used the samtools fixmate and view commands to remove any singleton reads. I then ran `kbbq` on the dataset with the options `--use-oq -g 214206308 -a .15`. I also ran GATK's BaseRecalibrator with the provided variant data as the known sites file and the `--use-original-qualities` flag set. I then used the ApplyBQSR tool with the `--use-original-qualities` flag to recalibrate the input file. I then compared the two recalibrated files by running the BaseRecalibrator tool again with the same options on both output datasets and using GATK's AnalyzeCovariates tool.

   E. In order to determine how misspecification of the database of variable sites affects the calibration of GATK's BQSR procedure, I simulated datasets with various levels of false negative and false positive variants. In this case, false negative variants in the database of variable sites causes a site that should be ignored to not be ignored, greatly increasing the number of bases that GATK

False Positive Rate	RMSE
0	0.60
20	0.58
40	0.53
60	0.49
80	0.49
100	5.50

Table 2: The root mean squared error of quality score for reads recalibrated using a database of variable sites with different false positive rates. The false negative rate for each dataset is 0%.

classifies as sequencing errors. On the other hand, false positive variants remove a site from GATK classification, so the affect is likely to be small except for large false positive rates. To create each false negative dataset, an appropriate number of sites from the VCF were randomly sampled with BCFTools and the shuf program. To create each false positive dataset, all sites from the VCF were extracted in BED format with the BCFTools query command, then subtracted from the BED of confident regions with bedtools subtract (Quinlan & Hall 2010). The appropriate number of sites were then sampled with the shuf program and appended to the sites from the VCF to generate the BED file of all sites to exclude. These files were then provided as input to GATK's BaseRecalibrator tool and calibration was evaluated with GATK's AnalyzeCovariates tool.

F. GATK method test

1. To simulate a situation where a researcher has sequenced a non-model organism, is using a reference that may not closely match the sample, and doesn't have a database of variable sites, I aligned the test data to the chimp reference genome (Waterson *et al.* 2005) usign NextGenMap (Sedlazeck *et al.* 2013).
2. In this situation, GATK recommends calling an initial set variants with high confidence and using these variants as the database of variable sites.
3. To see how this affects the results of GATK's BQSR, I did so using HaplotypeCaller with the `-stand-call-conf 50` argument. The default value for this argument is 40, and is a phred-scaled confidence threshold for reporting a variant. I then used the resulting variants with BaseRecalibrator to train a recalibration model. To evaluate this model using the truth set, I used the model to recalibrate the reads as they were aligned to the human reference. Thus, the sole difference between this recalibration method and the standard using the correct reference and database of variable sites is the trained model; the realignment has no impact on benchmarking the calibration. I then used AnalyzeCovariates as above to obtain the calibration data.

3 Results

I. To identify how errors in the database of variable sites affects calibration, I simulated different datasets with known false positive rates, plotted the calibration and calculated the root mean squared error (RMSE) of the data recalibrated with the model trained using each database as the known sites input to GATK BaseRecalibrator. These plots are shown in figure 1, and the RMSE of the quality score for each dataset is shown in table 2. For all these datasets, the false negative rate is 0. As the false positive rate increases, the degree of miscalibration doesn't change significantly except for the 100% false positive rate dataset, which is very poorly calibrated; however, this calibration is likely an artifact (see section 4).

I. I also simulated different databases of variable sites with differing false negative rates and similarly used it to recalibrate my dataset. In these datasets, the false positive rate is 0%. The plotted calibration and RMSE of the recalibrated data is shown in figure 2 and table 3. As the false negative rate increases, the degree of miscalibration also steadily increases.

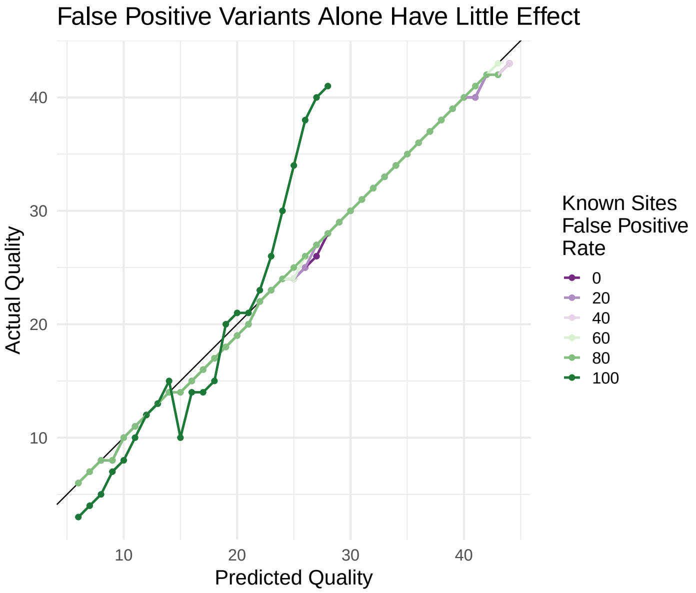

Figure 1: Base quality score calibration for a range of false positives in the database of variable sites. The false negative rate for all datasets is zero. Increasing the false positive rate does not significantly impact the quality of the calibration. The poor calibration at a 100% false negative rate is likely an artifact (see Section 4).

False Negative Rate	RMSE
0	0.60
20	1.32
40	2.10
60	2.57
80	2.90
100	3.20

Table 3: Root mean squared error of quality score for reads recalibrated with a database of variable sites simulated with the given false negative rate. The false positive rate for each dataset is 0%.

I. To see if there were any interactive effects of false positive rate and false negative rate, I also simulated datasets with varying false positive and false negative rates. The RMSE of the calibrated scores is reported in Table 4 and summarized in Figure 3. These datasets were simulated separately from the above datasets, so there are slight differences in the calibration of the resulting reads. As before, the number of false negatives significantly impacted calibration quality. In contrast to the false positive only dataset with a 0% false negative rate, increasing the false positive rate also increased the amount of error in the calibration. Thus, false positives in the database of variable sites seem to enhance miscalibration caused by false negatives.

I. Using GATK's recommended method of using high-confidence variants when a database of variable sites is unavailable produced poorly calibrated data with a RMSE of 1.66. This RMSE is similar to the RMSE of a simulated dataset with a false negative rate of approximately 20%and a false positive rate between 0 and 20%. However, the shape of this recalibrated data is also interesting; it seems that the major effect of recalibration was squeezing high quality scores toward the mean, as the highest recalibrated score in the dataset is 34 whereas using the truth set of variants yielded a highest recalibrated score of 44. This calibration and the calibration using the truth set, using KBBQ, and the raw read quality are plotted in Figure 5 and the RMSEs of each data set is listed in Table 5

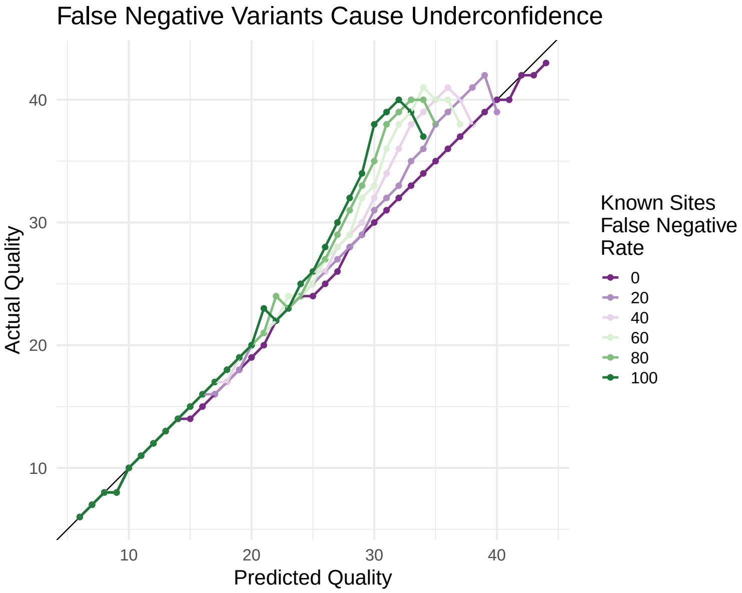

Figure 2: Base quality score calibration for a range of false negatives in the database of variable sites. The false positive rate for all datasets is zero. Increasing the false negative rate significantly decreases the quality of the calibration, causing increasing underconfidence in quality scores as the false negative rate rises.

FPR	FNR					
	0	20	40	60	80	100
0	.641	1.45	2.08	2.73	2.99	3.38
20	.599	1.73	2.54	2.96	3.39	3.76
40	.555	2.02	2.71	3.50	3.84	4.16
60	.531	2.58	3.37	4.27	4.51	4.73
80	.593	3.52	4.62	5.38	5.73	6.32
100	8.05	22.0	24.3	26.4	25.8	28.9

Table 4: Root mean squared error of base quality score for data calibrated with databases of variable sites containing different levels of false positives and false negatives. The columns indicate false positive rates, the rows indicate false negative rates. The values in each cell are the RMSE of the quality scores for the reads recalibrated with the database of variable sites with false positive and false negative rate appropriate for its row and column. See Figure 4 for a graphical representation. The data in the 100% false positive rows are likely artifacts; see section 4 for more information.

Calibration Method	RMSE
Chimp-GATK	1.66
GATK	0.60
KBBQ	0.96
Raw	4.05

Table 5: Root mean squared error of quality score for reads recalibrated using different methods. Chimp-GATK is the result of calibrating the reads using the model trained on the reads aligned to the chimp genome along with the variants called using that alignment. GATK is the result of using GATK's BaseRecalibrator with the truth set of variants. KBBQ is the result of using the KBBQ tool, which requires only reads and no reference or variant set. Raw is the calibration of the uncalibrated data.

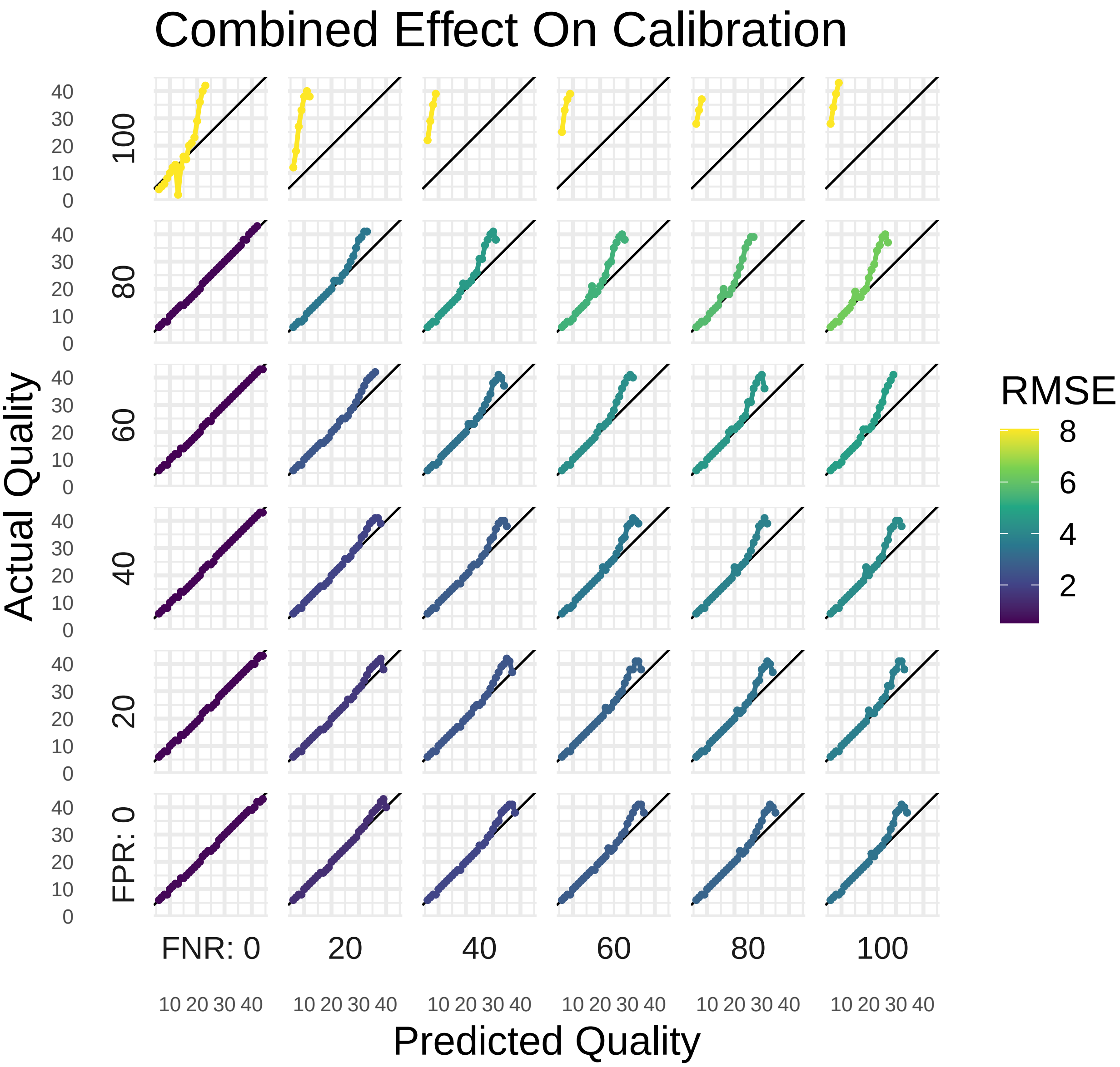

Figure 3: Calibration of base quality scores of reads recalibrated using a database of variable sites with varying ranges of false negatives and false positives. The RMSE of the quality scores of the resulting calibration are used to color each line. Across the columns are each false negative rate, and each row represents a false positive rate. Except for at a false negative rate of 0, increasing either the false positive rate or the false negative rate increases the RMSE. See Table 4 for the RMSE values and the discussion in Section 4 about false positive rates of 100%

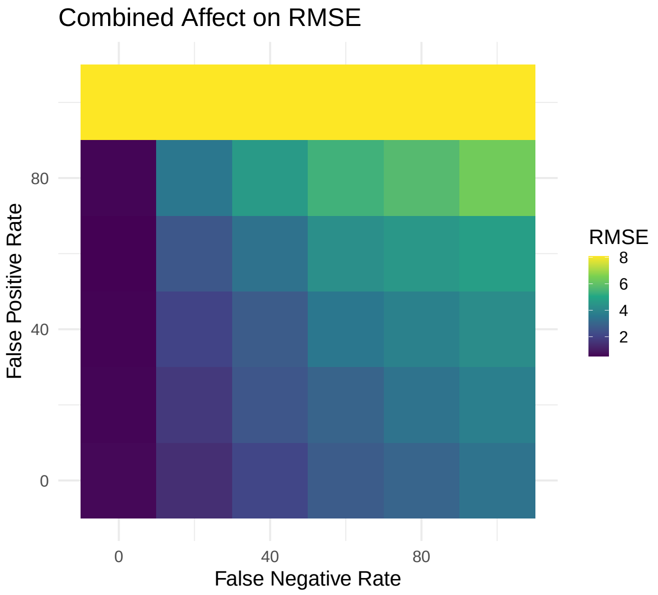

Figure 4: Heat map of base quality calibration of varying ranges of false negatives and false positives. Databases of variable sites with differing false positive and false negative rates were constructed and the RMSE of the quality scores of the resulting calibration were calculated. Across the columns are each false negative rate, and each row represents a false positive rate. Except for at a false negative rate of 0, increasing either the false positive rate or the false negative rate increases the RMSE. See Table 4 for the RMSE values and the discussion in Section 4 about false positive rates of 100%

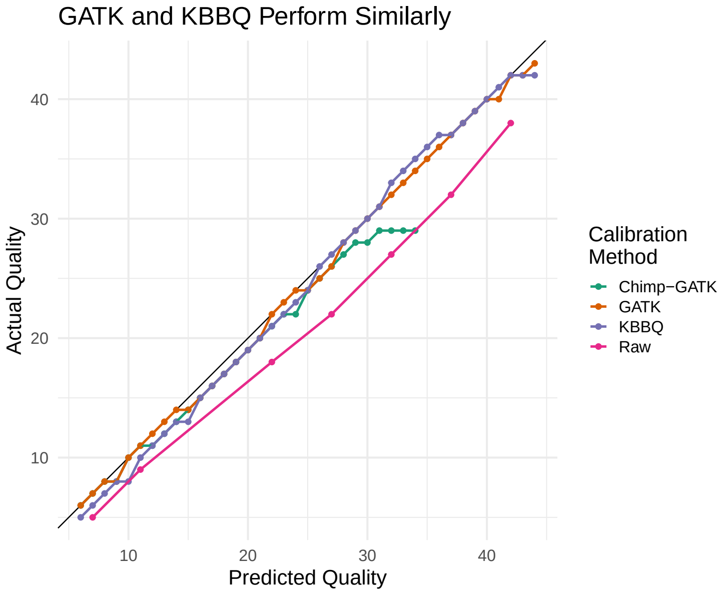

Figure 5: Comparison of calibration methods. Chimp-GATK is the result of calibrating the reads using the model trained on the reads aligned to the chimp genome along with the variants called using that alignment. GATK is the result of using GATK's BaseRecalibrator with the truth set of variants. KBBQ is the result of using the KBBQ tool, which requires only reads and no reference or variant set. Raw is the calibration of the uncalibrated data.

4 Discussion

I. These results show that GATK BaseRecalibrator is particularly vulnerable to false negatives (Figure 2) in the database of variable sites, but is robust to false positives if the false negative rate is near 0 (Figure 1). At the same time, when the false negative rate is not near 0, false positives will start to impact the calibration quality (Figure 3). Thus, when this database is unavailable and construction is required, it may be better to be liberal in deciding which sites may be variable to reduce the false negative rate as much as possible. This is in contrast to the GATK recommendation to use only the most confident sites when a database is unavailable.

II. The only rate that shows significant deviation from a 0% false positive rate in the false-positive-only data is the 100% false positive rate. Though a 100% false positive rate with a 0% false negative rate implies every site should be considered variable and ignored, the model curiously still has a source of errors it uses to recalibrate. Upon further investigation, this effect is driven by reads with alignments that begin with an insertion, as these inserted bases are not ignored by BaseRecalibrator when the first position of the site that should be ignored is equal to the first aligned position in the read. Thus, this line is a technical artifact. So long as there are enough bases available to analyze, the false positive rate doesn't significantly affect the performace of BaseRecalibrator at a false negative rate of 0. The calibrations of other datasets with a false positive rate of 100% would also be affected by this artifact; however, in a real dataset it's unlikely to ever achieve a false positive rate of 100%, so this artifact is unlikely to significantly affect real data.

III. Ultimately, if the false negative and false positive rates of the database of variable sites is high, GATK's procedure for BQSR *can* cause miscalibration of the data worse than using raw quality scores, though this can only happen with very large error rates. In this dataset, the raw data has a RMSE of about 4, which is similar to a simulated dataset with a false negative rate between 40-60% and a false positive rate between 60-80%. In a real situation it's unlikely that error rates like this will occur, so BQSR will not severely destroy the input data. However, at even modest false negative and false positive error rates BQSR can cause undesirable miscalibration that could feasibly impact variants called from the data. Interestingly, at all error rates the calibration of quality scores below 25 is almost always correct or 1-off the true value. It seems that errors in the database of variable sites have a larger effect on higher quality predictions than lower ones.

IV. At the same time, the calibration model trained with the chimp-aligned data shows performance worse than a simulated dataset with a false negative rate of approximately 20%. While the RMSE of 1.66 is not particularly high, the plotted calibration shows very poor performance at higher quality scores, and the maximum quality score the model assigned was 34 in contrast to other methods, which had a maximum assigned score of 44. Additionally, all quality scores assigned above 30 were more than one off the true quality score. And while all the simulated datasets caused underconfidence in the calibration, this calibration method caused overconfidence.

V. This suggests that an effect not captured in the simulated data was present in this data. Overconfidence indicates errors that should be counted by the model are instead missed and therefore there are more errors in actuality than the model predicts. Thus the set of variants used to skip errors is too aggressive and contains many false positives. So why is a similar effect not observed in the simulated data? This is likely because of how the simulated datasets were constructed; non-variable sites were selected independently and at random to be added to the set of purported variable sites. Thus a site containing an error and a site not containing an error are selected proportionally. In contrast, on a real dataset a variant calling algorithm would not make false positive errors independently; it is much more likely to classify a site as variable that has a sequencing error than to classify a site as variable that has no errors. That is: in this simulation, $P(\text{classified positive} \mid \text{actual negative})$ is independent of $P(\text{sequencing error})$, but on an empirical dataset $P(\text{classified positive} \mid \text{actual negative})$ is likely not. Thus in reality false positives probably cause overconfidence in a similar manner that false negatives cause underconfidence. This overconfidence is observed in this calibration method.

VI. As the chimp-aligned calibration shows, GATK BQSR struggles on real data from non-model organisms. False negatives are likely to be numerous in almost all but the most well-studied samples (Bobo *et* *al.* 2016), and false positives are similarly likely when using poor quality, draft reference genomes that cause alignment errors. This means in most datasets, while its performance may be acceptable if the false negative and false positive rate are sufficiently low, GATK BaseRecalibrator is not the best recalibration method. This is shown in Figure 5, which shows `kbbq` performing nearly as well as GATK BQSR with perfect knowledge of variable sites. However, `kbbq` doesn't use any reference or any variant site information to do recalibration. `kbbq` also performs much better than GATK's recommended procedure when a database of variable sites is not available, as shown in the Chimp-GATK calibration.

5 Conclusion

I. Base quality score recalibration is an important procedure to ensure base quality scores are accurate before variant calling. However, the most popular method for doing BQSR is not easy to do if the sequenced organism is a non-model organism. I simulated sets of variable sites with varying false positive and false negative rates to use with BQSR. While it seems the simulated false positives are somewhat different from those that appear in real datasets, it is clear that false negatives severely reduce the quality of the calibration using GATK's method. I developed the software tool `kbbq` to recalibrate base quality scores without a reference or database of variable sites to overcome these deficiencies. Since it doesn't use a database of variable sites or a reference, the quality of these resources is immaterial to the quality of the resulting calibration. Finally, I emulated GATK's procedure for calibration when a database of variable sites is unavailable by aligning benchmark data to the chimp genome and calling variants to use as the database of variable sites. This method produces a calibration much worse than `kbbq`. Thus, when a database of variable sites or reference is unavailable or of poor quality, `kbbq` is an effective method for base quality score recalibration.

References

1. Bobo, D., Lipatov, M., Rodriguez-Flores, J. L., Auton, A. & Henn, B. M. *False Negatives Are a Significant Feature of Next Generation Sequencing Callsets* preprint (Bioinformatics, July 26, 2016). `http://biorxiv.org/lookup/doi/10.1101/066043` (2020).
2. Cabanski, C. R. *et al.* ReQON: a Bioconductor package for recalibrating quality scores from next-generation sequencing data. *BMC bioinformatics* **13,** 221. ISSN: 1471-2105 (Sept. 4, 2012).
3. Callahan, B. J. *et al.* DADA2: High-resolution sample inference from Illumina amplicon data. *Nature Methods* **13,** 581–583. ISSN: 1548-7105. `https://www.nature.com.ezproxy1.lib.asu.edu/articles/nmeth.3869` (2019) (July 2016).
4. Chung, J. C. S. & Chen, S. L. Lacer: accurate base quality score recalibration for improving variant calling from next-generation sequencing data in any organism. *bioRxiv,* 130732. `https://www.biorxiv.org/content/10.1101/130732v2` (2019) (Apr. 27, 2017).
5. Cibulskis, K. *et al.* Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature Biotechnology* **31,** 213–219. ISSN: 1546-1696. `https://www.nature.com/articles/nbt.2514` (2019) (Mar. 2013).
6. Cock, P. J. A., Fields, C. J., Goto, N., Heuer, M. L. & Rice, P. M. The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic Acids Research* **38,** 1767–1771. ISSN: 1362-4962 (Apr. 2010).
7. Ewing, B., Hillier, L., Wendl, M. C. & Green, P. Base-calling of automated sequencer traces using phred. I. Accuracy assessment. *Genome Research* **8,** 175–185. ISSN: 1088-9051 (Mar. 1998).
8. Ewing, B. & Green, P. Base-Calling of Automated Sequencer Traces Using Phred. II. Error Probabilities. *Genome Research* **8,** 186–194. ISSN: 1088-9051, 1549-5469. `http://genome.cshlp.org/content/8/3/186` (2019) (Mar. 1, 1998).
9. Garrison, E. & Marth, G. Haplotype-based variant detection from short-read sequencing. *arXiv:1207.3907 [q-bio].* arXiv: `1207.3907`. `http://arxiv.org/abs/1207.3907` (2019) (July 20, 2012).

10. Li, H., Bloom, J. M., *et al.* A synthetic-diploid benchmark for accurate variant-calling evaluation. *Nature Methods* **15,** 595–597. ISSN: 1548-7091, 1548-7105. `http://www.nature.com/articles/s41592-018-0054-7` (2020) (Aug. 2018).

11. Li, H., Handsaker, B., *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)* **25,** 2078–2079. ISSN: 1367-4811 (Aug. 15, 2009).

12. Li, R. *et al.* SOAP2: an improved ultrafast tool for short read alignment. *Bioinformatics* **25,** 1966–1967. ISSN: 1367-4803, 1460-2059. `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btp336` (2019) (Aug. 1, 2009).

13. Meacham, F. *et al.* Identification and correction of systematic error in high-throughput sequence data. *BMC Bioinformatics* **12,** 451. ISSN: 1471-2105. `https://doi.org/10.1186/1471-2105-12-451` (2020) (Nov. 21, 2011).

14. Nakamura, K. *et al.* Sequence-specific error profile of Illumina sequencers. *Nucleic Acids Research* **39,** e90. ISSN: 0305-1048. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3141275/` (2020) (July 2011).

15. Ni, S. & Stoneking, M. Improvement in detection of minor alleles in next generation sequencing by base quality recalibration. *BMC genomics* **17,** 139. ISSN: 1471-2164 (Feb. 27, 2016).

16. Poplin, R. *et al.* Scaling accurate genetic variant discovery to tens of thousands of samples. *bioRxiv.* `http://biorxiv.org/lookup/doi/10.1101/201178` (2019) (July 24, 2018).

17. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26,** 841–842. ISSN: 1367-4803. `https://academic.oup.com/bioinformatics/article/26/6/841/244688` (2020) (Mar. 15, 2010).

18. Sedlazeck, F. J., Rescheneder, P. & von Haeseler, A. NextGenMap: fast and accurate read mapping in highly polymorphic genomes. *Bioinformatics* **29,** 2790–2791. ISSN: 1367-4803. `https://academic.oup.com/bioinformatics/article/29/21/2790/195626` (2020) (Nov. 1, 2013).

19. Song, L., Florea, L. & Langmead, B. Lighter: fast and memory-efficient sequencing error correction without counting. *Genome Biology* **15,** 509. ISSN: 1474-760X. `https://doi.org/10.1186/s13059-014-0509-9` (2020) (Nov. 15, 2014).

20. Van der Auwera, G. A. *Geraldine Van der Auwera on Twitter* Twitter. `https://twitter.com/VdaGeraldine/status/1296181178534440963` (2020).

21. Waterson, R. H., Lander, E. S., Wilson, R. K. & The Chimpanzee Sequencing and Analysis Consortium. Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature* **437,** 69–87. ISSN: 1476-4687. `https://www.nature.com/articles/nature04072` (2020) (Sept. 2005).

22. Zook, J. M., Samarov, D., McDaniel, J., Sen, S. K. & Salit, M. Synthetic spike-in standards improve run-specific systematic error analysis for DNA and RNA sequencing. *PloS One* **7,** e41356. ISSN: 1932-6203 (2012).