Comparing the Performance of Scalpel to GATK-HaplotypeCaller Using Simulated Reads

By
Matthew Lueder
Winter, 2017

Comparing the Performance of Scalpel to GATK-HaplotypeCaller Using Simulated Reads

By
Matthew Lueder

A project submitted in partial fulfillment of the requirements for the degree of
Master of Science in
Medical and Bioinformatics

at
Grand Valley State University

Winter, 2017

_____

Your Professor **Date**

Table of Contents

Abstract

The ability to accurately call variants from next-generation sequencing data (NGS) is a necessity for the success of NGS in clinical genomics. Therefore, there is a need for continuous in-depth reporting on the accuracy of state-of-the-art variant calling algorithms. In this paper, the performance of two local de novo reassembly-based variant calling tools are benchmarked using a simulated dataset. Genome Analysis Tool Kit HaplotypeCaller (GATK-HC) is consistently reported to be one of the best performing variant callers. Scalpel is a newer tool which has recently been reported to outperform GATK-HC in calling insertion/deletion elements (INDELs). The goal of this study is to provide an up to date and in-depth comparison of these two variant callers using a realistic simulated dataset. Simulated reads were generated using the tools VarSim and ART, then aligned to a reference genome using BWA-MEM. Precision, recall, and F_1-scores were calculated by comparing variants called by GATK-HC and Scalpel to a truth-set of variants using PrecisionFDA's comparison tool. GATK-HC was observed to have higher precision and recall for single nucleotide polymorphisms (SNPs) and INDELs.

Introduction

Next-generation sequencing (NGS) refers to a set of recent DNA sequencing technologies which offer substantial increases in throughput over traditional sanger-based approaches. NGS first appeared around the onset of the 21st century (1). Since its appearance, the cost to sequence a human sized genome has decreased at a rate faster than Moore's law (2). NGS achieves incredibly high throughputs by making use of the idea of concurrency. Hundreds of millions of short genomic fragments are sequenced in parallel (1).

The output from a next generation sequencer is a file which contains the predicted sequence for part of each fragment (known as reads) along with a separate quality score for each base sequenced. This quality score predicts the probability that the base was correctly sequenced. This output however does not tell you where in the genome each sequencing read originated from. Additional steps are required to figure this out. In the case that the species being sequenced has been sequenced previously and a reference genome is available, this is typically solved by aligning each read to the reference genome.

After aligning the reads to a reference genome and creating a genome assembly, it is common to be interested in how the sequenced genome differs from the reference. Differences found in the genome are known as variants and are discovered in a process known as variant calling. Variants can take the form of single nucleotide polymorphisms (SNPs), insertion or deletion elements (INDELs), and larger scale structural variation (3). An SNP is a variation of a single nucleotide at a specific genomic location. INDELs are variants which originate from an insertion or deletion mutation. Structural variation includes large INDELs, duplications, copy-number variants (CNVs), inversions, and translocations.

There are a wide variety of variant calling algorithms. Some algorithms are designed to target a specific type of variant. Some tools are designed to discover only CNVs, some are designed to uncover multiple types of structural variation, and others are specific to SNPs/INDELs (4). Variant calling algorithms can be further split into germline and somatic callers. Germline callers look for inherited variants. Germline callers often discover variants by comparison to a reference genome, as in the case

outlined above. Somatic variant callers look for DNA alterations which occurred after conception and are not present in the germline (5). Somatic variant calling is most common in cancer studies, and therefore most somatic variant calling tools are designed to look for variants by comparing assemblies derived from two different samples from the same individual: a tumor sample, and a regular tissue sample (6). Cancer studies often use somatic variant calling to identify driver and passenger genes (4). Germline variant calling is often used to identify causative genes in Mendelian disorders (4) and in studies looking to elucidate large scale genetic variation within and between populations (7).

Many traditional medical treatments are designed to best treat the "average patient" and as a result, certain treatments work well for certain patients but not others. Precision medicine is an attempt to change this by individually tailoring treatment plans to a person's genome, lifestyle, and environment (8). The idea behind precision medicine is not new, however with the recent advances in NGS and patient data storage, there is a large push to identify relationships between genetic variants and disease, and to use this data to develop personalized medicine (7). In order for genomics-based precision medicine to be successful it is imperative that the pipelines and algorithms used to discover genetic variants produce accurate results.

This study compares the accuracy of two popular variant calling algorithms: Genome Analysis Tool Kit HaplotypeCaller (GATK-HC) and Scalpel. Both tools identify SNPs and INDELs through local de novo assembly with de Bruijin or de Bruijn-like graphs (9) (10). These algorithms use alignment information to supplement the reassembly of small genomic regions by localizing the analysis into computationally tractable regions (9) (10). This reassembly uses methods developed for de novo sequencing, meaning it is performed without requiring a reference genome.

Scalpel uses a sliding window approach, in which de Bruijn assembly is performed within a window of a certain size. This window is moved along to different genomic locations by a predefined interval, and de Bruijn assembly is performed at each step. In highly repetitive regions, Scalpel iteratively recreates the de Bruijn graph, increasing k-mer size until a 'repeat-free' graph is built. Constructed graphs are exhaustively explored to identify paths representing assembled sequences. These

assembled sequences are aligned to the reference genome with using the Smith–Waterman algorithm to identify variation (9).

GATK-HC works by identifying 'active regions' based on significant evidence of variation in the input alignment. It then performs de novo reassembly within the identified active regions through the use of a de Bruijn-like graph. This assembly is used to identify candidate haplotypes, which are then aligned against the reference haplotype using the Smith–Waterman algorithm to potential variants. Then, individual reads are aligned to each candidate haplotype in a pairwise fashion with the PairHMM algorithm. This is done to evaluate the amount of supporting evidence for each candidate haplotype and results in a matrix containing the likelihoods each haplotype given each read. These scores are then marginalized to produce a set of scores for each potential variant. Bayes theorem is then used to generate the likelihood of each possible genotype from these scores. The most likely genotype is selected (10).

To perform a comparison between Scalpel and GATK-HC, I used a simulated dataset designed to mimic output from a next-generation sequencer. This simulated dataset contains reads with artificial SNPs, INDELs, and structural variation sampled from a dataset of real variants. A file listing the set of true variants was generated during the simulation. Simulation was performed with the tools ART and VarSim. Reads from this dataset were aligned to a human reference genome with BWA-MEM and variant calling was performed with GATK-HC and Scalpel. Called variants were then compared to the truth set and precision, recall, and F1 scores were calculated. I compare how well GATK-HC and Scalpel perform at the task of calling all variants, only SNPs, and only INDELs.

Background and Related Work

One reported advantage of local reassembly based variant calling algorithms is increased sensitivity and robustness for detection of large INDELs (11). Other variant calling tools try to infer genotypes directly from the genome assembly created from a read mapping algorithm. These algorithms perform worse with large INDELs because large INDELs can complicate and distort the correct mapping (11). This was the motivation for looking exclusively at assembly-based callers.

A study published in Nucleic Acids Research by Yi et al. compared the SNP calling performance of multiple variant callers with Illumina exome sequencing data (12). This study used family pedigree information and SNP array information for validation of variants. The variant callers compared in this study include GATK-HC, Samtools, VarScan, CASAVA, CLCbio, and Partek. The data used in this study was generated from a lymphocyte samples taken from two families with known pedigrees. Mendelian inheritance error checking (MIEC) was performed on this data to determine whether called SNPs passed or failed Mendelian inheritance rules. Error rates were calculated by dividing the number of SNPs which passed MIEC by the number that failed. In addition, the authors generated data from the same samples with an SNP array and compared SNPs found to those found by variant callers. Error rates in this experiment were based on whether the variant caller was consistent with SNPs detected on the array. The authors concluded that "GATK[-HaplotypeCaller] performed either in the best or in the top tier for most, if not all, of the comparison metrics and schemes and also was the most consistent across different evaluative tests" (12). These results show that GATK-HC outperforms other common tools in its ability to detect SNPs, however, this study did not show how well it performs at detecting INDELs and did not include Scalpel.

A study published in Nature Methods by Narzisi et al directly compared Scalpel and GATK-HC (13). This paper looked at Scalpel's, GATK-HC's, and SOAPindel's (another de novo assembly based caller) ability to detect INDELs in exome-capture data. To do this they first sequenced the exome of an individual with a severe case of Tourette syndrome and obsessive-compulsive disorder and aligned the sequencing reads to a reference genome with BWA. The aligned reads were then used as input into various variant callers and concordance among calls between the pipelines was studied. They

found that there was only 37% concordance between all three pipelines. To validate calls made by the three pipelines, targeted resequencing with Illumina MiSeq was performed on 1000 INDELs: 400 INDELs from the intersection of all three pipelines (including 200 INDELs with a size ≥ 30 bp) and 200 INDELs specific to each pipeline. Scalpel was reported to have a validation rate of 76% for INDELs greater than 5 bp in length, compared to 27% for GATK-HC. The overall INDEL validation rate for GATK-HC was only 22%, whereas 77% of INDELs called by scalpel were reported to be true positives (13).

The results of this study suggest that Scalpel vastly outperforms GATK-HC in INDEL calling, at least in terms of precision/positive predictive value. This has large implications because GATK is commonly regarded as the 'Gold Standard' tool for variant calling within the bioinformatics community (12). However, this study has some major limitations. Its experimental design only allows for insight into the precision of called INDELs; it does not provide insight into recall and does not report on SNP calling performance. In order to fairly compare variant calling algorithms, it is necessary to: i) take both precision and recall into account and ii) look at both INDELs and SNPs.

To estimate the precision and recall of tested variant callers, I took the approach of generating simulated data. When using simulated data, is important to use data which closely models real sequencing data, including sequencing errors and genetic variation. The tools VarSim and ART were used for this purpose. VarSim is a framework developed for the purpose of assessing NGS alignment and variant calling accuracy (14). VarSim uses a user-provided reference genome to generate a perturbed diploid genome with artificially induced genetic variation sampled from previously reported variation (previously reported variation also provided by the user). By sampling previously reported variation, VarSim ensures that the simulated genome is biologically relevant. VarSim generates the full spectrum of variants – SNPs, small INDELs, and all types of structural variation. Variants induced by VarSim are output to a truth VCF file (14). This allows for the assessment of both SNP and INDEL calls.

To generate sequencing reads from this perturbed genome, VarSim sits on top of ART. ART generates artificial NGS reads for a provided genome by emulating the sequencing process (15). The qualities of reads produced by actual NGS experiments are

variable, and so are the qualities of bases within reads. ART attempts to capture this variability in the reads it simulates using technology specific read error models and base quality profiles that have been parameterized from empirical observations (15). This makes simulations generated by ART significantly more realistic than simulations created from tools which induce sequencing errors randomly such as wgsim (15). ART can generate both single and paired-end reads and can emulate three NGS platforms: SOLiD, 454, and Illumina.

As mentioned earlier, VarSim requires the user to provide a set of variants to sample from. For this, I used a set of 5.4 million "Platinum" high-confidence benchmark variants originally produced in a study by Eberle et al. (16). This study generated whole-genome sequence data from 17 family members in a 3-generation pedigree and used this data in multiple pipelines to create a call set. Variants were then filtered based on MIEC. The larger pedigree used in this study inferred an increased ability to detect errors and asses the accuracy of variants compared to the standard trio analysis (16). The authors further filtered variants based on number of reads supporting the flanking sequence using a k-mer approach [for more info see Eberle et al.]. The final set of variants consisted of approximately 4.7 million SNPs and 700 thousand INDELs.

Program Requirements and Implementation

VarSim and ART were used to simulate HiSeq 2500 reads. Human reference genome GRCh37 was used as a foundation to generate a perturbed genome with variants sampled from the "Platinum" variant catalogue released by Eberle et al. (16). Simulated fragments created from the reference genome had a mean length of 500 bp and a standard deviation of 50 bp. Paired end reads generated from the fragments had a length of 150 bp. Enough reads were generated to achieve a mean coverage of 30. The rest of VarSim's parameters were left in the default state. Approximately 277.75 million reads pairs were generated. The set of truth variants consisted of 4,406,783 variants including 3,730,452 SNPs and 678,917 INDELs. VarSim version 0.5.2 was used.

Simulated reads were mapped to reference genome GRCh37 using BWA-MEM version 0.7.12. This was done using 32 cores provided PrecisionFDA's (17) cloud computing platform. Shorter split hits were marked as secondary and remaining parameters were left as default. BWA-MEM was the selected read mapping algorithm because of its popularity and because it is recommended by the creators of GATK and Scalpel (9) (10). Output from BWA-MEM was piped directly into biobambam2's bamsormadup tool. Bamsormadup sorts reads and marks PCR duplicates. Only 3 reads were left unmapped and 77 thousand read pairs were marked as duplicates.

The sorted and duplicate marked BAM file output by bamsormadup was then used as input into the variant callers. Variant calling was localized to exome regions defined by a BED file distributed by the European Bioinformatics Institute. This was done to reduce computation time. Reference genome GRCh37 was also used as input into both GATK-HC and Scalpel.

Scalpel works in two phases. First scalpel-discovery is used to call variants with user-defined parameters. This was done with default parameter values (using version 0.5.3). The result of this is a database containing all found variants. In the next step, scalpel-export is used to extract variants from this database and create a VCF file. This two-step process allows for easy filtration of called variants based on several user-defined parameters. Users can filter based on quality and variant type. I took advantage of this feature to create 3 VCF files: one containing all called variants, one containing only

SNPs, and one containing only INDELs. Splitting variants by type was necessary for comparison purposes later on.

Version 3.5 of GATK-HC was used to call variants with default parameters in the alternate pipeline. This resulted in a single VCF file containing all found variants. VCTtools was then used to create two additional filtered VCF files: one containing only SNPs and one containing only INDELs. VCFtools was also used to split the truth set VCF in a similar manner.

Variants called by GATK-HC and Scalpel were compared to each other using a comparison tool offered as a web application by PrecisionFDA. This tool is based on vcfeval 3.5.1 by Real Time Genomics (18). It was built to conceptually resemble 'Comparison Method 3' of the GA4GH benchmarking standards (19). This comparison tool reports precision, recall, F_1-score, number of true-positives, number of false-positives, and number of false negatives. It also allows users to compare genomic regions specified by a BED file. I used the exome targets bed file (used previously in the variant calling step) to filter out test set variants located outside of the exome. The entire analysis pipeline is depicted in figure 1.

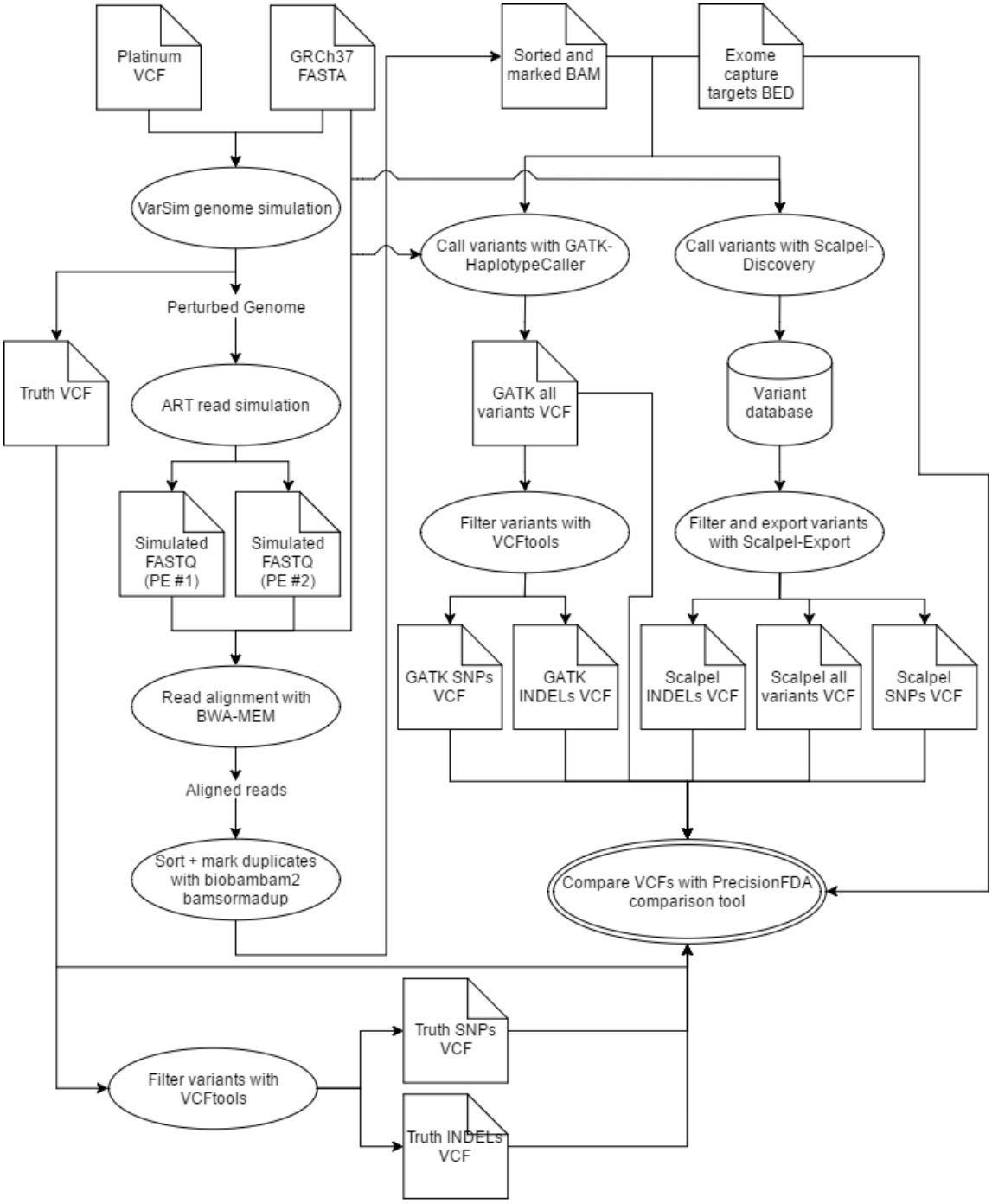

Figure 1. Flowchart representing project workflow. Files used as input/output are represented by square boxes. Tasks are represented by ellipses. Databases are represented by cylinders. The final task is shown as a double ellipse. Resources with an arrow pointing to a task represent that the resource was used in the task. Tasks with arrows pointing to a resource represent that the resource was produced by the task

Results, Evaluation, and Reflection

GATK-HC found 37,260 variants in total, including 34,843 SNPs and 2,427 INDELs. The number of variants found by Scalpel was significantly fewer. Scalpel found 29,232 variants in total including 27,415 SNPs and 1,833 INDELs. Total concordance between the two pipelines was 69.37% (figure 2). Concordance between pipelines for SNPs was higher than concordance for INDELs (70.12% vs 58.01%). The concordance of INDELs between GATK-HC and Scalpel closely mirrors what was found in the Narzisi et al study, which found a concordance of 59.45% (13). 93.16% of variant calls made by Scalpel are consistent with GATK-HC, while only 73.09% of GATK-HC's calls are consistent with Scalpel. This could be indication that GATK-HC possess superior recall ability over Scalpel but could also indicate that Scalpel possess superior precision.

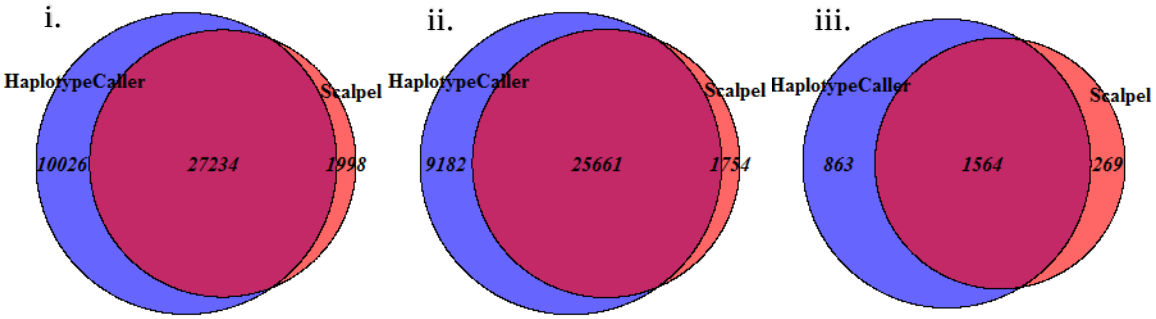

Figure 2. Concordance between GATK-HaplotypeCaller and Scalpel for: i) all variants ii) SNPs iii) INDELs

Precision, Recall, and F_1-scores were calculated from GATK-HC and Scalpel by comparison to the truth set generated by VarSim. Results indicate that GATK-HC consistently outperformed Scalpel (Table 1). The F_1-score, which is a measure of accuracy that takes both precision and recall into consideration, was substantially higher for GATK-HC for both INDELs and SNPs (97.44% vs. 81.11% for SNPs, 91.30% vs. 72.37% for INDELs).

The two pipelines differed most dramatically in terms of recall. GATK-HC had a recall of 97.23%, while Scalpel had a recall of 71.94%, when considering all variants. This difference in recall is even more pronounced when considering only INDELs. This shows that a significant portion of GATK-HC specific calls are true positives.

Table 1: Precision, recall, and F_1-scores for GATK-HaplotypeCaller and Scalpel. Results are shown for: 1) all variants 2) SNPs and 3) INDELs.

	All Variants (1)	SNPs (2)	INDELs (3)
GATK-HaplotypeCaller			
Precision	97.03%	97.40%	90.71%
Recall	97.23%	97.49%	92.03%
F_1-score	97.13%	97.44%	91.30%
Scalpel			
Precision	91.52%	92.06%	83.48%
Recall	71.94%	72.50%	63.87%
F_1-score	80.56%	81.11%	72.37%

GATK-HC had precisions of 97.03%, 97.40%, and 90.71% for all variants, SNPs, and INDELs respectively. This means 2.6% of called SNPs where false positives, which is consistent with the Yi et al. study which reported error rates of 3.27% and 2.53% for SNP calls made by two versions of GATK-HC (12). In Contrast, Scalpel achieved precisions of 91.52%, 92.06%, 83.48% for all variants, SNPs, and INDELs respectively. These results are significantly different than the results reported in the Narzisi et al study, which found Scalpel-specific INDELs to have higher validation rates than GATK-HC-specific INDELs (13). There are several potential reasons for this difference. Most notably, the version of GATK used in my analyses are different than the versions used in the Narzisi et al. study. My study used GATK version 3.5, while Narzisi et al. used versions 2.4.3 and 3.0. GATK is under rapid development and it is possible that GATK-HC's ability to call INDELs dramatically increased over subsequent version changes. The version of Scalpel used in this study was also different. I used the latest version of Scalpel (0.5.3), while Narzisi et al. used version 0.1.1 beta. In addition, Narzisi et al. performed an additional filtering step in which they removed INDELs showing a high coverage unbalance (13). This additional filtering step may have increased the precision of its INDEL calls, giving it an advantage over GATK.

As mentioned earlier, GATK is commonly regarded as the 'Gold Standard' variant calling tool within the bioinformatics community (12). The results found in this study are consistent with this view. While in the past Scalpel may have outperformed GATK at INDEL calling, these results suggest that this is no longer the case. GATK-HC performed better in terms of both precision and recall for INDELs and SNPs.

Future Work

While these results provide strong evidence that GATK-HC outperforms Scalpel, this was not an exhaustive test. Scalpel has 11 parameters which could potentially alter its performance, and they were all left in the default state. These parameters include custom k-mer size to be used in the construction of the de Bruijn graph, coverage thresholds, sliding window size, and the step size for the sliding window. GATK-HC also has many adjustable parameters with the potential to alter calling accuracy, including minimum base quality scores for variants, active region size, and heterozygosity for computing prior likelihoods. Further study could be done to determine how adjusting these parameters affect the performance of GATK-HC and Scalpel.

In this study, Scalpel was used in 'single' mode. Single mode is used for cases in which only a single sample is available. However, Scalpel also has two other modes: de novo and somatic. De novo mode is used to call de novo germline variants in nuclear families of four people. This means genome alignment data is required for a mother, a father, and two siblings. Somatic mode is used to detect somatic mutations given data for a tumor/normal pair. I could not find any literature reporting Scalpel's performance for these modes. This could be the subject of further study.

Finally, there could be further study done to see how additional steps in the pipeline affect variant calling performance of Scalpel and GATK-HC. Additional steps could include filtering reads before alignment based on quality scores, base quality score recalibration (BQSR), and applying quality filters after the variant calling procedure. Filtering reads could improve the quality of the alignment which could improve Scalpel's and GATK-HC's ability to accurately call variants. BQSR is a step recommended in GATK's best practices (20). This step is performed after an alignment is created and sorted, and PCR duplicates are marked. The purpose of this step is to detect systematic errors in the quality score of each base call, and to adjust the quality score accordingly. This step has the potential to improve GATK-HC's performance, but would not affect Scalpel's performance because Scalpel does not directly use base quality score information (9). At the end of the pipeline, quality filters can be applied to called variants to increase precision. However, these filters cannot be used to increase recall, which was Scalpel's biggest weakness.

Bibliography

1. Barba M, Czosnek H, Hadidi A. Historical Perspective, Development and Applications of Next-Generation Sequencing in Plant Virology. Viruses. 2014 October; 6.
2. Wetterstrand K. Genome.gov. [Online].; 2016 [cited 2017 April 9. Available from: https://www.genome.gov/sequencingcostsdata/.
3. Laurie S, Fernandez-Callejo M, Marco-Sola S, Trotta JR, Camps J, Chacon A, et al. From Wet-Lab to Variations: Concordance and Speed of Bioinformatics Pipelines for Whole Genome and Whole Exome Sequencing. Human Mutation. 2016 September; 37(12).
4. Pabinger S, Dander A, Fischer M, Snajder R, Sperk M, Efremova M, et al. A survey of tools for variant analysis of next-generation sequencing data. Briefings in Bioinformatics. 2013 January; 15(2).
5. National Cancer Institute at the National Institutes of Health. NCI Dictionary of Genetics Terms. [Online]. [cited 2017 April 9. Available from: https://www.cancer.gov/publications/dictionaries/genetics-dictionary.
6. Krøigård AB, Thomassen M, Lænkholm AV, Kruse T, Larsen MJ. Evaluation of Nine Somatic Variant Callers for Detection of Somatic Mutations in Exome and Targeted Deep Sequencing Data. PLoS ONE. 2016 March; 11(3).
7. Lu YF, Goldstein D, Angrist M, Cavalleri G. Personalized Medicine and Human Genetic Diversity. Cold Spring Harbor Perspectives in Medicine. 2014 July; 4.
8. Hodson R. Nature Outlook Precision Medicine Supplement. Nature. 2016 September; 537(7619).
9. Fang H, Bergmann E, Arora K, Vacic V, Zody M, Iossifov I, et al. Indel variant analysis of short-read sequencing data with Scalel. Nature Protocol. 2016 November; 11(12).
10. Broad Institute. Genome Analysis Toolkit Guide. [Online].; 2009 [cited 2017 April 8. Available from: https://software.broadinstitute.org/gatk/documentation/article.php?id=4148.
11. Fang H, Wu Y, Narzisi G, O'Rawe J, Jimenez Barrón L, Rosenbaum J, et al. Reducing INDEL calling errors in whole genome and exome sequencing data. Genome Medicine. 2014; 6(89).
12. Yi M, Zhao Y, Jia L, He M, Kebebew E, Stephens R. Performance comparison of SNP detection tools with illumina exome sequencing data—an assessment using both family pedigree information and sample-matched SNP array data. Nucleic Acids Research. 2014 May; 42(12).
13. Narzisi G, O'Rawe J, Iossifov I, Fang H, Lee Yh, Wang Z, et al. Accurate de novo and transmitted indel detection in exome-capture data using microassembly. Nature Methods. 2014 August; 11(10).
14. Mu J, Mohiyuddin M, Li J, Bani Asadi N, Gerstein M, Abyzov A, et al. VarSim: a high-fidelity simulation and validation framework for high-throughput genome sequencing with cancer applications. Bioinformatics. 2015 December; 31(9).
15. Huang W, Li L, Myers J, Marth G. ART: a next-generation sequencing read simulator. Bioinformatics. 2012 December; 28(4).
16. Eberle M, Fritzilas E, Krusche P, Källberg M, Moore B, Bekritsky M, et al. A reference data set of 5.4 million phased human variants validated by genetic inheritance from sequencing a three-generation 17-member pedigree. Genome Research. 2017; 27.
17. U.S. Food and Drug Administration. PrecisionFDA. [Online].; 2017 [cited 2017 March 30. Available from: https://precision.fda.gov.
18. Cleary J, Braithwaite R, Gaastra K, Hilbush B, Inglis S, Irvine S, et al. Comparing Variant Call Files for Performance Benchmarking of Next-Generation Sequencing Variant Calling Pipelines. Technical Report. 2015 August.
19. GA4GH Benchmarking Team. GA4GH Benchmarking Tools and Standards. [Online].; 2017 [cited 2017 April 9. Available from: https://github.com/ga4gh/benchmarking-tools.
20. Prabhakaran A, Shifaw B, Naik M, Narvaez M, Van der Auwera G, Powley G, et al. Infastructure for Deploying GATK Best Practices Pipeline. Technical Report. Intel; 2016.