BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 11 2013, pages 1361–1366
doi:10.1093/bioinformatics/btt172

Genome analysis

Advance Access publication April 24, 2013

A support vector machine for identification of single-nucleotide polymorphisms from next-generation sequencing data

Brendan D. O'Fallon*, Whitney Wooderchak-Donahue and David K. Crockett*

ARUP Institute of Clinical and Experimental Pathology, 500 Chipeta Way, SLC, UT 84102, USA

Associate Editor: Michael Brudno

ABSTRACT

Motivation: Accurate determination of single-nucleotide polymorphisms (SNPs) from next-generation sequencing data is a significant challenge facing bioinformatics researchers. Most current methods use mechanistic models that assume nucleotides aligning to a given reference position are sampled from a binomial distribution. While such methods are sensitive, they are often unable to discriminate errors resulting from misaligned reads, sequencing errors or platform artifacts from true variants.

Results: To enable more accurate SNP calling, we developed an algorithm that uses a trained support vector machine (SVM) to determine variants from .BAM or .SAM formatted alignments of sequence reads. Our SVM-based implementation determines SNPs with significantly greater sensitivity and specificity than alternative platforms, including the UnifiedGenotyper included with the Genome Analysis Toolkit, samtools and FreeBayes. In addition, the quality scores produced by our implementation more accurately reflect the likelihood that a variant is real when compared with those produced by the Genome Analysis Toolkit. While results depend on the model used, the implementation includes tools to easily build new models and refine existing models with additional training data.

Availability: Source code and executables are available from github.com/brendanofallon/SNPSVM/

Contact: brendan.d.ofallon@aruplab.com or david.crockett@aruplab.com

Received on January 9, 2013; revised on April 2, 2013; accepted on April 5, 2013

1 INTRODUCTION

High-throughput sequencing technologies produce large quantities of relatively low-quality data. Discrimination of true sequence variants from variants produced alignment, base-calling or platform-specific errors presents a significant bioinformatic challenge. In projects seeking to identify disease-causing variants, even a relatively modest number of false-positive calls may significantly impede analysis, as commonly used variant-filtering strategies often enrich for false-positive calls (Nielsen *et al.*, 2011). For instance, discarding variants above a certain frequency in the population preferentially retains false variants, as nearly all false-positive calls are likely to be rare.

Traditional variant-calling techniques, such as those available in the Genome Analysis Toolkit (GATK; McKenna *et al.*, 2010) or the samtools package (Li *et al.*, 2009), imagine that bases aligning to a reference position are drawn from a binomial distribution. Under the binomial assumption, the number of non-reference bases aligning to a position is approximately

$$P\{X = x|c\} = \binom{c}{x} p^x (1-p)^{c-x} \quad (1)$$

where x is the number of reads with non-reference bases aligning to the site, c is the total number of reads aligning and p is typically near 0.5 or 1.0, depending on the zygosity of the variant (base-calling and alignment errors prevent p from attaining either value exactly). A number of modifications have been described to model the impact of, for instance, base-calling errors or alternative zygosities (DePristo *et al.*, 2011; Shen *et al.*, 2010). More recently, a number of improvements have been described that use population haplotype and linkage information to refine the accuracy of calls (e.g. Browning and Browning, 2009; Howie *et al.*, 2009; Li *et al.*, 2011), although these methods are most useful for imputing genotypes from low-coverage data in humans. Despite these extensions, binomial-based methods do not naturally incorporate errors resulting from misaligned reads or platform-specific artifacts. In addition to generating a greater number of false-positive calls, these methods may overestimate the confidence of a variant call in regions of high coverage. In particular, under the strict binomial model, errors become less likely as read depth increases, leading to high-quality (confidence) scores for variants with high coverage. However, alignment errors may occur with similar probability at all read depths, and these variants resulting from these errors may be given erroneously high-quality scores when read depth is high.

The probability that a single-nucleotide polymorphism (SNP) is real is influenced by many statistical factors, including properties of the reference sequence, the type of substitution observed and the number, orientation and composition of the reads mapping to the region. For instance, false variants often exhibit strand bias, such that the ratio of variant bases on forward- to reverse-oriented strands differs from that for reference bases. Similarly, false variant calls may occur with greater frequency on reads with mismatches at other positions, or near the ends of reads, or in regions of low sequence complexity or with low genome 'mappability' scores. Inclusion of all such factors into a mechanistic variant-calling framework is difficult because it requires determination of how each factor influences the likelihood that a variant is real as well as how all such factors interact.

Machine-learning techniques offer one route to incorporation of multiple factors affecting variant likelihood into a single model (DePristo *et al.*, 2011). If sets of 'true' (and, possibly, 'false') variants are known in advance, their statistical properties

*To whom correspondence should be addressed.

© The Author 2013. Published by Oxford University Press. All rights reserved. For Permissions, please e-mail: journals.permissions@oup.com

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

may be assessed to form an educated guess regarding the state of a query variant. The Variant Quality Score Recalibration (VQSR) procedure (DePristo *et al.*, 2011) is a semisupervised technique that fits multidimensional Gaussian distributions to a collection of suspected true variants. Variants whose properties differ from those of the true variants are deemed more likely to be false positives. While the VQSR procedure improves the quality of variant calls, it suffers from several limitations. For example, VQSR requires tens of thousands of variants to accurately fit the distributions, and inclusion of more than a few (roughly 10) features may cause the algorithm to fail.

An alternative type of machine learning technique known as the support vector machine (SVM; Boser *et al.*, 1992; Cortes and Vapnik, 1995) addresses some of the difficulties encountered in earlier models. SVMs are numerical classification techniques designed to identify an arbitrarily defined class to which a query data point belongs, and have been previously used in bioinformatics applications from cancer subtype classification (Lee and Lee, 2002) to splice site prediction (e.g. Baten *et al.*, 2006). In a manner similar to VQSR, a vector of features is collected for all possible sites at which a variant may exist. The SVM is then trained on collections of sites known to contain true and false variants. Unlike VQSR, SVM training produces a 'model' that may then be used repeatedly to call variants from query datasets. SVMs can incorporate large numbers of features and, after training, an SVM does not require a large number of variants for precise calling; it may be used to classify a single query variant.

In this work, we demonstrate that an appropriately trained SVM can be used to accurately determine the positions of SNPs from next-generation sequencing data, and we present a software implementation designed to make such calculations practical for bioinformaticians. Our algorithm takes as input a.SAM or.BAM formatted alignment of sequencing reads and emits as output a.VCF formatted file containing positions of likely variants. The SVM is used to predict whether an alignment column contains a sequence variant of any zygosity, and additional non-SVM calculations are performed to determine the most likely zygosity of each variant.

2 METHODS

Our SNP-identification algorithm involves two steps. First, a SVM must be generated by specifying an input.BAM or.SAM-formatted sequence alignment and two.VCF files containing the locations of sites known to contain true variants and false variants. Using these data, an SVM is then trained and stored as a reusable 'model' that may be used to predict the variants in a query.BAM file. Second, the trained model can be used to predict which sites contain true variants in a query.BAM file. Both steps are described in detail below.

To conduct the SVM training and classification procedures, we use the previously developed and freely available LIBSVM library (v. 3.12; Chang and Lin, 2011). While LIBSVM provides several SVM implementations, SNPSVM exclusively uses the C-SVC formulation with a Radial Basis Function (RBF) kernel. While we do not offer a detailed description of the general SVM algorithm, a non-technical introduction can be found in (Noble, 2006), with further reading available in (Schoelkopf *et al.*, 2004). The procedure involves two tuning parameters, C and γ. To select optimal values of these parameters, we performed a grid search of parameter space using 4-fold cross-validation accuracy as the function to optimize. The grid search was performed with a utility included with LIBSVM (grid.py), and was repeated on several different training sets. We examined values of C between 2^{-4} and 2^{20} and values of γ in 2^{-14}–2. Examination of the cross-validation surface revealed a broad plateau of values near 98.6–98.9% corresponding to values of C in $2^3 - 2^8$ and values of γ in $2^{-8} - 0$. Because cross-validation accuracy appeared relatively insensitive to C and γ in this region and high values of C significantly increased training time, we chose to fix C at $C = 10.0$ and $\gamma = 0.01$ throughout.

2.1 Generation of trained models

Training runs take as input a.SAM or.BAM formatted alignment file and.VCF-formatted lists of true- and false-positive variant sites and emit as output a 'model' file. Training runs consist of two phases. In the first phase, the input training variant files are scanned, and for each training site, the input alignment is queried and feature data written to a training data file (a complete list of the features assessed is given below). In the second phase, the SVM is trained on the data produced in the first phase. Training involves numerical identification of the multidimensional plane that separates the two classes of training data yielding the greatest margin between true and false calls, referred to as the maximum-margin hyperplane. The numerical techniques used in the training stage are described in detail in (Chang and Lin, 2011).

Generation of an accurate model for variant calling the SVM requires a large set of training data where the true class (variant or invariant) of all suspected variants is known. To obtain sets of such variants, we turned to an in-house sample on which exome sequencing had been performed twice on the same instrument (an Illumina HiSeq 2000). The sample was a Caucasian female of eastern European origin. We initially determined variants according to the GATK's best practices guidelines, including alignment with BWA (0.6.1; Li and Durbin, 2009), indel realignment, base quality score recalibration and removal of polymerase chain reaction (PCR) duplicates using samtools 0.1.18. Variants were then called using the GaTK's UnifiedGenotyper tool, with the quality threshold for emitting potential variants set to 0.01. As our set of 'true' training variants, we retained all SNPs that had been called in both runs and that were present at 10% frequency or greater in the 1000 Genomes project data (version 1 of the data released on 21 May 2011). In all, we identified 35 567 such variants. The ratio of transitions to transversions in these data was 2.591, suggesting little contamination with false-positive calls.

To build a collection of false-positive variant calls, we used two strategies. First, we collected all variants that had been called in one run but not the other, and then eliminated all variants that had been seen at any frequency in the 1000 Genomes data. We then retained only those variants with quality scores <10.0. Across both samples, we identified ~11 000 variants with a Ti/Tv ratio of 0.75. Second, we tabulated all sites that had been previously observed at least eight times in an internal database of 57 exomes, but which had not been previously detected in the 1000 Genomes data. We identified 2642 such sites, with a Ti/Tv ratio of 0.60.

2.2 SNP calling on query alignments

Once a model has been trained, it may be used to call variants in query alignments. This step involves specifying a reference genome, a model and an input.BAM formatted alignment. SNP calling proceeds in three steps. In the first step, the input alignment is examined and, for all sites at which a variant may exist, measurements of all desired features are recorded (see *Feature selection* for a description of all features). In the second step, the existing SVM model is used to predict the class (either variant or invariant) of all query sites determined in the first step. In the third step, predictions are combined with alignment information about the input.BAM to generate a.VCF file.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

2.3 Sensitivity and specificity

Accurate determination of the sensitivity and specificity of a variant caller requires at least one data reference set in which variants have been identified with high reliability. For our gold-standard dataset, we used a previously described consensus of nine independently sequenced genomes and two exomes, all from the HapMap trio sample NA12878. The consensus set has been shown to have significantly higher sensitivity and specificity than variant sets produced with a single technology or variant caller alone (Zook,J.M. et al., unpublished data). To compare performance of the variant callers and investigate the properties of SNPSVM, we sequenced the exome of NA12878 on an Illumina HiSeq 2000 instrument using Agilent's SureSelect XT (version 4) exome capture kit and 2×100 paired-end reads. Reads were aligned with BWA (0.6.1; Li and Durbin, 2009), base quality scores recalibrated with the GATK (v1.6), PCR duplicates removed with samtools (v0.1.18; Li *et al.*, 2009) and local realignment was performed with the GATK to produce a 'final' alignment file. All variant calling was performed for the target region only on the final alignment.

Variant calls from the final alignment file were produced using SNPSVM, the GATK's UnifiedGenotyper, FreeBayes, Samtools/mpileup and the UnifiedGenotyper calls improved using VQSR. Because we were interested in identifying variants across a wide range of quality scores, threshold quality scores for all callers were lowered substantially for each tool. For the VQSR procedure, we used the quality/depth (QD), HaplotypeScore, ReadPosRankSum, Fisher strand bias (FS), MQRankSum and HRun features produced by the GATK. Because VQSR requires large sets of input variants for model training, we supplied an additional five exome variant sets to the recalibrator, each prepared using the same enrichment, sequencing, alignment and variant-calling procedure as the original NA12878 sample. In total, some 922 425 variants were supplied to the recalibration engine. Sensitivity and specificity curves were generated by altering the 'quality' (Q) score for all curves except the VQSR-generated curve, where the VQSLOD value was used.

2.4 Feature selection

Many possible features may influence the probability that a site contains a true variant. To determine a parsimonious set of features that yields high sensitivity and specificity, we used two strategies. First, beginning with a minimal set of three features that yielded poor performance, we sequentially added new features to quantify the impact of the feature on performance. Performance was measured by the area under an ROC curve (AUC) constructed using the consensus NA12878 truth set, relative to the AUC for the minimal set alone. Second, we conducted a leave-one-out procedure in which all features were included initially, and the effect of a given feature was computed by excluding it from the feature set, computing the area under the ROC as above and determining the loss of performance compared with the full set. Table 1 lists all features with a brief description and the relative impact on performance. Features that caused the greatest decrease in accuracy when removed were deemed to be the most informative. Only the 'Nearby base quality' feature caused an increase in accuracy when removed. A description of all features can be found in the Appendix.

Overall, this analysis indicates that the most important factors for determining true from false variants are the probabilities that the fraction of variant reads was drawn from a binomial distribution with parameter close to 0.5 or 1.0. The 'Error probability' feature computes the likelihoods of observing the given variant allele fraction conditional on the true variant fraction being 0.1, 0.5 or 1.0, then computes the relative likelihood that the true fraction was 0.1. This feature is meant to capture cases in which sequencing or alignment errors result in a small fraction of false variant bases at a given site. Other features with high impact include those relating to the base qualities observed, the mapping quality of reads with variant and reference bases and the total read depth at a site. Features relating to strand bias and position of the variant on the reads have less overall impact and only marginally improved accuracy.

Based on these analyses, we chose to include all features except the 'Nearby base quality' feature, which appeared to reduce performance. Unless otherwise mentioned, all analyses used the remaining 15 features.

Table 1. Features and their relative impact on AUC

Feature	Leave-out effect[a]	Add-in effect[b]
Error probability	0.9937	1.310
Binomial probability	0.9955	NA
Base quality mean	0.9959	NA
Base quality sum	0.9967	NA
Read mapping quality	0.9968	1.0291
Total depth	0.9975	NA
Dinucleotide repeat count	0.9992	1.001
Read position variance	0.9993	1.002
Read position mean	0.9995	1.012
Strand bias	0.9996	1.000
Area mismatch total	0.9999	1.002
Homopolymer run count	1.000	1.004
Nucleotide diversity	1.000	1.007
Read mismatch counts	1.000	1.018
Allele balance	1.001	1.0114
Nearby base quality	1.004	1.004

[a]AUC after feature removal relative to original AUC.
[b]AUC after feature addition relative to original AUC.

3 RESULTS

3.1 Sensitivity and specificity

A fundamental difference between SNPSVM and other variant callers is that SNPSVM relies on a variant-calling 'model' to separate true from false variants. As SNPSVM is designed to create and use a variety of models, the sensitivity and specificity of a given call set may vary substantially depending on the model used. While recognizing that results will differ for those who create their own model, we here quantify sensitivity and specificity using the model developed in Section 2.1. ROC curves generated for four variant callers [SNPSVM, UnifiedGenotyper (McKenna *et al.*, 2010), samtools (Li *et al.*, 2009) and FreeBayes (Garrison and Marth, 2012) and UnifiedGenotyper with VQSR] are shown in Figure 1. The GATK's UnifiedGenotyper, samtools and FreeBayes all display similar levels of sensitivity and specificity, as expected given the similar SNP-calling model used. The sensitivity of variant calls made by the UnifiedGenotyper is increased significantly when VQSR is used, particularly when the false-positive rate is greater than about 2×10^{-5}. At all levels of specificity, SNPSVM achieved greater sensitivity than the other callers, in some cases, substantially so. For instance, at specificity levels high enough to generate two false-positive calls in every 10 000 bases examined, SNPSVM identified 87.5% of true-positive variants, while the UnifiedGenotyper identified 77.9%, UnifiedGenotyper with VQSR 76.4%, Samtools 75.0% and FreeBayes 75.3%. Similarly, at specificity levels near three false positives per

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

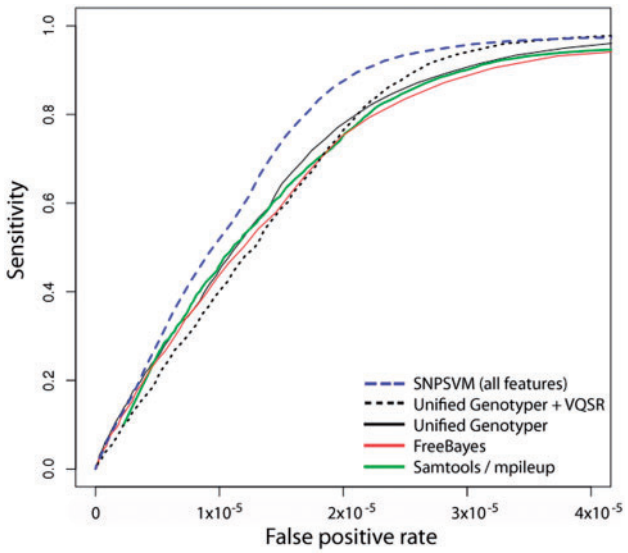

Fig. 1. Specificity and sensitivity of several variant-calling algorithms calculated from NA12878 truth set

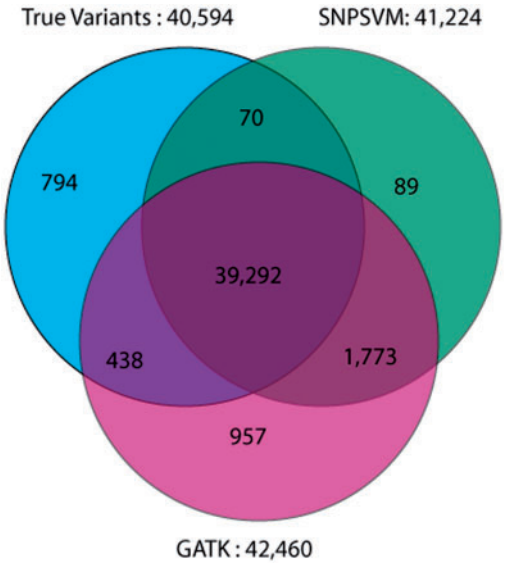

Fig. 2. Intersection of NA12878 variant calls for several algorithms

10 000 bases, SNPSVM identified 95.6% of all true positives, while the UnifiedGenotyper, UnifiedGenotyper + VQSR, Samtools and FreeBayes found 90.6%, 94.1%, 89.5% and 88.7%, respectively. Conversely, at a sensitivity level of 95% (such that 95% of all true SNPs are detected), SNPSVM yields roughly 2.8 false-positive calls per 10 000 bases examined, whereas the UnifiedGenotyper, VQSR, FreeBayes and Samtools yield 3.8, 3.14, 4.6 and 4.5 false-positive calls per 10 000 bases, respectively.

At default settings, SNPSVM yields somewhat less sensitivity but significantly increased specificity when compared with the GATK (Fig. 2). SNPSVM identified 96.9% of all true positives and 1862 false positives (4.5% of all calls), for 21.1 true positives per false-positive call. In contrast, the GATK identified 97.8% of all true positives but some 2730 false positives (6.5% of all GATK calls), or 14.5 true positives per false positive.

3.2 Performance on differing data types

One concern of using the SVM technique to identify SNPs is that the procedure may fail on datasets with differing error profiles. Different enrichment procedures, such the RainDance emulsion PCR technique and Agilent's Haloplex hybridization capture, may yield different types of errors and therefore may confound a SNPSVM model. To assess the impact of alternative enrichment strategies on the accuracy of variant calling, we sequenced an 800 Kb region of individual NA12878 using the Haloplex enrichment protocol, then called variants on this data using the same model used in the sensitivity and specificity analysis.

Haloplex data differ strikingly from SureSelect data. In particular, the Haloplex procedure involves creating DNA fragments through the use of restriction endonucleases in place of sonication. After alignment, many reads will share identical start and end sites, and thus will likely be flagged as PCR duplicates and removed from the analysis. Removing such reads yields low coverage, thus the analysis pipeline must be modified to retain potential PCR duplicates, in turn increasing false-positive rates due to PCR errors. Additionally, in our data, the Haloplex procedure did not always correctly trim adapter sequences from the reads, generating more potential false positives.

After aligning the reads with BWA (0.6.1) and performing base quality score recalibration and local indel realignment with the GATK (version 1.6), we called variants using both SNPSVM and the GATK's UnifiedGenotyper (using a quality cutoff of Q30) and compared the results with the NA12878 consensus truth set. The truth set contained 746 variants in the target region with a Ti/Tv ratio of 2.51. SNPSVM identified 690 variants with a Ti/Tv ratio of 2.49, of which 655 matched the truth set. Among the 35 variants identified by SNPSVM not in the truth set, the Ti/ Tv ratio was 1.05, suggesting a mix of false and true variant calls. In contrast, the GATK identified 956 variants above quality 30.0 with a Ti/Tv ratio of 1.63, in which 701 matched the truth set. In the 255 variants identified by the GATK not in the truth set, the Ti/Tv ratio was 0.55, indicating that nearly all were false positives. Overall, the GATK detected 46 more true positives than did SNPSVM at the expense of including an additional 220 false positives. Thus, despite the fact that the SVM was trained on data with an error profile that likely differed from the Haloplex data, SNPSVM was nonetheless able to accurately discriminate false-positive from true-positive variant calls.

3.3 Performance and scalability

To assess the speed and multithreaded performance of our implementation, we called SNPs on the exome from sample NA12878, restricting the calls to the 51 MB targeted by the exome capture kit. The.BAM file contained approximately 52 million reads. Overall our implementation was 10–20% slower than the GATK's UnifiedGenotyper (Fig. 3). For both callers, the time taken to process the input file reduced in near-linear fashion with the number of threads provided, with each doubling of thread count decreasing run time by 40–45%.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

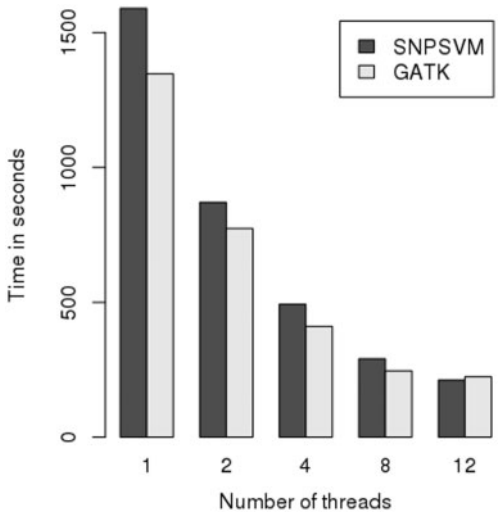

Fig. 3. Time taken to call SNPs on a single exome dataset for differing numbers of threads

3.4 Interpretation of quality scores

Many variant callers associate each variant with a single value that quantifies the confidence of the call, referred to as a quality score. Quality scores are typically phred scaled, such that a score of 10 signifies a 1 in 10 chance of the call being incorrect, while a score of 50 indicates a 1 in 100 000 chance of an incorrect call. In this section, we investigate the distribution and accuracy of these quality scores for the UnifiedGenotyper and SNPSVM, again using the NA12878 consensus set to determine true- and false-positive variant calls.

The initial call set produced by the UnifiedGenotyper contained many SNPs called with high confidence. Some 41 064 of 43 624 SNPs (94%) were assigned quality values greater than 100, indicating that false-positive calls should occur with a frequency of less than 10^{-10}. However, comparison with the NA12878 truth set suggested that 2063 (4.7%) of these SNPs were false positives. Thus, the UnifiedGenotyper significantly overestimates the likelihood that the SNPs it calls are real.

In contrast, SNPSVM did not produce any variant calls with a quality score greater than 60.3, and the mean quality of all calls was 19.8 (the mean quality for calls made from the UnifiedGenotyper was 1092.0). Of SNPs with qualities between 10 and 20, some 4.0% were false positives, and for variants between 20 and 30, 2.3% were false positive, figures that more closely match the expected frequency of errors (1–10% for Q10–20, 0.1–1% for Q20–30). Surprisingly, of the 340 variants with quality between 10 and 20 produced by the UnifiedGenotyper, 77% were false, while 68% of variants with qualities between 20 and 30 were false.

These results are based on the analysis of a single sample using only one possible model of variant calling, and thus results may vary substantially across data produced for differing platforms, enrichment techniques, samples and models. Nonetheless, our analysis used a commonly used platform (paired-end reads on an Illumina HiSeq 2000) and sample, and we believe the high-quality scores reported by the UnifiedGenotyper are representative of typical use.

4 DISCUSSION

Machine-learning techniques such as SVMs offer several advantages over traditional SNP-calling procedures. Most importantly, machine learning techniques allow for multiple and diverse measurements to be incorporated into a statistically robust model of variant calling, allowing for errors produced by miscalled bases, erroneously mapped reads and platform-specific artifacts to be detected in a single statistical procedure. By including features such as strand bias, read position, base quality measurements and mismatch counts into a single variant assessment algorithm, significantly greater sensitivity and specificity may be achieved. In addition, inclusion of these multiple error sources results in quality scores that more accurately reflect the probability that a variant is real when compared with the GATK's UnifiedGenotyper.

Another advantage of the SVM-based technique is that the variant-calling model may be continually improved as more training data become available. External information regarding true and false variant calls from trusted sources, for instance, Sanger sequencing methods or SNP chips, can be used to refine the variant-calling model and improve the accuracy of all future calls. For sequencing centers or laboratories that process many samples and routinely use Sanger sequencing to confirm certain variants, such data are likely to be widely available. In our implementation, we have included several features to aid incorporation of newly acquired data into an existing model to facilitate the refinement procedure.

While we have demonstrated that an SVM-based technique can yield more accurate results than traditional variant-calling techniques, the performance of our algorithm depends critically on the model used. As more accurate training sets are developed and more informative features identified, the sensitivity and specificity may be improved substantially. Additionally, because models and training data are stored as simple text files, they may be easily distributed among researchers once developed.

ACKNOWLEDGEMENTS

The authors thank Jacob Durtschi and Rebecca Margraf for helpful discussion of false-positive variant calls, Justin Zook for providing the NA12878 consensus calls and the ARUP Institute for Clinical and Experimental Pathology for funding this research.

Funding: ARUP Institute for Clinical and Experimental Pathology.

Conflict of Interest: none declared.

REFERENCES

Baten,A.K.M.A. *et al.* (2006) Splice site identification using probabilistic parameters and SVM classification. *BMC Bioinformatics*, **7**, S15.

Boser,B.E. *et al.* (1992) A training algorithm for optimal margin classifiers. In: *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. ACM Press, pp. 144–152.

Browning,B.L. and Browning,S.R. (2009) A unified approach to genotype imputation and haplotype phase inference for large data sets of trios and unrelated individuals. *Am. J. Hum. Genet.*, **84**, 210–223.

Chang,C.C. and Lin,C.-J. (2011) LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, **2**, 27.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

Cortes,C. and Vapnik,V. (1995) Support-vector network. *Mach. Learn.*, **20**, 273.
DePristo,M.A. *et al.* (2011) A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nat. Genet.*, **43**, 491–498.
Garrison,E. and Marth,G. (2012) Haplotype-based variant detection from short-read sequencing. http://arxiv.org/abs/1207.3907.
Howie,B.N. *et al.* (2009) A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. *PLoS Genet.*, **5**, e1000529.
Lee,Y. and Lee,C.-K. (2002) Classification of multiple cancer types by multicategory support vector machines using gene expression data. *Bioinformatics*, **19**, 1132–1139.
Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with BurrowsWheeler transform. *Bioinformatics*, **25**, 1754–1760.
Li,H. *et al.* (2009) The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.
Li,Y. *et al.* (2011) Low-coverage sequencing: implications for design of complex trait association studies. *Genome Res.*, **21**, 940–951.
Nielsen,R. *et al.* (2011) Genotype and SNP calling from next-generation sequencing data. *Nat. Rev. Genet.*, **12**, 443–451.
Noble,W. (2006) What is a support vector machine? *Nat. Biotechnol.*, **24**, 1565–1567.
McKenna,A. *et al.* (2010) The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.*, **20**, 1297–1303.
Schoelkopf,B. *et al.* (ed.) (2004) *Kernel Methods in Computational Biology*. MIT Press, Cambridge, MA.
Shen,Y. *et al.* (2010) A SNP discovery method to assess variant allele probability from next-generation resequencing data. *Genome Res.*, **20**, 273–280.

APPENDIX A

A1 Description of all features

Error probability: Log of probability that the number of reads with variant base was sampled from a binomial distribution with parameter 0.05 relative to probability that the parameter was 0.5 or 0.99.

Binomial probability: Probability that the number of reads with variant base was sampled from a binomial distribution with parameter 0.5 or 1.0, not 0.005, computed as $1 - p_{0.005}/(p_{0.005} + p_{0.5} + p_1)$, where p_x is the likelihood of observing the number of variant reads when the parameter of the binomial distribution is x.

Base quality mean: Mean quality of reference bases and variant bases (two values).

Base quality sum: Sum of all quality scores of reference bases and variant bases (two values).

Mapping quality: Mean mapping quality of reads with reference base and reads with variant base (two values).

Total depth: Total converage at site, capped at 2000 and log transformed.

Dinucleotide repeat counter: Number of dinucleotide repeats extending in both directions (two values).

Read position variance: Variance of position in read where variant allele occurs.

Read position mean: Mean position in read where variant allele occurs.

Strand bias: χ^2 value for test for strand bias of reads with variant bases versus reads with reference bases.

Area mismatch total: Mean number of variant bases per read across all reads mapping to site.

Homopolymer run: Total length of homopolymer run in both directions (two values).

Nucleotide diversity: Deviation in 20 bp window of reference base frequencies from genome-wide average.

Read mismatches: Number of mismatches on reads with variant base and reads with reference base (two values).

Allele balance: Fraction of reads containing variant allele to total read depth.

Nearby base qualities: Mean base qualities of bases aligning in 2 bp window surrounding site (four values) (Not used in final version).

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016