SNPredict: A Machine Learning Approach for Detecting Low Frequency Variants in Cancer

Vatsal Mehra
Marquette University

SNPREDICT: A MACHINE LEARNING APPROACH FOR DETECTING LOW FREQUENCY VARIANTS IN CANCER

by

Mehra, Vatsal

A Thesis submitted to the Faculty of the Graduate School,
Marquette University,
in Partial Fulfillment of the Requirements for
the Degree of Master of Science

Milwaukee, Wisconsin
August 2016

ABSTRACT

Mehra, Vatsal

Marquette University, 2016

SNPredict: A machine learning approach for detecting low frequency variants in cancer

Cancer is a genetic disease caused by the accumulation of DNA variants such as single nucleotide changes or insertions/deletions in DNA. DNA variants can cause silencing of tumor suppressor genes or increase the activity of oncogenes. In order to come up with successful therapies for cancer patients, these DNA variants need to be identified accurately. DNA variants can be identified by comparing DNA sequence of tumor tissue to a non-tumor tissue by using Next Generation Sequencing (NGS) technology. Detecting variants in cancer is a challenging problem because many of these variant occurs only in a small subpopulation of the tumor tissue. It becomes a challenge to distinguish these low frequency variants from sequencing errors, which are common in today's NGS methods. Several algorithms have been designed and implemented as a tool to identify such variants in cancer. However, it has been previously shown that there is low concordance in the results produced by these tools. Moreover, the number of false positives tend to significantly increase when these tools are faced with low frequency variants.

This study presents SNPredict, a single nucleotide polymorphism (SNP) detection pipeline that aims to utilize the results of multiple variant callers to produce a consensus output with higher accuracy than any of the individual tool with the help of machine learning techniques. By extracting features from the consensus output that describe traits associated with an individual variant call, it creates binary classifiers that predict a SNP's true state and therefore help in distinguishing a sequencing error from a true variant.

ACKNOWLEDGEMENTS

Mehra, Vatsal

I would like to acknowledge Dr. Serdar Bozdag for his mentorship during my graduate training and during the course of this project. I would also like to acknowledge member of his lab – David McKean for providing helpful ideas and giving useful feedback while working on this project. I would like to thank Dr. Valerie Trapp-Stamborski from Blood Center of Wisconsin and Dr. Mary Shimoyama from Medical College of Wisconsin for helping me shape some of the ideas and providing datasets used in this project.

TABLE OF CONTENTS

List of Tables

List of Figures

I. Introduction

Next Generation Sequencing (NGS) has immensely enhanced our knowledge of processes involved in cancer. Malignant tumors have complex architecture and detecting somatic mutations resulting from such tumor growths is of utmost importance from a therapeutic point of view. In recent years, there has been a significant effort towards discovering such mutations (Meyerson et al. 2010), however mining these mutations using traditional variant calling tools has not been straightforward for multiple reasons. First, cancer cells constantly introduce new variations in their sub populations and hence any single variation usually exists in low frequency (Ding et al. 2010; Stephens et al. 2012). This subclonal variation is offset to some extent by increasing sequencing depth in NGS technologies however this can be cumbersome given the increase in costs associated with deep sequencing. Second, widely reported sequencing errors with NGS technologies further complicate mining of such mutations (Dohm et al. 2008). Sequencing methods are by nature imperfect and prone to errors that lead to incorrect base calling which can affect proper alignment of short reads but more importantly this may also prevent correct identification of variants that are present in low frequency. These sequencing errors are highly variable and can differ between different NGS platforms such as whole genome sequencing and targeted exome sequencing, different lanes within a panel, as well different genomic locations or sequence motifs across the chromosomes (Allhoff et al. 2013). Even sample isolation, formalin fixation and other library preparation methods can introduce nucleotide changes that may contribute to sequencing errors (Williams et al. 1999; Robasky et al. 2014). Such errors can obfuscate the results of the existing variant callers, thereby increasing the risk of flagging such false positives as mutations of clinical

value. Third, tumor impurity can also skew detection of mutations. For instance, infiltration of non-cancerous cells may conceal low frequency mutations inherent to cancerous cells. This is especially relevant in liquid tumors such as blood tumors, which are the focus of this study.

Table1: Terminology used for building and evaluating performance of classifiers

Term	Description
True Positive (TP)	Variant calls that were correctly predicted as somatic mutations
False Positive (FP)	Variant calls that were predicted as somatic mutation but were not present in the variants validated by Blood Centre of Wisconsin
True Negatives	Variant calls that were correctly predicted as non-somatic variants as they were also not captured by Blood Centre of Wisconsin
False Negatives	Variant calls that are clinically relevant but not predicted as such by the pipeline
Sensitivity (S)	TP / (TP + FP)
Precision (P)	TP/ (TP + FN)
Training set	Pseudo tumor dataset used to build classifiers
Test set	Pseudo tumor dataset used to test the performance of classifiers
Target set	Real tumor dataset used to evaluate the performance of trained classifiers

Several variant callers have been introduced in the past few years (Koboldt et al. 2012; Saunders et al. 2012) in an effort to address aforementioned issues. Broadly, all these variant callers share a high degree of similarity. They require a reference genome, a pair of matched tumor and normal sample as inputs, followed by execution of statistical

approaches like Bayes theorem or logistic regression to find the posterior likelihood of a particular variant being somatic or not based on some priors and then finally among those selected, they apply various filters based on the characteristics or "features" associated with that variant (surrounding genomic motif, strand bias, distance to 3` end etc.) in order to remove the false positives. These approaches work considerably well provided the variant frequency is not very low (for e.g. less than 10%) and the short reads associated with a particular variation are high in the tumor sample. However, for tumors with low frequency variations, false positives begin to significantly outnumber the clinically relevant variations as will be highlighted in this study. Moreover, a previous study that compared five Illumina SNP detection pipelines showed that the existing variant callers have only about 57% concordance in calling variations (O'Rawe et al. 2013). Similarly, a comparative analysis for detecting single nucleotide variants found substantial differences in the number and characteristics of the calls produced by different variant callers (Roberts et al. 2013). It has also been noted that the performance variation among these variant callers can be a function of allelic fraction of the mutation in the tumor samples specific to an investigation (Xu et al. 2014) which further increases complexity in choosing a consistent tool across different types of tumors.

Another drawback with existing somatic variant callers is that their performances are usually evaluated with a pair of matched normal and tumor samples i.e. the assumption that both samples come from the same patient. This requirement is hard to achieve especially in a clinical setting where often investigators have no alternative but to use unmatched samples. This problem can potentially increase the number of false positives

because of the higher number of ambiguous non-somatic variants that just reflect differences between two individuals.

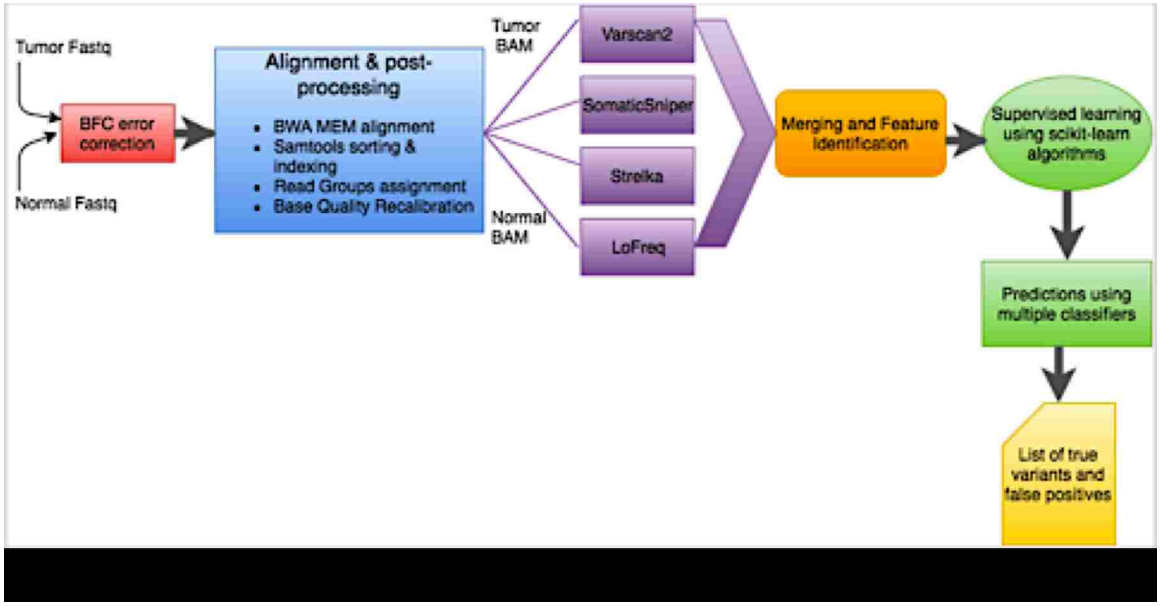
Given these challenges, there is a dire requirement for a tool or a pipeline that potentially alleviates such issues and reliably predicts clinically relevant variants. This study presents a novel somatic variant detection pipeline that overcomes the shortcomings of individual variant callers using a multistep process (Figure 1). The pipeline is initiated by considering paired end reads from an unmatched pair of normal & tumor samples as inputs to a pre-processing step that filters potential Illumina sequencing errors. This filtering of potential errors is followed by alignment of reads to a reference genome and processing of these aligned reads including assignment of each read to a *read group* (for downstream tools) and re-evaluation of the base quality. Processed reads are then subjected to variant calling using a combination of variant callers namely VarScan2 (Koboldt et al. 2012), SomaticSniper (Larson et al. 2012), Strelka (Saunders et al. 2012) and Lofreq (Wilm et al. 2012) in order to determine somatic single nucleotide polymorphisms (SNPs). These variant calling outputs are then combined together in a *merging step* to increase sensitivity, thereby resulting in a super set of all possible somatic variants present in the tumor. Since this super set consists of many false positives especially in cases where frequency of actual variants is low, it is essential to have a module that can aid the investigators in highlighting these false positives.

The *merging step* is followed by a *learning step* in which a training set is used to identify "features" associated with a particular call in the super set. In total, eleven features are used from four somatic callers. These models are then used over a target set of real blood

tumors to predict somatic mutations. The accuracy of the models used was calculated using F1 score, which is the harmonic mean of the Sensitivity (S) and Precision (P):

[redacted]

The terminology used in this study for these and other associated terms are described in Table 1.



[redacted]

The usage of machine learning techniques to predict variants is not novel. Previously, Fang et al. created SomaticSeq that used an Adaptive Boosting model to construct a classifier based on decision trees (Fang et al. 2015). Similarly, Kim et al. also proposed a method that combines results from multiple variant callers and then implements a logistic regression with feature-weighted linear stacking (FWLS) to improve accuracy of that combined set. (Kim et al. 2014). However, both these studies failed to test their algorithms on unmatched samples. In addition to testing the algorithm on unmatched samples, the present study also uses multiple learning algorithms like K-nearest

neighbors, Logistic Regression and Linear SVC in order to assign higher confidence to the predictions that are consistent across multiple algorithms.

II. Materials & Methods

Generation of Next-Gen Sequencing data

Collection of tumor samples, extraction of genomic DNA and sequencing of samples were all performed at The Blood Center of Wisconsin (BCW). Previously de-identified patient samples were sequenced at BCW for the purposes of diagnostic test validation. Briefly, BCW extracted genomic DNA from blood tumor samples, which were then sequenced on the Illumina Mi-Seq platform. Paired-end sequencing was performed, with an average read length of 150 base pairs. Sequencing was restricted to genomic loci strongly associated with cancer mutations in leukemia patients, based on previous literature (Chang & Li 2013). Targeted sequencing was performed to enrich for these genomic loci with maximum depth of coverage of up to 10,000X. This pre-existing sequencing data was analyzed for this project.

To aid the development of machine learning classifiers and extraction of standard features, training datasets were generated which contained known genetic mutations associated with Acute Myeloid Leukemia (AML). Twenty seven such training datasets were generated by spiking in multiple SNPs into normal samples. Once the machine learning classifiers were established based on training datasets, the efficiency of classifiers was validated by running the classifiers against test datasets. Target datasets were generated from 22 true tumor samples, on which Sanger sequencing was performed to reveal identity of true SNPs, thereby enabling validation of machine classifiers.

Five non-cancerous samples were sequenced similarly as described above and were used for both identification of SNPs with unmatched samples and also for creating spiked-in

training datasets. The training datasets were created at BCW using an R script that spiked in reads generated from real tumor samples into normal samples with the normal-tumors ratios of 80:20 and 90:10.

Preprocessing, alignment & post-processing of short reads

Reads generated from NGS were pre-processed with **bfc** (Li 2015) , a software that corrects multiple errors inherent with Illumina reads by eliminating singleton k-mers that generally imply sequencing errors. Reads were aligned to the human genome build 37 (GRCh37) using **bwa mem** (Li & Durbin 2010) algorithm. Following alignment, aligned reads were compressed, sorted and indexed using Samtools (Li 2011). To allow compatibility with downstream software tools, aligned reads were assigned *read groups* using Picard Tools (Van der Auwera et al. 2013). Read groups help GATK tools to identify whether a set of reads were sequenced together on a specific lane and therefore aid in compensating for differences across subsequent sequencing runs.

Variant calling algorithms generally rely on quality scores of individual bases produced by the sequencing platform. Hence it is important to correct for possible over/under estimations of as a result of technical errors. For this purpose, aligned reads were passed through GATK's *Base Recalibrator* (Van der Auwera et al. 2013), which applies machine learning to model technical errors and then recalculates the quality score of individual bases.

Variant Calling Tools

All existing variant callers evaluate their own performance using matched normal-tumor sample which is not always accessible in a clinical setting. Hence, its essential to evaluate the performance of existing somatic callers on unmatched samples. For this purpose, seven variant callers were tested against datasets generated from three unmatched normal-tumor datasets. The specific variant callers tested were *VarScan2* (Koboldt et al. 2012)*, Strelka* (Saunders et al. 2012)*, CaVEMan* (Gerstung et al. 2014)*, Scalpel* (H. Fang et al. 2015)*, SomaticSniper* (Larson et al. 2012), *MuTect* (Cibulskis et al. 2013) and LoFreq (Wilm et al. 2012). The accuracy was evaluated based on the ability of each caller to detect SNPs and indels (variants introduced by insertion or deletion of multiple bases). Selected callers are designed based on different strategies to detect somatic variants and collectively have their own strengths and weaknesses. Out of these seven somatic callers, only *VarScan2, Strelka, SomaticSniper and Lofreq* were used for the machine learning analysis in this This selection was made because *Scalpel* focuses only on indels (not part of this study) while *CaVEMan*'s output directory structure makes it difficult to use for merging variant results from a large set of samples. Although *MuTect* primarily focuses on SNPs, its filtered variant results are in *txt* format which makes it incompatible with the downstream tool PyVCF (explained below). Following are brief descriptions for each of the tools and their usage in the context of this study:

Varscan2

VarScan2 (Koboldt et al. 2012) is a mutation caller for targeted exome and whole genome sequencing data. The software assigns a particular genotype as a heterozygous or

homozygous variant depending on if the variant base has at least a minimum variant frequency of 0.20 for all reads (This default was modified to 0.05 for this study in order to capture low frequency variants). Following this, the software performs a Fisher exact test between the reference and variant supporting reads for both normal & tumor samples and calls each variation as either somatic or germline. This categorization is done based on whether the normal sample was a homozygous/heterozygous reference. Lastly, several filters like variant position in supporting read relative to read length, distance to 3' end, fraction of variant reads from forward strand (to avoid strand bias), mapping quality difference between reference and variant are applied to correct for location and mapping quality of reads to eliminate false positives. As input, the software requires pileup files that consist of base pair information at each chromosomal location. Pileup files for both the normal and tumor samples were generated using **samtools mpileup** program, which were then passed along with a reference genome to generate variant calling results containing both SNPs and indels (only SNP output used in this study). To filter non-somatic calls, *processSomatic* command was used.

SomaticSniper

SomaticSniper (Larson et al. 2012) provides SNP detection by doing a Bayesian comparison of genotype likelihoods in the tumor and normal sample. It further uses somatic detection filters like location of site from a predicted indel, mapping quality, read coverage and consensus quality. It provides numerous options (e.g. changing prior probabilities of finding somatic mutation) that can be used along with the main program *bam-somaticsniper* to make the algorithm more or less sensitive. Each of the called variant is given a '*somatic score*' which is phred-scale posterior probability that the

variant is somatic or not. Using SomaticSniper's parameters, calls with somatic score less than 20 and reads with mapping quality less than 1 were filtered.

LoFreq

LoFreq is a SNP & indel detector that utilizes base call quality by modeling sequencing errors in order to distinguish them from true variants. Using viral, bacterial and human datasets, it was shown to predict variants below the sequencing error rate (Wilm et al. 2012). LoFreq can also utlize a user-provided dbSNP variant file to remove known germline variants. It's sensitivity can be adjusted by changing the parameter *tumor-mtc-alpha,* which signifies the value of alpha for the bonferroni test for the tumor. To increase the sensitivity, this value was increased from 1.0 (default) to 1.5.

Strelka

Strelka (Saunders et al. 2012) can predict both somatic SNPs and small indels. It uses a Bayesian probability model in which normal sample is considered mixture of germline variant with noise, while tumor sample is considered a mixture of normal sample with somatic variation. This assumption towards nature of normal and tumor samples, allows for better accuracy in highly impure tumor samples. It requires that reads are aligned using the *bwa aligner* and also uses a *config* file that can be used to vary various filters like min. allele fraction, mapping quality, prior probability of any site being a somatic mutation and the expected rate of heterozygosity in normal sample etc. All these parameters were kept to default values except *isSkipDepthFilters,* which was changed from 0 to 1 as required for data generated from targeted sequencing.

Scalpel

Scalpel (H. Fang et al. 2015) only provides indel detection. It uses a de-Bruijn graph based strategy to find insertions and deletions for any kind of sample. For the purpose of this study, the somatic mode of operation was chosen. It also requires a reference genome and a bed file along with tumor-normal sample set. Once the indels are provided, an "export" option is used to filter the relevant indels depending on the criteria required. All parameter values were kept to default.

CaVEMan

CaVEMan (cancer variants through expectation maximization) (Stephens et al. 2012) only detects SNPs. The algorithm uses a Bayesian classifier to estimate posterior probability for a genotype at each base and then applies post processing filters. The workflow involves implementation of multiple scripts for creating a config file, which consists of location of the normal, tumor BAM files and reference fasta file. This is followed by creation of separate segments using *caveman split* command, which breaks up the analysis for each of the sequence chromosome. After merging of segments using *caveman merge*, *caveman estep* is the command to finally call variants. For this study, the parameter —min-base-qual was adjusted to 30 to filter all bases with lesser quality. All other parameters were kept to default values.

MuTect

MuTect (Cibulskis et al. 2013) detects somatic point mutations by first preprocessing the reads to filter the reads with low quality and mismatches which is then followed by usage of a Bayesian classifier that checks whether tumor is non-reference at a site. These sites

are then checked for their non-existence in normal sample. Post processing is the final step in which artifacts of NGS are eliminated. Since it requires aligned reads to have *read groups* associated with them, *picardtools* was used to fulfill that requirement. Default values were used for all available parameters.

Merging variant calling results

For creation of training data, machine learning and prediction, four somatic variant callers were selected: VarScan2, SomaticSniper, Lofreq and Strelka. The training data consisted only of SNP calls and the indel calls produced by VarScan2, Lofreq and Strelka were ignored for this study. After running the variant calling pipeline over 27 pseudo tumor datasets, the VCF (Variant Calling Format) results for each of these pseudo tumor dataset from each tool were first concatenated and sorted using Unix command line tools, which was followed by merging of these concatenated results using Pandas (McKinney 2011) which is a Python data analysis library. To use consensus among somatic callers for a particular variant as one of the features to aid in prediction results, each variant call was assigned a weight which was calculated by simply adding the number of tools that reached consensus for that particular call with values ranging from 4 (maximum consensus) to 1 (minimum consensus). Merging was performed by using *position* of the variant as the common feature among all variant callers. Many variant call positions did not overlap after the merge as the consensus among callers was less than four. Therefore, Pandas introduced missing value *NaN* in the merged table for such calls. For example, if Varscan and SomaticSniper called a position 128202677 but Strelka and LoFreq did not, for this particular location, features associated with Strelka and LoFreq would have *NaN* entries. These missing values were converted into the mean value for that feature which

is a standard practice used when creating learning models with data that has absent entries.

PyVCF

In order to extract features that would help the learning algorithms to make better predictions, it was essential to find out whether the true positives (TPs) and false positives (FPs) had any characteristic traits that would vary significantly. For this purpose, *PyVCF* (Kelleher et al. 2013) which is a variant call format (VCF) parser built in Python was used. It parses the VCF file and stores each variant call and its characteristics in a *record object*. This record object consists of multiple functions that allow access to features associated with a variant call like position, quality, reference and alternate base, tumor information like depth, variant quality etc. *PyVCF* was used for parsing the VCF output from each of the variant callers and the resulting *data frames* were finally merged as described above. All plots in this study were created using Python's visualization library *Matplotlib* (Hunter et al. 2014).

Scikit-learn

To create classifiers and perform other machine learning analysis, *scikit-learn* library (Pedregosa et al. 2011) for Python was used to get access to numerous functions appropriate for data mining and data analysis. Since this study treats the problem of identifying TPs and FPs as a classification problem, three supervised learning algorithms were used for the analysis: logistic regression, linear support vector machine (SVM) and k-nearest Neighbor (kNN).

Logistic Regression (LR)

Logistic regression (LR) is a linear model of classification in which a logistic function is used to model probabilities for a particular outcome of a single trial. The type of LR used for this study was binary LR since the outcome for dependent variables only has two possible outcomes. The LR method simply calculates the log of the ratio of the odds of possible outcomes.

Linear Support Vector Machine (SVM)

Linear SVM algorithm simply builds a binary linear classifier that consists of categories that are clearly divided in the feature space. These categories are formed using training dataset in such a way so as to have a clear distinction between them. The predicted data points are then mapped onto one of these categories based on whichever side they fall on.

k Nearest Neighbors (kNN) Classification

kNN is an instance based learning model where instances of training data are simply stored instead of constructing an internal model. These instances form a feature space with different classes built for each possible output. A particular object in question is assigned a class based on the class of its *k* nearest neighbors. In this study, the classifiers were built with *k=7*.

Privacy of patient data

The MCW/FH IRB committee, which serves both BCW and Medical College of Wisconsin (MCW), reviewed the project and determined that it does not meet criteria for human subject research at 45 CFR 26.102 and therefore does not require further review

by the MCW/FH IRB. The Marquette University (MU) office of research compliance confirmed that based on the MCW/FH IRB decision, an MU IRB would not be needed, as the activity would not constitute research involving “human subjects”.

III. Results and Discussion

Performance of Variant Callers on unmatched normal-tumor samples

In order to evaluate the performance of existing variant calling algorithms on pairs of unmatched normal-tumor datasets, seven somatic variant callers were tested on 3 Acute Myeloid Leukemia (AML) samples *S19, S31* and *S41* that consist of variants with varying frequency. Other than evaluating performance on unmatched pairs of normal-tumor data, another essential component of this comparison was to replicate the already reported discrepancy between the variant calling results from different callers. BCW used Sanger sequencing to validate 6 somatic mutations in *S19* and 3 mutations both in *S31* and *S41* which were used as true positives for this comparison analysis.

Table 2: Performance evaluation of variant callers on unmatched normal-tumor sample

Samples	MuTect	Somatic Sniper	Strelka	VarScan2	Caveman	Scalpel	Lofreq
S19	0.2	0.14	0.06	0.27	0.25	0.01	0.12
S31	N/A	N/A	0.25	0.01	N/A	0.02	0.04
S41	0.33	0.09	0.09	0.05	0.1	0.01	0.4

To measure the performance of each variant caller, the TP to FP ratio was compared across six variant callers. As can be noted from Table 2, performances of these callers across all tumors were considerably low. VarScan2 performed best in capturing most SNPs (5 out of 6) in *S19* but the variant frequency of all mutations present in this tumor

was greater than 30% which is not always the case with AML. *S31* provided a better test since the variant frequency was as low as 8%. With this tumor, Strelka outperformed the other callers by identifying maximum number of true positives with relatively less number of false positives. As Mutect, SomaticSniper and CaVEMan capture only SNPs, their performance on *S31* was not evaluated since BCW only validated indel locations for this tumor sample. *S41* also consisted of mutations with frequency as low as 8% and although Caveman captured maximum TPs, Lofreq had the highest TP to FP ratio as it was able to filter a large number of FPs.

Table 3: Features from variant callers used for building classifiers

Feature	Description
Read Depth (all 4 callers)	Read depth of variant supporting base
VAF (Varscan2 & LoFreq)	Variant allele frequency for the base in tumor sample
Somatic Score (SomaticSniper)	Phred scale score signifying the likelihood of a particular call being somatic. Higher value means high probability of call being somatic
Somatic Score (VarScan2)	Score in Phred scale derived from somatic p-value
Quality Score (Strelka)	Quality score reflecting the joint probability of a somatic variant and NT
Quality Score (Lofreq)	Phred-scaled quality score for the assertion made in ATL field. High number reflects high confidence call.
Weight	Consensus among variant callers for a particular call

Admittedly, these three real tumor datasets do not consist of enough mutations to highlight the low frequency that is characteristic of blood tumors. However, it was essential to evaluate the discrepancy between the results from existing variant callers especially when they are dealt with an unmatched normal-tumor dataset. This small scale

comparison analysis corroborates previous work (Roberts et al. 2013) and highlights that there is a strong need to combine the results of these variant callers as their individual usage will likely lead to incomplete and misleading results.

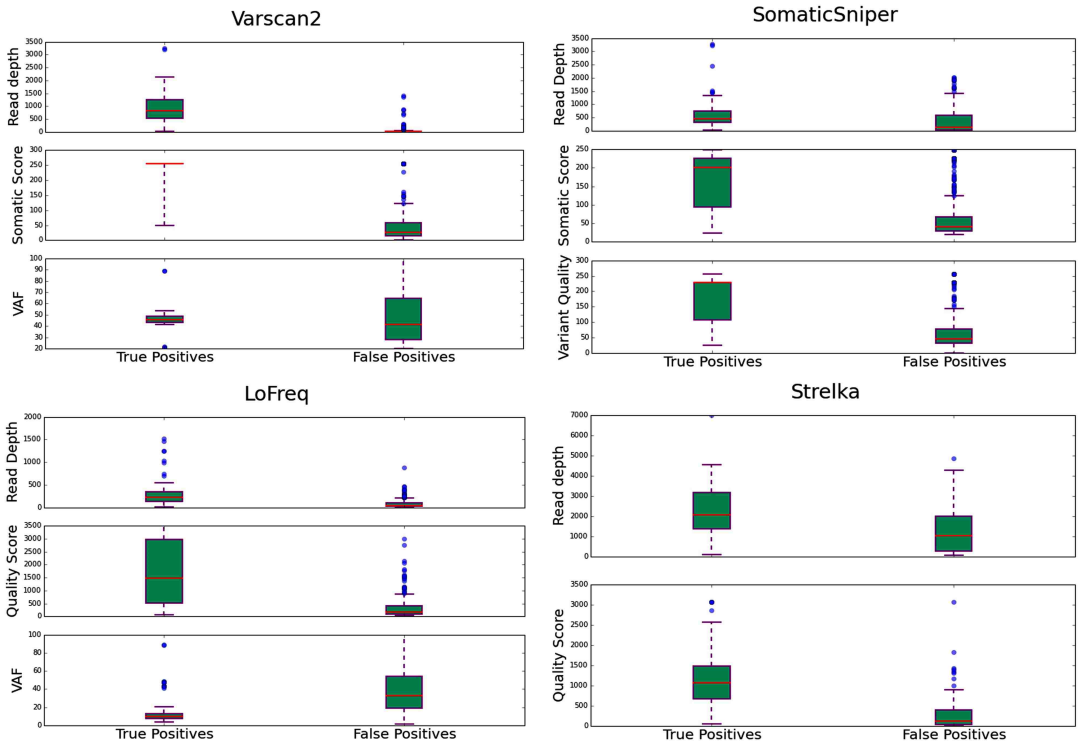


Figure 2: Feature variation among true positives and false positives for different variant callers

Feature variation among true positives and false positives

To identify the features or characteristics of a particular variant call that may be useful in predicting the final state (TP or FP) of that variant, traits associated with each variant were extracted from the VCF files produced by each of the variant callers. These traits and their corresponding descriptions are presented in Table 3. This information was then plotted as box plots to visualize the feature distinction among TPs and FPs (Figure 2).

Each caller's TPs and FPs showed similar patterns for some traits but some traits' variation across TP and FP differed across tools. VarScan2 showed significant correlation with higher values of *read depth* and *somatic score* corresponding with TPs but no such correlation was observed with *Variant Allele Frequency* (VAF). SomaticSniper showed clear differences among TPs and FPs with higher values of s*omatic score* and Variant Quality correlating with TPs but the average variant *read depth* values for the two sets were almost equal. For LoFreq, higher r*ead depth* and Phred *quality score* was correlated with TPs but higher VAF was associated with FPs which could be due to its primary focus on catching low frequency variants.

Strelka also showed positive correlation for high Read Depth and Quality Score with TPs.These results indicate that the VCF files do contain useful information associated with a particular variant call that can aid the learning algorithms to train classifiers for predicting a true variation and distinguishing it from a sequencing error.

Classifier creation and prediction using spiked-in datasets

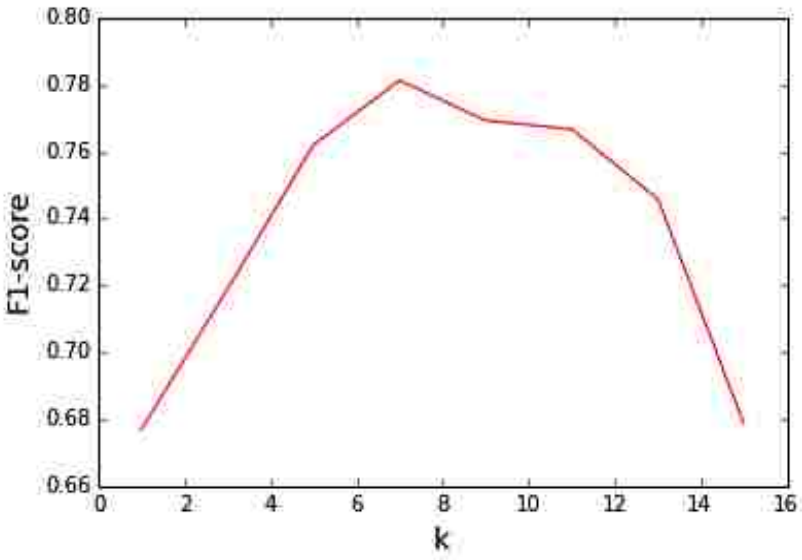


Figure 3: Performance variation of k-Nearest Neighbor (kNN) with varying values of k

In total, there were 308 mutation calls that were collectively identified by the somatic callers for the 27 pseudo tumor datasets. To build the classifiers, a total of 11 features (Table 3) from the four

aforementioned variant callers were used.

Hence, the merged dataset's dimensions were *308 x 11*. It was divided into a *training set* and a *test set* with a ratio of 3:2 using scikit learn's function *train_test_split*.

All three classifiers were then trained by fitting the *training* set. The state of the variant calls in the *test set* were predicted as either a 0 (true mutation) or 1 (sequencing error) using the trained classifiers. Finally, to evaluate the performance of these classifiers, their F-1 scores were plotted. Performance of the kNN classifier was observed over odd values of k (Figure 3) and F1-score was highest with k=7 and this is the value that was chosen for cross validation purposes with kNN. This value of k also averages out any bias that would arise from only the very close neighbors (k = 3).

Stratified k-fold cross validation analysis was used for all classifiers' performance evaluation as it has been previously noted that rearranging the dataset such that each pair of training and test set is a good representative of the whole, helps in reducing bias when compared to regular cross validation (Kohavi 1995). The results are plotted in Figure 4. kNN with k=7 performed the best on the training set with average F1-scores of 74% (74% sensitivity & 75% precision) with 10-fold cross validation and 75% (77% sensitivity & 75% precision) with 15-fold cross validation.

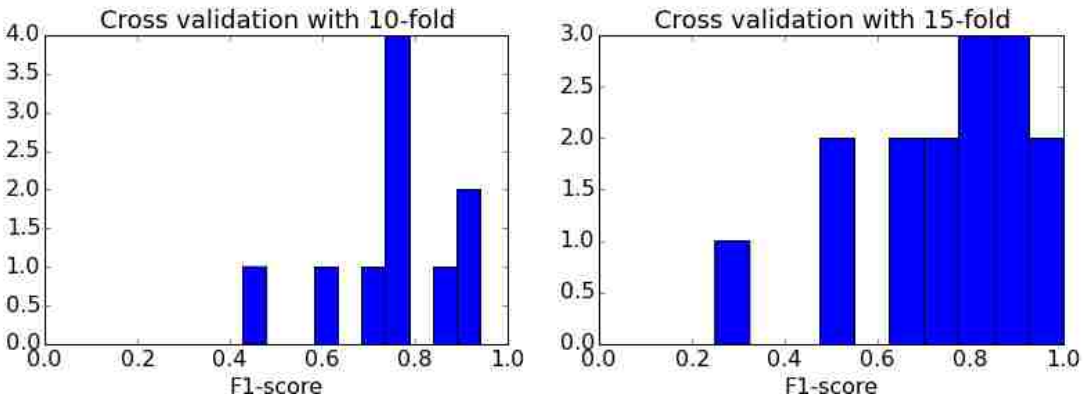


Figure 4: Cross validation with kNN on spiked-in tumors

Next, the performance of Logistic Regression classifier was tested and cross validated. The results are plotted in Figure 5. This classifier achieved average F1-scores of 65% (77% sensitivity and 58% precision) with 10-fold and 63% (79% sensitivity and 56% precision) with 15-fold cross validations.

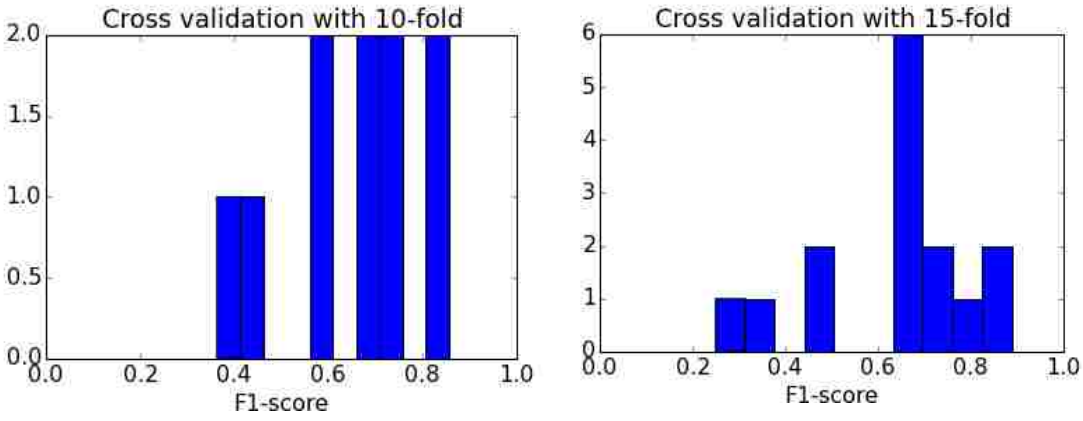


Figure 5: Cross validation with Logistic Regression on spiked-in tumors

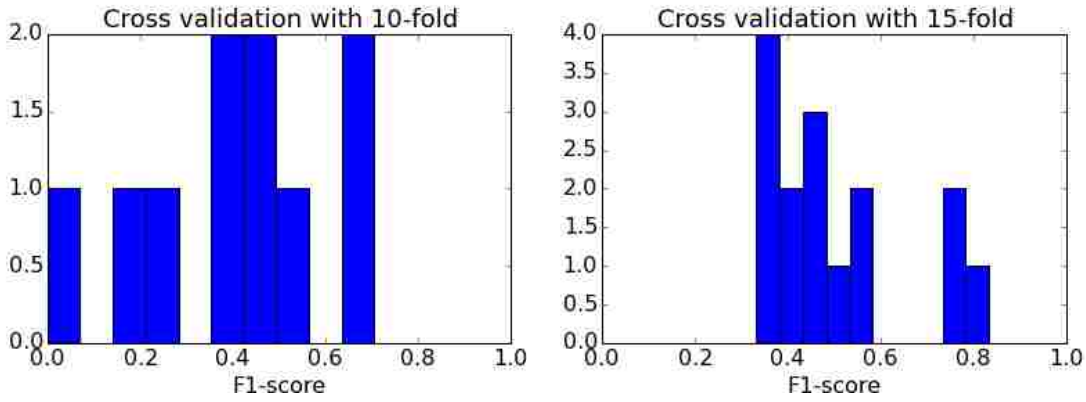


Figure 6: Cross validation with Linear SVC on spiked-in tumors

Finally, the training dataset was used to evaluate the performance of Linear SVC classifier which is plotted in Figure 6. This classifier's performance was worse compared to the other two with F1-scores of only 43 (58% sensitivity & 40% precision) with 10-fold and 45 (57% sensitivity & 46% precision) with 15-fold cross validations.

Tool	Sensitivity	Precision	F1-score
SNPredict (kNN)			
SNPredict (Log Reg)	82	54	61
SNPredict (LinearSVC)			
Varscan2	10.5	19.5	14
SomaticSniper			
LoFreq	24	56	34
Strelka			

The performance of these classifiers was also compared with each of the individual variant calling tools and SNPredict clearly outperformed each of these individual tools (Table 4). SNPredict's performance with kNN and LR outperformed every other algorithm with Strelka performing slightly better compared to SNPredict with Linear SVC. Linear SVC's weak performance highlights the fact that the variation of values for features among true positives and false positives is not clearly distinct for all features.

Classifier creation and prediction using real tumor datasets

After training and building the classifiers using the pseudo tumor dataset, the pipeline was then tested on the real tumor datasets. Before using the classifiers, SNPs in the merged dataset underwent a filtering process which was performed by running the variant calling pipeline over a pair of normal-normal sample. The SNP dataset collected from all variant callers using this normal-normal pair were checked against the merged SNP dataset from real tumors for potential overlapping SNPs. There were 61 such SNPs that were common in these two sets which were filtered from the classifier prediction analysis as they were clearly false positives.

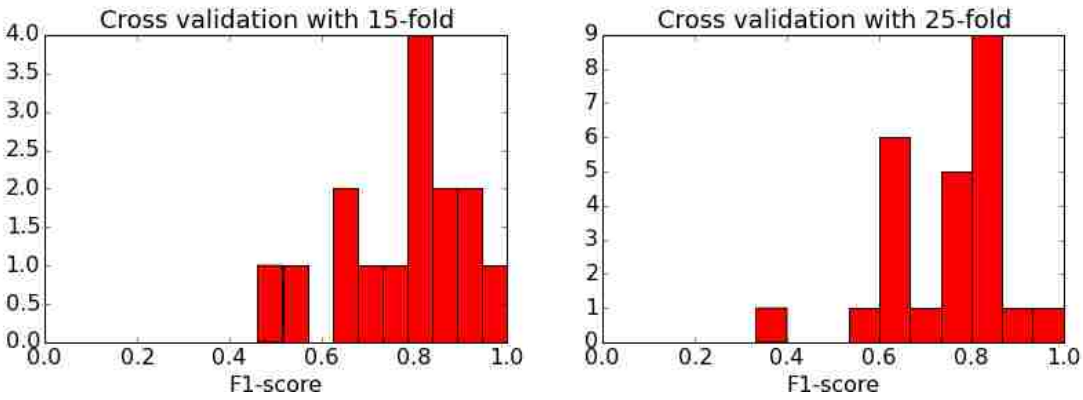


Figure 7: Cross validation with kNN on real tumors

The F1-scores achieved by SNPredict on real tumors with kNN, Logistic Regression and Linear SVC are plotted in Figure 7,8 and 9 respectively. *kNN* again did well with F1-scores of 77 (84% sensitivity and 76% precision) with 15-fold CV and 76 (83% sensitivity and 76% precision) with 25-fold CV. *Logistic Regression* also showed good prediction performance with F1-scores of 73 (76% sensitivity and 71% precision) with 15-fold CV and 71 (78% sensitivity and 70% precision) with 25-fold CV. *Linear SVC* again showed weak performance with F1-scores of 29 (31% sensitivity and 40% precision) with 15-fold CV and 20 (21% sensitivity and 27% precision) with 25-fold CV.

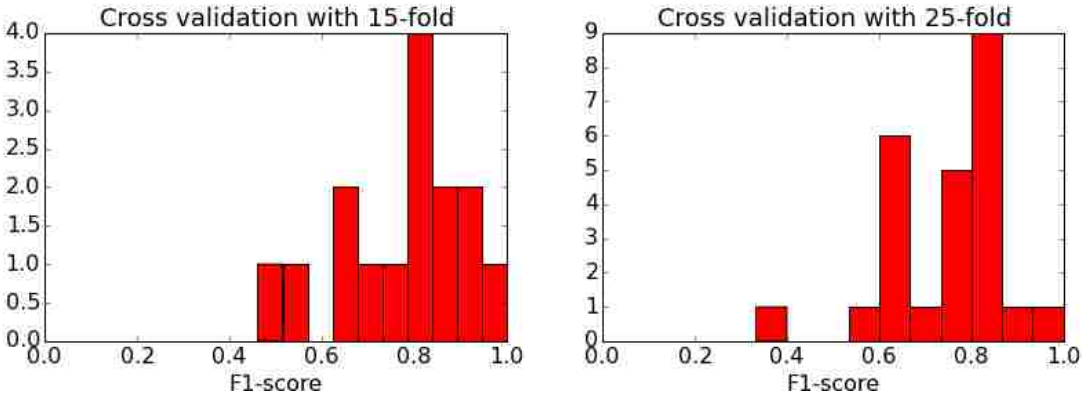


Figure 8: Cross validation with Logistic Regression on real tumors

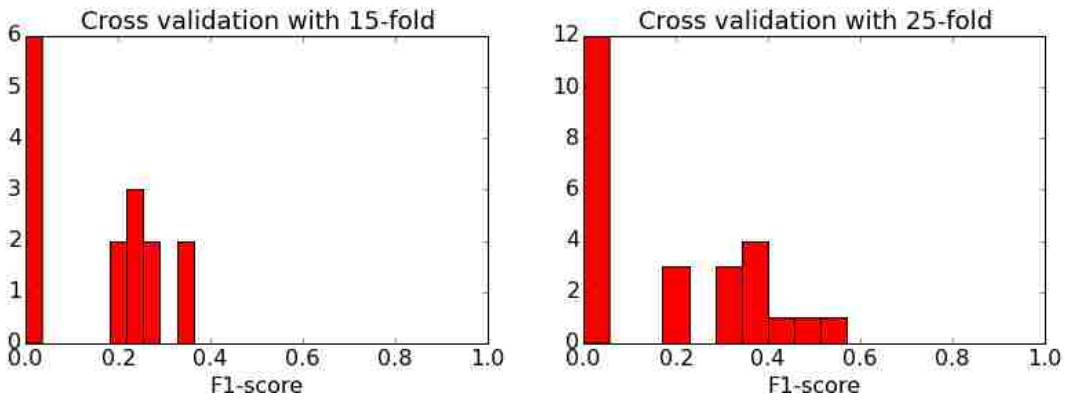


Figure 9: Cross validation with Linear SVC on real tumors

Table 5: Performance of SNPredict's classifiers built using spiked-in data on real tumor datasets (in %)

Algorithm	Sensitivity	Precision	F1-score
SNPredict (kNN)	13	93	23
SNPredict (Log Reg)	14	100	24
SNPredict (Linear SVC)	13	99	23

In order to evaluate the similarity in features associated with TPs and FPs in spiked-in tumor datasets and features associated with TPs and FPs in real tumor datasets, classifiers built using spiked-in tumor datasets were used for predicting true SNPs and sequencing errors in real tumor datasets. The performance results of all three algorithms used by SNPredict are plotted in Table 5. Poor F1-scores highlight the fact that the values of features associated with TPs and FPs in spiked-in tumors may not reflect the true

complexity involved in real tumors and hence one should be careful before building machine learning classifiers using pseudo-tumors in order to make predictions on real tumors.

Since a mutation is called a False Positive if its predicted as true mutation by SNPredict but not identified by Sanger sequencing at BCW, for validation of SNPredict's results, it was essential to use a secondary source that documents known cancerous mutation. One such source is Database of Curated Mutations (DoCM) (Krogan et al. 2015) which maintains lists of known, disease-causing mutations including SNPs associated with various cancers. Real tumors in this study were from the patients suffering from AML and hence SNPs associated with this cancer were downloaded and compared against. As expected, all mutations reported by BCW were also documented at DoCM. However, SNPredict was also able to identify eight additional SNPs that were not reported by BCW (and hence incorrectly identified as false positives in this analysis) but were present in the DoCM. The location and the corresponding frequency of these eight SNPs is highlighted in Table 6. In order to further verify that these SNPs were indeed actual variation in the tumor, BAM files associated with tumors were visualized and compared against the normal

Table 6: SNPs captured by SNPredict that were missed by BCW pipeline

Chr: Position	Variant Allele Frequency (%)
2: 25457242	44
2: 198267359	31
12: 25398284	37
13: 28602340	35
13: 28608281	16
13: 28602381	1.09
17: 7578394	10.4
17: 7574113	1.6

samples using Integrated Genome Viewer (IGV) (Thorvaldsdóttir et al. 2013). As can be noticed from Figures 10, 11 and 12, there is indeed base change for samples S19, S33 and S43.

Interestingly, there were six more SNPs collected by the union of variant callers and present in DoCM (but not BCW), which were incorrectly called as sequencing errors by the classifiers. It is important to note that some of the calls predicted by SNPredict as "false positives" is likely a consequence of BCW's Sanger validation only for AML specific genomic locations. Further validation across more genomic locations and especially for the regions where SNPredict incorrectly predicts SNPs, will provide a better set of ground truth to evaluate SNPredict's performance.

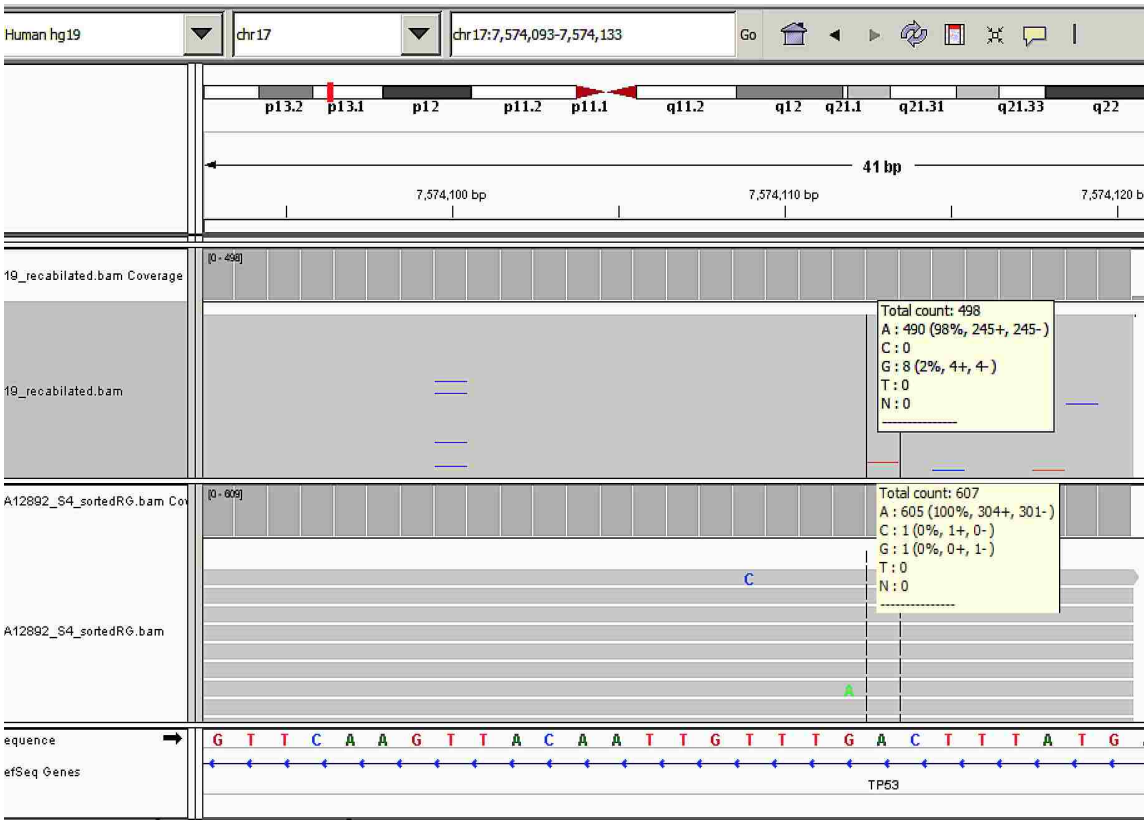


Figure 10: IGV visualization of low frequency SNP at chr17:7574113.
SNPredict's prediction of a low frequency variant (2% VAF) in sample S19 at position 7574113 of chromosome 17. This chromosomal location corresponds to genomic locus of TP53.

Figure 11: IGV visualization of low frequency SNP at chr17: 7578394.
IGV visualization of SNPredict's prediction of a low frequency variant (10% VAF) in sample S43 at position 7578394 of chromosome 17. This chromosomal location also corresponds to genomic locus of TP53.

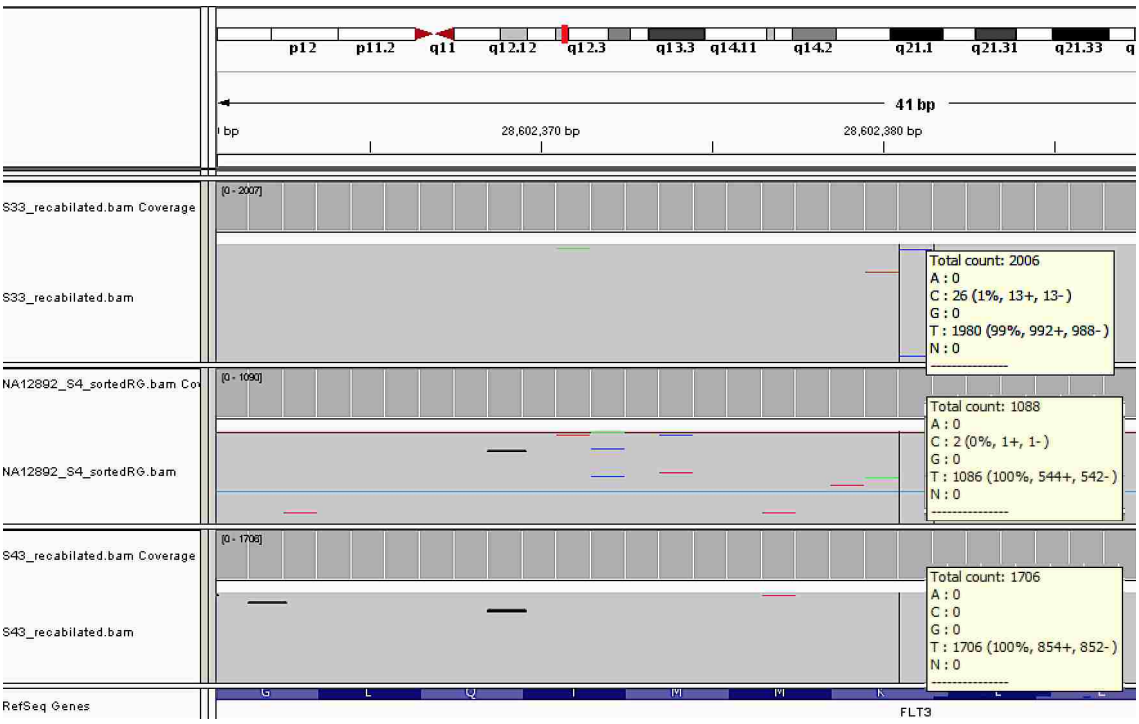


Figure 12: **IGV visualization of low frequency SNP at chr13: 28602381.**
SNPredict's prediction of a low frequency variant (1% VAF) in sample S33 at position 28602381 of chromosome 13. This chromosomal location corresponds to genomic locus of FLT3. For comparison, sample S43 was also visualized at the same location but no such variant was found.

IV. Conclusions

Different variant callers use different algorithms to discriminate between true variant and sequencing errors. As was highlighted in this study, the performance of these callers vary depending on the variant frequency of the mutations present in the tumor and their usage for unmatched normal-tumor samples leads to incomplete results. Therefore, none of these existing callers should be used without the aid of other callers for calling somatic SNPs. The pipeline presented in this study takes advantage of some of these somatic callers by not only combining their results but by also extracting features from their individual outputs to model classifiers that can predict somatic mutations with better accuracy than each of the individual tools.

With spiked-in datasets, two out of three SNPredict's classifiers significantly outperform existing somatic callers. Even with real tumor samples, kNN and Logistic Regression both achieve high F1-scores. It is essential to note that the classifiers built using spiked-in tumor samples do not provide an accurate picture of the real tumor samples and hence it is recommended that SNPredict be used to predict mutations for tumors that are similar to the ones used as training data for building the classifiers. One major finding of this study is the accurate prediction of eight true SNPs - including 3 SNPs with variant frequency as low as 1% - not captured by BCW.

In summary, this pipeline can be a valuable source for clinical centers that currently execute only a single variant caller for the purpose of somatic variant detection. SNPredict's high performance over existing variant callers on unmatched samples highlights its usefulness for clinical institutes like BCW that lack the matched normal-

tumor sample set. Going forward, its strength should be further increased by not only adding more somatic SNP callers to the analysis but also include indel and copy number variations to predict broader kinds of somatic variations present in cancer.

V. Bibliography

1. Allhoff, M. et al., 2013. Discovering motifs that induce sequencing errors. *BMC bioinformatics*, 14(5), p.1. Available at: http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-S5-S1

2. Van der Auwera, G.A. et al., 2013. From FastQ data to high confidence variant calls: the Genome Analysis Toolkit best practices pipeline. *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, 43, pp.11–33. Available at: http://doi.wiley.com/10.1002/0471250953.bi1110s43

3. Chang, F. & Li, M.M., 2013. Clinical application of amplicon-based next-generation sequencing in cancer. *Cancer genetics*, 206(12), pp.413–419. Available at: http://linkinghub.elsevier.com/retrieve/pii/S2210776213001427

4. Cibulskis, K. et al., 2013. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nature biotechnology*, 31(3), pp.213–219. Available at: http://www.nature.com/doifinder/10.1038/nbt.2514

5. Ding, L. et al., 2010. Analysis of Next Generation Genomic Data in Cancer: Accomplishments and Challenges. *Human Molecular Genetics*, 19, p.ddq391. Available at: http://www.hmg.oxfordjournals.org/cgi/doi/10.1093/hmg/ddq391

6. Dohm, J.C. et al., 2008. Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic acids research*, 36(16), p.e105. Available at: http://nar.oxfordjournals.org/lookup/doi/10.1093/nar/gkn425

7. Fang, H. et al., 2015. Indel variant analysis of short-read sequencing data with Scalpel. *bioRxiv*, p.028050. Available at:

http://biorxiv.org/lookup/doi/10.1101/028050

8. Fang, L.T. et al., 2015. An ensemble approach to accurately detect somatic mutations using SomaticSeq. *Genome Biology*, 16(1), p.197. Available at: http://genomebiology.com/2015/16/1/197

9. Gerstung, M., Papaemmanuil, E. & Campbell, P.J., 2014. Subclonal variant calling with multiple samples and prior knowledge. *Bioinformatics (Oxford, England)*, 30(9), pp.1198–1204. Available at: http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btt750

10. Hunter, J. et al., 2014. The Matplotlib Development Team. Matplotlib: Python Plotting—Documentation. 2013. Available at: http://scholar.google.com/scholar?q=related:SOTXjoUZjZcJ:scholar.google.com/&hl=en&num=20&as_sdt=0,5

11. Kelleher, J., Ness, R.W. & Halligan, D.L., 2013. Processing genome scale tabular data with wormtable. *BMC bioinformatics*, 14(1), p.356. Available at: http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-356

12. Kim, S.Y., Jacob, L. & Speed, T.P., 2014. Combining calls from multiple somatic mutation-callers. *BMC bioinformatics*, 15(1), p.154. Available at: http://www.biomedcentral.com/1471-2105/15/154

13. Koboldt, D.C. et al., 2012. VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, 22(3), pp.568–576. Available at: http://genome.cshlp.org/content/22/3/568.full

14. Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Ijcai*. Available at: https://pdfs.semanticscholar.org/0be0/d781305750b37acb35fa187febd8d

b67bfcc.pdf

15. Krogan, N.J. et al., 2015. The cancer cell map initiative: defining the hallmark networks of cancer. *Molecular cell*, 58(4), pp.690–698. Available at: http://linkinghub.elsevier.com/retrieve/pii/S1097276515003445

16. Larson, D.E. et al., 2012. SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics (Oxford, England)*, 28(3), pp.311–317. Available at: http://bioinformatics.oxfordjournals.org/content/28/3/311.full

17. Li, H., 2011. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics (Oxford, England)*, 27(21), pp.2987–2993. Available at: http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btr509

18. Li, H., 2015. BFC: correcting Illumina sequencing errors. *Bioinformatics (Oxford, England)*, 31(17), pp.2885–2887. Available at: http://bioinformatics.oxfordjournals.org/lookup/doi/10.1093/bioinformatics/btv290

19. Li, H. & Durbin, R., 2010. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)*, 26(5), pp.589–595. Available at: http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btp698

20. McKinney, W., 2011. pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*. Available at: http://www.dlr.de/sc/Portaldata/15/Resources/dokumente/pyhpc2011/submissions/pyhpc2011_submission_9.pdf

21. Meyerson, M., Gabriel, S. & Getz, G., 2010. Advances in understanding cancer genomes through second-generation sequencing. *Nature Reviews Genetics*, 11(10), pp.685–696. Available at: http://www.nature.com/doifinder/10.1038/nrg2841

22. O'Rawe, J. et al., 2013. Low concordance of multiple variant-calling pipelines: practical implications for exome and genome sequencing. *Genome* …. Available at: http://www.biomedcentral.com/content/pdf/gm432.pdf

23. Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, pp.2825–2830. Available at: http://www.jmlr.org/papers/v12/pedregosa11a.html

24. Robasky, K., Lewis, N.E. & Church, G.M., 2014. The role of replicates for error mitigation in next-generation sequencing. *Nature Reviews Genetics*, 15(1), pp.56–62. Available at: http://www.nature.com/doifinder/10.1038/nrg3655

25. Roberts, N.D. et al., 2013. A comparative analysis of algorithms for somatic SNV detection in cancer. *Bioinformatics (Oxford, England)*, 29(18), pp.2223–2230. Available at: http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/btt375

26. Saunders, C.T. et al., 2012. Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics (Oxford, England)*, 28(14), pp.1811–1817. Available at: http://bioinformatics.oxfordjournals.org/content/28/14/1811.full

27. Stephens, P.J. et al., 2012. The landscape of cancer genes and mutational processes in breast cancer. *Nature*, 486(7403), pp.400–404. Available at: http://www.nature.com/doifinder/10.1038/nature11017

28. Williams, C. et al., 1999. A High Frequency of Sequence Alterations Is Due to Formalin Fixation of Archival Specimens. *The American Journal of Pathology*, 155(5), pp.1467–1471. Available at: http://linkinghub.elsevier.com/retrieve/pii/S0002944010654612

29. Wilm, A. et al., 2012. LoFreq: a sequence-quality aware, ultra-sensitive variant caller for uncovering cell-population heterogeneity from high-throughput sequencing datasets. *Nucleic acids research*, 40(22), Available at: http://nar.oxfordjournals.org/content/early/2012/10/18/nar.gks918.full

30. Xu, H. et al., 2014. Comparison of somatic mutation calling methods in amplicon and whole exome sequence data. *BMC genomics*, 15(1), p.244. Available at: http://www.biomedcentral.com/1471-2164/15/244