SNVSniffer: An Integrated Caller for Germline and Somatic SNVs Based on Bayesian Models

Yongchao Liu*, Martin Loewer†, Srinivas Aluru*, and Bertil Schmidt‡
*School of Computational Science & Engineering
Georgia Institute of Technology, Atlanta, GA 30332, USA
Emails: {yliu, aluru}@cc.gatech.edu
†Translational Oncology
Johannes Gutenberg University Medical Center gGmbH Mainz, Mainz 55131, Germany
Email: martin.loewer@tron-mainz.de
‡Institute of Computer Science
Johannes Gutenberg University Mainz, Mainz 55128, Germany
Email: bertil.schmidt@uni-mainz.de

***Abstract*—The discovery of single nucleotide variants (SNVs) from next-generation sequencing (NGS) data typically works by aligning reads to a given genome and then creating an alignment map to interpret the presence of SNVs. Various approaches have been developed to call whether germline SNVs (or SNPs) in normal cells or somatic SNVs in cancer/tumor cells. Nonetheless, efficient callers for both germline and somatic SNVs have not yet been extensively investigated. In this paper, we present SNVSniffer, an integrated caller for germline and somatic SNVs from NGS data based on Bayesian probabilistic models. In SNVSniffer, our germline SNV calling models allele counts per site as a multinomial conditional distribution. Meanwhile, our somatic SNV calling relies on NGS tumor-normal sample pairs, and introduces a hybrid approach combining a subtraction approach with a joint sample analysis which models tumor-normal allele counts per site as a joint multinomial conditional distribution. Moreover, we investigate a lightweight tumor purity estimation approach, which demonstrates high accuracy on synthetic tumors. Compared to some leading SNP callers (SAMtools, GATK and FaSD) and somatic SNV callers (VarScan2, SomaticSniper, JointSNVMix2, MuTect), SNVSniffer demonstrates comparable or even better accuracy at faster speed. SVNSniffer, the synthetic tumor-normal data and the supplementary information are available at http://snvsniffer.sourceforge.net.**

I. Introduction

Next generation sequencing (NGS) technologies provide affordable, reliable and high-throughput sequencing of DNA, and make it possible to comprehensively catalog genetic variations in human genomes. Single nucleotide variation is one of the most common genetic variations in human individuals. Variants can be further interpreted as germline SNVs, i.e. single nucleotide polymorphisms (SNPs), in normal cells or somatic SNVs in cancer/tumor cells. Up to date, a variety of computational methods have been developed to call germline or somatic SNVs from NGS read data, the majority of which typically work by three steps: (*i*) align NGS sequence reads from one or more samples to the genome; (*ii*) call variants using probabilistic methods (e.g. Bayesian model); and (*iii*) assess the statistical significance of the called variants and report the results. Step (*i*) can be fulfilled by leading NGS read aligners (e.g. [1], [2], [3] and [4]). Steps (*ii*) and (*iii*) are usually realized by standalone callers.

A few single-sample SNV callers have been developed for NGS, and representative callers include MAQ [5], SOAPsnp [6], SAMtools [7], SNVMix [8], GATK [9], and FaSD [10]. MAQ, SOAPsnp and FaSD model allele counts at each site as a binomial distribution, while SNVMix uses a mixed binomial distribution. All of the four callers identify SNVs by computing Bayesian-based posterior probabilities. Both SAMtools and GATK employ Bayesian likelihood and provide support for the processing of pooled data. It should be noted that these SNV callers actually can be applied to identify any single-nucleotide genetic variation in an individual, including both germline and somatic variants, albeit originally targeting SNPs. Refer to [11] for more details about the state-of-the-art research on genotyping and single-sample SNV calling.

Compared to germline SNV calling, somatic SNV calling is more challenging since its objective is to identify alleles that appear in the tumor, but do not occur in the host's germ line. In other words, we have to additionally distinguish germline polymorphisms from somatic ones at the sites containing variants. One approach [8] is to first call SNVs in the tumor using conventional SNP callers and then screen the predicted SNVs against public SNP databases, e.g. dbSNP [12]. Unfortunately, this approach is challenged by the considerable number of novel SNVs found in individuals, e.g. [13] reported that 10∼50% of SNVs per individual are novel events. In this case, germline mutations uncatalogued in public databases would be falsely identified as somatic mutations.

A more reliable approach to detecting somatic mutations is to call variants in both a tumor sample and its matched normal sample. Approaches used by existing somatic SNV callers can be classified into two categories: simple subtraction and joint sample analysis. The simple subtraction approach separately genotypes the normal and tumor samples at each site and then classifies the site as somatic if the genotype in the normal is homozygous reference and the genotype in the tumor contains alternative alleles to the reference base. This also suggests that callers based on simple subtraction can directly use well-established single-sample SNV callers such as SAMtools and GATK. This simple subtraction approach may provide reasonable prediction for sample pairs with high somatic allele frequency and data purity. However, it has

978-1-4673-6799-8/15/$31.00 ©2015 IEEE

been observed that somatic mutations are prevalent at a low frequency in clinical samples [14]. In this case, any tendency to mistake germline mutations for somatic ones may potentially contaminate the discovery of somatic SNVs. On the other hand, there are variations in somatic allele frequencies from site to site or sample to sample, which are often caused by substantial admixture of normal cells in the tumor sample, copy number variations and tumor heterogeneity. In this regard, a joint analysis of both samples is expected to be capable of further improving performance, by facilitating simultaneous tests for alleles in both samples and enabling more comprehensive representation of tumor impurity and noisy data.

Recently, several somatic SNV callers have been developed based on joint sample analysis, including VarScan2 [15], SomaticSniper [16], JointSNVMix2 (JSM2) [17], Strelka [18], MuTect [19] and FaSD-somatic [20]. VarScan2 pioneers to employ Fisher's exact test to evaluate the statistical significance of allele frequency differences in the two samples. Nonetheless, intrinsically, VarScan2 still employs the simple subtraction approach as the core. For SomaticSniper, JSM2, Strelka, MuTect and FaSD-somatic, all of them adopt Bayesian models to simultaneously analyze the tumor-normal pair, but with different formula or specific procedures.

SomaticSniper estimates the Bayesian posterior probability for each joint genotype, given the alignment map and genotype prior probabilities, and then applies a somatic score cutoff to remove false positives. JSM2 assumes that allele counts in a tumor-normal pair follow a binomial distribution at each site, and models the joint genotypes across all sites as a multinomial distribution. Moreover, this caller also introduces a machine learning approach based on expectation maximization to train the parameters for the mixed binomial model. Strelka apparently models the joint allele frequencies of the two samples to compute somatic variant probabilities, but actually relies on genotype inference as well. This is because the somatic variant probabilities do not distinguish somatic variant types. Hence, Strelka has to associate somatic calls with a joint probabilistic model of somatic variation and the normal genotype inferred by a conventional single-sample Bayesian approach.

MuTect does not call genotypes, but directly manipulates allele frequencies in a tumor-normal pair. It explains the aligned alleles at each site in the tumor by means of two models: a reference model and a variant model. The reference model assumes that no variant exists at the site and any alternative allele to the reference base is caused by sequencing noise. The variant model assumes that the site contains a true variant besides sequencing noise. Variants are identified by computing the log likelihood ratio of the two models. FaSD-somatic adopts a very similar approach to SomaticSniper, but introduces transition and transversion probabilities into the Bayesian model as well as implements a different somatic score formula. A recent comparative analysis of existing somatic SNV callers (except for MuTect and FaSD-somatic) using tumor-normal pairs can be obtained from [21].

In this paper, we present SNVSniffer, an integrated caller for both germline and somatic SNVs by computing Bayesian posterior probabilities to infer genotypes and call SNVs. For germline SNV calling, we model allele counts at each site to follow a multinomial conditional distribution and select the most likely genotypes based on Bayesian posterior probabilities. For somatic SNV calling, we consider the allele counts in the normal to be the result of the mixture of diploid germline variants and noise (e.g. from sequencing cycles or alignment process), and the allele frequencies in the tumor to be caused by the existence of normal cells and somatic variants (besides noise). In this case, by modeling the joint allele counts from the tumor-normal pair as a joint multinomial distribution, we have introduced a hybrid approach by combining a subtraction approach with a joint sample analysis. Moreover, we investigate a lightweight tumor purity estimation approach to predict the proportion of sequencing data from tumor cells, given a tumor-normal pair. In addition, SNVSniffer demonstrates comparable or better accuracy (in terms of F-score) at faster speed, compared to some leading SNP callers (SAMtools, GATK and FaSD) and somatic SNV callers (VarScan2, SomaticSniper, JSM2, MuTect) through our evaluations.

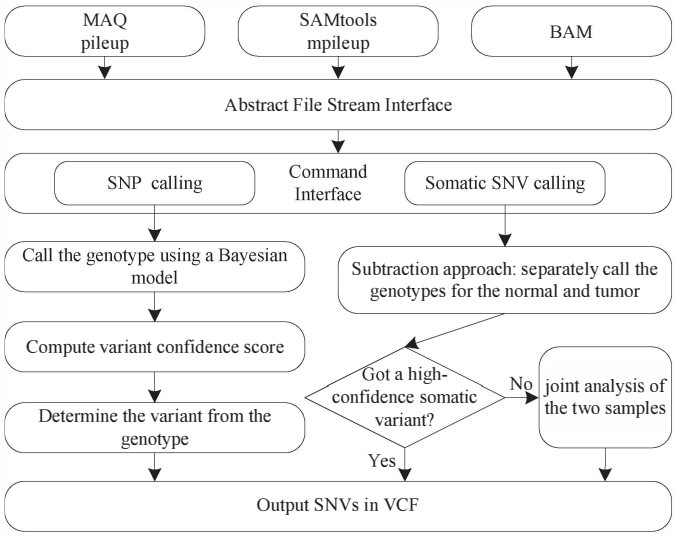

Fig. 1. Program workflow for germline and somatic SNV calling

II. METHODS

SNVSniffer calls germline and somatic SNVs based on genotype inference from Bayesian posterior probabilities. Our caller accepts three input file formats: the `pileup` format from MAQ, the `mpileup` format from SAMtools, and the `BAM` format [7], all of which are centrally managed by an abstract file stream interface (AFSI) (see Figure 1). The `pileup` and `mpileup` formats are natively supported and our AFSI directly loads data from the input. However, when it comes to `BAM`, SNVSniffer spawns a separate child process for each `BAM` file. Each child process runs an instance of SAMtools to produce `mpileup` data stream from the corresponding `BAM` file. In this scenario, each child process writes the data stream to its own FIFO (first in first out) special file, and then uses the FIFO file to communicate with the AFSI.

A. Genotype Inference

1) Allele counts modeling: We model allele counts at each site of the genome as a multinomial distribution conditioned on genotypes. At site *i*, we define X_i to denote the allele count vector, $X_{i,j}$ to denote the aligned allele over Σ={A, C, G, T} from read *j*, γ to denote the reference base, and $G_k = G_k^1 G_k^2 \in$

{AA, CC, GG, TT, AC, AG, AT, CG, CT, GT} to denote a genotype for diploid genomes ($1 \leq k \leq K$ and *K* is the total number of genotypes, i.e. 10 in our case).

In SNVSniffer, the probability $P(X_i|G_k)$ of observing allele count vector X_i at site *i* is defined as

$$P(X_i|G_k) \propto \prod_{j=1}^{N} \prod_{c \in \Sigma} P(X_{i,j}|G_k)^{I(X_{i,j}=c)} \quad (1)$$

where *N* is the number of alleles covering the site, N_x is the number of allele $x \in \Sigma$, and $I(X_j = x)$ is an indicator function whose value is 1 if $X_{i,j}$ equals *x* and 0 otherwise. For each aligned allele $X_{i,j}$, the probability $P(X_{i,j}|G_k)$ of observing this allele at the site is defined as

$$P(X_{i,j}|G_k) = \alpha P(X_{i,j}|G_k^1) + (1-\alpha)P(X_{i,j}|G_k^2) \quad (2)$$

by taking into account the factors such as sequencing bias between distinct haploid chromosomes, base-calling errors and alignment quality. α denotes the proportion of reads sequenced from the G_k^1 haploid chromosome, and is set to 0.5 in our implementation based on the assumption that the two haploid chromosomes are impartially sequenced. $P(X_{i,j}|G_k^b)$ means the probability of observing $X_{i,j}$ at the G_k^b haploid chromosome (*b* = 1 or 2), and is defined as

$$P(X_{i,j}|G_k^b) = \begin{cases} \omega_{i,j} & \text{if } X_{i,j} = G_k^b \\ (1-\omega_{i,j})\Phi(X_{i,j}, G_k^b) & \text{otherwise} \end{cases} \quad (3)$$

where $\Phi(\cdot)$ is a 2-dimensional probability table, with $\Phi(X_{i,j}, G_k^b)$ representing the probability of G_k^b being the true chromosomal base given that $X_{i,j}$ is miscalled or misaligned, and $\omega_{i,j}$ is the accuracy (or weight) of $X_{i,j}$. This equation is inspired by [9], but has two major differences. On one hand, [9] classifies alleles covering each site as whether reference base or non-reference variant, and thereby partitions the full set of genotypes over three genotype categories: homozygous reference, heterozygous reference and homozygous variant. On the contrary, SNVSniffer does not perform such allele classifications. Instead, we consider the total of *K* possible genotypes, which can also be further classified into four categories: homozygous reference, heterozygous reference, homozygous variant and heterozygous variant. On the other hand, mapping quality scores are additionally introduced to our computation. In SNVSniffer, $\omega_{i,j}$ is calculated as

$$\omega_{i,j} = 1 - \frac{2 \times 10^{-0.1(B_q + M_q)}}{10^{-0.1B_q} + 10^{-0.1M_q}} \quad (4)$$

where B_q is the base quality score and M_q is the mapping quality score.

As for $\Phi(X_{i,j}, G_k^b)$, a naïve approach is to use non-informative prior, i.e. set $\Phi(X_{i,j}, G_k^b)$ to $1/(|\Sigma|-1)$ for each allele $X_{i,j}$ that is not equal to G_k^b. Alternatively, we can also inspect the error profiles of different sequencing technologies and then derive $\Phi(\cdot)$ for use. In SNVSniffer, we have used the probabilities for Illumina sequencing given in [9].

2) Genotype selection: Having gained $P(X_i|G_k)$ for each genotype G_k, we compute the posterior probability $P(G_k|X_i)$ of the true genotype being G_k given X_i, based on the Bayes' theorem, where $P(G_k|X_i)$ is computed as

$$P(G_k|X_i) = \frac{P(X_i|G_k)P(G_k)}{\sum_{l=1}^{K} P(X_i|G_l)P(G_l)} \propto P(X_i|G_k)P(G_k) \quad (5)$$

Subsequently, we single out the genotype with the largest posterior probability as the "true" genotype at the site. In general, we need to show that the larger probability of the selected genotype is statistically significant compared to the others. [22] proposed the use of Dixon's *Q* test [23] which originally targets the detection of outliers. The *Q* test examines the ratio of the absolute difference between the largest and the second largest numbers, to the range of evaluated numbers, and then compute a *P*-value at a specific confidence level to guide whether to reject or accept the hypothesis. In our algorithm, we have attempted to use the *Q* test to evaluate the statistical significance of the most likely genotype. However, through our evaluations we did not find any significant differences in the performance of genotype calling, when compared to the case without using the *Q* test. Considering the computational overhead of this *Q* test, we have disabled this test, and instead directly select the genotype with the largest probability.

3) Genotype prior probabilities: In Equation (5), we require a prior probability for each genotype G_k. In our algorithm, we have considered three implementations of prior probabilities: non-informative priors, priors derived from heterozygous mutation rate θ [16], and priors derived from both θ and transition/transversion (T_i/T_v) ratio [20]. Specifically, θ means the expected rate of heterozygous point mutations in the population of interest and its estimated value is close to 10^{-3} between two distinct human haploid chromosomes [6]. T_i/T_v ratio is around 2.0∼2.1 for the whole human genome shown by the recent human genome studies, particularly the 1000 genomes project [13]. For non-informative priors, each genotype is assumed to have the same prior probability $\frac{1}{K}$. For the θ-only priors, they are defined as

$$P(G_k) = \begin{cases} \theta & \text{if } G_k \text{ is heterozygous ref.} \\ \theta^2 & \text{if } G_k \text{ is heteozygous var.} \\ \theta/2 & \text{if } G_k \text{ is homozygous var.} \\ 1 - \sum_{l=1}^{K} P(G_l)I(G_l) & \text{if } G_k \text{ is homozygous ref.} \end{cases} \quad (6)$$

where $I(G_l)$ returns 0 if G_l is homozygous reference, and 1, otherwise.

For the T_i/T_v-based priors, they are similarly defined to Equation (6), but additionally check whether the genotype G_k has a transition or transversion mutation relative to the reference base at each site. Intuitively, more accurate results can be yielded if the priors used are consistent with the ground truth, and otherwise, misleading results may be caused by the use of unrealistic priors. Through our evaluations, it is interesting to find that none of the three priors is able to consistently show superior performance. In this regard, we have chosen the θ-only priors as the default setting, since it has been more often observed to have better performance in our limited number of tests. In addition, we have pre-computed the priors for every combination of reference bases with genotypes in order to improve speed.

4) Variant confidence score: To trade off sensitivity and specificity, we have introduced a variant confidence score (VCOS) to measure our confidence of the correctness of the variants called. VCOS is only computed for the inferred genotypes that are not homozygous reference, i.e. genotypes must have ≥ 1 alternative allele to γ.

$$VCOS = \frac{-\sum_{j=1}^{N} I(X_{i,j} \in G_k) \cdot \log(P(X_{i,j}|\gamma))}{\sum_{j=1}^{N} I(X_{i,j} \in G_k)} \quad (7)$$

where $I(X_{i,j} \in G_k)$ equals 1 if $X_{i,j}$ is an allele in the inferred genotype G_k, and 0, otherwise. $P(X_{i,j}|\gamma)$ means the probability of substituting the reference base γ for the aligned allele $X_{i,j}$, and is computed as

$$P(X_{i,j}|\gamma) = \begin{cases} S_{ti} & \text{if } X_{i,j} \text{ is a transition of } \gamma \\ S_{tv} & \text{if } X_{i,j} \text{ is a transversion of } \gamma \\ 1-\theta & \text{if } X_{i,j} = \gamma \end{cases} \quad (8)$$

by incorporating heterozygous mutation rate θ and the T_i/T_v ratio δ in the population of interest. In Equation (8), $S_{ti} = \delta\theta/(1+\delta)$ and $S_{tv} = 0.5\theta/(1+\delta)$. We have set δ to 2.0 for human samples.

Based on VCOS, our caller classifies the variants called into three categories: high-confidence, low-confidence and false positives, depending on how many alternative alleles to γ are there in the corresponding genotypes. A variant is deemed as high-confidence if its VCOS is $\geq HC(G_k)$, as low-confidence if its VCOS is $< HC(G_k)$ but $\geq LC(G_k)$, and as false positives, otherwise. The score threshold $HC(G_k)$ is computed as

$$HC(G_k) = \begin{cases} -\frac{1}{2}\log((1-\theta) \times S_{ti}) & \text{Case 1} \\ -\log(S_{ti}) & \text{Case 2} \end{cases} \quad (9)$$

and $LC(G_k)$ computed as

$$LC(G_k) = \begin{cases} -\phi\log(1-\theta) - \frac{1}{2}(1-\phi)\log(S_{ti} \times S_{tv}) & \text{Case 1} \\ -\frac{1}{2}\psi\log(S_{ti} \times S_{tv}) - (1-\psi)\log(1-\theta) & \text{Case 2} \end{cases} \quad (10)$$

where `Case 1` means that G_k is heterozygous reference and `Case 2` means that G_k does not contain γ. Note that we have constrained the values of ϕ and ψ to ensure that $HC(G_k)$ is always $\geq LC(G_k)$.

B. Overview of Somatic SNV Calling

We call somatic variants from paired tumor-normal samples sequenced from tumor and normal tissues of the same individual, respectively. In this scenario, normal sample can act as a control in order to better distinguish SNVs that are unique to the tumor (somatic variants) from those present in the matched normal (germline polymorphisms). In our algorithm, we have adopted a hybrid approach benefiting from the combination of an independent subtraction analysis and a joint sample analysis, with genotype inference as the core. Based on the genotypes inferred for both samples, SNVs detected in the tumor are classified into four mutation types: `Somatic`, `LOH` (loss of heterozygosity), `Germline`, and `Unknown`. Table I shows the type classification, where G_k^N and G_k^T $(1 \leq k \leq K)$ denote the genotypes from the normal and tumor, *A* and *B* denote the reference base γ and the non-reference variant ($\neq \gamma$) in the diploid genotypes, respectively. Moreover, the SNVs are reported in the well-established `VCF` format [24].

TABLE I. TYPE CLASSIFICATION OF SOMATIC SNVS

$G_k^N \backslash G_k^T$	AA	AB	BB
AA	Wild	Somatic	Somatic
AB	LOH	Germline	LOH
BB	Unknown	Unknown	Germline

C. Subtraction Analysis

Our subtraction analysis first calls mutations from the normal and tumor samples separately using the aforementioned Bayesian probabilistic models and then contrasts the results like a simple subtraction. This approach can provide reasonable predictions if there exists little noise and variant alleles have large enough frequencies (e.g. exceeding the expected) to be detected. In SNVSniffer, given a site, the subtraction analysis works as follows : (*i*) call the genotype G_k^N from the normal. If G_k^N is not homozygous reference, the VCOS is computed for the genotype. G_K^T is processed in the same way; and (*ii*) if G_k^N is homozygous reference and G_k^T has high-confidence variants, a `Somatic-type` SNV is reported for this site, and otherwise, leave it to the subsequent joint sample analysis.

D. Joint Sample Analysis

As mentioned above, somatic allele frequencies in the tumor may have considerable variability, often caused by the presence of normal cells in the tumor sample, copy number variations and tumor heterogeneity. In such cases, the simple subtraction analysis may become less effective since variant alleles might have considerably low frequencies compared to the expected frequencies. In this regard, a joint model to simultaneously analyze both samples will likely lead to an increased ability to detect shared signals, which arise from germline polymorphisms or sequencing cycles, as well as weakly observed real somatic variants. In SNVSniffer, this joint analysis is applied after the preceding subtraction analysis.

1) Joint genotype selection: At each site *i* of the genome, we model allele counts of the tumor-normal pair as a joint multinomial conditional distribution, given a joint genotype G_k^N and G_t^T, where $1 \leq k < K$ and $1 \leq t < K$. Assume that X_i represents the allele count vector observed in the normal and Y_i the allele count vector observed in the tumor. The posterior probability $P(G_k^N, G_t^T|X_i, Y_i)$ of the joint genotype G_k^N and G_t^T, given X_i and Y_i, is computed as

$$P(G_k^N, G_t^T|X_i, Y_i) \propto \begin{array}{l} P(X_i|Y_i, G_k^N, G_t^T) \\ P(Y_i|G_k^N, G_t^T)P(G_k^N, G_t^T) \end{array} \quad (11)$$

As mentioned above, the normal sample is assumed not to contain any read sequenced from tumor cells. This indicates that in our algorithm, X_i is independent of both Y_i and G_t^T. Hence, $P(X_i|Y_i, G_k^N, G_t^T)$ can be re-written as $P(X_i|G_k^N)$ and Equation (11) can therefore be simplified as

$$P(G_k^N, G_t^T|X_i, Y_i) \propto \begin{array}{l} P(X_i|G_k^N) \\ P(Y_i|G_k^N, G_t^T)P(G_k^N, G_t^T) \end{array} \quad (12)$$

where $P(X_i|G_k^N)$ is computed using Equation (1). As for $P(Y_i|G_k^N, G_t^T)$, it is equal to $P(Y_i|G_t^T)$ if the tumor sample contains no normal cell (meaning that Y_i is independent of G_k^N), and thereby can be computed using Equation (1). However, in practice, the tumor sample has a probability of incorporating normal cells and it would be more realistic to taken into account the tumor purity, which represents the expected percentage of alleles from tumor cells at each site, in our computation. For simplicity, our algorithm assumes $P(Y_i|G_k^N, G_t^T)$ equals $P(Y_i|G_t^T)$, but employs the tumor purity as an indicator to trade off sensitivity and specificity, especially at the sites with low variant fractions.

2) Joint genotype prior probabilities: In Equation (11), $P(G_k^N, G_t^T)$ can be re-written as $P(G_k^N, G_t^T) = P(G_t^T|G_k^N)P(G_k^N)$. To compute $P(G_k^N, G_t^T)$, one approach is treating the genotypes of the two samples as completely independent events, where $P(G_k^N, G_t^T)$ can be computed as $P(G_k^N)P(G_t^T)$. Another approach is assuming that the genotypes of both samples are dependent. The latter is more realistic since the two samples are sequenced from the same individual and tend to share germline polymorphisms. In this regard, we have used the conditional probability $P(G_t^T|G_k^N)$ proposed in [16] by assuming that the genotypes of the two samples are dependent.

3) Post-processing procedure: Having calculated the most likely genotypes G_k^N and G_t^T using Equation (11), our post-processing procedure is based on four steps: (*i*) if the new G_k^N is not identical to the one called by the previous subtraction analysis, compute its VCOS if it is not homozygous reference. If the new G_k^N has high-confidence variants, it is retained to replace the old one computed by the subtraction approach, and otherwise, G_k^N is deemed as homozygous reference because of the lack of confidence. G_t^T is processed in the same way; (*ii*) if the new G_k^N is identical to the one called by the previous subtraction analysis, compute its VCOS with a relaxed constraint if it is not homozygous reference. This relaxed computation considers a genotype as non-false-positive if the number of variants in the genotype exceeds a minimum threshold conditioned on the read depth at the site. The new G_t^T is also processed likewisely; (*iii*) if both G_k^N and G_t^T are classified as false positives, the called variant will be discarded and otherwise, retained; and (*iv*) classify the variant and report the result in `VCF`.

E. Tumor Purity Estimation

Assuming that the tumor sample is a mixture of normal and tumor cells, we have investigated a lightweight approach to estimating the tumor purity, given a tumor sample. This estimation approach is designed based on the rational that at a site with a somatic SNV, the fraction of the reads containing the somatic variant, among all of the reads covering the site, in principle provides an estimate of the proportion of tumor cells from which the somatic variant originates. In SNVSniffer, we have only considered the `Somatic`-type mutation positions at which the normal genotype is homozygous reference and the tumor genotype is heterozygous reference. Given the inferred tumor genotype $G = G^1G^2$ at a site, the purity β is estimated as

$$\beta = \min\{1, \frac{2f(G^b)}{f(\gamma) + f(G^b)}\} \quad (13)$$

where G^b (b = 1 or 2) is the very allele of G, which is not equal to γ, and the function $f(b)$ returns the number of occurrences of base $b \in \Sigma$ at the site of the tumor sample. This equation assumes that there is no sequencing errors in the reads and the two strands of the diploid genome are impartially sequenced. After gaining a list of purity values, we sort all of the values in ascending order and then take the mean square root of the mean and the median values as the estimated purity.

F. Diploid Tumor-Normal Sample Simulation

Since real data lacks the ground truth of somatic variants, we have implemented a somatic simulator (a subprogram of our caller) that is able to synthesize a tumor diploid genome and its matched normal diploid genome at a specific somatic mutation rate, and then generate tumor-normal pairs by sequencing reads from the simulated tumor-normal genome pair. In our simulator, we have adopted a similar method to the one proposed in [5] to simulate the normal genome from a reference genome, and used a method inspired by the virtual-tumor idea [19] to simulate a tumor genome from the simulated normal genome as well as generated reads from the simulated tumor genome given a specific tumor purity.

Our simulator generally works by four steps: (i) simulate a diploid genome for the normal sample from a reference genome at a specific germline mutation rate; (ii) exert a somatic mutation rate on the simulated normal genome to generate a tumor genome with somatic point mutations, where the normal genome must have a homozygous reference genotype at each somatic site; (iii) sequence reads from the simulated normal genome to produce a normal sample comprising paired-end Illumina-like reads; (iv) sequence reads from the simulated tumor genomes to produce a tumor sample containing paired-end Illumina-like reads as well. It needs to be stressed that since a tumor sample has a probability of containing reads from normal cells, a tumor read covering a certain somatic site should have a probability of being replaced by a read covering the same site from the normal genome, in order to mimic the impurity of the tumor sample. In our simulator, at each somatic site the number of normal reads covering the site follows a binomial distribution $B(N, p)$, where $0 \leq p \leq 1$. This leads to an expected tumor purity $\beta = 1 - p$, because at each site the expected number of alleles from tumor cells is $N(1-p)$ and therefore $\beta = N(1-p)/N = 1 - p$.

III. Performance Evaluation

We have evaluated SNVSniffer in terms of both germline and somatic SNV calling, and have further compared it to some leading tools accordingly. For synthetic data, since the ground true is known beforehand, we have used the metrics: recall, precision and *F*-score to measure SNV calling accuracy. Recall is computed by dividing the number of correct predictions by the number of true SNVs, and precision by dividing the number of correct predictions by the number of predictions. *F*-score is a weighted average of recall and precision and defined as $2 \times \text{recall} \times \text{precision}/(\text{recall} + \text{precision})$. In addition, in the `VCF`-format output, a called variant is deemed as correct if the site position matches the ground truth, regardless of the alternative allele column (i.e. column `ALT`).

Unless otherwise specified, all tests are conducted on a workstation with two Intel Xeon X5650 2.67 GHz hex-core

TABLE II. PERFORMANCE AND RUNTIMES FOR SNP CALLING USING SMASH

Dataset	Caller	Recall	Precision	*F*-score	Time(s)
Venter	SNVSniffer	**98.0**	98.4	98.2	**61**
	SAMtools	97.8	98.5	98.1	2,060
	GATK	**98.0**	**99.1**	**98.5**	3,630
	FaSD	**98.0**	97.4	97.7	2,026
Contaminated-Venter	SNVSniffer	96.6	**96.9**	96.7	**63**
	SAMtools	97.6	85.1	90.9	2,073
	GATK	**97.8**	96.7	**97.2**	3,941
	FaSD	96.4	96.5	96.4	2,043

CPUs and 96 GB RAM, running Linux operating system (Ubuntu 14.04). Runtime is measured in wall clock time (unless otherwise specified) and each caller runs a single thread. For VarScan2, FaSD and SNVSniffer the input files are in `mpileup` format, while for the other callers the input files are in `BAM` format. Detailed parameters have been given in the supplementary information. In addition, the values of recall, precision and *F*-score have been multiplied by 100, and all of the best values in tables have been highlighted in bold.

A. SNP Calling Performance Evaluation

We have used the SMASH benchmarking toolkit [25] to compare SNVSniffer (v1.0.25) to the leading SNP callers: SAMtools (v0.1.19), GATK (v3.2-2) and FaSD (version number is not given). In this test, we have used two synthetic datasets of SMASH: `Venter` and `Contaminated-Venter`, and used only the reads aligned to the human chromosome 20 (UCSC hg19) for the sake of speed. In this test, we have directly used the alignment files in SMASH, rather than re-align the raw reads.

Table II shows the performance and runtimes for SNP calling (only SNP statistical results, excluding insertions or deletions). For the Venter dataset, SNVSniffer, GATK and FaSD all achieve the best recall. GATK performs best in terms of both precision and *F*-score, while FaSD performs worst for the two metrics. For the Contaminated-Venter dataset, SNVSniffer yields the best precision, while GATK has the best recall and *F*-score. In particular, SNVSniffer and GATK are both superior to FaSD for each metric. As for speed, SNVSniffer outruns any other caller for each dataset. For the Venter dataset, SNVSniffer achieves a speedup of 33.7, 59.4 and 33.2 over SAMtools, GATK and FaSD, respectively. For the Contaminated-Venter dataset, SNVSniffer runs $33.2\times$, $63.1\times$ and $32.7\times$ faster than SAMtools, GATK and FaSD, respectively.

B. Somatic SNV Calling Performance Evaluation

We have used synthetic tumor-normal pairs, either from simulated or real sequencing data, to compare SNVSniffer to selected leading somatic SNV callers: VarScan2 (v2.3.7), SomaticSniffer (v1.0.4), JSM2 (v0.8-b2), MuTect (v1.1.4). In this evaluation, NGS reads are aligned by BWA (v0.7.5a), since neither Bowtie2 [2] nor CUSHAW2 [3] on average gave as good performance as BWA. Moreover, JSM2 is observed not to explicitly report somatic sites as other callers do. Instead, it outputs the probabilities of the joint genotypes. In this regard, Roth *et al.* [17] suggest to compute the probability $P(\text{Somatic})$ of a site being a somatic position as $P(AA, AB) + P(AA, BB)$. In our evaluation, a site will be retained if its $P(\text{Somatic}) \geq 0.9$.

TABLE III. CALLING PERFORMANCE AND RUNTIMES FOR SOMATIC SNVS CALLING

Metric	Error	SNVSniffer	VarScan2	SomaticSniper	JSM2	MuTect
Recall	2.0%	**93.57**	87.74	85.59	89.76	93.42
	1.5%	**94.67**	88.08	87.67	90.28	93.64
	1.0%	**95.59**	88.34	89.60	90.67	93.85
Precision	2.0%	96.82	96.20	99.95	**100.00**	85.82
	1.5%	96.82	96.17	99.96	**100.00**	88.12
	1.0%	96.36	96.00	99.97	**100.00**	87.61
F-score	2.0%	**95.17**	91.78	92.21	94.60	89.46
	1.5%	**95.73**	91.95	93.41	94.89	90.80
	1.0%	**95.97**	92.01	94.50	95.11	90.62
Time(s)	2.0%	**141**	2 586	179	1 311	3 831
	1.5%	**139**	2 542	177	1 248	3 508
	1.0%	**138**	2 494	174	1 156	3 440

TABLE IV. RECALL AND RUNTIMES (IN HOUR) FOR REAL TUMORS

	SNVSniffer	VarScan2	SomaticSniper	JSM2	MuTect
T1	66.29/**1.31**	38.29/5.75	61.14/1.58	0.00/10.41	**91.43**/22.87
T2	78.43/**1.16**	35.29/4.73	63.73/1.33	0.00/9.48	**93.14**/19.37
T3	69.11/**1.12**	59.35/4.30	75.61/1.34	0.00/7.39	**95.93**/19.27
T4	90.32/1.59	83.87/5.26	87.10/**1.39**	0.00/10.00	**96.77**/24.16
T5	68.75/**1.52**	43.75/4.75	68.75/1.55	0.00/8.11	**97.92**/20.50

For each value *x*/*y*, *x* denotes the recall and *y* the runtime in hour.

1) On synthetic tumors from simulated data: We have simulated three tumor-normal pairs (1.0%, 1.5% and 2.0% error rates) from the human chromosome 21 (UCSC hg38). Each sample has a $30\times$ coverage of its corresponding chromosome, and contains 100-bp Illumina-like paired-end reads with 500 ± 50 insert size. For each tumor, we set the purity to 0.9 and the ratio of heterozygous to homozygous alternative alleles to 5:1 at somatic sites.

Table III shows the calling performance and runtimes, where all of the `Somatic`-, `LOH`- and `Unknown`-type variants are counted in and the runtimes of SNVSniffer include the times spent on tumor purity estimation. For each sample pair, SNVSniffer performs best in terms of both recall and *F*-score, and JSM2 best in terms of precision. As the error rate decreases, the recall increases for each caller, but the precision does not show a consistent trend. As for *F*-score, except MuTect, all other callers achieve improved performance. In terms of speed, SNVSniffer runs fastest and MuTect slowest for each sample pair on the same hardware. On average, SNVSniffer runs $18.2\times$, $1.3\times$, $8.9\times$, and $25.8\times$ faster than VarScan2, SomaticSniper, JSM2, and MuTect, respectively.

2) On virtual tumors from real data: Secondly, we have assessed the performance of each somatic SNV caller using ten "virtual tumors" (refer to [19]), which are synthesized from the real sequencing data of two human individuals by following the procedure described in [19]. A total of 4,436 somatic mutations have been implanted, with the normal genotype being homozygous reference and the corresponding tumor genotype being heterozygous reference at each somatic site. Their tumor purities range from 0.1 to 1.0 uniformly.

Figure 2 shows the performance as a function of tumor

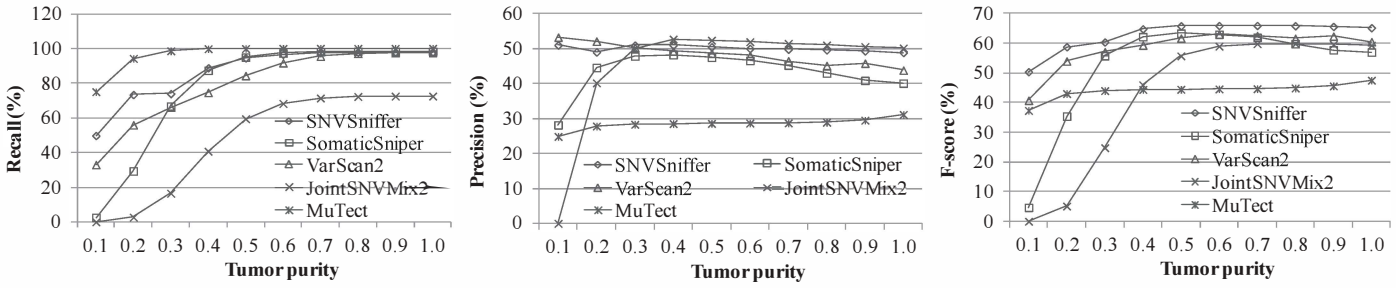

Fig. 2. Performance changes on virtual tumors from real sequencing data as tumor purity varies

purity for each caller. For each case, SNVSniffer achieves the best *F*-score and MuTect the best recall. It is worth mentioning that MuTect successfully identified all of the somatic sites for ≥ 0.7 purities, whereas none of the other callers is able to make it. In terms of recall, SNVSniffer is always superior to both JSM2 and VarScan2. SNVSniffer achieves larger recall than SomaticSniper for ≤ 0.4 purities, but is outperformed by the latter for the rest. In terms of precision, JSM2 is best for ≥ 0.4 purities, SNVSniffer best for purity 0.3, and VarScan2 best for the rest. Meanwhile, SomaticSniper is always inferior to SNVSniffer. In terms of *F*-score, JSM2 is worst for each case. VarScan2 outperforms SomaticSniper for eight purities, while the latter performs better for the remaining two ones.

3) On real tumors: Thirdly, we have used five whole genome sequencing tumor-normal sample pairs for the Ovarian serous cystadenocarcinoma disease. All of them are acquired from the TCGA project with the accession identifiers: TCGA-13-0885-01A-02W-0421-09 (T1), TCGA-13-1481-01A-01W-0549-09 (T2), TCGA-13-1488-01A-01W-0549-09 (T3), TCGA-24-1417-01A-01W-0549-09 (T4), and TCGA-24-1424-01A-01W-0549-09 (T5), respectively. For this test, all callers are executed on a supercomputer with each node having four AMD Opteron 6272 2.1 GHz 16-core CPUs, and the results shown here are in part based on the data generated by the TCGA Research Network (http://cancergenome.nih.gov). Table IV shows the recall and the runtimes (measured in CPU time in order to remove the effect of other jobs scheduled to the same nodes). For each sample, MuTect demonstrates the best recall ($> 90\%$ each) and SNVSniffer performs second best ($> 66\%$ recall each). Interestingly, JSM2 does not manage to identify any true variant for each case. Moreover, SomaticSniper ($> 61\%$ recall each) outperforms VarScan2 ($> 35\%$ recall each). The variant concordance between callers has been shown using Venn diagrams in the supplementary information. As for speed, SNVSniffer is superior to all other callers for each case with an exception that SomaticSniper is fastest on T4.

C. Accuracy of Tumor Purity Estimation

Finally, we have evaluated the performance of our lightweight tumor purity estimation approach (see Equation (13)) using the aforementioned ten virtual tumors. For these tumors, the expected purities are 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0, while the corresponding estimated purities are 0.24, 0.29, 0.34, 0.42, 0.51, 0.59, 0.69, 0.78, 0.87 and 0.95, respectively. Since each value pair in the two groups are not exactly identical, we have used the paired *t*-test, at a *P*-value threshold 0.05, to test whether the difference between the two groups is statistically significant, and got a two-tailed *P*-value 0.35 that is > 0.05, indicating that the difference is not statistically significant and thereby underscoring the accuracy of our estimation.

IV. Conclusion

In this paper, we have presented SNVSniffer, an integrated germline and somatic SNV calling algorithm using Bayesian probabilistic models. Although the Bayesian model has been commonly used in SNV calling, an integrated method for both germline and somatic SNV calling has not yet been extensively investigated in literature. This is one prominent feature that distinguishes SNVSniffer from other callers. In SNVSniffer, we call SNVs by means of genotype inference and select the most likely genotypes based on Bayesian posterior probabilities. Our germline SNV calling models allele counts per site to follow a multinomial distribution, while our somatic SNV calling models the paired tumor-normal allele counts as a joint multinomial distribution. Furthermore, for somatic SNV calling, we have introduced a hybrid approach by combining a subtraction analysis with a joint sample analysis, and also proposed a lightweight estimation approach to predict the proportion of sequencing data from tumor cells. This approach demonstrates high accuracy on synthetic tumors through our evaluations.

As for variant calling, SNVSniffer achieves comparable or better accuracy (in terms of F-score) and faster speed, compared to some leading SNP callers (i.e. SAMtools, GATK, and FaSD) and somatic SNV callers (i.e. VarScan2, SomaticSniper, JSM2 and MuTect). In particular, with respect to somatic SNV calling, SNVSniffer is consistently the top caller. Firstly, evaluations using synthetic tumors from simulated data showed that SNVSniffer performs best in terms of both recall and *F*-score along with relatively high precision for each tumor-normal sample pair. MuTect yields the worst precision and *F*-score for each case. Secondly, evaluations using virtual tumors from real data demonstrated that for each case MuTect achieves the best recall, but along with the worst precision. In contrast, SNVSniffer always produces the best *F*-score, while still holding relatively high precision. Thirdly, on real tumors, MuTect and SNVSniffer are consistently exposed as the best and the second best, respectively. Finally, SNVSniffer runs fastest (though very close to SomaticSniper) mainly due to its less complex computations. The use of `mpileup` format may be another contributor to the fast speed. The reason why this format is used is that SNV calling is expected to become

more efficient by seamlessly integrating read alignment with SNV calling, thereby bypassing intermediate representations such as `SAM/BAM/mpileup`, in future work.

Nonetheless, for our somatic SNV calling, there are still some limitations and challenges. First, the normal sample is assumed to be an admixture of germline mutations and noise. This assumption does not always hold since contamination may occur in normal cells. Second, our tumor purity estimation approach might not work well if a tumor sample has many homozygous-variant somatic mutations, which can be explained as follows. Given such a low-purity tumor sample, the tumor genotype at a homozygous-variant somatic site is likely called as heterozygous reference, due to the abundance of reference bases coming from normal cells at the site. In this case, simply considering that the variants originate merely from one strand of the chromosome will double the true purity, thus overestimating its value. Third, our caller does not support the discovery of insertions or deletions. Fourth, our caller does not take into account some more complex genomic variations in cancer such as copy number variations and sub-clonal populations. In this regard, we expect that explicit modeling of these genomic complexities might lead to enhanced calling accuracy as well as interpretability for somatic variant prediction. How to address such limitations and challenges is part of our future work. As the sequencing of matched tumor-normal samples is becoming a popular routine in cancer research, we still demand more accurate yet efficient calling algorithms for somatic SNVs at practical levels of tumor purity.

ACKNOWLEDGMENT

We acknowledge funding by SRFN Johannes Gutenberg University Mainz and the Carl-Zeiss-Foundation.

Conflict of interest: none declared.

REFERENCES

[1] H. Li and R. Durbin, "Fast and accurate long-read alignment with burrows–wheeler transform," *Bioinformatics*, vol. 26, no. 5, pp. 589–595, 2010.

[2] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with bowtie 2," *Nature methods*, vol. 9, no. 4, pp. 357–359, 2012.

[3] Y. Liu and B. Schmidt, "Long read alignment based on maximal exact match seeds," *Bioinformatics*, vol. 28, no. 18, pp. i318–i324, 2012.

[4] Y. Liu, B. Popp, and B. Schmidt, "Cushaw3: sensitive and accurate base-space and color-space short-read alignment with hybrid seeding," 2014.

[5] H. Li, J. Ruan, and R. Durbin, "Mapping short dna sequencing reads and calling variants using mapping quality scores," *Genome research*, vol. 18, no. 11, pp. 1851–1858, 2008.

[6] R. Li, Y. Li, X. Fang, H. Yang, J. Wang, K. Kristiansen, and J. Wang, "Snp detection for massively parallel whole-genome resequencing," *Genome research*, vol. 19, no. 6, pp. 1124–1132, 2009.

[7] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin *et al.*, "The sequence alignment/map format and samtools," *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.

[8] S. P. Shah, R. D. Morin, J. Khattra, L. Prentice, T. Pugh, A. Burleigh, A. Delaney, K. Gelmon, R. Guliany, J. Senz *et al.*, "Mutational evolution in a lobular breast tumour profiled at single nucleotide resolution," *Nature*, vol. 461, no. 7265, pp. 809–813, 2009.

[9] M. A. DePristo, E. Banks, R. Poplin, K. V. Garimella, J. R. Maguire, C. Hartl, A. A. Philippakis, G. Del Angel, M. A. Rivas, M. Hanna *et al.*, "A framework for variation discovery and genotyping using next-generation dna sequencing data," *Nature genetics*, vol. 43, no. 5, pp. 491–498, 2011.

[10] F. Xu, W. Wang, P. Wang, M. J. Li, P. C. Sham, and J. Wang, "A fast and accurate snp detection algorithm for next-generation sequencing data," *Nature communications*, vol. 3, p. 1258, 2012.

[11] R. Nielsen, J. S. Paul, A. Albrechtsen, and Y. S. Song, "Genotype and snp calling from next-generation sequencing data," *Nature Reviews Genetics*, vol. 12, no. 6, pp. 443–451, 2011.

[12] S. T. Sherry, M.-H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, and K. Sirotkin, "dbsnp: the ncbi database of genetic variation," *Nucleic acids research*, vol. 29, no. 1, pp. 308–311, 2001.

[13] . G. P. Consortium *et al.*, "An integrated map of genetic variation from 1,092 human genomes," *Nature*, vol. 491, no. 7422, pp. 56–65, 2012.

[14] M. Meyerson, S. Gabriel, and G. Getz, "Advances in understanding cancer genomes through second-generation sequencing," *Nat. Rev. Genet.*, vol. 11, no. 10, pp. 685–696, Oct 2010.

[15] D. C. Koboldt, Q. Zhang, D. E. Larson, D. Shen, M. D. McLellan, L. Lin, C. A. Miller, E. R. Mardis, L. Ding, and R. K. Wilson, "Varscan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing," *Genome research*, vol. 22, no. 3, pp. 568–576, 2012.

[16] D. E. Larson, C. C. Harris, K. Chen, D. C. Koboldt, T. E. Abbott, D. J. Dooling, T. J. Ley, E. R. Mardis, R. K. Wilson, and L. Ding, "Somaticsniper: identification of somatic point mutations in whole genome sequencing data," *Bioinformatics*, vol. 28, no. 3, pp. 311–317, 2012.

[17] A. Roth, J. Ding, R. Morin, A. Crisan, G. Ha, R. Giuliany, A. Bashashati, M. Hirst, G. Turashvili, A. Oloumi *et al.*, "Jointsnvmix: a probabilistic model for accurate detection of somatic mutations in normal/tumour paired next-generation sequencing data," *Bioinformatics*, vol. 28, no. 7, pp. 907–913, 2012.

[18] C. T. Saunders, W. S. Wong, S. Swamy, J. Becq, L. J. Murray, and R. K. Cheetham, "Strelka: accurate somatic small-variant calling from sequenced tumor–normal sample pairs," *Bioinformatics*, vol. 28, no. 14, pp. 1811–1817, 2012.

[19] K. Cibulskis, M. S. Lawrence, S. L. Carter, A. Sivachenko, D. Jaffe, C. Sougnez, S. Gabriel, M. Meyerson, E. S. Lander, and G. Getz, "Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples," *Nat. Biotechnol.*, vol. 31, no. 3, pp. 213–219, Mar 2013.

[20] W. Wang, P. Wang, F. Xu, R. Luo, M. P. Wong, T.-W. Lam, and J. Wang, "Fasd-somatic: a fast and accurate somatic snv detection algorithm for cancer genome sequencing data," *Bioinformatics*, vol. 30, no. 17, pp. 2498–2500, 2014.

[21] N. D. Roberts, R. D. Kortschak, W. T. Parker, A. W. Schreiber, S. Branford, H. S. Scott, G. Glonek, and D. L. Adelson, "A comparative analysis of algorithms for somatic snv detection in cancer," *Bioinformatics*, p. btt375, 2013.

[22] N. You, G. Murillo, X. Su, X. Zeng, J. Xu, K. Ning, S. Zhang, J. Zhu, and X. Cui, "Snp calling using genotype model selection on high-throughput sequencing data," *Bioinformatics*, vol. 28, no. 5, pp. 643–650, 2012.

[23] W. J. Dixon, "Analysis of extreme values," *The Annals of Mathematical Statistics*, pp. 488–506, 1950.

[24] P. Danecek, A. Auton, G. Abecasis, C. A. Albers, E. Banks, M. A. DePristo, R. E. Handsaker, G. Lunter, G. T. Marth, S. T. Sherry *et al.*, "The variant call format and vcftools," *Bioinformatics*, vol. 27, no. 15, pp. 2156–2158, 2011.

[25] A. Talwalkar, J. Liptrap, J. Newcomb, C. Hartl, J. Terhorst, K. Curtis, M. Bresler, Y. S. Song, M. I. Jordan, and D. Patterson, "Smash: a benchmarking toolkit for human genome variant calling," *Bioinformatics*, vol. 30, no. 19, pp. 2787–2795, 2014.