Noname manuscript No.
(will be inserted by the editor)

Empirical Bayes Single Nucleotide Variant Calling For Next-Generation Sequencing Data

Ali Karimnezhad* · **Theodore J. Perkins**

Received: date / Accepted: date

Abstract One of the fundamental computational problems in cancer genomics is the identification of somatic single nucleotide variants (SNVs) from DNA sequencing data. Many statistical models and software implementations for SNV calling have been developed in the literature, yet, they still disagree widely on real datasets. Based on an empirical Bayesian approach, we introduce a local false discovery rate (LFDR) estimator for SNV calling. Our approach learns model parameters without prior information, and simultaneously accounts for information across all sites in the genomic regions of interest. We also propose another LFDR-based algorithm that reliably prioritizes a given list of mutations called by any other variant-calling algorithm.We use a suite of gold-standard cell line data to compare our LFDR approach against a collection of widely used, state of the art programs. We find that our LFDR approach approximately matches or exceeds the performance of all of these programs, despite some very large differences among them. Furthermore, when prioritizing other algorithms' calls by our LFDR score, we find that by manipulating the type I-type II tradeoff we can select subsets of variant calls with minimal loss of sensitivity but dramatic increases in precision.

Keywords Empirical Bayes · DNA · Local False Discovery Rate · Multiple Hypothesis Testing · Single Nucleotide Variant.

*A. Karimnezhad (Corresponding Author)
Department of Mathematics and Statistics
University of Ottawa, Ottawa, Canada
E-mail: a.karimnezhad@uottawa.ca

T. J. Perkins
Department of Biochemistry, Microbiology and Immunology
University of Ottawa, Ottawa, Canada
E-mail: tperkins@ohri.ca

1 Introduction

Recent developments in next generation sequencing (NGS) technologies provide an insight into the task of mutation calling [1] and have made it possible to characterize the genomic alterations in a tumor in an unbiased manner [2]. With the advancement of NGS technologies, the number of large-scale projects (especially cancer projects) dealing with point mutation in various tumor types has been increased rapidly, and many bioinformatics tools have been developed.

Several packages with different algorithms have been introduced in recent years to increase the accuracy in the mutation detection procedure. Perhaps, SAMtools [3] is the most cited package for manipulating and converting alignments to different formats. It is also used for sorting and merging alignments, generating per-position information in pileup and mpileup formats, as well as calling single nucleotide variants (SNVs) and short insertion and deletions (INDELS). VarScan2 [4], a newer version of VarScan [5], reads SAMtools's pileup or mpileup output and detects SNVs, INDELs and copy number variations using separate commands. Mutect2 [6], an algorithm based on a Bayesian classifier, is claimed to be highly sensitive in the detection of very low frequency SNVs. VarDict [7] calls different types of mutations, including SNVs, multiple-nucleotide variants, INDELs, complex and structural variants at the same time. Pisces [8], which includes a variant-collapsing algorithm to unify variants broken up by read boundaries, basic filtering algorithms, and a simple Poisson-based variant confidence-scoring algorithm, is aimed to reduce noise or increase the likelihood of detecting true variants. In a recent study, Karimnezhad et al. [9] compared the performance of the above-mentioned variant callers on different sets of NGS datasets sequenced on different platforms. For a recent up-to-date list of mutation callers, readers may also refer to Xu [10], where 46 mutation callers are reviewed.

Most variant callers in the literature rely on several pre-defined but adjustable parameters such as minimum base call quality (BQ), minimum mapping quality (MQ), strand bias threshold, etc. Remarkably, many of the parameters as well as their default values are not the same among the existing variant callers. For example, there is no minimum allele frequency defined in Mutect2. See Table 1 for a list of selected parameters along with their default values in some selected algorithms. These parameters can strongly affect the output of the variant-calling algorithms. Best values for the parameters may in general depend on the type of sequencing data, the type of mutations sought (somatic or germline), noise characteristics of the dataset, etc. As such, most users/analysts may not have enough expertise to adjust those pre-determined parameters properly. On the other hand, relying on default values for parameters may result in unreliable results and is problematic when comparing programs, because their default parameters differ.

The above approaches are either conventional Bayesian or frequentist methods, and do not take multiplicity and testing efficiency issues into account. Moreover, many programs fail to output a criterion by which SNV calls can

Table 1: Default values for options in different mutation callers. A dash means that the corresponding parameter was not defined in the caller's settings.

Option	Mutect2	VarScan2	SAMtools	VarDict	Pisces
threshold for allele frequency	–	0.01	–	0.05	0.01
min BQ score	10	15	13	22.5(Illumina) 15(PGM)	20
BQ score threshold	18	–	–	–	—
max BQ score	–	–	–	–	100
min MQ	20	null	0	null	1
mean min MQ	–	null	–	null	–
maximum min MQ	null	–	–	–	–
min coverage	–	8	–	–	10
maximum coverage	–	–	250	–	–
supporting reads to call a variant	–	2	–	–	–
minimum variant quality score	–	–	–	–	15
threshold for variant quality score filter	–	–	–	–	20
strand bias filter	–	–	–	–	0.5
minimum reads to strand bias	2	–	–	–	–

be ranked, such as p-values, so that adjusting preference between type I and type II errors or constructing ROC curves is not naturally supported. In a recent study, Zhao et. al [11] have developed an optimal empirical Bayesian testing procedure to detect variants in NGS data, which is based on pooling a normalized amount of DNA from multiple samples. Due to a capacity issue, they assume samples may be distributed and sequenced independently in $M > 1$ pools and each pool consists of N individuals. Although having M pools with N individuals may be cost-effective, however, our available samples (similar to many other clinical labs) instruct us to base the model only on one DNA sample for one individual.

In this paper, we use an empirical Bayesian approach to develop a local false discovery rate (LFDR) estimator for SNV detection. In contrast to existing algorithms, our novel approach calls SNVs not just on a site-by-site basis, but by simultaneously using information across all the sites to build a probabilistic model of the data.

To the best of our knowledge, LFDR estimation has not been employed in the task of variant-calling, but it has been well developed in a variety of other contexts, and different strategies have been introduced in the literature. To name a few LFDR estimation based approaches, readers may refer to Pan et al. [12], Efron et. al [13], Efron [14], Padilla and Bickel [15], Yang et al. [16], Karimnezhad and Bickel [17], and Karimnezhad [18].

Obviously, all variant-calling algorithms determine mutations based on some observed evidence, but since evidence supporting variants (including number of reference and alternative read counts) differs from site to site, the

confidence in calling a site as a variant site varies. We fill this gap by introducing an LFDR-based algorithm that meaningfully scores variants called by any variant caller and prioritizes them from most to least probable variants. This helps with significantly reducing many false positives.

The structure of this work is as follows. In Section 2, we provide a detailed presentation of the model, method and the algorithm we propose. In Section 3, we use a suite of gold-standard cell line data to evaluate the performance of our proposed LFDR approach against a collection of widely used, state of the art programs: MuTect2, SAMtools, VarScan2, VarDict, and Pisces. In Section 4, we propose a modified version of the LFDR algorithm so that it can prioritize variant called by any desired variant caller. Some discussion and concluding remarks are provided in Section 5.

2 Statistical Model and Methods

We consider analyzing a single DNA sample extracted from a tumor, mixture of tumor and normal tissue from a patient, or tissue from a healthy patient, where only germline mutations are expected to be seen. We also assume that the data has been mapped to a reference genome, which specifies one of A, C, G, or T bases for every position. Each mapped read shows one of the four possible bases (A, C, G, or T) for the same position. With no loss of generality, we also suppose, for a replicate (technical or biological), that sequencing covers p sites (individual positions in the genome) of which p_0 sites are non-mutant. For each locus i, $i = 1, \ldots, p$, we assume that there are K_i (short) known number of reads covering the locus i. At each locus i, we suppose there are four possible bases (A, C, G, T) of which R_i random reads carry the reference allele. We further suppose that of the remaining $K_i - R_i$ reads, M_i random number of reads carry the most alternative (dominant) allele and, we assign the remaining random alternative reads to X_{1i} and X_{2i}. We briefly refer to these random variables by $\boldsymbol{X}_i = (R_i, M_i, X_{1i}, X_{2i})$. Generally, the larger M_i is, the more support we have for an alternative allele being present at site i in the DNA being sequenced. Indeed, if the data had no sequencing or mapping errors, $M_i > 0$ could only arise as the result of an alternative allele being present. However, it is well-recognized that sequence data does contain errors. Although those errors can arise in many ways, the standard model error employed by the community is the error in mistakenly reporting an alternative allele as one of the other three possible alleles, and vice versa. We refer to this error by e. Obviously, if there is no error, mutant sites are expected to be those $p - p_0$ sites, for which M_i is positive.

To relate the observed data at site i, $\boldsymbol{X}_i$, to the model parameters, μ_i, θ_i, we assume that each read is drawn independently from either the reference or alternative allele pools. However, we also assume an independent chance e of read error in each read at each site, which randomly changes the correct allele to one of the other three options. Then, as in Mutect2 [6], we define four probabilities. The probability that a randomly chosen read covering site

i shows the reference allele is $p_{R_i} = \theta_i \frac{e}{3} + (1-\theta_i)(1-e)$. This is explained as the sum of the probabilities that the DNA actually contained the reference allele at site i and it was correctly read, $(1-\theta_i)(1-e)$, and probability that the DNA actually contained the alternative allele but it was misread, and by chance, it was misread as the reference allele, $\theta_i \frac{e}{3}$. Similarly, the probability that a random read covering site i shows the alternative allele is $p_{M_i} = \theta_i(1-e) + (1-\theta_i)\frac{e}{3}$, which arises either as correct reading of the alternative allele or misreading the reference allele as the alternative. Finally $p_{X_{1i}} = p_{X_{2i}} = \frac{e}{3}$ is the chance that one of the two other alleles that are not reference or alternative occurs at site i. Note that although these probabilities only include θ_i and not μ_i, they implicitly depend on μ_i in that $\theta_i = 0$ if and only if $\mu_i = 0$. Putting these together, the probability of the total data at site i, $\boldsymbol{X}_i$, is multinomial with K_i tries (reads) and probabilities of reference, alternative, and the other two alleles. At this point, we diverge from the model of Mutect2 by proposing a new hierarchical model governing the parameters μ_i and θ_i, which ultimately allows us to develop our empirical Bayesian LFDR approach:

$$\begin{cases} \boldsymbol{X}_i|\theta_i \sim \text{Multi}\left(K_i, p_{R_i}, p_{M_i}, p_{X_{1i}}, p_{X_{2i}}\right), \\ \theta_i|\mu_i \sim \mu_i g(\theta_i), \\ \mu_i \sim \text{Ber}(1-\pi_0), \end{cases} \tag{1}$$

where $p_{R_i} = \theta_i \frac{e}{3} + (1-\theta_i)(1-e)$, $p_{M_i} = \theta_i(1-e) + (1-\theta_i)\frac{e}{3}$, $p_{X_{1i}} = \frac{e}{3}$, $p_{X_{2i}} = \frac{e}{3}$, $i = 1, \ldots, p$.

To discover whether site i is a mutant site, we propose testing the null hypothesis $H_{0i} : \mu_i = 0$ against the alternative hypothesis $H_{1i} : \mu_i = 1$, $i = 1, \ldots, p$. To do so, we focus on estimating $\psi_i \equiv P(\mu_i = 0|\boldsymbol{X}_i)$, the posterior probability that the null hypothesis H_{0i} is true. Once estimated, it is compared with a pre-specified threshold leading to either rejecting or failing to reject the null hypothesis. This quantity is well-known in the literature as LFDR.

With the above settings, the following probability functions are derived under the null and the alternative hypotheses, respectively,

$$\begin{aligned} P(\boldsymbol{X}_i|\mu_i = 0) &= \binom{K_i}{R_i, M_i, X_{1i}, X_{2i}} (1-e)^{R_i} \left(\frac{e_{ij}}{3}\right)^{M_i + X_{1i} + X_{2i}} \\ &= \binom{K_i}{R_i, M_i, X_{1i}, X_{2i}} (1-e)^{R_i} \left(\frac{e_{ij}}{3}\right)^{K_i - R_i} \end{aligned} \tag{2}$$

and

$$\begin{aligned} P(\boldsymbol{X}_i|\mu_i = 1) &= \int P(\boldsymbol{X}_i|\mu_i = 1, \theta_i) g(\theta_i|\mu_i = 1) d\theta_i \\ &= \int_0^1 \binom{K_i}{R_i, M_i, X_{1i}, X_{2i}} \left(\theta_i \frac{e}{3} + (1-\theta_i)(1-e)\right)^{R_i} \\ &\quad \times \left(\theta_i(1-e) + (1-\theta_i)\frac{e}{3})\right)^{M_i} \left(\frac{e}{3}\right)^{X_{1i}+X_{2i}} g(\theta_i) d\theta_i. \end{aligned} \tag{3}$$

Now, using the Bayes formula, the LFDR can be expressed by

$$\psi_i = \frac{\pi_0 P(\boldsymbol{X}_i|\mu_i = 0)}{\pi_0 P(\boldsymbol{X}_i|\mu_i = 0) + (1-\pi_0)P(\boldsymbol{X}_i|\mu_i = 1)}, \tag{4}$$

where $P(\boldsymbol{X}_i|\mu_i = 0)$ and $P(\boldsymbol{X}_i|\mu_i = 1)$ are given by (2) and (3), respectively, and π_0 is the proportion of non-mutant sites. Both the parameters π_0 and $g(\theta_i)$ (in $P(\boldsymbol{X}_i|\mu_i = 1)$) are unknown and need to be estimated before making any inference.

Now, let δ_i be a binary decision rule corresponding to ith set of hypotheses H_{0i} and H_{1i}. We assume that $\delta_i = 1$ if the null hypothesis H_{0i} is rejected, and $\delta_i = 0$, otherwise. But since such binary decisions can lead to some errors, we define the following loss function when testing the null hypothesis at site i, i.e.,

$$L(\mu_i, \delta_i) = \begin{cases} 0 & \delta_i = \mu_i = 1 \,\text{or}\, \delta_i = \mu_i = 0, \\ l_I & \delta_i = 1,\ \mu_i = 0, \\ l_{II} & \delta_i = 0,\ \mu_i = 1, \end{cases}$$

where l_I and l_{II} are loss values incurred due to making type I and type II errors, respectively, see Karimnezhad and Bickel [17]. Also, see Zhao et. al [11] where a specific version of this loss with $l_I = \lambda$ and $l_{II} = 1$ is used.

To take all p sites in a sample to account, suppose that $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_p)$ represents a vector of estimators of $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)$ and measure the aggregated loss by $L(\boldsymbol{\delta}, \boldsymbol{\mu}) = \sum_{i=1}^{p} L(\mu_i, \delta_i)$, which takes both type-I and type-II errors into account. Now, to derive a Bayesian decision rule in the above-mentioned hypothesis testing problem, let $\rho_{\boldsymbol{X}}(\boldsymbol{\mu}, \boldsymbol{\delta}) = E[L(\boldsymbol{\mu}, \boldsymbol{\delta})|\boldsymbol{X}]$ denote the posterior risk of choosing a decision vector $\boldsymbol{\delta}$, where $\boldsymbol{X} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_p)$. Then, similar to Karimnezhad and Bickel [17], it can be verified that the Bayesian decision vector $\boldsymbol{\delta}^B = (\delta_1^B, \ldots, \delta_p^B)$ with

$$\delta_i^B = \begin{cases} 0 & \psi_i > \frac{l_I}{l_I + l_{II}}, \\ 1 & \psi_i \leq \frac{l_I}{l_I + l_{II}}, \end{cases} \tag{5}$$

where ψ_i is given by (4), minimizes the posterior risk w.r.t. $\boldsymbol{\delta}$.

The above equation suggests that mutant sites can be determined by estimating ψ_i, $i = 1, \ldots, p$, and then comparing it with the threshold $\frac{l_I}{l_I + l_{II}}$. However, estimating ψ_i is challenging due to the complicatedness of the form of $P(\boldsymbol{X}_i|\mu_i = 1)$ as well as $g(\cdot)$, the distribution of AFs.

One immediate solution to estimating $g(\cdot)$ would be to assign a non-informative prior to the alternative AFs θ_i. In a similar situation, Zhao et. al [11] consider a uniform distribution on the interval $(0, a)$, and then they estimate the value of a as well as π_0 using the traditional method of moments. While simple, this assumption seems unrealistic. Figure 1 represents the distribution of AFs at known mutant as well as non-mutant sites from TST-GM12878 data (see Section 3 for more information).

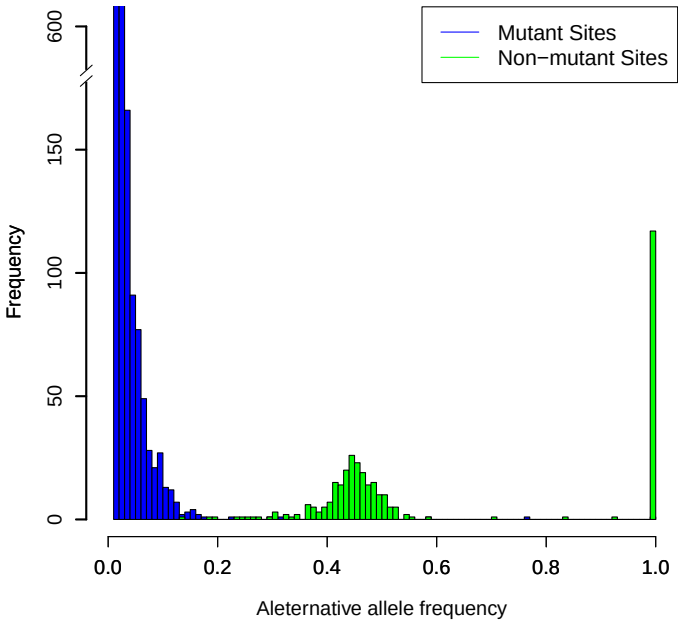


Fig. 1: Histogram of distribution of alternative AFs from TST-GM12878 data.

From Figure 1, we observe that the assumption that AFs take any number between 0 and 1 with equal probability is unrealistic. Estimating π_0 may be connected to the estimation of $g(\cdot)$ and consequently, a bad estimate of $g(\cdot)$ may directly lead to a bad estimate of π_0. This indeed is true in some datasets with ambiguous AF ranges, including our datasets. Clearly, from this figure, AFs of mutant sites is distributed around either 0.5 or 1. This highlights the importance of finding a suitable estimate of $g(\cdot)$. Dependency of estimated values of parameters of LFDR to each other can be seen in different algorithms in the literature. See for example Zhao et. al [11] where their proposed estimator of π_0 is a function of the estimated value of the upper bound a in the uniform prior assigned to the alternative AF. For other examples, see Pan et al. [12] and Karimnezhad and Bickel [17].

If it was possible to fit a known density function to $g(\cdot)$, then perhaps one reasonable solution could be to apply the Monte-Carlo integration technique. But, as Figure 1 suggests, the alternative AFs do not seem to follow one of the well-known distributions in the literature. As an innovative approach, we suggest estimating $g(\cdot)$ empirically, based on the alternative AFs at a set of known or suspected mutated sites. This raises the question of which set of sites to use.

An immediate solution to the above problem would be to take all sites for which M_i counts are positive. But as many non-zero alternative AFs in NGS studies turn out to correspond to artifacts, we need to find and exclude them from our list so that a precise empirical distribution for $g(\cdot)$ can be estimated. Suppose that $\mathcal{I}_s$ stands for the set of s indices of those non-zero θ_i corresponding to mutant sites, and $\widehat{\boldsymbol{\theta}}_s$ is the vector of the corresponding empirical AFs. In our seetings, s represents the number of elements of the set $\mathcal{I}_s$. Now, let $G(\cdot)$ represent the cumulative distribution function (CDF) that corresponds to $g(\cdot)$. Then, the corresponding empirical CDF is given by $\widehat{G}_s(t) = \frac{1}{s}\sum_{l\in\mathcal{I}_s} 1_{\theta_l \le t}$, $t \in \Re$. Obviously, due to the strong law of large numbers, $\widehat{G}_s(t)$ converges to $G(t)$ as $n \to \infty$ almost surely, for every value of t, and this implies that $\widehat{G}_s(t)$ is a consistent estimator. Then, one may estimate $P(\boldsymbol{X}_i|\mu_i = 1)$ by

$$P_{\widehat{\boldsymbol{\theta}}_s}(\boldsymbol{X}_i|\mu_i = 1) \simeq \frac{1}{s}\sum_{l\in\mathcal{I}_s}\left\{\binom{K_i}{R_i, M_i, X_{1i}, X_{2i}}\left(\theta_l\frac{e}{3} + (1-\theta_l)(1-e)\right)^{R_i}\right.$$
$$\left.\times\left(\theta_l(1-e) + (1-\theta_l)\frac{e}{3})\right)^{M_i}\left(\frac{e}{3}\right)^{X_{1i}+X_{2i}}\right\}. \qquad (6)$$

Since $\mathcal{I}_s$ includes many artifacts, s needs to be optimally learned. In the next subsection, we propose an algorithm that allows for repeatedly updating the set of mutant sites and leads to a precise estimate of $g(\cdot)$.

2.1 An empirical Bayes mutation detection procedure

To discover mutant sites, we propose the procedure outlined in Algorithm 1. The proposed algorithm, like many existing procedures in the literature, including VarScan2 [4], Mutect2 [6], VarDict [7] and Pisces [8], is comprised of three main steps.

In the pre-processing step, we propose excluding low quality bases from the input file (either FASTQ or BAM format). Thus, observations in our model are only those bases that pass a minimum average BQ and average MQ threshold. BQ corresponds to an error rate of $e = 10^{-\frac{BQ}{10}}$, see for example Cai et al. [6] and Dunn, et al. [8]. Thus taking $BQ = 20$ corresponds to an error rate of 0.01 which can be inferred as expecting one miscalled base in reading 100 bases. MQ is related to aligning reads to a reference genome, and it usually varies between 0 and 60. Similar to Pisces, we impose a threshold of 20, and 30 to the minimum BQ and MQ, respectively. We also reflect the corresponding error rate in Step 2.2 of our model by taking $e = 0.01$. This assumption facilitates the speed of computation, as our data analyses convinced us, it has an ignorable effect on the number of detected mutations. Different publicly available softwares such as SAMtools[3] and bam-readcount (https://github.com/genome/bam-readcount) can be applied to exclude such low quality reads. Once low quality bases are excluded, the counts K_i, R_i, M_i, X_{1i} and X_{2i} need to be formed.

Although in equation (6) we only include all non-zero empirical AFs, one may tweak the algorithm by just assuming that AFs follow, for example, a uniform distribution on the interval $(0, 1)$, or a subset of it. Thus, $\mathcal{I}_s^j$ in Step 2.6 of the algorithm, may be alternatively taken to be a set of N (for example 1000) sites for which their AFs are randomly generated from a uniform distribution on the interval $(0, 1)$. However, once the estimation procedure enters Step 2.10, the updated $\mathcal{I}_s^j$ set will only depend on the actual empirical AFs. A third approach would be to estimate LFDRs by focusing only on the original uniformly sampled AFs, without updating the set of indices $\mathcal{I}_s^j$ in step 2.10, i.e., for all j, $\mathcal{I}_s^j = \mathcal{I}_s^1$. We refer to this approach by "uniform" estimation. For comparison purposes, we use these three methods in our data analysis and will refer to them by "empirical", "uniform/empirical" and "uniform" estimation of $g(\cdot)$, respectively. In Step 2.10, we took $\epsilon = 0.001$.

In the post-processing step, we take 0.01 and 10 as default AF threshold (AFT) and read depth threshold (DPT) values, respectively. This is not uncommon, and many variant callers impose such thresholds. For example, Pisces [8] has the same default values for these thresholds. We remark that one may apply this filtering in Step 1. However, the final estimated π_0 will reflect the proportion of non-mutant sites w.r.t. the filtered input file rather than the original one.

3 Performance evaluation

We now focus on evaluating our proposed mutation calling algorithm on some data generated by clinical assays. We employ DNA sequencing data generated from two well-characterized Coriell cell lines, GM12877 and GM12878, studied by Karimnezhad et al. [9]. The corresponding genomes have been well characterized and thus, there are lists of known mutations compared to the reference human genome that help us measure whether or not mutations detected by our proposed algorithm are correct. Among such lists, we rely on a list of known mutations published by Eberle et al. [19] where a comprehensive and genome-wide catalog of high-confidence variants mutations for a collection of Coriell cell lines, including GM12877 and GM12878 is presented (available through https://www.ncbi.nlm.nih.gov/gap/ under accession phs001224.v1.p1). The data we used was sequenced in two different ways: 1) on an Illumina NextSeq500 sequencer using Illumina's TruSight170 targeted sample preparation method (TST170 for short), and on an Ion Torrent PGM sequencer using an Oncomine Focus targeted panel (OF for short).

The TST170 data spans 514761 total bases, covering parts of 170 genes, and includes six technical replicates of each of the Coriell cell lines GM12877 and GM12878. By intersecting the list of known mutations with the TST170 genomic regions, we determined that our GM12877 and GM12878 data should contain 336 and 343 mutations, respectively. The OF panel covers 29008 total bases in 47 genes, and includes three technical replicates of each of the Coriell cell lines GM12877 and GM12878. There should be 24 and 26 known mutations

present in each replicate of the GM12877 and GM12878 data, respectively. For more information regarding the datasets and also sequencing platforms, readers may refer to Karimnezhad et al. [9].

For a given single replicate, let $TP = \sum_{i=1}^{p} \delta_i^B \mu_i$, $FP = \sum_{i=1}^{p} \delta_i^B (1-\mu_i)$, $FN = \sum_{i=1}^{p} (1-\delta_i^B)\mu_i$ represent the total number of true positives, false positives, and false negatives, respectively. We measure the performance of the algorithms by computing precision or positive predictive value $Prec = \frac{TP}{TP+FP}$ and sensitivity $Sens = \frac{TP}{TP+FN}$. A good algorithm is expected to have high $Prec$ and $Sens$ rates.

To apply the algorithm on the TST170 as well as OF replicates, we used the bam-readcount package to calculate the counts K_i, R_i, M_i, X_{1i} and X_{2i}, and then followed Steps 1-3 of Algorithm 1. We then compared the list of final detected variants with the lists of known mutations. Also, to investigate the impact of a chosen LFDR threshold in Step 2.1 of the algorithm on the detection accuracy, we picked 10^{-300}, 10^{-200}, 10^{-100}, 10^{-50}, 10^{-20}, 10^{-10}, 10^{-2} and 0.5.

We measured the performance of the proposed approaches by calculating TP, FP, FN, and consequently $Prec$ and $Sens$ values for different datasets and LFDR thresholds. As a second performance measurement, we compared the proposed algorithm with Mutect2[6], VarScan2[4], SAMtools [3], VarDict[7] and Pisces [8]. Illumina offers Pisces for the analysis of TST170 data through their customized pipeline. The application accepts paired-end fastq files as inputs, generates BAM files, and after aligning to the reference human genome (hg19) by the Isaac aligner [20], uses Pisces [8] to generate a list of mutations. Then, the final list of mutations is generated after some internal filtering. We should add that, to compare the performance of our proposed algorithm with Pisces, as well as VarScan2, Mutect2, SAMtools and VarDict, we used the same aligned BAM files to reduce some possible alignment errors. Because differences in default parameters of the five algorithms in Table 1 are potential sources of discrepancies in the list of final mutations, we set their parameters as similar as possible. We set the minimum variant AF to 0.01. For MuTect2, which does not have such a parameter, we post-filtered the results to remove MuTect2 calls with apparent frequency less than 0.01. We set minimum base call quality and minimum mapping quality to 20. Finally, we set the minimum coverage for a called variant at 10 reads.

Figure 2 represents the average $Prec$ and $Sens$ values over replicates based on calls made by the LFDR approach for different thresholds, as well as the other five variant callers. From this figure, we observe that for all the LFDR thresholds, $Sens$ values are high while $Prec$ is increased by decreasing the LFDR threshold. Indeed, when the LFDR threshold is high, say 0.5 for example, many false positives are allowed to be in the list of variants called. However, a strict LFDR threshold, say 10^{-50} for example, leads to few(er) false positives and consequently an improved $Prec$ is gained. All the three approaches methods performed nearly the same. Especially, when the LFDR

Algorithm 1 LFDR-based variant-calling algorithm.

Step 1. Pre-processing. Exclude reads not passing minimum average base call quality (BQ) and minimum average mapping quality (MQ) thresholds.

Step 2. Estimation.

Step 2.1 Specify the LFDR threshold $l_I/(l_I + l_{II})$.

Step 2.2 Specify the error rate e, $i = 1, \ldots, p$.

Step 2.3 For each $i = 1, \ldots, p$, calculate $P(\boldsymbol{X}_i|\mu_i = 0)$.

Step 2.4 Take $j = 1$. This represents an iteration number.

Step 2.5 Choose an initial value for $\pi_0 \in (0, 1)$, set $\widehat{\pi}_0^j = \pi_0$.

Step 2.6 Define $\mathcal{I}_s^j$ to be the set of all sites for which M_i is positive, and let $\widehat{\boldsymbol{\theta}}_s^j = \widehat{\boldsymbol{\theta}}_s$ represent the vector of the corresponding empirical AFs.

Step 2.7 For each $i = 1, \ldots, p$, calculate $P_{\widehat{\boldsymbol{\theta}}_s^j}(\boldsymbol{X}_i|\mu_i = 1)$.

Step 2.8 Estimate LFDRs by

$$\widehat{\psi}_i^j = \frac{\widehat{\pi}_0^j P(\boldsymbol{X}_i|\mu_i = 0)}{\widehat{\pi}_0^j P(\boldsymbol{X}_i|\mu_i = 0) + (1 - \widehat{\pi}_0^j) P_{\widehat{\boldsymbol{\theta}}_s^j}(\boldsymbol{X}_i|\mu_i = 1)}, \quad i = 1, \ldots, p.$$

Step 2.9 Compute the Bayes rule

$$\delta_i^{B,j} = \begin{cases} 0 & \widehat{\psi}_i^j > \frac{l_I}{l_I + l_{II}}, \\ 1 & \widehat{\psi}_i^j \leq \frac{l_I}{l_I + l_{II}}. \end{cases}$$

Step 2.9 Step up j by one, i.e., $j = j + 1$.

Step 2.10 Reset $\mathcal{I}_s^j$ to be the set of indices i where $\delta_i^{B,j-1} = 1$, and update $\widehat{\boldsymbol{\theta}}_s^j$ based on the new $\mathcal{I}_s^j$.

Step 2.11 Define $\widehat{\pi}_0^j = 1 - \frac{1}{p}\sum_{i=1}^p \delta_i^{B,j-1}$.

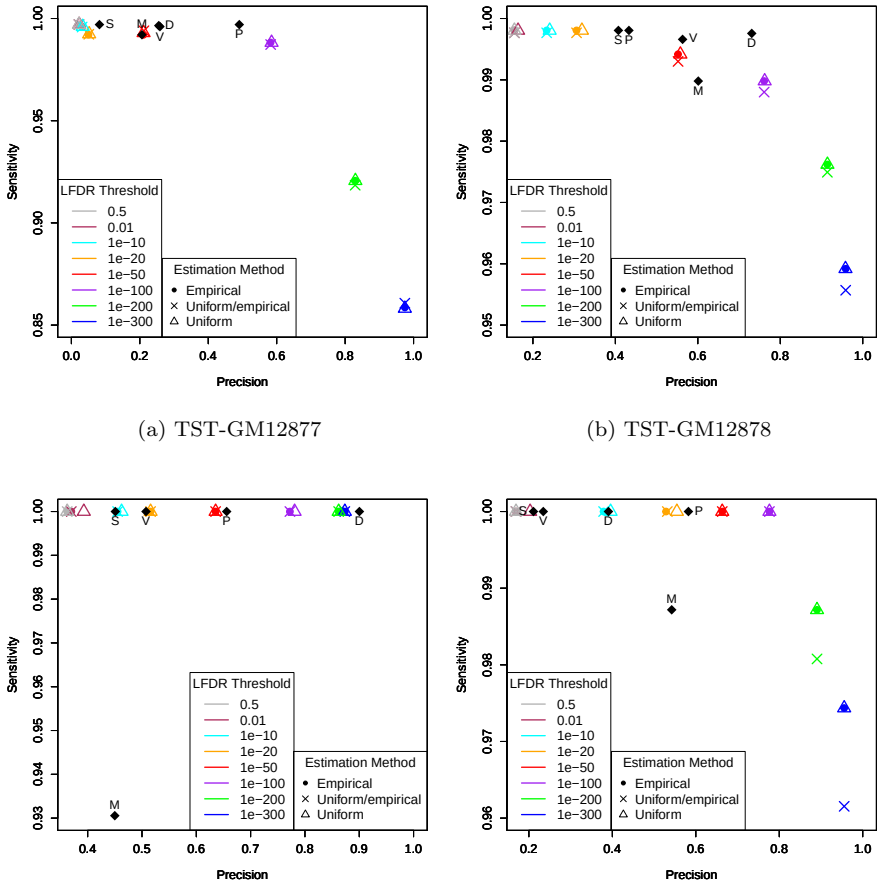
Step 2.12 Repeat Steps 2.7-2.11 until the difference between the new estimate of $\widehat{\pi}_0$ and its previous value does not exceed a small number ϵ, i.e., $|\widehat{\pi}_0^j - \widehat{\pi}_0^{j-1}| < \epsilon$. Refer to the last j by j^*.

Step 2.13 Now, any site i for which $\delta_i^{B,j^*} = 1$ is a mutant site.

Step 3. Post-Processing. Exclude sites with AF and read depth (or coverage) not passing a pre-determined threshold.

threshold is below 10^{-50}, the three approaches led to the same $Prec$ and slightly different $Sens$ values.

Comparing with the other variant callers, we observe that the LFDR approach has equal or better $Sens$ for some LFDR thresholds compared to all other algorithms. And for other thresholds, it has equal or better $Prec$. For example, from panel (a) in Figure 2 we observe that the LFDR approach with a threshold of 10^{-50} outperforms all the other five variant callers in terms $Prec$. However, it reports a smaller $Sens$. When looking at other panels, slightly different but remarkable performance is observed. For example, by choosing the same LFDR threshold, we observe in panel (b) that the LFDR approach outperforms MuTect2 in terms of $Prec$, and has almost the same $Sens$, and outperforms the remaining four variant callers in terms of $Prec$. However, its $Sens$ is a bit smaller than the others. But when looking at panel (d), we observe that the LFDR approach with the same LFDR threshold outperforms all the other variant callers in terms of both $Prec$ and $Sens$. We observe that



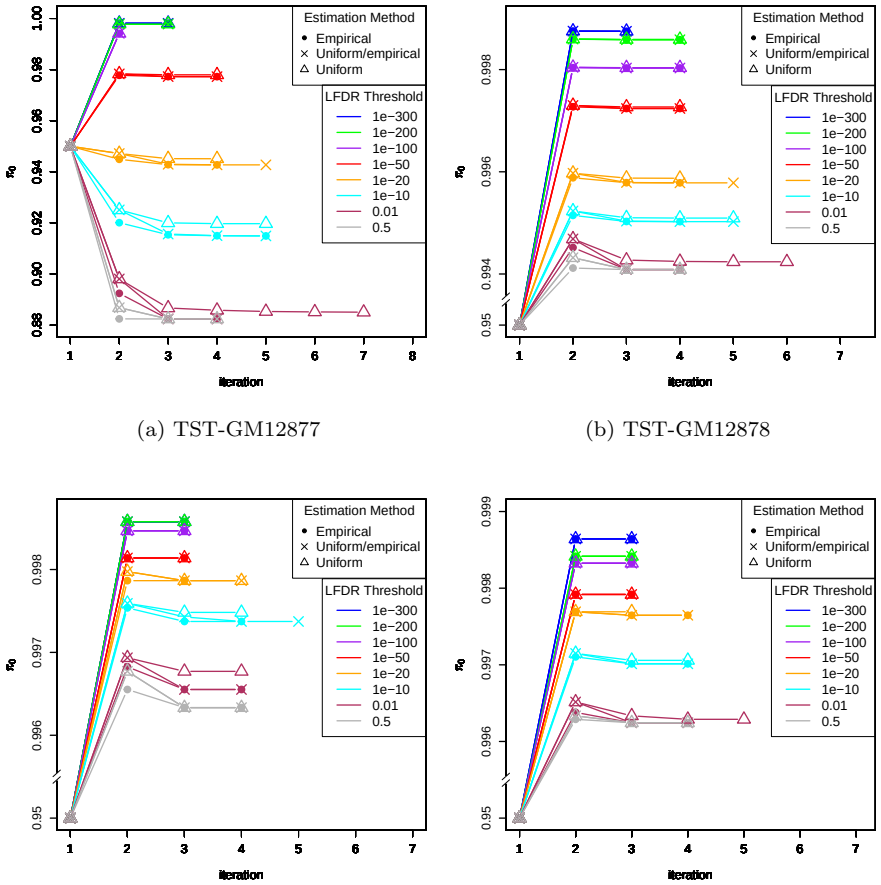
(a) TST-GM12877 (b) TST-GM12878

(c) OF-GM12877 (d) OF-GM12878

Fig. 2: Average of *Prec* and *Sens* values over replicates based on calls made by the LFDR approach for different LFDR thresholds. Average of *Prec* and *Sens* values over replicates based on calls made by the other five variant callers were added for comparison purposes. M, V, S, D and P stand for MuTect2, VarScan2, SAMtools, VarDict and Pisces, respectively.

having a varying LFDR threshold allows for a wide range of tradeoff between *Sens* and *Prec*.

Figure 3 represents convergence of estimated π_0 in replicate 1 of each dataset for different values of LFDR thresholds. From this figure we observe that the algorithm converged mostly in 3-8 iterations, and most remarkably, for stringent LFDR thresholds (10^{-100} or less), it converged in 3 iterations. We also note that in each panel, those stringent LFDR thresholds resulted in the



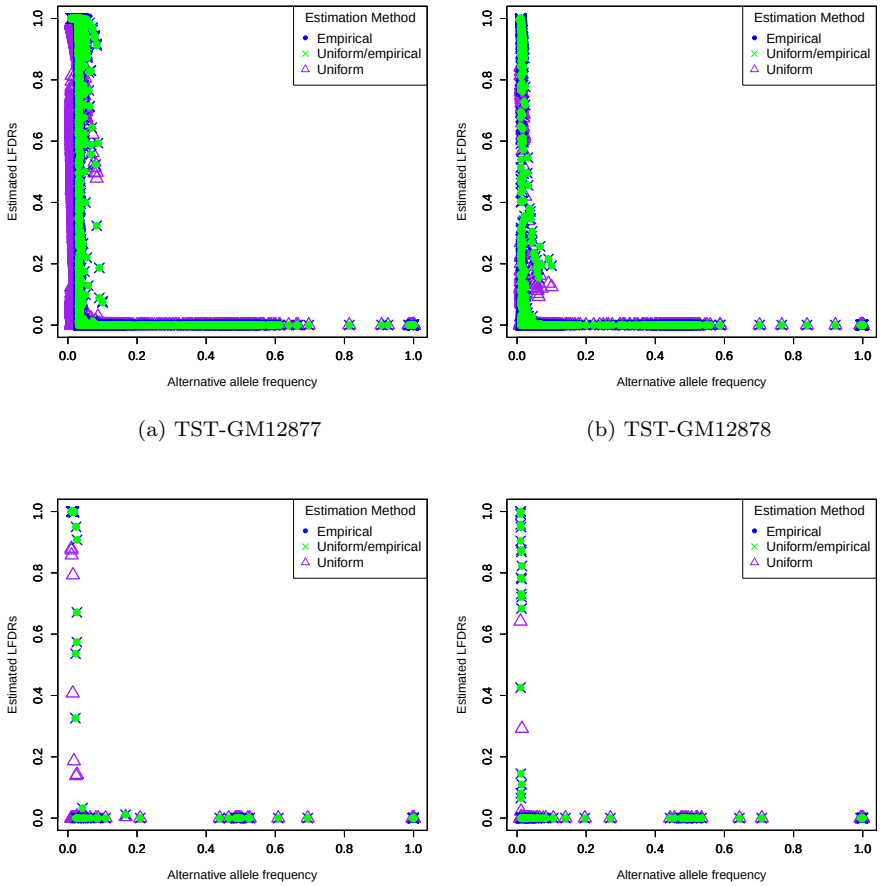
(a) TST-GM12877 (b) TST-GM12878

(c) OF-GM12877 (d) OF-GM12878

Fig. 3: Estimated π_0 in replicate 1 of each dataset for different values of LFDR thresholds. The solid grey line represents the true π_0.

greatest accuracy in estimating π_0. In fact, the lower is the LFDR threshold, the more accurate is the corresponding estimated π_0. Comparing with Figure 2, we also observe that the most accurate estimated π_0 leads to the highest *Prec*. However, this is not neccessarily true when looking at *Sens* values.

We also investigated the impact of each method on the magnitude of LFDR estimates. Figure 4 represents estimated LFDRs for replicate 1 of each of our datasets based on taking 10^{-300} as the LFDR threshold. This figure reflects that both the empirical and uniform/empirical approaches coincide on detecting the same variants. The corresponding estimated LFDRs are mostly either close to zero (that corresponds to mutant sites) or one (that corresponds to



(a) TST-GM12877

(b) TST-GM12878

(c) OF-GM12877

(d) OF-GM12878

Fig. 4: Estimated LFDRs for replicate 1 of each dataset based on taking 10^{-300} as the LFDR threshold.

non-mutant sites), revealing that both approaches were able to nicely categorize the sites as either mutant or non-mutant sites. But, LFDRs estimated using the uniform approach did not follow this structure and there are many sites with medium estimated LFDRs. This concludes that both the empirical and uniform/empirical approaches outperform the uniform approach in classifying sites to non-mutant as well mutant categories.

4 LFDR as a variant prioritization tool

We suggest that our LFDR calculation can be applied to prioritize any list of mutations called by any mutation caller from most to least probable mutation. This can be done by modifying Algorithm 1 so that $\widehat{\pi}_0$ is estimated as the number of reference sites according to the caller, out of all those assayed, and $g(\cdot)$ is estimated as the empirical distribution of variant sites. The LFDR calculation is then applied to all variant sites using these estimates of π_0 and $g(\cdot)$. Algorithm 2 outlines the prioritization steps.

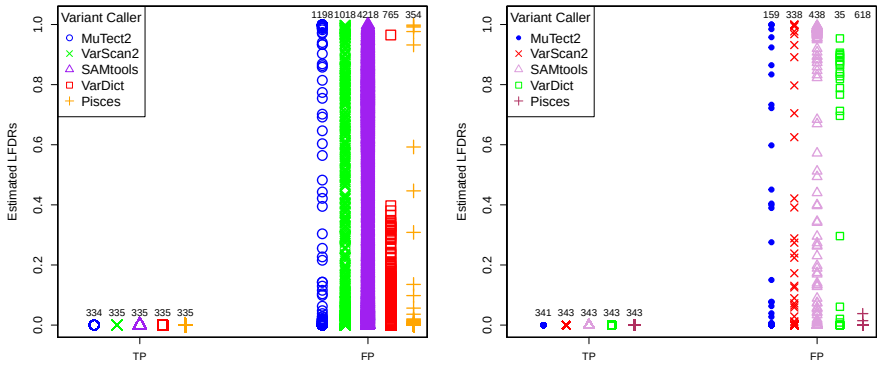
Algorithm 2 LFDR-based prioritizing algorithm for calls made by a desired variant caller.

Step 1 Specify the error rate e, $i = 1, \ldots, p$.
Step 2 For each $i = 1, \ldots, p$, calculate $P(\boldsymbol{X}_i|\mu_i = 0)$ in equation (2).
Step 3 Set $\widehat{\pi}_0 = 1 - \frac{p_0}{p}$, where p represents the total number of sites in a genomic region of interest of which p_0 sites are declared by any chosen variant caller to be non-mutant sites.
Step 4 In equation (6), define $\mathcal{I}_s$ to be the set of all $p - p_0$ sites deemed to be variant sites by the chosen variant caller, and let $\widehat{\boldsymbol{\theta}}_s$ represent the vector of the corresponding AFs.
Step 5 For each $i = 1, \ldots, p$, approximate $P_{\widehat{\boldsymbol{\theta}}_s}(\boldsymbol{X}_i|\mu_i = 1)$ in (6).
Step 6 Estimate the corresponding LFDRs by

$$\widehat{\psi}_i = \frac{\widehat{\pi}_0 P(\boldsymbol{X}_i|\mu_i = 0)}{\widehat{\pi}_0 P(\boldsymbol{X}_i|\mu_i = 0) + (1 - \widehat{\pi}_0) P_{\widehat{\boldsymbol{\theta}}^s}(\boldsymbol{X}_i|\mu_i = 1)}, \quad i = 1, \ldots, p.$$

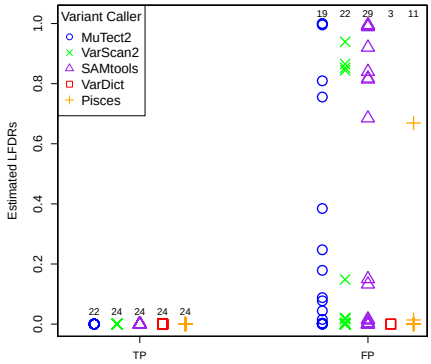
Following the steps in Algorithm 2, we calculated the LFDR values for variants detected by the five variant callers (MuTect2, VarScan2, SAMtools, VarDict and Pisces) across replicate 1 of all the datasets. Figure 5 represents the corresponding LFDRs and whether the variants are TPs or FPs, along with the number of TPs and FPs. From the figure we observe that all those TPs led to either zero or very close to zero estimated LFDR values, as expected. For example, we observe from panel (a) that of the total 1745 variants detected by MuTect2, 333 variants with estimated LFDR values of zero or very close to zero are TPs, and 1412 variants with estimated LFDRs varying between zero and one are FPs. Comparing these numbers with 336, the number of known variants expected to be present in TST-GM12878 data, we realize that imposing a small LFDR threshold would dramatically reduce the number of FPs, and consequently *Prec* gets significantly improved. The lower is the LFDRs threshold, the more FPs are eliminated from the list of variants. Therefore, the proposed algorithm, with less complication compared to Algorithm 1, can prioritize results of other variant callers by simply eliminating FPs based on their LFDR values.

Naturally, prioritizing another variant caller's output, and potentially deciding that some of those calls are below threshold, cannot increase *Sens*. *Sens* increase could only be achieved by adding in missed true variants. However,

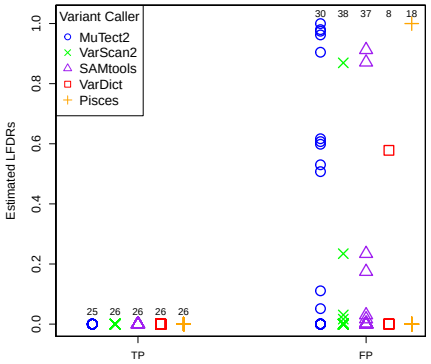


(a) TST-GM12877

(b) TST-GM12878



(c) OF-GM12877

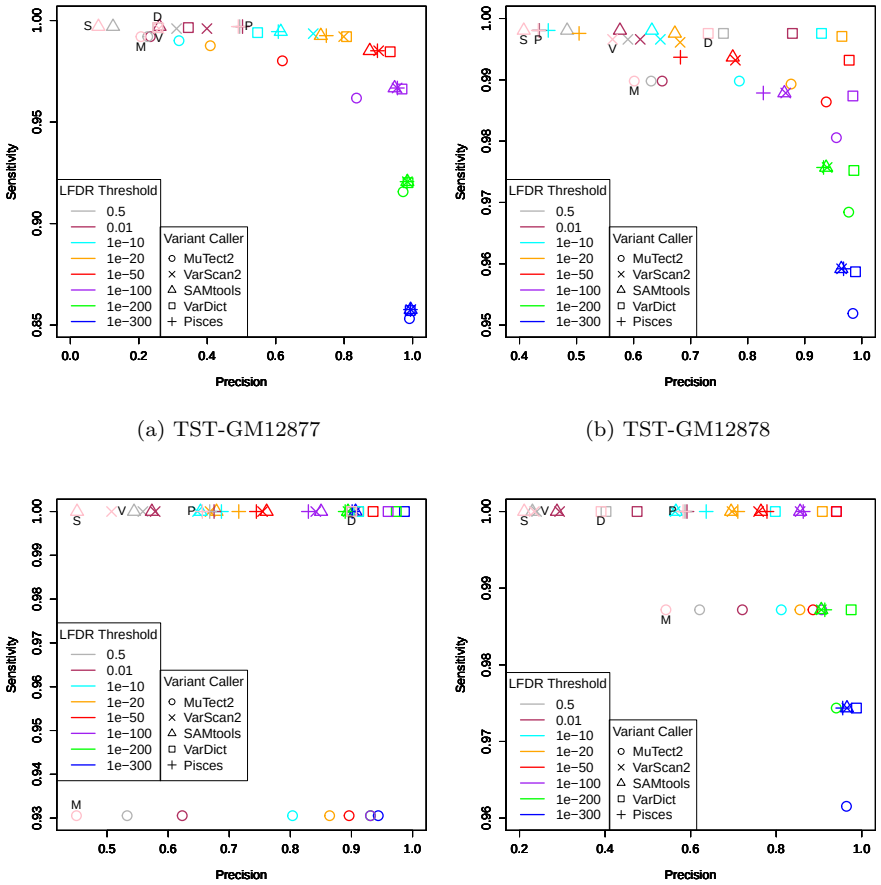


(d) OF-GM12878

Fig. 5: Estimated LFDRs for TP and FP calls made by the five varian callers in replicate 1 of each dataset.

prioritization and thresholding has the potential to eliminate many FPs, and thus increase *Prec*, hopefully with little or no loss to *Sens*.

We applied this prioritization approach to all calls made by the five variant callers in all replicates in the four datasets. We then calculated average *Prec* and *Sens* values for different LFDR thresholds as shown in Figure 6. For comparison purposes, we also included *Prec* and *Sens* values calculated for each individual variant caller in Figure 2. From Figure 6 we observe that prioritizing variants using the LFDR method successfully leads to a significant increase in *Prec* values of the variant callers with minimal loss of *Sens* values for any threshold chosen between 0.5 and 10^{-50}. As an example, from Figure



(a) TST-GM12877 (b) TST-GM12878

(c) OF-GM12877 (d) OF-GM12878

Fig. 6: Average of *Prec* and *Sens* values for calls prioritized by the LFDR approach. The pink symbols refer to the *Prec* and *Sens* values of each individual variant caller calculated in Figure 2.

6(a), VarDict led to *Prec* and *Sens* values of 0.256 and 0.997, respectively, and prioritizing its calls using the LFDR approach with a threshold of 0.05, led to 0.656 and 0.997 as new *Prec* and *Sens* values. Thus, LFDR prioritization appears to be a powerful approach to further filter the output of other variant-calling algorithms.

5 Discussion and Concluding Remarks

In this paper we introduced a novel LFDR-based approach that can be built into a single-nucleotide variant caller, or can be used to prioritize variants called by other algorithms. The algorithm requires some pre-determined numbers including the error rate e, the threshold $l_I/(l_I + l_{II})$, and an initial value for π_0. Although, the algorithm is applicable to different error rates for each site i, we took a global error rate $e = 0.01$. This corresponds to a base call quality of 20 which has been set as the default parameter in some well performed softwares including Pisces [8]. We considered 8 different LFDR thresholds, and noticed that, thresholds between 10^{-100} and 10^{-20} lead to better results in term of $Prec$ and $Sens$ values on the datasets we studied. To establish the generality of this observation, further testing would be needed on other datasets from different sources or of different types.

We used three approaches to estimate $g(\cdot)$ in equation (3). As expected, both the empirical and uniform/empirical approaches performed the same and identified the same variants. This is because both approaches lead to the same $\mathcal{I}_s$ set. The uniform approach also led to very similar results. This could be because we are ignoring the magnitude of LFDR values and we just compare them with the chosen thresholds. One possible explanation could also be the fact that the data we used in our project do not have any genuine mutations at low AFs, and thus, accurately estimating $g(\cdot)$ has little effect on the performance. However, as we showed in Figure 4, the uniform approach is not able to firmly categorize sites to either mutant and non-mutants sites.

Both of our LFDR algorithms have few parameters to adjust and thus it is easier to tune, compared to the five variant callers we used in this study. Like the five variant callers we used, our algorithm uses some pre-specified numbers for BQ, MQ and error rate e, but it does not require any other parameter to adjust, except the LFDR threshold. And yet as an advantage, the threshold can be controlled by users/analysts and is easy to interpret, due to the fact that the LFDR is a probability and varies only on the interval $[0, 1]$. The closer to zero (one) it is, the more confidence is gained in detecting variant (non-variant) sites.

We end our discussion with highlighting that in some situations, one may notice conflicts in calling a site a variant site. This may happen when multiple replicates are studied simultaneously and a variant caller detects a mutation at a specific site in one replicate and does not call that variant in the other replicate(s). One simple way to resolve this inconsistency would be to take the intersection of the variants called in all replicates. However, this may exclude some important, if borderline, variants, and in general ignores the confidence any one replicate gives us. In an extreme case, if a replicate failed to have data in a region, and thus no variants were called, that lack of data would effectively overrule the positive data in other replicates. Alternatively, one may seek to take advantage of advanced decision-theoretic approaches. In a different but still applicable context, Karimnezhad and Bickel [17] developed an empirical Bayesian approach that takes advantage of prior information from multiple

reference classes and leads to a unique decision in conflicting situations. This could be a potential future research problem.

Data Availability

All sequencing data is available from the Sequence Read Archive under project accession PRJNA614006. Human genome version hg19/GRCh37 is available from the UCSC Genome Browser website (genome.ucsc.edu).

Acknowledgements

The authors wish to acknowledge the technical support of the Ottawa Hospital Research Institute/University of Ottawa Bioinformatics Core Facility and StemCore Laboratories for their expert technical assistance. The sequencing data was generated by StemCore Laboratories at the Ottawa Hospital Research Institute. This Project is funded by the Government of Canada through Genome Canada and Ontario Genomics, and by the Government of Ontario, under Genome Canada grant GAPP 6450. This work was supported by NSERC Discovery Grant RGPIN-2019-06604 to TJP.

References

1. Wong, K.M., Hudson, T.J., McPherson, J.D.: Unraveling the genetics of cancer: genome sequencing and beyond. Annual review of genomics and human genetics 12, 407-430 (2011).
2. Morgensztern, D., Devarakonda, S., Mitsudomi, T., Maher, C., Govindan, R.: Mutational events in lung cancer: Present and developing technologies. In: IASLC Thoracic Oncology (Second Edition), pp. 95-103. Elsevier, (2018)
3. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R. (2009). The sequence alignment/map format and samtools. *Bioinformatics*, **25** (16), 2078-2079.
4. Koboldt, D. C., Zhang, Q., Larson, D. E., Shen, D., McLellan, M. D., Lin, L., Miller, C. A., Mardis, E. R., Ding, L., Wilson, R. K. (2012). Varscan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome research*, **22** (3), 568-576.
5. Koboldt, D.C., Chen, K., Wylie, T., Larson, D.E., McLellan, M.D., Mardis, E.R., Weinstock, G.M., Wilson, R.K., Ding, L.: Varscan: variant detection in massively parallel sequencing of individual and pooled samples. Bioinformatics 25(17), 2283-2285 (2009)
6. Cai, L., Yuan, W., Zhang, Z., He, L., Chou, K. C. (2016). In-depth comparison of somatic point mutation callers based on different tumor next-generation sequencing depth data. *Scientific reports*, **6**, 36540.

7. Lai, Z., Markovets, A., Ahdesmaki, M., Chapman, B., Hofmann, O., McEwen, R., Johnson, J., Dougherty, B., Barrett, J. C., Dry, J. R. (2016). Vardict: a novel and versatile variant caller for next-generation sequencing in cancer research. *Nucleic acids research*, **44**(11), 108-108.
8. Dunn, T., Berry, G., Emig-Agius, D., Jiang, Y., Iyer, A., Udar, N., Stromberg, M. (2016). Pisces: An accurate and versatile single sample somatic and germline variant caller. In: Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, pp. 595-595 (2017). ACM.
9. Karimnezhad, A., Palidwor, G. A., Thavorn, K., Stewart, D. J., Campbell, P. A., Lo, B., Perkins, T. J. (2020). Accuracy and reproducibility of somatic point mutation calling in clinical-type targeted sequencing data. *BMC medical genomics*, **13**(1), 1-14.
10. Xu, C. (2018). A review of somatic single nucleotide variant calling algorithms for next-generation sequencing data. Computational and structural biotechnology journal, 16, 15-24.
11. Zhao, Z., Wang, W. and Wei, Z. (2013). An empirical Bayes testing procedure for detecting variants in analysis of next generation sequencing data. *The Annals of Applied Statistics*, 2229-2248.
12. Pan, W., Lin, J., Le, C. T. (2003). A mixture model approach to detecting differentially expressed genes with microarray data. *Functional and Integrative Genomics* **3** (3), 117–124.
13. Efron, B., Tibshirani, R., Storey, J. D., Tusher, V. (2001). Empirical Bayes analysis of a microarray experiment. *Journal of the American statistical association* **96**(456), 1151-1160.
14. Efron, B. (2012). *Large-scale inference: empirical Bayes methods for estimation, testing, and prediction* (Vol. 1). Cambridge University Press.
15. Padilla, M., Bickel, D. R. (2012). Estimators of the local false discovery rate designed for small numbers of tests. *Statistical Applications in Genetics and Molecular Biology* **11**(5) Art. 4.
16. Yang, Y., Aghababazadeh, F. A., Bickel, D. R. (2013). Parametric estimation of the local false discovery rate for identifying genetic associations. *IEEE/ACM Trans Computational Biology and Bioinformatics* **10**:98-108.
17. Karimnezhad, A., Bickel, D. R. (2018). Incorporating prior knowledge about genetic variants into the analysis of genetic association data: An empirical Bayes approach. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, doi:10.1109/TCBB.20 18.2865420.
18. Karimnezhad, A. (2022). A Simple Yet Efficient Parametric Method of Local False Discovery Rate Estimation Designed for Genome-Wide Association Data Analysis. *Statistical Methods & Applications* **31**:159–180.
19. Eberle, M. A., Fritzilas, E., Krusche, P., Kallberg, M., Moore, B. L., Bekritsky, M. A., Kruglyak, S. (2017). A reference dataset of 5.4 million phased human variants validated by genetic inheritance from sequencing a three-generation 17-member pedigree. *Genome Research* **27** (1),

157-164.
20. Raczy, C., Petrovski, R., Saunders, C. T., Chorny, I., Kruglyak, S., Margulies, E. H., Chuang, H. Y., Kallberg, M., Kumar, S. A., Liao, A., et al. (2013). Isaac: ultra-fast whole-genome secondary analysis on illumina sequencing platforms. *Bioinformatics*, **29**(16), 2041-2043.