Bioinformatics Advance Access published July 13, 2016

Bioinformatics, 2016, 1–9
doi: 10.1093/bioinformatics/btw383
Advance Access Publication Date: 22 June 2016
Original Paper

Genome analysis

Vecuum: identification and filtration of false somatic variants caused by recombinant vector contamination

Junho Kim[1], Ju Heon Maeng[1], Jae Seok Lim[2], Hyeonju Son[1], Junehawk Lee[3], Jeong Ho Lee[2] and Sangwoo Kim[1,*]

[1]Severance Biomedical Science Institute, Brain Korea 21 PLUS Project for Medical Sciences, Yonsei University College of Medicine, Seoul 03722, South Korea, [2]Graduate School of Medical Science and Engineering, KAIST, Daejeon 34141, South Korea and [3]Department of Convergence Technology Research, Korea Institute of Science and Technology Information, Daejeon 34141, South Korea

*To whom correspondence should be addressed

Associate Editor: Bonnie Berger

Received on April 10, 2016; revised on May 28, 2016; accepted on June 14, 2016

Abstract

Motivation: Advances in sequencing technologies have remarkably lowered the detection limit of somatic variants to a low frequency. However, calling mutations at this range is still confounded by many factors including environmental contamination. Vector contamination is a continuously occurring issue and is especially problematic since vector inserts are hardly distinguishable from the sample sequences. Such inserts, which may harbor polymorphisms and engineered functional mutations, can result in calling false variants at corresponding sites. Numerous vector-screening methods have been developed, but none could handle contamination from inserts because they are focusing on vector backbone sequences alone.

Results: We developed a novel method—Vecuum—that identifies vector-originated reads and resultant false variants. Since vector inserts are generally constructed from intron-less cDNAs, Vecuum identifies vector-originated reads by inspecting the clipping patterns at exon junctions. False variant calls are further detected based on the biased distribution of mutant alleles to vector-originated reads. Tests on simulated and spike-in experimental data validated that Vecuum could detect 93% of vector contaminants and could remove up to 87% of variant-like false calls with 100% precision. Application to public sequence datasets demonstrated the utility of Vecuum in detecting false variants resulting from various types of external contamination.

Availability and Implementation: Java-based implementation of the method is available at http://vecuum.sourceforge.net/

Contact: swkim@yuhs.ac

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The detection of somatic variants from massively parallel sequencing data is a core analysis in genomic studies. Recent advances in sequence analysis have remarkably improved the accuracy of somatic variant detection to consider low-allelic fraction mutations (Cibulskis *et al.*, 2013; Kim *et al.*, 2013) and genetic subpopulations in a sample (Roth *et al.*, 2014). This improvement has enabled researchers to interrogate genetic causes that act in a minute circumstance in various diseases. For instance, studies on neurodevelopmental malformations identified disease-causing variants at a very low-allelic fraction (down to a few percent; Jamuar *et al.*, 2014; Lee *et al.*, 2012; Lim *et al.*, 2015; Poduri *et al.*, 2013), while several metagenomic studies revealed small subpopulations of microbial pathogens regarding carcinogenesis (Castellarin *et al.*, 2012; Kostic *et al.*, 2012; Salyakina and Tsinoremas, 2013). There is also a growing possibility for the clinical use of high-resolution next-generation sequencing (NGS) in diagnosing genetic diseases that are hardly detectable by conventional clinical testing (Shirley *et al.*, 2013; Wilson *et al.*, 2014).

However, extending the range of somatic mutation analysis accompanies increased vulnerability to false findings that result from numerous factors including machine errors, mapping ambiguity, imprecise call models and observational stochasticity (inevitable amplification or the reduction of alternative alleles in an observational process). Of these, external sample contamination is a significant risk that can generate critical erroneous calls that perplex the investigators even at an infinitesimal amount. For example, the cause of genetic anomalies observed in various genome-sequencing data has been identified to be an inclusion of unexpected cell line DNA (Cantalupo *et al.*, 2015) and viral contamination (Hue *et al.*, 2010; Kjartansdottir *et al.*, 2015; Naccache *et al.*, 2014; Xu *et al.*, 2013; Zhi *et al.*, 2014). Recent studies showed that microbial contamination also exists even in large-scale international projects (Laurence *et al.*, 2014; Strong *et al.*, 2014). A series of findings in experimentally well-controlled studies indicate that external contamination needs to be assessed by more systematic and computational approaches in quality control (Strong *et al.*, 2014).

External contamination by recombinant vector is a continuously occurring problem in sequencing experiments, which has been reported from traditional PCR-based sequencing to the NGS sequencing era (Borst *et al.*, 2004; Lopez-Rios *et al.*, 2004; Tang *et al.*, 2013, 2015; Tao *et al.*, 2015). Recombinant vectors are extraordinary problematic contaminants compared with others, because of the presence of recombinant inserts. Unlike, vector backbone sequences that are easily distinguishable in xenogeneic genomes, the sequences of recombinant inserts is hardly separable once mixed in sample DNA to form homogeneous short-read mapping in NGS sequencing; recombinant inserts are usually designed for transfection based on the genetic sequence of the target organism in order to express a transcript in target cells. Moreover, vector inserts are often engineered to harbor the intended mutations for *in vitro* or *in vivo* validation, thus generating variant-like alternative alleles at functionally important sites (Fig. 1A, yellow marks). Such false variants can result in serious pitfalls in interpreting the results of sequence analysis and assessing the occurrence of disease causing mutations. So far, numerous methods have been developed for the identification and filtration of vector-contaminated reads and are used as a quality control process in sequence analysis (Falgueras *et al.*, 2010; Li and Chou, 2004; Schmieder and Edwards, 2011; White *et al.*, 2008; https://sourceforge.net/projects/seqclean/). However, none could filter out such variant-containing reads generated from vector inserts, because all previous attempts have been confined to searching reads containing vector backbone sequences (Fig. 1A, blue reads). Therefore, a novel approach that guarantees the removal of the entire vector contaminant is required for more robust somatic mutation analysis in NGS.

Here, we introduce Vecuum that computationally estimates vector-originated reads including vector backbones and recombinant inserts, and identifies the resultant false variants caused by inherent contamination. Vecuum identifies the existence and sites of vector contamination by detecting vector backbone sequences in an ultra-high-speed screening step based on BWA-fastmap. Vecuum then attempts to capture short reads from recombinant inserts using

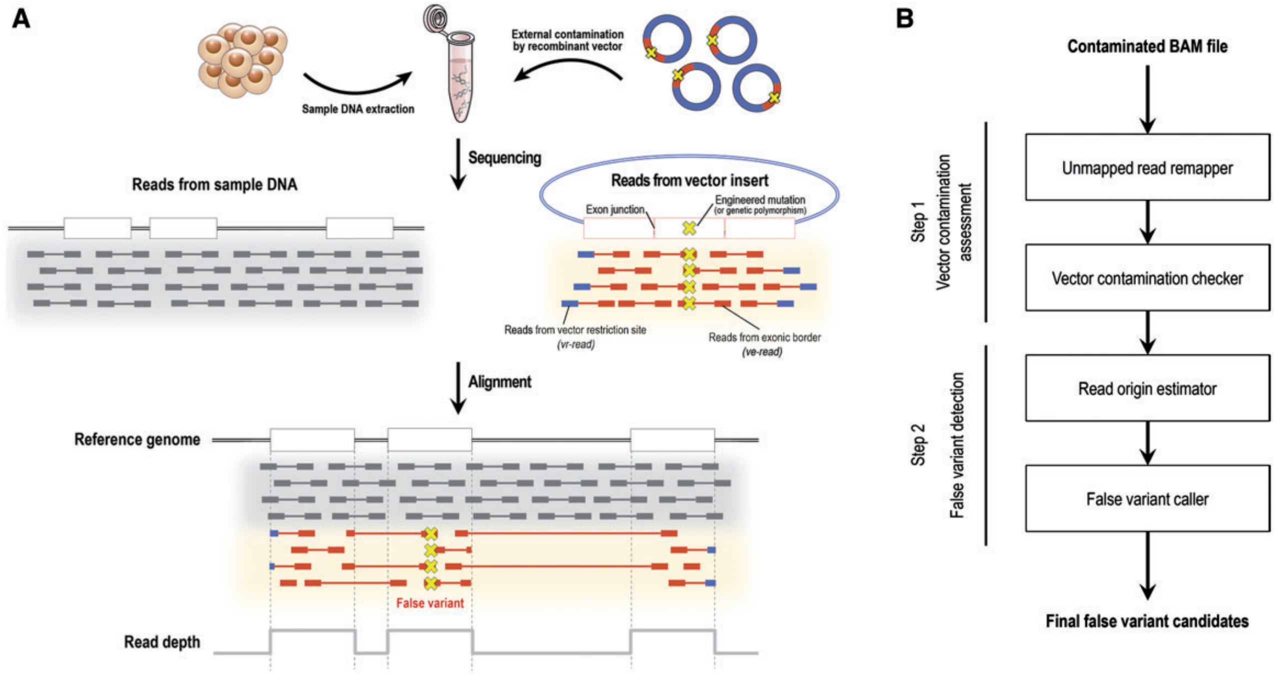
A
Sample DNA extraction
External contamination by recombinant vector
Sequencing
Reads from sample DNA
Reads from vector insert
Exon junction
Engineered mutation (or genetic polymorphism)
Reads from vector restriction site (vr-read)
Reads from exonic border (ve-read)
Alignment
Reference genome
False variant
Read depth
B
Contaminated BAM file
Step 1 Vector contamination assessment
Unmapped read remapper
Vector contamination checker
Step 2 False variant detection
Read origin estimator
False variant caller
Final false variant candidates

Fig. 1. Overall Vecuum workflow. (**A**) Sequence reads of a contaminated specimen consists of DNA originating from the sample (grey reads) and recombinant vector (blue and red reads). Due to ordinary mapping of reads from vector inserts, engineered mutations can be observed as low-frequent variants (yellow marks). Since recombinant inserts do not include intronic sequences in general, unique mapping patterns are generated by vector-originated reads at exon junctions (clipped and discordant reads), which provides a clue to identify the false variants. (**B**) The analysis consists of two major steps with four sub-processes. In the vector contamination assessment step, Vecuum checks for vector contamination and predicts insert regions. False variants are then examined within the predicted insert regions in the next step. The identified false variants with probability scores are provided as a result

intron-less mapping signatures (split or discordantly mapped reads at exon junctions with a full recovery in transcript mapping, Fig. 1A, red discordant reads); vector inserts for molecular cloning are generally constructed from intron-less processed cDNAs. Finally, Vecuum identifies false variants based on observation bias of mutant alleles in vector-originated reads using statistical tests. In a carefully designed simulation, we demonstrated that Vecuum not only outperforms previously reported methods, but also reduces the computation time in the vector identification problem. Vecuum also accurately identified the genomic regions of contamination and detected most of the false variants, which has not been attempted in the previous methods. These performances were further validated by real spike-in experimental data. By applying Vecuum to public sequence datasets, we revealed that various types of external contaminations can be detected including mammalian expression vectors, mouse genes in xenografts and prepped mRNA (cDNA) libraries. Based on these results, we expect that Vecuum will improve the reliability of low-frequent somatic variant calls in NGS data by providing a novel quality control method for external contamination.

2 Methods

2.1 Method outline

The overall workflow of Vecuum is shown in Figure 1. In general, vector-originated reads are hardly distinguishable from those of sample origin, especially in intra-species contamination. Nevertheless, some portion of mapped reads can be considered to be vector-originated based on two kinds of evidences. First, reads generated from vector restriction sites are likely to be clipped in alignment due to the inclusion of vector backbone sequences (Fig. 1A, blue reads). Second, reads sequenced across the exonic borders within a vector insert can be also clipped at the exon junction sites (with respect to the reference genome) and/or form a discordant paired-end mapping (in terms of distance between two ends) due to the absence of intronic sequences (Fig. 1A, red clipped/discordant reads). Both types of reads and their mates are defined as *plausible vector reads* and play a key role in the overall process in Vecuum.

Vecuum takes the alignment of paired-end sequencing (BAM) from potentially contaminated samples as the input. The analysis consists of two consecutive steps: (i) vector contamination assessment and (ii) false variant detection. In vector contamination assessment, Vecuum collects entire clipped reads of input BAM with the remapping process (see Supplementary Methods) and attempts to detect reads that are originated from vector restriction sites (*vr-reads*) by examining the inclusion of vector backbone sequences. The unclipped (aligned) portions of *vr-reads* are used to locate the genomic position of the vector insert. If sufficient numbers of *vr-reads* are secured at potential insert sites, Vecuum confirms vector contamination and moves on to false variant detection. Once vector insert positions are determined, reads from vector exonic border (*ve-reads*) are extracted at the insert region based on read clipping and transcript mapping (see below for detailed methods). The *vr-reads* and *ve-reads* finally comprise plausible vector reads. Utilizing them, Vecuum further classifies all the mapped reads in the insert region as vector and sample originated according to their mapping patterns. Finally, Vecuum checks if mismatches are significantly biased for vector origin to identify vector driven false variants.

2.2 Estimating vector insert regions

Vecuum firstly examines the input data to determine the existence of vector contamination and the corresponding genomic locations. To do this, Vecuum collects all the clipped reads and queries their full sequences to a custom vector database to find reads containing vector backbone sequences. The custom database was constructed using 1629 vector sequences from UniVec (http://www.ncbi.nlm.nih.gov/tools/vecscreen/univec/) and 48 089 sequences from AddGene (http://www.addgene.org).

In keeping with the continuous demand, numerous methods have been developed and applied to vector sequence identification from low to high throughput sequencing data (Falgueras *et al.*, 2010; Li and Chou, 2004; Schmieder and Edwards, 2011; White *et al.*, 2008). Most of the methods implement the sequence query (to vector sequences) module based on nucleotide BLAST (BLASTN), which is a major time-consuming step. Instead, we used the BWA-fastmap program (Li, 2012), which is specifically designed to conduct ultra-fast short-read alignment without allowing gaps. Conceptually, the custom vector database is used as the reference genome (with numerous contigs), and matches are determined by aligning the query sequences. After the query is complete, reads from vector restriction sites (*vr-reads*) are classified from the clipped read set based on the following criteria: (i) clipped length ≥ 20 bp, (ii) the length of vector-matched subsequence ≥ 20 bp, (iii) mapping quality ≥ 30, (iv) clipped subsequence is matched with vector sequence and (v) the mate read is not mapped outside the clipped subsequence. The last condition is a mate constraint to avoid misclassification of sample reads that are partially homologous to vector sequences. The default values of matching parameters are derived from previous vector screening methods, and all thresholds are adjustable by user input.

Based on the classified *vr-reads*, genomic positions of vector inserts are estimated. A clipping position supported by more than three *vr-reads* is initially considered a candidate vector insert site. The direction of read clipping determines whether the candidate site is the 5′ or the 3′ end of the insert (see Supplementary Fig. S1). If a pair of 5′ and 3′ candidate sites is detected in a gene, the interval region is the place where the vector inserts are located and false variants may exist. Sites with only one insert end (5′ or 3′) are regarded as false signatures generated by sequence homology.

Once a vector insert region is inferred, Vecuum conducts another round of *vr-read* search at the 5′ and 3′ insert ends. Here, *vr-reads* that did not pass the filters previously (e.g. insufficient clipped length) are rescued if their clipped position matches the insert sites (see Supplementary Fig. S2) while mapping quality and mate read constraints are preserved to prevent the misclassification of sample reads.

2.3 Separation of vector and sample-originated reads

Each mapped read within vector-insert regions is separated into three classes depending on the inferred source DNA: (i) a read from sample DNA, (ii) a read from the vector insert (*ve-read*) and (iii) a read from an unknown source. Reads that are mapped to intronic regions with >5 bp are considered to be sample originated (Fig. 2A, grey reads) because vector-insert sequences are assumed to be spliced and do not contain intron sequences. We use two different signatures to identify *ve-reads* (Fig. 2A, red reads). First, we align all the clipped reads at exon-junctions to the reference transcriptome. We regard a clipped read as a *ve-read* if the entire sequence is mapped without clipping. Likewise, discordant read pairs that are recovered in transcriptome mapping within a range of three standard deviations from the average insert size are also considered as *ve-reads*. We additionally check whether each corresponding exon of the discordant read pairs possesses clipped *ve-reads* at both ends, to

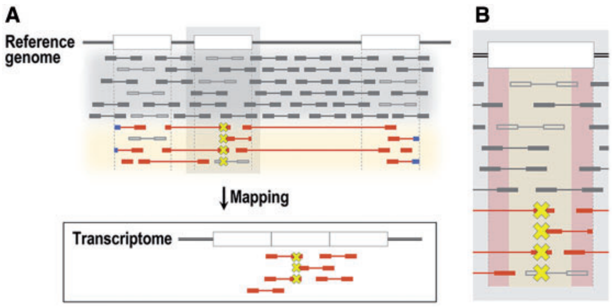

Fig. 2. Estimation of read origin and false variant detection. (**A**) All reads within the insert region are classified as vector- (blue and red reads) and sample- originated reads (gray reads) according to their mapping patterns at exon junctions. Clipped or discordant reads at exon junctions are regarded as *ve-reads* if the mapping pattern is recovered as normal with reference transcriptome mapping. Read pairs entirely mapped within an exon are marked as unknown (uncolored reads). (**B**) False variants are detected based on biased distribution of mutant alleles to vector-originated reads, as evaluated with one-tailed Fisher's exact test. The detected mutations are categorized into primary and secondary variants according to their location, which decides whether both types (clipped and discordant) of anomalous reads can support the variant or not

distinguish them from structural variant-derived discordant pairs. Since any source (sample or vector) can generate read pairs that are entirely mapped within an exon (Fig. 2A, uncolored reads), classification is not available *per se*. These reads are marked as those of unknown origin.

2.4 Detecting vector-induced false variants

For each exon within the identified vector inserts, we first screen the potential variant positions by counting the number of reads with mismatches from non-reference nucleotides (B alleles). Since the sequencing error rate of NGS is expected to be <0.01, all genomic positions with more than B allele frequency (BAF) of 0.01 with base quality ≥20 are initially considered; these positions are either true mutations or vector-induced false variants. We hypothesize that the distribution of B alleles would be significantly biased to the reads of vector origin (identified from the previous section) if the variant is falsely induced. The significance (*P*-value) of bias is measured using one-tailed Fisher's exact test. Simply, we test whether a candidate position possesses variant (whose BAF is above sequencing error rate) *only for vector reads*. We define a variant is falsely induced if all of the following conditions are satisfied at the corresponding genomic position: (i) the number of plausible vector reads (*vr-read* and *ve-read*) with the B allele >3 and the BAF for vector reads >0.01, (ii) the number of sample-originated reads with the B allele ≤3 or the BAF for sample reads ≤0.01, (iii) the number of sample-originated reads with the reference allele ≥5 and (iv) *P*-value of one-tailed Fisher's exact test <0.01 or all sample-originated reads containing the reference allele. Thresholds for condition (i) and (ii) are set to exclude false calls from sequencing errors (Lim *et al.*, 2015). The minimum cut-off values for the number of sample-originated reads and the plausible vector reads are determined to satisfy the significance level < 0.01 in Fisher's exact test for the minimal case. Default values of parameters were tested under multiple different criteria (see Supplementary Methods). Reads with the B allele of unknown origin are initially excluded from the significance test, but are later regarded as plausible vector reads if the position is called as a false variant.

Vecuum calls two different types of false variants according to the supporting evidence. Variants within one read length from exon junctions are covered by both clipped reads and discordant reads (Fig. 2B, red shaded area), while variants that are located more than one read length away from both exon junctions can be only covered by discordant reads (Fig. 2B, yellow shaded area). Since the evidence level is higher in the former form, we regard with the two evidences as primary and secondary, respectively. Finally, Vecuum reports the list of genomic positions of vector contamination and the predicted false variants annotated with the evidence level. Cleaned alignment (BAM) is provided optionally by filtering out all the vector-originated reads from the input data.

2.5 Data preparation for validation

2.5.1 Simulated data

To test the performance of Vecuum, we generated simulated datasets that mimic vector contamination. We first constructed artificial recombinant vectors using the CLC Genomics Workbench (http://www.clcbio.com) following its '*in silico* cloning workflow' (see Supplementary Methods for details). In total, 19 mammalian expression vector backbones were *recombinated* with 51 cancer genes to build 969 initial recombinant vectors. Of these, seven were excluded due to the absence of appropriate restriction sites, leaving 962 vectors for testing. Each recombinant vector was then simulated to contain one point mutation at a random position within its gene region (vector insert). The mutation simulation was repeated two times independently for each recombinant vector, to generate 1924 artificial recombinant vectors with a unique mutation.

Paired-end whole-exome sequencing (WES) data of a blood sample (∼250× coverage) were served as the sample DNA. For each recombinant vector sequence, simulated paired-end reads were generated with the same read length and fragment size as in the sample WES data (101 × 2 bp read length and 170 ± 60 bp fragment size). GemSim (McElroy *et al.*, 2012) was used to generate simulated reads for ∼1000× coverage (enough for down-sampling) with the Illumina paired-end error model with a tagging corresponding vector ID at every read name for further identification. For each simulated read set from one recombinant vector, we randomly extracted a subset of reads 10 times, with different down-sampling rates. The 10 read subsets were then mixed with the normal WES data independently to simulate contamination at different levels, thus building 19 240 BAM files of artificially contaminated samples. We further filtered out 7694, where less than three reads with B alleles were mapped at the mutation sites; the contamination level of these samples is limited in NGS observation and would be harmless in variant calling. Finally, a total of 11 546 artificially contaminated data were prepared for performance evaluation.

2.5.2 Real spike-in data

Spike-in data were prepared by intentionally mixing vectors into a control sample. Ten recombinant plasmid vectors containing unique mutations were added to the gDNA of a normal blood sample (Supplementary Table S1). To mimic low-level contamination, recombinant vectors were diluted to 1:30 from their original concentration in stock, to attain the ratio of 1:200 with respect to the blood gDNA. Contaminated DNA was sequenced using Illumina HiSeq 2000 following the manufacturer's standard protocols. The detailed protocol is available in Supplementary Methods.

2.5.3 Public sequence data

Since most previous publications of vector contamination were unusual case reports and most public datasets have undergone quality control processes, the prevalence of vector contamination in public

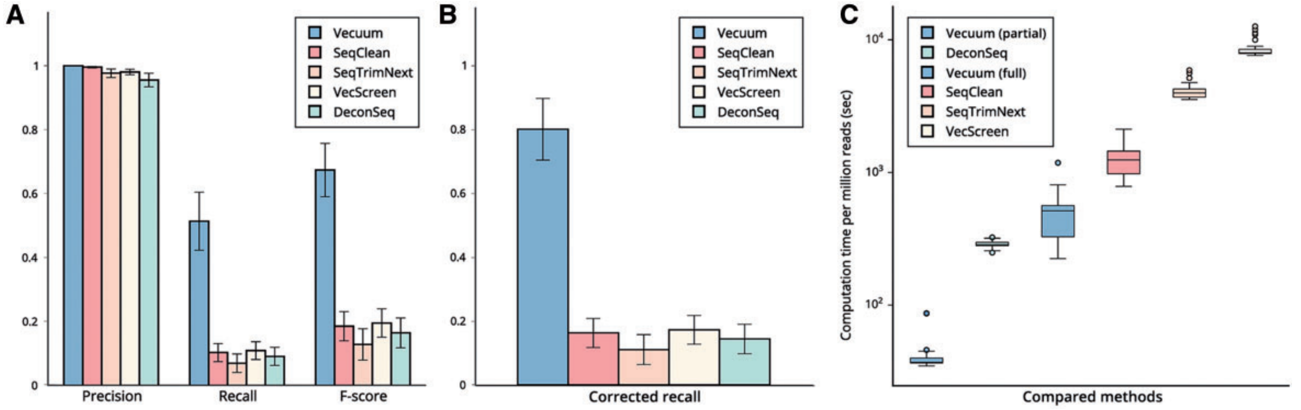

Fig. 3. Performance comparison for identification of vector-originated reads. (**A**) Precision, recall and *F*-score of detected vector-originated reads were measured for each method. Vecuum outperformed other tools in recall according to the detection of *ve-reads*. (**B**) Practical recalls were recalculated by excluding unknown reads, which are theoretically inseparable. Vecuum successfully called most of the detectable reads, but other tools barely identified them. (**C**) Computation time per million reads taken by each method to detect vector-originated reads. A total of 2000 datasets from entire simulated sets were randomly selected and tested. To compare the performance for the same task, the time taken by a subprocess of Vecuum to detect the reads containing vector sequence is additionally measured and compared [annotated as Vecuum (partial)]

datasets is difficult to expect. Therefore, we focused on finding real cases that demonstrate the *proof-of-concept* for our suggested problem. Deep WES datasets from Lim *et al.* (SRP055482) were assessed by Vecuum with the authors' permission, which were suspected of the inclusion of false variants due to unknown reasons. We also tried to find cases with false variants from publicly released datasets in the Sequence Read Archive (SRA). We randomly downloaded public paired-end WES/WGS datasets and tested them with Vecuum; several interesting cases were obtained from mouse xenograft studies (SRP056402 and SRP060313). One in-house WES dataset suspected of cDNA contamination from an anonymous individual was additionally used for further analysis of false variants from other types of contamination.

3 Results

3.1 Performance tests with simulated data

We tested the performance of Vecuum in identifying vector contamination and falsely induced mutations using 11 546 *in silico* simulation data. The simulation was carefully designed to represent realistic structures of expression vectors that contain engineered mutations (see Supplementary Methods for detailed design). Accuracy and computation time were measured and were compared with conventional tools with further analyses regarding several factors that affect the final performance.

3.1.1 Performance test for vector contamination assessment

We first evaluated the accuracy of vector-origin read detection (Fig. 3). As the true origins of reads have been annotated in the simulated data (hidden in the test), we could calculate the precision (called # of true vector reads/total # of called vector reads), recall (called # of true vector reads/total # of true vector reads) and *F*-score [2 × precision × recall/(precision + recall)] of Vecuum and other tools used for comparison including VecScreen (http://www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html), SeqClean (https://sourceforge.net/projects/seqclean/), SeqTrimNext (Falgueras *et al.*, 2010), and DeconSeq (Schmieder and Edwards, 2011).

Vecuum showed a near perfect precision (0.999), which means that almost all reads classified as vector-origin were truly of vector-origin. Other tools include a small amount of the false detection where some sample-origin reads are misclassified to be of vector-origin (precision 0.955–0.995). In general, we noted that tools based on BLASTN (VecScreen, SeqClean and SeqTrimNext) have better precision than the one based on BWA (DeconSeq). We expect that BLASTN-based methods could be further optimized for vector sequence identification using empirically fine-tuned parameters. Nevertheless, despite being built on BWA, Vecuum could prevent misclassification by its unique filters such as read-clipping orientation and mate-read mapping (see Methods).

The performance gap is more remarkable between Vecuum and other tools in terms of recall (Fig. 3A). Conventional tools could identify ∼10% of the mapped vector-originated reads (recall = 0.069–0.109) by considering only vector restriction sites. Vecuum, on the contrary, successfully detected more than half of the true vector reads (recall = 0.513). Note that, this seemingly imperfect performance is actually due to read pairs of *unknown-origin* (see Methods) whose both ends are entirely mapped within a single exon (Fig. 2A, uncolored reads). These reads are theoretically inseparable unless additional sequence variations are included. In other words, these inseparable, perfectly mapped read pairs without any mismatches are not influential in sequence analysis even if retained in the data. When we excluded those reads, the practical recall was increased to 0.802 for Vecuum while that of other tools remained far below 0.2 (Fig. 3B).

We next examined the computation time needed for analysis (Fig. 3C). We randomly selected 2000 samples out of 11 546 simulated data to measure the average run time per million mapped reads. Although Vecuum performs many more additional analyses than other tools including unmapped read search and remapping, clipped read extraction, fastq-BAM conversion and false variant calling, overall computation time was much less than that required by BLASTN-based tools (512.8 versus 1244.6–8454.3, 2.4- to 16.5-fold decrease) and was comparable with that of DeconSeq. The key to the improved computational efficiency is the use of BWA-fastmap in vector sequence matching (see Methods). In an additional analysis using a part of Vecuum that conducts vector sequence search only, the overall time reduction was 30- to 200-fold [Fig. 3C, denoted as Vecuum (partial)].

Finally, we assessed the performance in estimating the genomic positions of contaminated regions. Since no previous methods have provided such information, accuracy was measured only for

Vecuum. We considered that an estimated region is correct if its genomic position fully covered the corresponding true coding sequence. Out of 11 546 vector inserts, Vecuum correctly estimated the genomic positions of 10 748 inserts without any wrong calls (precision = 1.0, recall = 0.931). We found that most of the false-negatives were generated in samples in which insufficient numbers of *vr-reads* were mixed in the data-generation step (the mixture rate was randomly assigned for each sample, see Methods). With the above three *vr-reads* for each restriction site, Vecuum could identify the accurate genomic position in >99% samples.

3.1.2 Performance test for false variant detection

Before the evaluation of false variant detection for simulated datasets, we first checked the effect of vector-insert inclusion on somatic mutation calling. Of 11 546 mutation sites, each of which is an artificially switched sequence contained in an individual simulation set (see Methods), we found that 11 502 (~99.6%) were initially called as somatic mutations by MuTect (Cibulskis *et al.*, 2013) including 9665 (~83.7%) that passed all the filters. The average variant allele frequency of the 9665 mutations was 0.443 ranging from 0.013 to 0.796, confirming that vector insert contamination leads to false discovery of somatic mutations even at a low frequency (Fig. 4A). Other conventional callers including VarScan2 (Koboldt *et al.*, 2012) and Strelka (Saunders *et al.*, 2012) also showed similar results on somatic mutation calling; 10 196 (~88.3%) and 8831 (~76.5%) false variants were survived as final somatic candidates from VarScan2 and Strelka, respectively.

Next, we ran Vecuum on 11 546 simulated datasets to check its performance in detecting the false calls. In Vecuum, false variant detection is basically attempted for regions that are estimated as vector contaminated in the previous step. Thus, 798 mutations caused by unidentified vector contaminants (see the previous section) were excluded and were finally regarded as failures in the evaluation. The results were compared with the intended mutation sites to measure precision, recall, and *F*-score. Of 11 546 false mutations, 10 150 mutations were successfully detected by Vecuum (recall = 0.879) with only 430 calls that were not included in the intended mutation list (precision = 0.959; Fig. 4B, blue bars), which were finally confirmed as true vector-originated mutations upon further inspection (see below). The receiver operation characteristic (ROC) curve was drawn based on the *P*-value of the calls (Fig. 4C) to confirm a high classification power with the area under curve (AUC) of 0.911.

We further assessed that the relationship between detection performance and BAF at the mutation sites. Mutations were divided into eight bins based on BAF values (from 0.0–0.1 to 0.7–0.8, the maximum BAF was < 0.8), and performance was measured separately for each bin (Fig. 4D, blue bars). Precision was almost perfect in mutations with BAF > 0.1 (precision = 0.983–1), and most unintended calls were observed at BAF < 0.1 (precision = 0.770). There also was a weak positive correlation between BAF and recall, but Vecuum showed a reliable recall rate at BAF >0.1 (recall = 0.868–0.930) with the relatively low recall at BAF < 0.1 (recall = 0.725) due to failed estimation of contaminated regions.

We checked the source of the 430 mutations that were called by Vecuum but were never generated artificially in the preparation of simulation data. Interestingly, all of 430 mutations were originated from vectors. From manual inspection, we found two additional mechanisms (other than engineered mutation) that can generate false variants by vector contamination (Supplementary Fig. S3). One involves sequence polymorphism between the sample gene and the cDNA used to build the vector insert. These polymorphisms (most of them are SNPs) are intrinsically the same as engineered mutations with the exception of intentionality. The other mechanism is an

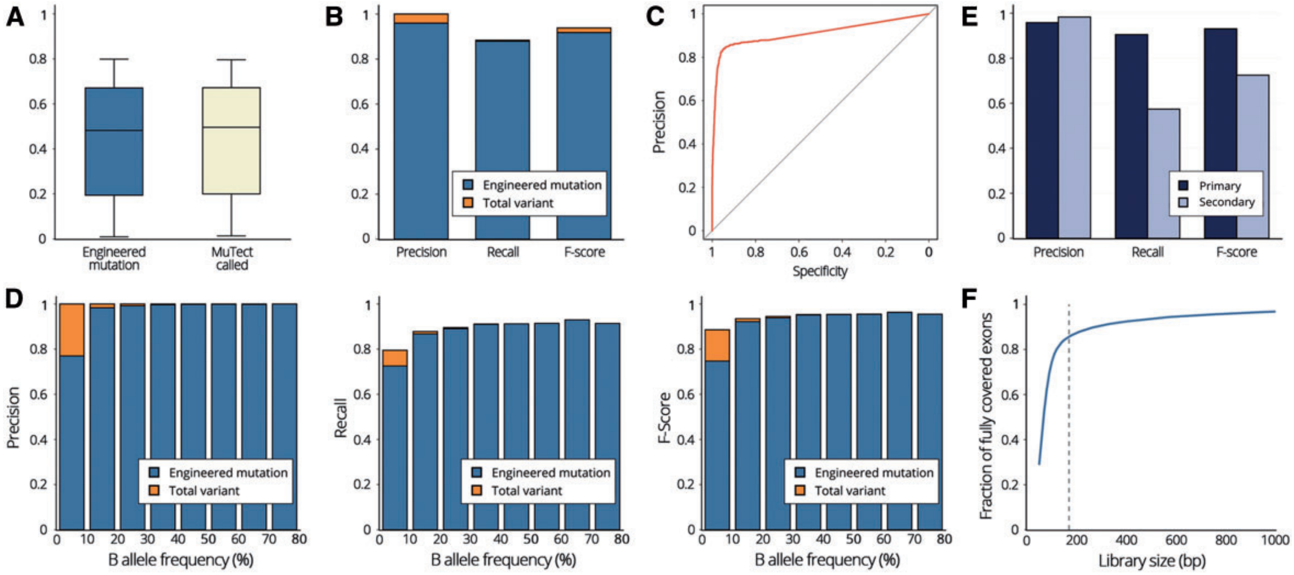

Fig. 4. Performance of false variant detection for simulated datasets. (**A**) Comparison of BAF distribution between 11 546 simulated mutations (vector-insert included) and 9665 MuTect calls. Most of the simulated mutations were called by MuTect regardless of BAF. (**B**) Precision, recall and *F*-score of the detected false variants by Vecuum (blue bars). With manual curation, all false-positive calls (calls that were not intentionally inserted) were confirmed as vector-induced mutations (see Supplementary Figure S3 for details). Considering additional vector-induced mutations as true answers, Vecuum achieved perfect precision with a high recall value (orange bars). (**C**) ROC curve drawn by *P*-value of Vecuum calls (AUC = 0.911). (**D**) Precision, recall and *F*-scores of the detected false variants for each BAF range. Performances with the curated answer set are additionally depicted (orange bars). Due to the failed estimation of contaminated regions, relatively low recall was observed for BAF <0.1. (**E**) Performance comparison between primary and secondary variants. Most false negatives were produced by secondary variants. (**F**) The rate of fully covered exons according to the given library size. About 86% of human exons can be fully covered by the simulated library size of 170 bp (dotted line). BAF, B allele frequency; ROC, receiver operating characteristic; AUC, area under the curve

alignment error in determining the exact clipping point (Supplementary Fig. S3, right). Vector reads generated across splice or restriction sites are usually clipped during alignment. When the number of base pairs to be clipped is small (e.g. 1), aligners occasionally prefer a mismatch rather than clipping; mismatches are also more tolerated at the end of reads. With the majority of vector reads that are correctly clipped, mismatches form a somatic-variant-like signature. These two extra mechanisms explain all 430 unintended calls, confirming that there is not a single wrong call for the all BAF ranges (Fig. 4B and D, orange bars).

We then analyzed the cause of false negatives. Most false negatives were produced by secondary variants that are located more than one read length away from the exon junctions (Fig. 4E). Out of 1396 false negatives, 798 mutations were located in the undetected regions (see the previous section). Among the remaining 598 mutations, 384 were secondary variants and most of them turned out to be unreachable mutations due to the large size of the variant containing exons. Since Vecuum detects false variants based on predicted vector reads, there is a limit of detection for false variants according to the library size (Supplementary Fig. S4). The called results for secondary variants might seem to lose too much sensitivity; however, only 52 of the 384 secondary variants were unique and the rest were identical variants with differing contamination levels. In addition, the total number of secondary variants was considerably small compared with the number of primary variants (901 and 10 645, respectively) because most human exons can be fully covered by typical read and library sizes. We could calculate the rate of fully covered exons by the given library size, and the result showed that 86% of the exons were fully covered by the simulated size (Fig. 4F). Considering the smaller size of the simulated library compared with typical sequencing design and the size limit of conventional cloning, we can expect to support reasonable performance of false variant detection for most vector contamination cases.

All taken together, we demonstrated that Vecuum successfully examines vector contamination and accurately estimates genomic positions of contaminated regions. Vecuum also showed superior performances for the detection of false variants, in addition to unrecognized mutations including genetic polymorphisms and clipping errors.

3.2 Performance test with experimental data

To test Vecuum with real sequencing data, we prepared an intended vector-contaminated DNA sample for whole-exome sequencing (see Material and Methods section for details). Ten recombinant vectors with each specific mutation were prepared and added to the gDNA of a normal blood sample (Supplementary Table S1). All the 10 sites were called as somatic mutations by conventional methods without Vecuum processing. Sequencing design including library size, read length and coverage were set as identical to the simulated datasets to control other sources of variables.

We first evaluated the performance for estimating genomic positions of contaminated regions. Vecuum reported six estimated insert regions, located in five genes (Supplementary Table S2). All inserted genes were correctly estimated by Vecuum without any false-positive calls, demonstrating the reliable performance for vector contamination assessment. Due to microhomology of exon 33 with the vector sequence, estimation of MTOR was split into two parts at the end of exon 34 (Supplementary Fig. S5).

Finally, we evaluated the accuracy of false variant detection for experimental data (Supplementary Table S3). Simply, Vecuum successfully detected all the induced mutations without a wrong call. Moreover, one extra site was detected by Vecuum, which turned out to be a false mutation call caused by a read-clipping artifact (Supplementary Fig. S6). As in the simulated data, we again found that there are various causes of false calling other than engineered mutations, which can be resolved by Vecuum. Another interesting finding is the large variance of BAF among mutant alleles (BAF = 0.07–1.0, stdev = 0.35), while the supplied amount of each plasmid was controlled to be equal (~5 ng per each plasmid, see Supplementary Methods). We assume that the large variance might result from different efficiencies of hybridization. Thus, the presented level of contamination in sequence data is highly unpredictable through sample preparation and sequencing procedures. Considering that extremely high level of BAF (up to 28 000× coverage with the variant allele) can be achieved with only 5 ng of vector DNA, false variants of lower allele frequency can be caused by an extremely small amount of vector contamination.

3.3 Application to public datasets

We applied Vecuum to our recently published deep WES datasets, which were sequenced from focal cortical dysplasia (FCD) patients (Lim *et al.*, 2015). FCD is a neurodevelopmental disorder with cortical malformation and intractable epilepsy, which is caused by somatic mutations (Crino, 2009; Lim *et al.*, 2015). In the previous study, we successfully discovered several brain somatic mutations with low-allelic frequency using in-depth bioinformatics analysis followed by *in vitro* and *in vivo* functional validation. In the meantime, we also found that several false-positive mutations mimicking the functional somatic mutations co-exist in part of the samples by excluding them in deep-targeted amplicon sequencing using primer sets covering both intronic and exonic regions. Although such false-positive mutations were strictly excluded from the final report through our QC procedures, we thought that these artifacts might arise from the genomic DNA preparation step, especially associated with vectors, because the mammalian expressing vectors containing false-positive mutations have been constructed for *in vitro* functional assays.

To test this possibility, we examined the eight sequence dataset of WES by Vecuum (SRP055482). We found that out of the eight deep WES data, three were detected as vector-contaminated (SRR1819827, SRR1819829 and SRR1819831). The estimated positions of vector inserts were identical for all three samples at MTOR (chr1: 11 167 437–11 319 466). SRR1819827 and SRR1819829 showed two identical false variants, caused by two different recombinant vectors (Fig. 5A, Supplementary Table S4 and Supplementary Fig. S7). In SRR1819831, one of two variants was missed due to the absence of a variant containing soft-clipped reads. Two additional false variants were called from SRR1819831 at two SNP sites, detected according to different genotypes from other samples (Supplementary Fig. S8). The BAF of false variants ranged from 0. 01 to 0.25, which are highly similar to those of previously reported true brain-specific mutations from various neurological disease studies (Jamuar *et al.*, 2014; Lee *et al.*, 2012; Lim *et al.*, 2015; Poduri *et al.*, 2013), thus increasing the ambiguity in separation. Compared with the high BAF caused by 5 ng of vector DNA in the experimental dataset, the low-allelic frequency (~1%) might be a result of extremely low amounts of vector DNA (e.g. aerosols generated from laboratory operations such as pipetting). These results, on the other hand, show the importance of validation sequencing with biological/sequencing duplicate especially on different platforms; in the previous study, all vector-induced mutations have been called as somatic

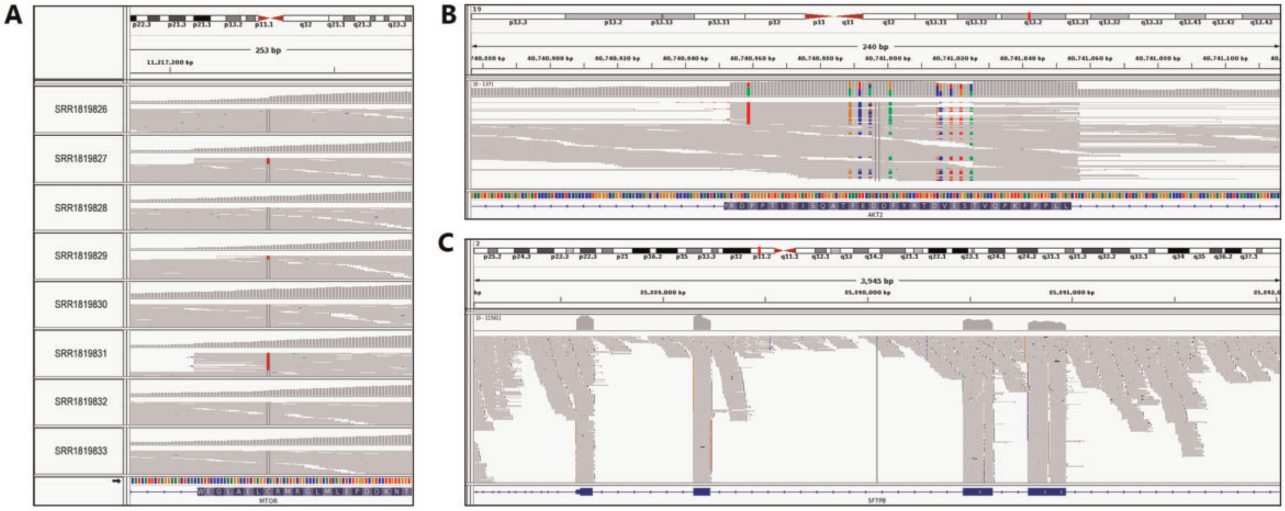

Fig. 5. Detected false variants from public datasets. (**A**) False variants induced from vector contamination. Three of the eight samples (SRR1819827, SRR1819829 and SRR1819831) were detected as vector-contaminated and showed variant alleles with low frequency at the predicted sites by Vecuum. (**B**) Xenograft-derived false variants from the prostate cancer study. MuTect and VarScan2 called all 10 variants in the figure as somatic, whereas Vecuum called them as falsely induced. Alignment with combined reference (human + mouse) removed all those variants, which supports their origin as mouse genome. (**C**) False variants induced from mRNA (cDNA) libraries. Abnormally excessive, discontinuous read-depth only for exons of specific genes supports the presence of cDNA contamination

mutations in hybrid-capture sequencing, but none was discovered in amplicon-based sequencing.

We applied Vecuum to other public sequence data to find more possible sources of contamination. For this purpose, Vecuum can be run for a false variant search without estimation of vector contamination and of contaminated locations (can be selected in option). In a prostate cancer study (SRP060313), we discovered numerous numbers of false variants in 10 out of the 13 samples, ranging from 651 to 8439. We found that the 10 samples with false variants were xenograft-derived, while other three were from human tissues. To check whether the called false variants were derived from cultured mouse cells, reads containing false variants were remapped to the mouse reference genome and perfect matches were confirmed for most of them (Fig. 5B and Supplementary Fig. S9). Since exonic sequences are far more highly conserved than introns between humans and mice, a part of sequence reads generated from mouse exon boundaries were clipped at the exon junctions in the alignment against the human reference sequence, ultimately leading to the generation of false variants. Out of the 413 genes with at least 1 false variant call, 393 had mouse ortholog genes annotated by the Metaphor database (van der Veen *et al.*, 2014). All false variants disappeared when a combined reference genome (human + mouse) was used for mapping, confirming its necessity (Tso *et al.*, 2014). Likewise, we identified a total of 5149 false mutations from 42 samples in a different xenograft-derived dataset (SRP056402) using the same mechanism. Based on the result, we assume that more public or laboratory level data may have a similar risk and further investigation is suggested.

Lastly, we applied Vecuum to a WES data from an anonymous individual, suspected of contamination by another unknown sample. This was first noted by a data-submitter based on an abnormally large number of somatic mutation calls, enriched in lung specific genes (i.e. surfactant genes), where the original tissue was not obtained from lung. We hypothesized potential contamination of prepped mRNA libraries (cDNA) based on read clipping patterns at exon junctions (Fig. 5C and Supplementary Fig. S10). Since cDNAs do not contain introns, like vector inserts, Vecuum could be applied instantly for a false variant search. We found 2204 false variants from the sample, of which 1626 (~74%) were annotated by dbSNP and the 1000 Genome database. Therefore, we could reconstruct the event using Vecuum that there was a contamination of cDNA from another individual, probably during sample preparation.

In conclusion, a series of applications reconfirms the variety of possible external contaminants (e.g. vector, xenograft genome and cDNA), the variety of mechanisms causing false variants (e.g. engineered mutations and polymorphisms in vector insert and read clipping artifacts) and the utility of Vecuum in handling and recovering the compromised samples.

4 Discussion

In this study, we developed a novel computational method, Vecuum, for the detection of false variants caused by vector contamination, which has not been explored by previously reported methods. We predicted vector-originated reads based on unique mapping patterns at exon junctions, and identified the false variants based on skewness of mutant alleles to predicted vector reads. Validation with simulated and experimental contamination datasets not only showed the outperformance of Vecuum against conventional assessment of vector contamination, but also showed reliable performance for false variant detection.

Occurrence of false somatic calls from external contamination has been occasionally reported. However, researchers often experience that many of the initial mutations from somatic variant analysis are not reproduced in confirmation steps such as Sanger sequencing, targeted deep sequencing and/or mass spectrometry-based panels (Pearce *et al.*, 2009) without a prominent feature of failure, and therefore remain unreported. Such publication bias suggests a good chance that contamination may exist more frequently than shown. Moreover, high-depth sequencing is necessary to identify small amounts of contamination. The rapid decline of sequencing costs and popularization of low-frequent somatic mutation analysis will call more attention to contamination issues, where Vecuum is a good fall-back option.

Application to public datasets showed the utility of Vecuum for various sources of contamination. In addition to external contamination that we demonstrated here, Vecuum even detected false variants caused by inherent pseudogenes. All types of genomic data generated by hybrid capture-based sequencing would be applicable by Vecuum including WES, WGS and ChIP-Seq data. To our knowledge, Vecuum is the first tool that performs an overall inspection of false variants. Judging from the perfect precision of the false variant detection achieved upon validations, the false variants identified by Vecuum can be considered as artifacts with high confidence regardless of their origin, which makes it applicable for robust quality control of somatic variant analyses.

Acknowledgements

The authors would like to thank Dong-Su Jang, MFA (Medical Illustrator, Medical Research Support Section, Yonsei University College of Medicine, Seoul, Korea) for his help with illustrations.

Funding

This research was supported by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number: HI14C1324).

Conflict of Interest: none declared.

References

Borst,A. *et al.* (2004) False-positive results and contamination in nucleic acid amplification assays: suggestions for a prevent and destroy strategy. *Eur. J. Clin. Microbiol. Infect. Dis.*, **23**, 289–299.

Cantalupo,P.G. *et al.* (2015) HeLa nucleic acid contamination in the cancer genome atlas leads to the misidentification of human papillomavirus 18. *J. Virol.*, **89**, 4051–4057.

Castellarin,M. *et al.* (2012) Fusobacterium nucleatum infection is prevalent in human colorectal carcinoma. *Genome Res.*, **22**, 299–306.

Cibulskis,K. *et al.* (2013) Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat. Biotechnol.*, **31**, 213–219.

Crino,P.B. (2009) Focal brain malformations: seizures, signaling, sequencing. *Epilepsia*, **50(Suppl 9)**, 3–8.

Falgueras,J. *et al.* (2010) SeqTrim: a high-throughput pipeline for preprocessing any type of sequence read. *BMC Bioinform.*, **11**, 38.

Hue,S. *et al.* (2010) Disease-associated XMRV sequences are consistent with laboratory contamination. *Retrovirology*, **7**, 111.

Jamuar,S.S. *et al.* (2014) Somatic mutations in cerebral cortical malformations. *N. Engl. J. Med.*, **371**, 733–743.

Kim,S. *et al.* (2013) Virmid: accurate detection of somatic mutations with sample impurity inference. *Genome Biol.*, **14**, R90.

Kjartansdottir,K.R. *et al.* (2015) Traces of ATCV-1 associated with laboratory component contamination. *Proc. Natl. Acad. Sci. USA*, **112**, E925–E926.

Koboldt,D.C. *et al.* (2012) VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.*, **22**, 568–576.

Kostic,A.D. *et al.* (2012) Genomic analysis identifies association of Fusobacterium with colorectal carcinoma. *Genome Res.*, **22**, 292–298.

Laurence,M. *et al.* (2014) Common contaminants in next-generation sequencing that hinder discovery of low-abundance microbes. *PLoS One*, **9**, e97876.

Lee,J.H. *et al.* (2012) De novo somatic mutations in components of the PI3K-AKT3-mTOR pathway cause hemimegalencephaly. *Nat. Genet.*, **44**, 941–945.

Li,H. (2012) Exploring single-sample SNP and INDEL calling with whole-genome de novo assembly. *Bioinformatics*, **28**, 1838–1844.

Li,S. and Chou,H.H. (2004) LUCY2: an interactive DNA sequence quality trimming and vector removal tool. *Bioinformatics*, **20**, 2865–2866.

Lim,J.S. *et al.* (2015) Brain somatic mutations in MTOR cause focal cortical dysplasia type II leading to intractable epilepsy. *Nat. Med.*, **21**, 395–400.

Lopez-Rios,F. *et al.* (2004) Evidence against a role for SV40 infection in human mesotheliomas and high risk of false-positive PCR results owing to presence of SV40 sequences in common laboratory plasmids. *Lancet*, **364**, 1157–1166.

McElroy,K.E. *et al.* (2012) GemSIM: general, error-model based simulator of next-generation sequencing data. *BMC Genom.*, **13**, 74.

Naccache,S.N. *et al.* (2014) Concerns over the origin of NIH-CQV, a novel virus discovered in Chinese patients with seronegative hepatitis. *Proc. Natl. Acad. Sci. USA*, **111**, E976.

Pearce,M. *et al.* (2009) Mutation profiling in tumor samples using the Sequenom OncoCarta^{TM} Panel. *Nature Methods*, **6**, 6.

Poduri,A. *et al.* (2013) Somatic mutation, genomic variation, and neurological disease. *Science*, **341**, 1237758.

Roth,A. *et al.* (2014) PyClone: statistical inference of clonal population structure in cancer. *Nat. Methods*, **11**, 396–398.

Salyakina,D. and Tsinoremas,N.F. (2013) Viral expression associated with gastrointestinal adenocarcinomas in TCGA high-throughput sequencing data. *Hum. Genom.*, **7**, 23.

Saunders,C.T. *et al.* (2012) Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics*, **28**, 1811–1817.

Schmieder,R. and Edwards,R. (2011) Fast identification and removal of sequence contamination from genomic and metagenomic datasets. *PLoS One*, **6**, e17288.

Shirley,M.D. *et al.* (2013) Sturge-Weber syndrome and port-wine stains caused by somatic mutation in GNAQ. *N. Engl. J. Med.*, **368**, 1971–1979.

Strong,M.J. *et al.* (2014) Microbial contamination in next generation sequencing: implications for sequence-based analysis of clinical samples. *PLoS Pathogens*, **10**, e1004437.

Tang,K.W. *et al.* (2013) The landscape of viral expression and host gene fusion and adaptation in human cancer. *Nature Communications*, **4**, 2513.

Tang,K.W. *et al.* (2015) Absence of cytomegalovirus in high-coverage DNA sequencing of human glioblastoma multiforme. *Int. J. Cancer*, **136**, 977–981.

Tao,Z.Y. *et al.* (2015) Vector sequence contamination of the *Plasmodium vivax* sequence database in PlasmoDB and *in silico* correction of 26 parasite sequences. *Parasit Vectors*, **8**, 318.

Tso,K.Y. *et al.* (2014) Are special read alignment strategies necessary and cost-effective when handling sequencing reads from patient-derived tumor xenografts? *BMC Genom.*, **15**, 1172.

van der Veen,B.E. *et al.* (2014) Metaphor: finding bi-directional best hit homology relationships in (meta)genomic datasets. *Genomics*, **104**, 459–463.

White,J.R. *et al.* (2008) Figaro: a novel statistical method for vector sequence removal. *Bioinformatics*, **24**, 462–467.

Wilson,M.R. *et al.* (2014) Actionable diagnosis of neuroleptospirosis by next-generation sequencing. *N. Engl. J. Med.*, **370**, 2408–2417.

Xu,B. *et al.* (2013) Hybrid DNA virus in Chinese patients with seronegative hepatitis discovered by deep sequencing. *Proc. Natl. Acad. Sci. US A*, **110**, 10264–10269.

Zhi,N. *et al.* (2014) Reply to Naccache et al: Viral sequences of NIH-CQV virus, a contamination of DNA extraction method. *Proc. Natl. Acad. Sci. USA*, **111**, E977.