Published in final edited form as:

Curr Protoc Bioinformatics. 2013 December ; 44: 15.4.1–15.4.17. doi:10.1002/0471250953.bi1504s44.

Using VarScan 2 for Germline Variant Calling and Somatic Mutation Detection

Daniel C. Koboldt^{*1}, David E. Larson¹, and Richard K. Wilson¹

Daniel C. Koboldt: dkoboldt@genome.wustl.edu; David E. Larson: dlaron@genome.wustl.edu; Richard K. Wilson: rwilson@genome.wustl.edu

¹The Genome Institute at Washington University in St. Louis, Missouri, USA, 63108

Abstract

The identification of small sequence variants remains a challenging but critical step in the analysis of next-generation sequencing data. Our variant calling tool, VarScan 2, employs heuristic and statistic thresholds based on user-defined criteria to call variants using SAMtools mpileup data as input. Here, we provide guidelines for generating that input, and describe protocols for using VarScan 2 to (1) identify germline variants in individual samples; (2) call somatic mutations, copy number alterations, and LOH events in tumor-normal pairs; and (3) identify germline variants, *de novo* mutations, and Mendelian inheritance errors in family trios. Further, we describe a strategy for variant filtering that removes likely false positives associated with common sequencing- and alignment-related artifacts.

Keywords

variant calling; mutation detection; trio calling; snvs; indels; varscan 2; next-generation sequencing

Introduction

The identification of sequence variants such as single nucleotide variants (SNVs) and small insertions/deletions (indels) represents a critical step in the analysis of resequencing data. The incredible throughput of next-generation sequencing (NGS) instruments has made it feasible to sequence the exomes or genomes of thousands of individuals in a relatively short time for a fraction of the cost of traditional capillary-based sequencing (Mardis 2008). Yet accurate variant calling remains challenging due to variable coverage, sequencing errors,

^{*}Contact Information Address: Campus Box 8501, 4444 Forest Park Blvd., St. Louis, MO 63108, Tel: +1 (314) 286-1814.

INTERNET RESOURCES: <http://varscan.sourceforge.net>

VarScan web site

<http://samtools.sourceforge.net>

SAMtools web site

<https://github.com/genome/bam-readcount>

bam-readcount web site

<http://samtools.sourceforge.net/SAM1.pdf>

SAM/BAM format specification

<http://vcftools.sourceforge.net/specs.html>

Variant Call Format (VCF) specification

alignment artifacts, and other issues (Koboldt, Ding et al. 2010). Many applications of NGS, such as the sequencing of pooled samples or tumor-normal pairs, require detection of variants at low frequencies, which are more difficult to detect using probabilistic methods that were designed for diploid samples.

We initially developed VarScan (<http://varscan.sourceforge.net>) for SNV/indel calling in individual and pooled samples (Koboldt, Chen et al. 2009). In subsequent releases, we expanded our tool's capabilities to include detection of somatic mutations and copy number alterations (CNAs) in tumor-normal pairs (Koboldt, Zhang et al. 2012) and *de novo* mutations in family trios (Figure 1).

Here, we present recommended protocols for germline, somatic, and trio strategies of variant detection using VarScan 2. Basic Protocol 1 describes how to use VarScan to call SNVs and indels in a single individual, or a pool of individuals that were sequenced together. Alternate Protocol 1 also describes germline variant calling, but across a cohort of individuals, also known as cross-sample variant calling. Basic Protocol 2 outlines the steps for analysis of tumor-normal pairs, including the detection of somatic mutations and LOH events. Alternate Protocol 1 describes exome-based copy number calling for identifying somatic CNAs in tumor-normal pairs. Basic Protocol 3 describes the identification of germline variants and *de novo* mutations in family trios. In Support Protocol 1, we outline a strategy for filtering false positive variants that we recommend for germline, somatic, and *de novo* mutation call sets.

Strategic Planning

VarScan is written in Java, so it runs on most operating systems where Java Virtual Machine (JVM) is installed. However, it is a command-line tool, meaning that it must be run in a terminal or command prompt. The main program is available for download as a JAR file from <http://varscan.sourceforge.net>. Once downloaded, VarScan can be run with the command:

```
java -jar VarScan.jar -h
```

The above command will show basic usage information. All VarScan operations are performed using a subcommand, e.g. `java -jar VarScan.jar mpileup2snp` for calling SNPs in one or more individuals.

Sequence Alignments

Before variants can be called, sequencing data must be aligned to a reference sequence. The choice of alignment algorithm and parameters is an important one, and well beyond the scope of this protocol. However, for practical purposes, alignments should be output in (or converted to) the binary alignment/map (BAM) format so that a SAMtools mpileup file (the input for most VarScan operations) can be obtained. For all of these protocols, you will need a sorted, indexed BAM file and an indexed reference sequence. See the SAMtools documentation (<http://samtools.sourceforge.net>) for details on how to generate these files.

SAMtools mpileup Format

The variant calling features of VarScan for single samples (`pileup2snp`, `pileup2indel`, `pileup2cns`) and multiple samples (`mpileup2snp`, `mpileup2indel`, `mpileup2cns`, and `somatic`) expect input in SAMtools `pileup` or `mpileup` format. In current versions of SAMtools, the `pileup` command has now been replaced with the `mpileup` command. For a single sample, these operate in a very similar fashion and the output is identical. The only exception is that the `mpileup` command performs Base Alignment Quality (BAQ) adjustment by default (we recommend disabling this for variant calling). When you give it multiple BAM files, however, SAMtools `mpileup` generates a multi-sample pileup format that must be processed with the `mpileup2*` commands in VarScan. To build an `mpileup` file, you will need:

- One or more BAM files (“myData.bam”) that have been sorted using the sort command of SAMtools.
- The reference sequence (“reference.fasta”) to which reads were aligned, in FASTA format.
- The SAMtools software package (<http://samtools.sourceforge.net>).

A few parameters for SAMtools `mpileup` are particularly relevant for variant calling, and should be considered before starting (Table 1).

Generate an `mpileup` file with the following command:

```
samtools mpileup -f [reference sequence] [parameters] [BAM file(s)]  
>myData.mpileup
```

Note, to save disk space and file I/O, you can redirect `mpileup` output directly to VarScan with a “pipe” command. Here are two examples.

One sample:

```
samtools mpileup -f reference.fasta myData.bam | java -jar  
VarScan.jar mpileup2snp
```

Multiple samples:

```
samtools mpileup -f reference.fasta sample1.bam sample2.bam | java -  
jar VarScan.jar mpileup2snp
```

Basic Protocol 1

Germline Variant Calling in Individual or Pooled Samples

Detection of SNVs and indels in an individual sample is a common task for NGS studies. With VarScan, it is possible to call both types of variants simultaneously using the `mpileup` output from SAMtools. At each position, VarScan looks for variants that meet user-defined minimum criteria for sequencing coverage, number of supporting reads, variant allele frequency (VAF), and Fisher's Exact Test p-value.

The subcommands for germline variant calling are `mpileup2snp` (for SNV calling), `mpileup2indel` (for indel calling), and `mpileup2cns` (for consensus or simultaneous SNV/indel calling).

Relevant user-defined parameters are listed in Table 2. Among these, the most important are the `--min-coverage` and `--min-var-freq` parameters, which essentially govern VarScan's sensitivity and specificity for variant calling. Conservative values (`--min-coverage 20 --min-var-freq 0.20`) yield high-confidence calls, but may miss variants that are under-covered or under-sampled in the dataset. For pooled samples, one should generally specify higher minimum coverage but lower variant allele frequency thresholds.

Necessary Resources

Software: SAMtools (<http://samtools.sourceforge.net>) for generating the input `mpileup`

Data: Aligned sequencing data in binary alignment map (BAM) format, with one BAM file for each sample (individual or pool) sequenced.

Call Variants from a BAM file

1. Run SAMtools `mpileup` on the BAM file (in this case, `sample.bam`):

```
samtools mpileup -B -q 1 -f reference.fasta sample.bam
>sample.mpileup
```

2. Run VarScan `mpileup2snp` to call SNVs.

```
java -jar VarScan.jar mpileup2snp sample.mpileup --min-coverage 10 --min-var-freq 0.20 --p-value 0.05
>sample.varScan.snp
```

3. Run VarScan `mpileup2indel` to call indels:

```
java -jar VarScan.jar mpileup2indel sample.mpileup --min-coverage 10 --min-var-freq 0.10 --p-value 0.10
>sample.varScan.indel
```

Note that we use slightly less stringent parameters for indel calling, which reflects the fact that these variants are more difficult to detect in NGS data. VarScan's sensitivity and range for indel detection are dictated by the underlying read alignments. For example, aligning paired-end 100-bp reads with the BWA algorithm typically allows for detection of indels in the 1-30 bp size range, corresponding to the gap size that the aligner allows. Alternate indel detection strategies, such as the split-read mapping approach employed by Pindel (Ye, Schulz et al. 2009), are recommended for larger indels.

4. Filter SNP calls to remove those near indel positions:

```
java -jar VarScan.jar filter sample.varScan.snp --indel-file sample.varScan.indel --output-file sample.varScan.snp.filter
```

5. Filter indels to obtain a higher-confidence set:

```
java -jar VarScan.jar filter sample.varScan.indel --min-reads2
4 --min-var-freq 0.15 --p-value 0.05 --output-file
sample.varScan.indel.filter
```

See Support Protocol 1 for additional filtering recommendations.

Alternate Protocol 1

Germline Variant Calling in a Cohort of Individuals

The ability of SAMtools to generate multiple-sample pileup (mpileup) output makes it possible to call variants across a set of many samples simultaneously. This approach, called cross-sample variant calling, is advantageous because it identifies all variant positions in a group of samples, and provides genotype calls for all samples at each one. Most users will prefer to obtain output in variant call format (VCF), which is compatible with many downstream analysis tools (Danecek, Auton et al. 2011).

Notably, the size of the output file increases exponentially with the number of samples included, since each sample must be genotyped at every variant position (and variant positions unique to that sample must be genotyped in all other samples). Nevertheless, a single master VCF file containing all variant calls in a cohort offers a convenient format for data sharing and downstream analysis.

The command and parameter settings for cross-sample variant calling are the same as for Basic Protocol 1. At each position, variant calling requirements are evaluated on each individual's mpileup data. Any positions at which at least one sample contains a qualifying variant are reported for the entire cohort.

Necessary Resources

Software: SAMtools (<http://samtools.sourceforge.net>) for generating the input mpileup

Data: Aligned sequencing data in binary alignment map (BAM) format, with one BAM file for each sample in the cohort. A sample list for use in VCF column headers (e.g. samples.txt), should also be provided. This is a text file, one sample per line, and the order of the samples should match the order of the BAM files given to the SAMtools command in step 1.

Perform Cross-sample Variant Calling

1. Run SAMtools mpileup on all BAM files in a single command:

```
samtools mpileup -B -q 1 -f reference.fasta sample1.bam
sample2.bam sample3.bam ... >cohort.mpileup
```

Any number of BAM files can theoretically be provided to SAMtools (just three are shown above), but users may encounter I/O issues when processing many (>100) BAM files simultaneously.

2. Run VarScan mpileup2snp to call SNVs.

```
java -jar VarScan.jar mpileup2snp cohort.mpileup --vcf-sample-  
list samples.txt --min-coverage 10 --min-var-freq 0.20 --p-  
value 0.05 --output-vcf 1 >cohort.varScan.snp.vcf
```

3. Run VarScan mpileup2indel to call indels:

```
java -jar VarScan.jar mpileup2indel cohort.mpileup --vcf-  
sample-list samples.txt --min-coverage 10 --min-var-freq 0.10  
--p-value 0.10 --output-vcf 1 >cohort.varScan.indel.vcf
```

Note that we use slightly less stringent parameters for indel calling, for reasons described in Basic Protocol 1.

Basic Protocol 2

Somatic Mutation Detection in Tumor-Normal Pairs

Next-generation sequencing of tumor samples and matched (normal) controls is a powerful strategy for studying the genetic basis of cancer initiation, development, and growth. The importance of a matched normal sample from the patient cannot be understated, because it allows researchers to distinguish acquired/somatic mutations (<0.01% of variants) from inherited germline variation (>99.99% of variants). It is also advisable that tumor and normal samples be sequenced under identical experimental conditions, to minimize potential batch effects that could affect analysis.

Given NGS data from a tumor-normal pair, VarScan can identify somatic mutations, germline variants, and LOH events by directly comparing normal and tumor data at every position of sufficient coverage. The VarScan somatic command accepts mpileup input from a normal and tumor sample (in that order); at every position meeting the minimum coverage requirement, it calls both samples independently to identify possible variants. If a variant is called in either sample, the genotypes and supporting data for both samples are compared to infer the somatic status (Germline, LOH, or Somatic).

Specifically, a Fisher's Exact Test of the read counts supporting reference and variant alleles in normal and tumor determines whether the evidence for a variant is different (and statistically significant) between normal and tumor. The user-specified threshold (--somatic-p-value) tells VarScan the value at which a difference between tumor and normal is deemed significant, meaning that the variant will be called somatic (if enriched in tumor) or LOH (if enriched in normal). This approach allows for detection of subtle differences between normal and tumor, such as low-frequency mutations due to subclonal populations or low tumor cellularity.

Necessary Resources

Software: SAMtools (<http://samtools.sourceforge.net>) for generating the input mpileup

Data: Aligned sequencing data in binary alignment map (BAM) format, with one BAM file for the tumor and one for the matched normal.

Perform Somatic Mutation Calling

1. Run SAMtools mpileup on the BAM files for normal and tumor:

```
samtools mpileup -B -q 1 -f reference.fasta normal.bam
tumor.bam >normal-tumor.mpileup
```

2. Run VarScan in somatic mode, providing the mpileup file (normal-tumor.mpileup) and a basename for output files (output.basename):

```
java -jar VarScan.jar somatic normal-tumor.mpileup
output.basename -min-coverage 10 -min-var-freq 0.08 -somatic-
p-value 0.05
```

See Table 3 for descriptions and recommended settings of VarScan somatic parameters. The above command will generate two output files, one for SNVs (output.basename.snp) and one for indels (output.basename.indel). See Table 4 for details on VarScan somatic output fields.

3. Run the processSomatic subcommand to divide the output into separate files based on somatic status and confidence:

```
java -jar VarScan.jar processSomatic output.basename.snp
java -jar VarScan.jar processSomatic output.basename.indel
```

This command will generate six files per input file. For SNVs, the output files will be:

```
output.basename.snp.Somatic – all somatic mutations
output.basename.snp.Somatic.hc – high-confidence somatic mutations
output.basename.snp.LOH – all LOH events
output.basename.snp.LOH.hc – high-confidence somatic events
output.basename.snp.Germline – all germline variants
output.basename.snp.Germline.hc – high-confidence germline variants
```

The subset of high-confidence variants is determined using a few empirically-derived criteria. For example, high-confidence somatic mutations have tumor VAF>15%, normal VAF<5%, and a somatic p-value of <0.03. These are user-adjustable.

4. Run an additional filter on the somatic mutations

```
java -jar VarScan.jar somaticFilter
output.basename.snp.Somatic.hc -indel-file
output.basename.indel -output-file
output.basename.snp.Somatic.hc.filter
```

The above command identifies and removes somatic mutations that are likely false positives due to alignment problems near indels. After this step, candidate somatic

mutations should also be filtered to remove other artifacts, as described in Support Protocol 1.

Support Protocol 1

Filtering to Remove False Positives

Many, if not most, of the false positive variant calls obtained by VarScan and similar programs are due to systematic artifacts from the alignment of relatively short (100 bp) read sequences to a reference genome. Reads from genomic regions that are not fully represented in the reference can be mapped to paralogous regions, often with recurrent substitutions or gaps at recurrent positions. Alternatively, reads may map to the correct location, but suffer local mis-alignment (especially near insertion/deletion variants) that gives rise to false positive variant calls. Other systematic artifacts may be attributed to the sequencing platform, such as reads with long runs of low-quality bases.

We have developed a strategy to filter likely false positives based on strand representation, mismatch quality sum (an indication of paralogous alignments), read position bias, mapping quality differences, and other criteria (Koboldt, Zhang et al. 2012). Because many of these criteria cannot be evaluated from the SAMtools mpileup file, we developed another program (bam-readcount) to retrieve the necessary metrics directly from the BAM files.

The filtering protocol described is recommended for all VarScan call sets (from mpileup2cns, somatic, or trio subcommands), but is particularly important for predicted mutations (somatic or *de novo*). Because these are usually rare events, their call sets are often enriched for false positives.

Necessary Resources

Software: The *bam-readcount* utility (<https://github.com/genome/bam-readcount>)

The *fpfilter.pl* accessory script (<https://sourceforge.net/projects/varscan/files/scripts/>)

Data: A list of variants (SNVs or indels) in VarScan format (tab-delimited; the first five columns must be chromosome, position, reference allele, variant allele).

An indexed BAM file from which to extract metrics. For individual variant calls, use the individual's BAM file. For somatic mutations, use the tumor BAM file. For Germline and LOH events in tumor-normal comparisons, use the normal BAM file, as both types events should be best represented in the normal sample.

The reference sequence to which reads in the BAM file were aligned.

Run the False Positive Filter:

1. Obtain metrics for the list of variants:

```
bam-readcount -q 1 -b 20 -f reference.fasta -l  
varScan.variants BAM_FILE >varScan.variants.readcounts
```

2. Run the FPfilter accessory script:


```
perl fpfilter.pl varScan.variants varScan.variants.readcounts  
-output-basename varScan.variants.filter
```

The above command would create two output files. Variants passing the filter are found in `varScan.variants.filter.pass` while variants that fail are printed to `varScan.variants.filter.fail` along with the reason for the failure. Filtering parameters in the `fpfilter.pl` script are set to recommended values for Illumina paired-end (2×100 bp) reads, but can be modified by the user in the script if desired.

Alternate Protocol 2

Somatic Copy Number Alteration Detection in Tumor-Normal Pairs

The digital read counts provided by next-generation sequencing are not only useful for estimating allele frequencies, but also informative for genomic copy number as well. Numerous methods have been developed to detect copy number variation (CNV) using whole-genome sequencing data while accounting for read mappability, GC content bias, and other factors (Koboldt, Larson et al.). Due to the prohibitive cost of whole-genome sequencing, however, many genetic studies of tumors are conducted by targeted sequencing, e.g. exome sequencing of a tumor-normal pair. Such datasets are valuable for rapid screening of protein-coding mutations and germline (inherited) variants. CNV calling with targeted sequencing data, however, is hampered by the inherent variability in capture efficiency from one region to the next.

When a tumor and its matched normal are sequenced under identical conditions, however, a direct comparison of exome sequence depth at each covered position can be used to infer relative copy number changes in the tumor. Because the tumor and normal samples are nearly identical, the effect of background genetic variation on capture efficiency should be minimal. It is therefore possible to identify somatic copy number alterations (gains and losses) in targeted sequencing data for tumor-normal pairs. This protocol describes how to conduct such an analysis with VarScan 2 and a popular segmentation algorithm implemented in R.

Necessary Resources

Software: SAMtools (<http://samtools.sourceforge.net>) for generating the input mpileup

The DNACopy R package from Bioconductor (<http://www.bioconductor.org/packages/2.3/bioc/html/DNACopy.html>)

Data: Aligned sequencing data in binary alignment map (BAM) format, with one BAM file for the tumor and one for the matched normal. To account for differences in input data, the user should compute the data ratio of uniquely mapped reads in the normal to uniquely mapped reads in the tumor (`normal_unique_mapped_reads / tumor_unique_mapped_reads`).

Detect Somatic Copy Number Changes

1. Run SAMtools mpileup on the BAM files for normal and tumor:

```
samtools mpileup -B -q 1 -f reference.fasta normal.bam
tumor.bam >normal-tumor.mpileup
```

2. Run VarScan in copynumber mode, providing the mpileup file (normal-tumor.mpileup) and a basename for output files (output.basename):

```
java -jar VarScan.jar copynumber normal-tumor.mpileup
output.basename -min-coverage 10 --data-ratio [data_ratio] --
min-segment-size 20 --max-segment-size 100
```

The above command will report all contiguous regions that meet the coverage requirement (10) in both normal and tumor. Only regions of at least 20 bp will be reported, and after reaching 100 bases, a new region will be started. These user-configured parameters should be adjusted with care, as they govern the confidence and resolution of reported regions. For each region, VarScan calculates the mean coverage in tumor and normal, the log₂ value of the ratio of tumor to normal mean coverages, and the GC content.

3. Adjust for GC content and extract potential homozygous deletions

```
java -jar VarScan.jar copyCaller varScan.output.copynumber -
output-file varScan.output.copynumber.called --homdel-file
varScan.output.copynumber.homdel
```

The above command computes a copy number change distribution across GC content values and performs an adjustment to account for observed bias. It also identifies regions with high coverage in the normal but low or no coverage in the tumor. These regions may represent experimental artifacts or possibly homozygous deletions. In either case, the copy number change cannot be reliably computed, so they should be analyzed separately. Note, this command can also be used to recenter the data in the event of a systematic inflation or deflation of log₂ ratio.

The output of this command is a list of regions with coordinates and ratio of copy number change (log₂ value); this is similar to the output of array-based copy number experiments, and therefore amenable to similar segmentation algorithms.

Perform Circular Binary Segmentation (CBS)

1. Load the DNACopy library and import VarScan called copynumber data into R.

```
library(DNACopy)

cn <- read.table("varScan.output.copynumber.called",header=T)
```

2. Create a copynumber object using the adjusted log₂ ratio, chromosome, and region start position from your input.

```
CNA.object <-CNA( genomdat = cn$adjusted_log_ratio, chrom = cn
$chrom, maploc = cn$chr_start, data.type = 'logratio')
```

3. Perform smoothing on the CNA object to remove outliers:

```
CNA.smoothed <- smooth.CNA(CNA.object)
```

4. Run CBS on the smoothed data:

```
segment <- segment(CNA.smoothed, verbose=0, min.width=2,
undo.SD=3)
```

Note that the `undo.SD` parameter follows recommendations from the authors of the DNACopy package, who suggest that change-points of less than two or three standard deviations should be removed. The user is encouraged to experiment with different `undo.SD` values to achieve a level of segmentation that is best fitted to the data. Empirical experience suggests that `undo.SD` values between 1 and 4 are usually appropriate.

5. Calculate change-point p-values for segments:

```
p.segment <- segments.p(segment)
```

6. Plot the results (optional)

```
plot(segment, type="w")
```

Note that the DNACopy library overrides the `plot()` function in R, to provide appropriate plotting of raw data and segmented results.

7. Output the segmented regions to a tab-delimited file:

```
write.table(p.segment,
file="varScan.output.copynumber.called.segments.p_value",
sep="\t")
```

See the DNACopy library documentation for details on the contents of the output.

Basic Protocol 3

Pedigree-Aware Calling of Family Trios

Next-generation sequencing of family pedigrees offers a powerful approach to studying inherited disease. There is particular interest in studying family trios (mother, father, and affected child) to identify transmitted alleles and/or *de novo* mutations that may confer susceptibility to disease. We have developed a trio-calling module for VarScan which leverages the family relationship to improve variant calling accuracy, identify apparent Mendelian Inheritance Errors (MIEs), and detect high-confidence *de novo* mutations.

Data from whole-genome sequencing studies of families have shown that the *de novo* mutation rate in humans is approximately 1.1×10^{-8} per haploid genome (Roach, Glusman et al. ; 1000 Genomes Project Consortium 2010). By this estimate, an individual's diploid genome harbors, on average, around 64 *de novo* mutations among 3.2 billion base pairs. Given the current size of the consensus coding sequence (CCDS) in humans (~34 million base pairs), we expect less than one *de novo* coding mutation per diploid individual.

Because of this extreme rareness, *de novo* mutations should be called conservatively. To address this, VarScan re-evaluates apparent *de novo* mutations in each parent using relaxed parameters and re-classifies those with some evidence in one or both parents as a germline variant. In a similar manner, VarScan attempts to reconcile apparent Mendelian Inheritance

Errors. The output of the trio subcommand is a single VCF in which all variants are classified as germline (transmitted or untransmitted), *de novo*, or MIE.

Necessary Resources

Software: SAMtools (<http://samtools.sourceforge.net>) for generating the input mpileup

Data: Aligned sequencing data in binary alignment map (BAM) format, with one BAM file each for father, mother, and child.

Perform Cross-sample Variant Calling

1. Run SAMtools mpileup on father, mother, and child BAM files in that order.

```
samtools mpileup -B -q 1 -f reference.fasta father.bam
mother.bam child.bam >trio.mpileup
```

2. Run VarScan mpileup2snp to call SNVs. 000747096 for definitions and recommended values of command-line parameters.

```
java -jar VarScan.jar trio trio.mpileup trio.mpileup.output --
min-coverage 10 --min-var-freq 0.20 --p-value 0.05 -adj-var-
freq 0.05 -adj-p-value 0.15
```

The above command will produce two VCF output files: one for SNPs (trio.mpileup.output.snp.vcf) and one for indels (trio.mpileup.output.indel.vcf).

See Table 6 for details on the VCF output format and custom field definitions.

Guidelines for Understanding Results

The most common issue reported by VarScan 2 users is a situation in which VarScan's call (or lack of call) does not match expectation. It should be noted that the --min-coverage and --min-var-freq parameters define hard thresholds for variant calling in VarScan 2. In somatic mode, if --min-var-freq is set to 0.20, then a mutation present in the tumor at 16% variant allele frequency will not be reported, regardless of the p-value. For sites of particular interest, users may wish to use the --validation 1 parameter in VarScan somatic, which tells the program to report all sites meeting the minimum coverage requirement, even if no variant was called (in which case the somatic status will be "Reference"). The variant allele frequencies can be useful for evaluating low allelic fraction mutations at key sites of interest.

The read counts and variant allele frequencies computed by VarScan may differ from those provided by other programs, such as the Integrative Genomics Viewer (IGV, (Robinson, Thorvaldsdottir et al.)). Notably, these discrepancies usually arise due to differences in minimum mapping and/or base quality thresholds employed by other programs, as well as the BAQ adjustment in the SAMtools mpileup command. By default, VarScan does not count bases with reported qualities below 15 or 20 (depending on the command); often the BAQ computation adjusts many base qualities to values below these thresholds. We also recommend a minimum mapping quality of 1 or 10 for most VarScan commands; this

parameter choice, too, will influence the reads that are included in the *mpileup* output and therefore the resulting read counts and variant allele frequencies.

Regardless of the approach used, variant calling in NGS data is an imperfect art and may still produce some false positives. Variants of interest that are discovered using NGS technologies, especially somatic or *de novo* mutations and novel genetic variants, should be carefully scrutinized. Visual manual review of the sequencing data in a viewer such as IGV (Robinson, Thorvaldsdottir et al.) is a common and inexpensive strategy for doing so. Ideally, however, novel variants should be experimentally validated in an orthogonal assay, such as targeted resequencing (to very high coverage or with a different sequencing platform) or genotyping.

The downstream analysis, annotation, and interpretation of sequence variants remain a challenging area of this field that extends beyond the scope of this article. However, a recent review of variant analysis tools for NGS data (Pabinger, Dander et al.) may offer some initial guidance in this regard.

The tables accompanying this protocol unit describe the usage and recommended settings for input parameters, as well as the output formats. For further details on VarScan native output formats, please visit the web site (<http://varscan.sourceforge.net>). For information about the Variant Call Format (VCF) specification (Danecek, Auton et al. 2011), please visit the VCFtools web site (<http://vcftools.sourceforge.net>).

Commentary

Background Information

NGS technologies have many applications, including both genome (whole, exome, or targeted region) sequencing and transcriptome sequencing (RNA-seq) of population cohorts, cases and controls, family pedigrees, and tumor-normal pairs. Many methods have been developed for the detection of SNVs (Wei, Wang et al.; Li, Handsaker et al. 2009; Li, Li et al. 2009; Shen, Wan et al. 2010) and indels (Li, Li et al.; Ye, Schulz et al. 2009; Albers, Lunter et al. 2010) as well as somatic mutations (Li ; Saunders, Wong et al. ; Larson, Harris et al. 2012) with NGS data.

The ability to adjust coverage and minimum allele frequency thresholds in VarScan 2 makes it advantageous for the detection of low allele frequency variants in high-depth datasets, such as RNA-seq data and tumor-normal exome pairs. Other approaches, such as the Bayesian statistical models employed by SAMtools/BCFtools (Li ; Li, Handsaker et al. 2009) and GATK (McKenna, Hanna et al. 2010), are also popular. Such models perform well in the analysis of diploid genomes, but may be hampered by datasets of extreme coverage depth or low allelic fractions. Indeed, a recent comparison of mutation detection tools for tumor subclone analysis (Stead, Sutton et al. 2013) found that VarScan 2 performed best overall with sequencing depths of 100×, 250×, 500×, and 1,000× required to accurately identify variants present at 10%, 5%, 2.5%, and 1%, respectively.

In some studies, such as family-based sequencing of several family members, a different variant caller that incorporates extended pedigree information (such as Polymutt (Li, Chen et

al.)) may be advantageous. For most applications, a comprehensive strategy combining VarScan 2 variant calls with those made by another method (such as SAMtools, GATK, or Pindel) is likely to provide the most comprehensive results. For example, our group routinely combines the results of VarScan 2, SomaticSniper (Larson, Harris et al. 2012), and Strelka (Saunders, Wong et al.) for somatic SNV calling in whole-genome sequencing data for tumor-normal pairs.

Parameter Choices and Troubleshooting

When investigating an unexpected result from VarScan 2, users may wish to work backwards from the call to the evidence used for that call to the underlying data. For example, to investigate a variant that is visible in IGV but not called by VarScan, the user might first run VarScan's *mpileup2cns* command with relaxed parameters, to compute the read depth and variant allele frequency for every sample at the variant position. Then, the user might ask:

- How many reads appear to support the variant in aligned reads? Are the mapping and base qualities of those reads sufficiently high?
- Does the variant allele frequency computed by VarScan fall below the minimum threshold? Does the p-value fail to meet the significance threshold?
- What does the raw SAMtools mpileup output look like for the position in question? Do the numbers of variant-supporting reads and base qualities match expectation?

For bug reports and questions/suggestions related to VarScan 2, users are encouraged to post to the “Help” forum on VarScan's web site (<http://sourceforge.net/p/varscan/discussion/1073559/>). Additional helpful information on VarScan may be found on the SeqAnswers (<http://www.seqanswers.com>) and BioStars (<http://www.biostars.org>) online forums.

Acknowledgments

This work was supported by grants from the National Institutes of Health: U54HG003079 (R.K. Wilson) and U01HG006517 (L. Ding).

Literature Cited

- 1000 Genomes Project Consortium, T. A map of human genome variation from population-scale sequencing. *Nature*. 2010; 467(7319):1061–73. [PubMed: 20981092]
- Albers CA, Lunter G, et al. Dindel: accurate indel calls from short-read data. *Genome Res*. 2010; 21(6):961–73. [PubMed: 20980555]
- Danecek P, Auton A, et al. The variant call format and VCFtools. *Bioinformatics*. 2011; 27(15):2156–8. [PubMed: 21653522]
- Koboldt DC, Chen K, et al. VarScan: Variant detection in massively parallel sequencing of individual and pooled samples. *Bioinformatics*. 2009
- Koboldt DC, Ding L, et al. Challenges of sequencing human genomes. *Brief Bioinform*. 2010; 11(5): 484–98. This review article offers guidelines for next-generation sequencing data analysis while highlighting some of the important challenges of human genome resequencing with NGS technologies. [PubMed: 20519329]
- Koboldt DC, Larson DE, et al. Massively parallel sequencing approaches for characterization of structural variation. *Methods Mol Biol*. 838:369–84. [PubMed: 22228022]

- Koboldt DC, Zhang Q, et al. VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.* 2012; 22(3):568–76. The VarScan 2 publication describes the algorithm's underlying methodology and showcases its performance (variant calling, mutation detection, somatic CNA detection, and false positive filtering) using exome data from tumor-normal pairs. [PubMed: 22300766]
- Larson DE, Harris CC, et al. SomaticSniper: identification of somatic point mutations in whole genome sequencing data. *Bioinformatics.* 2012; 28(3):311–7. [PubMed: 22155872]
- Li B, Chen W, et al. A likelihood-based framework for variant calling and de novo mutation detection in families. *PLoS Genet.* 8(10):e1002944. [PubMed: 23055937]
- Li H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics.* 27(21):2987–93. [PubMed: 21903627]
- Li H, Handsaker B, et al. The Sequence Alignment/Map format and SAMtools. *Bioinformatics.* 2009; 25(16):2078–9. [PubMed: 19505943]
- Li R, Li Y, et al. SNP detection for massively parallel whole-genome resequencing. *Genome Res.* 2009; 19(6):1124–32. [PubMed: 19420381]
- Li S, Li R, et al. SOAPindel: efficient identification of indels from short paired reads. *Genome Res.* 23(1):195–200. [PubMed: 22972939]
- Mardis ER. Next-generation DNA sequencing methods. *Annu Rev Genomics Hum Genet.* 2008; 9:387–402. [PubMed: 18576944]
- McKenna A, Hanna M, et al. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* 2010; 20(9):1297–303. [PubMed: 20644199]
- Pabinger S, Dander A, et al. A survey of tools for variant analysis of next-generation genome sequencing data. *Brief Bioinform.*
- Roach JC, Glusman G, et al. Analysis of genetic inheritance in a family quartet by whole-genome sequencing. *Science.* 328(5978):636–9. [PubMed: 20220176]
- Robinson JT, Thorvaldsdottir H, et al. Integrative genomics viewer. *Nat Biotechnol.* 29(1):24–6. [PubMed: 21221095]
- Saunders CT, Wong WS, et al. Strelka: accurate somatic small-variant calling from sequenced tumor-normal sample pairs. *Bioinformatics.* 28(14):1811–7. [PubMed: 22581179]
- Shen Y, Wan Z, et al. A SNP discovery method to assess variant allele probability from next-generation resequencing data. *Genome Res.* 2010; 20(2):273–80. [PubMed: 20019143]
- Stead LF, Sutton KM, et al. Accurately Identifying Low-Allelic Fraction Variants in Single Samples with Next-Generation Sequencing: Applications in Tumor Subclone Resolution. *Hum Mutat.* 2013
- Wei Z, Wang W, et al. SNVer: a statistical tool for variant calling in analysis of pooled or individual next-generation sequencing data. *Nucleic Acids Res.* 39(19):e132. [PubMed: 21813454]
- Ye K, Schulz MH, et al. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics.* 2009; 25(21):2865–71. [PubMed: 19561018]

Figure 1. VarScan variant calling applications by data type

VarScan identifies single nucleotide both variants (SNVs) and insertion/deletion variants (indels) using the SAMtools mpileup output from one or more BAM files.

Table 1
SAMtools mpileup parameters of note

Parameter	Description	Recommendation
-f STR	Reference sequence in FASTA format	Provide the reference sequence to which the reads were mapped.
-q INT	Minimum mapping quality	At least 1 to use only uniquely mapped reads, but setting higher values (e.g. 10) may be desirable.
-B	Disable BAQ adjustment	Using this parameter disables BAQ, and is the current recommended practice for variant calling with VarScan.
-r STR	Region for which mpileup is generated	Use this command to extract mpileup for one specific region of the genome. The format is <i>ref_name:start-stop</i> (e.g. 1:1001-2001)

Table 2
VarScan germline variant calling parameters for the mpileup2cns, mpileup2snp, and mpileup2indel subcommands. All conditions must be met for a variant to be called

Parameter	Description	Recommendation
--min-avg-qual INT	Minimum Phred base quality	At least 15 for Phred-scaled base qualities, or 20 for even more conservative calling.
--min-coverage INT	The minimum depth of coverage	The default is 8, but the highest confidence calls have 10x or 20x coverage.
--min-reads2 INT	The minimum number of variant supporting reads.	At least 4
--min-var-freq NUM	A decimal value indicating the minimum VAF for calling variants.	For individual samples, 0.20 to allow for variant under-representation. Lower for pooled samples (maybe ½ the expected frequency of a single heterozygote for the most sensitive detection).
--p-value NUM	The Fisher's Exact Test p-value threshold below which a variant call is deemed significant and will be reported.	0.05 or 0.10 are useful starting points. Note that a perfect heterozygote (50% of reads) requires 12x coverage to reach F.E.T. p-value of 0.05.
--min-freq-for-hom	Minimum VAF above which a variant will be called homozygous in a given sample.	The default (0.75) is fairly conservative, but addresses the fact that NGS alignments bias toward the reference allele.
--variants	Report only variant positions. Otherwise, report all positions meeting minimum coverage.	Set to 1 if the goal is to call SNPs and indels simultaneously (into a single output file) using mpileup2cns.
--output-vcf	Generate VCF output	Set to 1 if VCF output is desired, or leave unset for VarScan's native output format, which is more human readable.

Table 3
VarScan somatic mutation calling parameters for the somatic subcommand

Parameter	Description	Recommendation
<code>--min-avg-qual INT</code>	Minimum Phred base quality	At least 15 for Phred-scaled base qualities, or 20 for even more conservative calling.
<code>--min-coverage INT</code>	The minimum depth of coverage required (in both normal and tumor) for a position to be evaluated.	The default is 8, but it may be advisable to specify different thresholds for normal and tumor using the next two parameters instead.
<code>--min-normal-coverage</code>	The minimum depth of coverage required for the normal. Overrides <code>--min-coverage</code> .	6
<code>--min-tumor-coverage</code>	The minimum depth of coverage required for the tumor. Overrides <code>--min-coverage</code> .	8
<code>--min-var-freq NUM</code>	A decimal value indicating the minimum VAF for calling variants.	For individual samples, 0.20 to allow for variant under-representation. Lower for pooled samples (maybe ½ the expected frequency of a single heterozygote for the most sensitive detection).
<code>--p-value NUM</code>	The Fisher's Exact Test p-value for calling variants in the individual normal/tumor samples	Leaving this parameter at 0.99 is advisable, as it signals VarScan to skip FET computation for individual samples, and instead compare tumor and normal directly.
<code>--somatic-p-value NUM</code>	The Fisher's Exact Test p-value threshold for significant differences between normal and tumor	0.05 is a good starting point. With good coverage, heterozygous somatic mutations will reach this threshold easily, while false positives (e.g. under-called germline variants or systematic artifacts) will not.
<code>--min-freq-for-hom</code>	Minimum VAF above which a variant will be called homozygous in a given sample.	The default (0.75) is fairly conservative, but addresses the fact that NGS alignments bias toward the reference allele.
<code>--normal-purity NUM</code>	Expected purity of the normal sample	This should be 1, unless analyzing cancer types such as AML in which normal tissue may be contaminated with tumor cells. Setting a value below 1.0 tells VarScan to adjust (reduce) the VAF observed in normal according to expected tumor contamination.
<code>--tumor-purity NUM</code>	Expected purity of the tumor sample	Few tumor samples are 100% pure, and VarScan's default settings address that already. Set this to a value below 1 if you believe the tumor to have very low cellularity, i.e. 0.50. Doing so tells VarScan to adjust (increase) the observed VAF in tumor accordingly.
<code>--strand-filter</code>	Enable (1) or disable (0) the strand filter	If set to 1, VarScan will flag/remove somatic mutations for which the variant supporting reads in tumor are significantly biased toward one strand.
<code>--output-vcf</code>	Generate VCF output	Set to 1 if VCF output is desired, or leave unset for VarScan's native output format, which is more human readable.

Table 4
VarScan somatic output. The column headers for VarScan native output or field names for VCF output are described below

Native header	VCF field name	Description
chrom	CHROM	Chromosome or reference name
position	POS	Position from SAMtools pileup (1-based)
ref	REF	Reference base at this position
var	ALT	Variant base seen in tumor
normal_reads1	RD (col 10)	Reads supporting reference in normal
normal_reads2	AD (col 10)	Reads supporting variant in normal
normal_var_freq	FREQ (col 10)	Variant allele frequency in normal
normal_gt	GT (col 10)	Consensus genotype call in normal
tumor_reads1	RD (col 11)	Reads supporting reference in tumor
tumor_reads2	AD (col 11)	Reads supporting variant in tumor
tumor_var_freq	FREQ (col 11)	Variant allele frequency in tumor
tumor_gt	GT (col 11)	Consensus genotype in tumor
somatic_status	SS (col 8)	Somatic status (Germline, Somatic, LOH, Unknown)
variant_p_value	GPV (col 8)	Variant p-value from FET for germline events (tumor + normal vs reference)
somatic_p_value	SPV (col 8)	Somatic p-value from FET (tumor vs normal)
tumor_reads1_plus	DP4 (col 11)	Tumor reference-supporting reads on + strand
tumor_reads1_minus		Tumor reference-supporting reads on – strand
tumor_reads2_plus		Tumor variant-supporting reads on + strand
tumor_reads2_minus		Tumor variant-supporting reads on - strand
normal_reads1_plus	DP4 (col 10)	Normal reference-supporting reads on + strand
normal_reads1_minus		Normal reference-supporting reads on – strand
normal_reads2_plus		Normal variant-supporting reads on + strand
normal_reads2_minus		Normal variant-supporting reads on - strand

Table 5
VarScan trio calling parameters for the VarScan trio subcommand

Parameter	Description	Recommendation
--output-name STR	Basename for output VCF files	Provide if piping SAMtools mpileup into VarScan
--min-coverage INT	The minimum depth of coverage required (in both normal and tumor) for a position to be evaluated.	For high-confidence calling, specify 20.
--min-reads2	The minimum number of variant-supporting reads to call a variant	4
--min-avg-qual	Minimum Phred base quality required to count a base	20
--min-var-freq	Minimum variant allele frequency for a variant to be called	0.20
--p-value	Fisher's Exact Test p-value threshold required for variant calls	0.05
--adj-var-freq	Adjusted minimum VAF for re-calling family members	0.05
--adj-p-value	Adjusted FET p-value threshold for re-calling family members	0.10

Table 6
VarScan trio output. SNPs and indels are output to separate VCF files, each with the following fields

VCF field name	Description
CHROM	Chromosome or reference name
POS	Position from SAMtools pileup (1-based)
REF	Reference base at this position
ALT	Variant base seen in tumor
FILTER	mendelError if MIE, otherwise PASS
STATUS (col 8)	1=untransmitted, 2=transmitted, 3=denovo, 4=MIE
DENOVO	A flag for <i>de novo</i> mutations
GT	Genotype call
GQ	Genotype quality score
SDP	SAMtools raw read depth
DP	VarScan quality read depth
RD	Number of reads supporting reference base
AD	Number of reads supporting variant base
FREQ	Variant allele frequency
PVAL	P-value from Fisher's Exact Test (this sample vs reference).
RBQ	Average base quality of reference-supporting reads
ABQ	Average base quality of variant-supporting reads
RDF	Number of reference-supporting reads on + strand
RDR	Number of reference-supporting reads on - strand
ADF	Number of variant-supporting reads on + strand
ADR	Number of variant-supporting reads on - strand