Bioinformatics Advance Access published August 18, 2015

Application Note

svviz: a read viewer for validating structural variants

Noah Spies[1,2,3*], Justin M. Zook[4], Marc Salit[3,] and Arend Sidow[1,2]

[1]Department of Genetics and

[2]Department of Pathology, Stanford University, Stanford, CA

[3]Genome Scale Measurements Group, National Institute of Standards and Technology, Stanford, CA and

[4]Gaithersburg, MD

Associate Editor: Dr. Inanc Birol

ABSTRACT

Summary: Visualizing read alignments is the most effective way to validate candidate SVs with existing data. We present svviz, a sequencing read visualizer for structural variants (SVs) that sorts and displays only reads relevant to a candidate SV. svviz works by searching input bam(s) for potentially relevant reads, realigning them against the inferred sequence of the putative variant allele as well as the reference allele, and identifying reads that match one allele better than the other. Separate views of the two alleles are then displayed in a scrollable web browser view, enabling a more intuitive visualization of each allele, compared to the single reference genome-based view common to most current read browsers. The browser view facilitates examining the evidence for or against a putative variant, estimating zygosity, visualizing affected genomic annotations, and manual refinement of breakpoints. svviz supports data from most modern sequencing platforms.

Availability and Implementation: svviz is implemented in python and freely available from http://svviz.github.io/.

Contact: nspies@stanford.edu

Supplementary information: Supplementary data are available at Bioinformatics online.

1 INTRODUCTION

The human eye has an unparalleled ability to identify patterns from visual representations of data. While the identification of mutations from high-throughput sequencing has been largely automated, visual inspection of putative variants using tools such as the Integrative Genomics Viewer (IGV; Robinson 2011) remains an important step in ensuring the quality and relevance of these variant calls. However, existing read visualizing tools such as IGV are largely constrained by a reference-genome-centric display model. Hence, point mutations can be represented easily as mismatched bases within sequencing data, but more complex structural variants (SVs) including insertions, deletions, translocations and inversions are more difficult to parse visually against the linear reference genome sequence. Newer tools are able to represent short indels within sequencing data but do not help in representing larger SVs (Edmonson 2011; Gymrek 2014).

Support for SVs can be displayed within IGV by highlighting reads with certain characteristics, including read pairs mapping to distant regions of the genome or in unexpected orientations, or truncated alignments. However, it is difficult to identify from these highlighted, discordantly mapping reads whether they all agree with a putative variant, and if so, which variant. Furthermore, IGV relies on the quality and completeness of the alignments provided in input BAM files, which are produced en masse against a huge reference genome and hence may not optimally represent read support for a given variant. Finally, most existing viewers (a notable exception being TargetSeqView; Halper-Stromberg 2014) show all read data in the vicinity of a putative structural variant, making it difficult to discriminate reads supporting the SV, reads supporting the reference allele, and reads that are not relevant to an SV.

To overcome these limitations, we present svviz, a read visualizer for structural variants that sorts and displays only reads relevant to the current SV. As with IGV, svviz only visualizes variants and does not identify them. svviz runs locally on a standard OS X or Linux desktop machine, and requires as input read data, a reference genome, and structural variants. The flexible approach employed by svviz means it can display arbitrary SV types such as translocations, deletions and insertions, inversions and mobile element insertions. Visualizations are rendered in SVG (scalable vector graphics), an open web standard graphics format, and shown in a locally-hosted interactive web browser viewer or exported in publication-ready form. svviz supports read data in BAM format from any sequencing platform, including short-read [Illumina (Bentley 2008)] single- and paired-end as well as mate-pair or longer read [Pacific Biosciences (Eid 2009), Oxford Nanopore, or Illumina's synthetic long-reads] sequencing technologies. In batch mode, multiple SVs can be provided as input in the standard VCF file format, producing summary statistics and PDF or SVG visualizations for hundreds or thousands of SVs with a single command. Annotations such as gene models or repeats can be shown relative to each allele.

2 METHODS

svviz performs several pre-processing steps before visualizing a particular structural variant:

(1) Breakpoints for the input SV are processed to produce a representation of the unique genomic sequence of the SV.

(2) Reads are identified that map near all SV breakpoints.

(3) For paired end data, read mates are collected.

(4) Reads are realigned both to the alternate (SV) allele and to the reference allele by Smith-Waterman alignment (Zhao 2013).

(5) Reads are assigned to the reference or alternate allele if they better support one allele over the other; otherwise, they are labeled as ambiguous. The criteria for this step are described below.

(6) Reference and alternate alleles are visualized separately with individual tracks for each input sample for each allele. Ambiguous reads, typically mapping near but outside of the breakpoints, can also be visualized in a third set of tracks.

Realigned reads are assigned to the allele with the higher alignment score, or (if scores are identical), the allele with the better match to the empirical insert size distribution (derived from the input BAM file). Reads that cannot confidently be assigned to one allele or the other are instead marked as

*To whom correspondence should be addressed.

ambiguous, for example when read-pair orientations are incorrect or the alignment score is below that expected for the given sequencing platform.

This process extends the approach adopted in TargetSeqView (Halper-Stromberg 2014), enabling allele assignment of arbitrary length reads (eg, those produced by long-read technologies) and taking advantage of the insert size distribution, which can be more informative than the alignment score.

3 RESULTS

The Genome in a Bottle consortium (Zook 2014) has recently begun sequencing an Ashkenazi Jewish trio from the Personal Genome Project with a number of high-throughput sequencing platforms, providing a rich resource for identifying and validating variants using orthogonal experimental methods. Structural variants were called from Complete Genomics data for mother, father and son separately. From these variant calls, we randomly chose an 11.5kb inversion on chromosome 4 to visualize and validate using long mate-pair Illumina data.

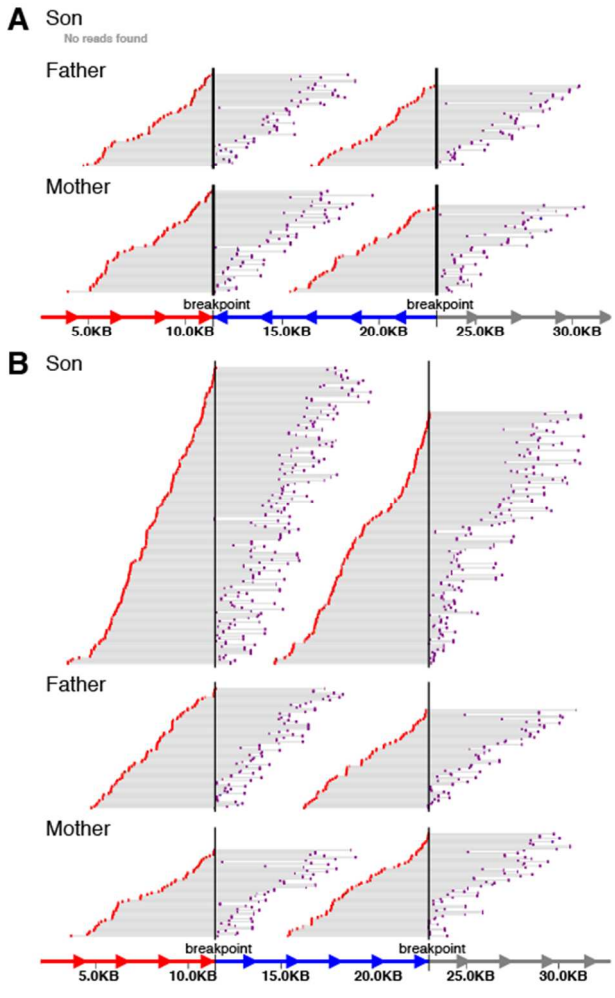

Fig. 1. svviz visualization of a chromosome 4 inversion. (a) Reads supporting the alternate allele in three individuals. Blue region with arrows pointing to the left demarcate the inverted region. (b) Reads supporting the reference allele, with non-inverted region again shown in blue at bottom (but arrows pointing to the right). Red reads are on the minus strand and purple reads are on the plus strand, with gray lines linking mate-pairs (note that the mate-pair data shown here are sequenced in −/+ orientation, and have an average insert size of ~6.5kb). Ambiguous reads, those unable to distinguish between the alleles, are not shown.

The visual representation is split into two sections. The top section (Figure 1a) shows reads supporting the alternate "inversion" allele, while the bottom section (Figure 1b) shows those supporting the reference allele. Each read is shown only once, relative to its assigned allele.

For the alternate allele, mate-pair reads tile across the breakpoints in both parents, while no reads were found in the son. All three individuals show ample coverage of the reference allele breakpoints. The number of reads assigned by svviz to each allele suggests the son is homozygous reference and both parents are heterozygous for the inversion. Figure S1 shows the same data represented in IGV, with likely non-reference reads colored maroon and blue, suggestive of a structural variant but difficult to interpret as an inversion.

Additional visualization examples are shown in the supplement: a putative 1200bp deletion for which svviz shows very little supporting evidence, and which we thus estimate is a false-positive (Figure S2); an inversion with PacBio reads spanning both breakpoints (Figure S3); svviz being used to refine imprecisely-called breakpoints (Figure S4); a mobile-element insertion (Figure S5); a fusion gene present in a cancer sample but not the matched normal sample (Figure S6); a heterozygous deletion with reads in flanking regions shown to demonstrate the reduction in read coverage within the deletion (Figure S7); and a screenshot of the web view, zoomed in to show a SNP present only in the alternate allele (Figure S8).

4 USAGE

svviz can be installed on OS X and linux using the single command "`sudo pip install svviz`" (requires python and pip to be installed; see the online documentation at `http://svviz.github.io/` for detailed installation instructions). It takes approximately 10–30 seconds to analyze and visualize a single variant in a single sample on a 2014 Mac Pro. The number of samples that can be visualized simultaneously is limited only by practical concerns; processing time scales with the number of samples and reads as well as the size of the variant and lengths of the reads.

The inversion shown in Figure 1 can be visualized using the command "`svviz demo`" and the mobile-element insertion in Figure S5 can be visualized using the command "`svviz demo 2`".

ACKNOWLEDGMENTS

We thank the Genome in a Bottle Consortium and Complete Genomics for making the data publicly available. Certain commercial equipment, instruments, or materials are identified in this document. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose.

REFERENCES

Bentley, D.R. et al. (2008). Accurate whole human genome sequencing using reversible terminator chemistry. Nature *456*, 53–59.

Edmonson, M.N. et al. (2011). Bambino: a variant detector and alignment viewer for next-generation sequencing data in the SAM/BAM format. Bioinformatics *27*, 865–866.

Eid, J. et al. (2009). Real-time DNA sequencing from single polymerase molecules. Science *323*, 133–138.

Gymrek, M. (2014). PyBamView: a browser-based application for viewing short read alignments. Bioinformatics *30*, 3405–3407.

Halper-Stromberg, E. et al. (2014). Visualization and probability-based scoring of structural variants within repetitive sequences. Bioinformatics *30*, 1514–1521.

Robinson, J.T. et al. (2011). Integrative genomics viewer. Nat. Biotechnol. *29*, 24–26.

Zhao, M. et al. (2013). SSW library: an SIMD Smith-Waterman C/C++ library for use in genomic applications. PLoS ONE *8*, e82138.

Zook, J.M. et al. (2014). Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. Nat. Biotechnol. *32*, 246–251.