NAR Genomics and Bioinformatics, 2024, **6**, lqae083
https://doi.org/10.1093/nargab/lqae083
Advance access publication date: 9 July 2024
Application Notes

OXFORD

SCI-VCF: a cross-platform GUI solution to summarize, compare, inspect and visualize the variant call format

Venkatesh Kamaraj [1,2,*] **and Himanshu Sinha** [1,2,3,4,*]

[1]Centre for Integrative Biology and Systems Medicine (IBSE), IIT Madras, Chennai 600036, Tamil Nadu, India
[2]Robert Bosch Centre for Data Science and Artificial Intelligence (RBCDSAI), IIT Madras, Chennai 600036, Tamil Nadu, India
[3]Wadhwani School of Data Science and Artificial Intelligence, IIT Madras, Chennai 600036, Tamil Nadu, India
[4]Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, IIT Madras, Chennai 600036, Tamil Nadu, India

[*]To whom correspondence should be addressed. Email: sinha@iitm.ac.in
Correspondence may also be addressed to Venkatesh Kamaraj. Email: vengatesh.vengatesh89@gmail.com

Abstract

As genomics advances swiftly and its applications extend to diverse fields, bioinformatics tools must enable researchers and clinicians to work with genomic data irrespective of their programming expertise. We developed SCI-VCF, a Shiny-based comprehensive analysis utility to summarize, compare, inspect, analyse and design interactive visualizations of the genetic variants from the variant call format. With an intuitive graphical user interface, SCI-VCF aims to bridge the approachability gap in genomics that arises from the existing predominantly command-line utilities. SCI-VCF is written in R and is freely available at https://doi.org/10.5281/zenodo.11453080. For installation-free access, users can avail themselves of an online version at https://ibse.shinyapps.io/sci-vcf-online.

Introduction

Genomics is a rapidly evolving field that has profoundly impacted biomedical and public health research. With the advent of next-generation sequencing (NGS), an increasing number of genomes from various species are being sequenced, enabling the applications of genomics to proliferate into diverse disciplines such as medicine (1), agriculture (2), microbiology (3), ecology (4), evolutionary biology (5), psychology (6) and anthropology (7). This is accelerated further by combining genomics with advancements in data science and artificial intelligence (8). Such interoperability is facilitated by well-defined data formats that represent diverse biological entities. One such data structure is the variant call format (VCF).

VCF is a standard file format to store the sequence variations in a genome such as single nucleotide polymorphisms (SNPs), insertions, deletions (INDELs), structural variants (SVs) and other assorted variants (9). It is a tab-delimited text file, with each row comprising information about a genomic variant site and metadata of these variants presented at the beginning of the file. The first eight columns of the headers are fixed and are named CHROM, POS, ID, REF, ALT, QUAL, FILTER and INFO. Figure 1A depicts the structure and contents of a generic VCF file. It is a widely adopted file format used in many bioinformatics tools that analyse genomic data. While the VCF has many advantages for storing and sharing genomic variants efficiently, it can be complex and arduous to understand, particularly for those unaccustomed to bioinformatics and genomic data analysis.

Several tools allow users to examine, analyse and interpret VCF files. However, they are predominantly command-line utilities and require significant programming expertise from the user. Data visualization is a ubiquitous method for quickly understanding complex information, but tools that visualize a VCF file are scarce. VIVA (10), vcfR (11) and vcflib (12) are some existing frameworks that allow the visualization of a VCF file. However, effectively using these variant visualization tools also requires proficiency in programming by the user.

As the applications of genomics burgeon into diverse fields, it calls for tools and software that empower clinicians and researchers to work with genomic data formats irrespective of their programming expertise. Here, we introduce SCI-VCF, a comprehensive toolkit with an intuitive graphical user interface (GUI) that lets users summarize, interpret and compare genomic variants from VCF files. It also equips users to design interactive visualizations of the VCF file in numerous ways. SCI-VCF is platform-agnostic and works seamlessly across any operating system. A web version of the tool is also made available for enhanced accessibility. While it is not feasible for a single GUI-based tool to support the wide-spanning analytical capabilities of the VCF file, SCI-VCF provides a well-founded framework that simplifies the core components of VCF analyses, thus increasing the approachability of genomics to novices.

Materials and methods

SCI-VCF is written in R, a free and open-source programming language for statistical computing, data analysis and visualization. The user interface and the routines are designed and developed in Shiny, an R framework to build interactive applications. VCF files are parsed in SCI-VCF using the *vcfR* library and are processed using the *tidyverse* packages. The static visualizations of the VCF files are developed with *ggplot2* (13), which operates on a grammatical theory of graphics. The static plots are then transformed into interactive

Received: March 7, 2024. **Revised:** June 3, 2024. **Editorial Decision:** June 24, 2024. **Accepted:** July 1, 2024

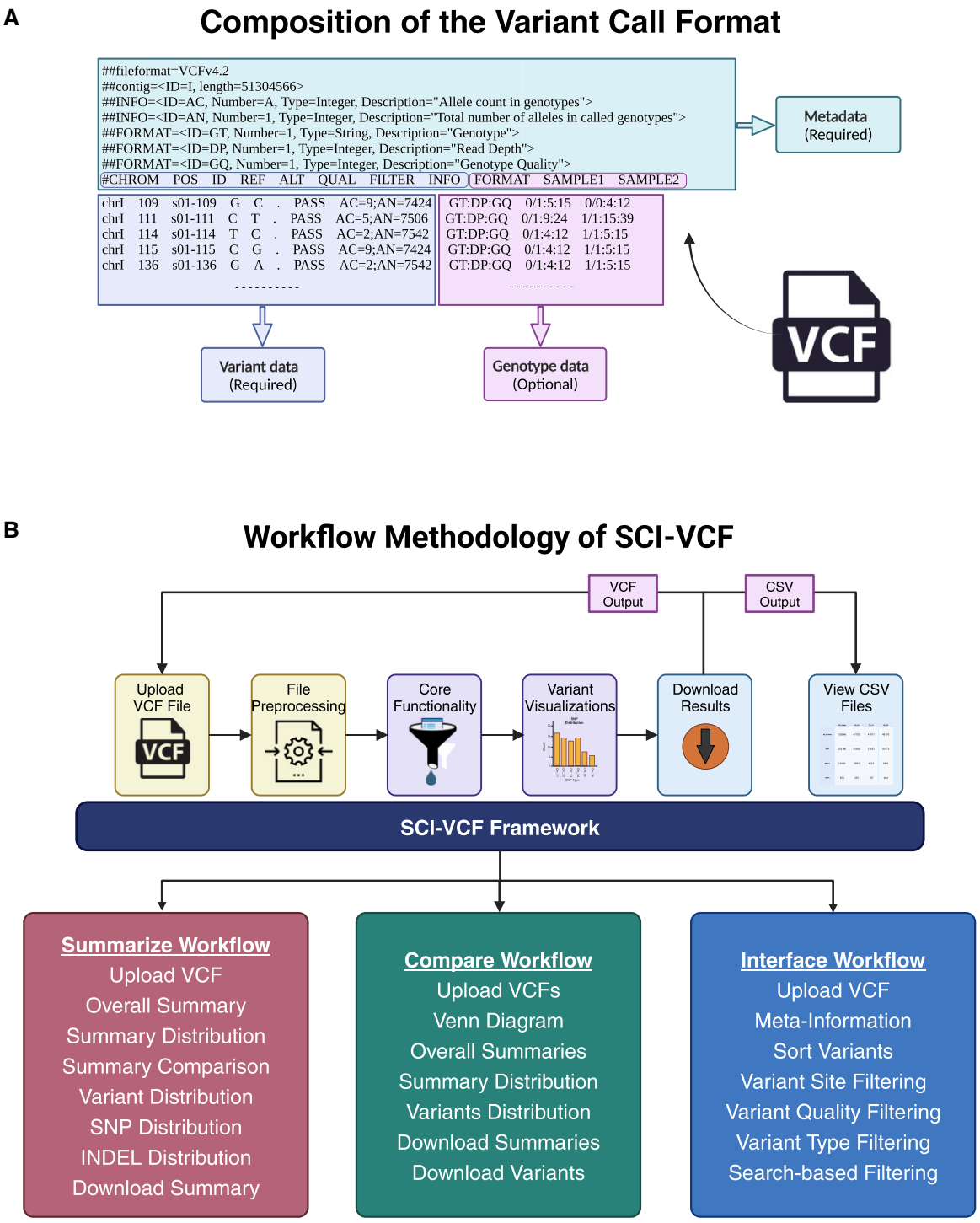

Figure 1. VCF analysis workflow in SCI-VCF. (**A**) Composition of the standard VCF file. Metadata and the information of variants in the first eight columns are mandatory. Genotype information is optional and can be present across multiple columns. (**B**) The workflow methodology of SCI-VCF describes the core functionalities offered by the tool. Created with Biorender.com

visualizations using *Plotly* for R. The area-proportional Venn diagrams are made with the *eulerr* package, and the interactive data tables enabling detailed inspection of VCF files are created using the *reactable* library.

SCI-VCF has three major workflows separated into respective modules in the navigation bar: Summarize, Compare and Interface. Submodules capturing the result of individual analyses are assembled under the major module's sidebars. Figure 1B elucidates the major workflows of the tool. Both compressed and uncompressed VCF files are accepted as valid inputs in all three major modules. A module to inspect comma-separated values (CSV) files has also been added to provide a tabular variant-level analysis system. Every plot generated can be customized to equip the user to change various aspects of the visualization. All the data and visualizations generated by the tool are downloadable for further examination. Sam-

ple VCF files are provided to get started with the tool from the ground up.

Results

SCI-VCF provides summary statistics of a VCF file in a graphical format

The first four columns of a VCF file can be used to identify the variants present in a normalized VCF file uniquely. Normalizing is achieved by breaking down comma-separated multiallelic sites into individual entries and removing duplicates. The nature of REF and ALT entries enabled the classification of variants into different variant types: SNPs, transitions, transversions, INDELs, insertions and deletions individually, and other multinucleotide polymorphisms, multiallelic sites and assorted variants that do not fit the other categories. The summary of a VCF file was generated by classifying variants and summing up unique entries in each category.

Statistics derived from this summary, such as the ratio of transitions to transversions (Ts/Tv), are generally used as quality control metrics to assess the confidence in the variants captured by the variant calling algorithm (14). The occurrence of variants throughout the genome is presented as an interactive histogram to recognize regions that are depleted or enriched with genetic variations. Such a plot aids in identifying the highly polymorphic and invariant sections of the genome, possibilities of selection pressures and genomic loci linked to fundamental biological functions and salient clinical features. With the distribution of INDEL sizes captured by SCI-VCF, it is possible to gauge the prevalence of frameshift mutations and structural variants. The spread of each variant type across different chromosomes is represented in diverse plots. By illustrating the synopsis and distribution of specific variant types across the genome, SCI-VCF helps researchers gain valuable insights into the genetic diversity, evolutionary history and disease susceptibility of an individual, a population or a species.

Comparison of a pair of VCF files to understand genetic diversity

Two VCF files were compared by interpreting the first eight columns of the files as two-dimensional heterogeneous tabular datasets. The area-proportional Venn diagram created assists in a quick overview of the commonalities, dissimilarities and relationships between the two variant calls. The shared and unique variant sets between the two files are summarized and visualized in multiple ways to advance further investigation by the user. Comparing VCF files and recapitulating the resulting variant sets enables the comprehension of the genetic diversity across individuals and populations, and aids in gaining crucial insights about them. For example, novel variants with potential clinical and biological significance could be uncovered by comparing the VCF files of an individual or population with a reference database such as the 1000 Genomes project (1KGP) (15) or the gnomAD dataset (16). Variant calling pipelines could be validated for consistency and accuracy by comparing their results with available high-confidence variant sets for thoroughly studied samples such as the Genome In A Bottle (GIAB) benchmark sets (17).

The genetic basis of various diseases is understood by juxtaposing the variants in individuals with and without a particular disease. This procedure helps identify the mutations associated with the condition. A similar approach is also used to study genotype–phenotype association and pin down genetic markers for phenotypic traits. SCI-VCF equips users with the framework to effortlessly compare VCF files and summarize the overlapping and dissimilar variants between individuals and populations. These comparisons and summaries are pivotal for understanding genetics, unravelling disease mechanisms, developing targeted interventions and advancing genomic research.

SCI-VCF provides in-depth visualization of the contents of a VCF file

SCI-VCF offers a framework to view, search, sort, identify and filter the contents of a VCF file. Entries in a file are filterable in terms of keywords, variant sites, quality scores and variant types. Variant sorting and quality filtering are standard analyses when dealing with VCF files, with quality filtering aiding in removing the low-quality variants that might have resulted from sequencing errors or mapping artefacts. The variant site and type filtering enable the prioritization of variants and help focus the analysis on specific regions of interest, such as genes, exons or regulatory elements. The keyword search was extended to each VCF column, enabling sophisticated filtration capabilities, including annotation-based variant extraction. These variant inspections can be beneficial when studying variants in candidate genes or genomic regions associated with a particular phenotype or disease.

The meta-information from VCF files is extracted and displayed in a searchable fashion to provide additional context about the variants, such as the reference genome used, databases used for variant annotation, descriptions of those annotations and possible pre-processing steps done with the VCF file. The filtered variants and the meta-information can be downloaded by the user for further study. While the results obtained in the VCF file type can then be summarized and compared using the respective modules in SCI-VCF, the results downloaded in a CSV file type can be inspected in-depth in the 'View CSV Files' module.

All data visualizations in SCI-VCF are presented as interactive plots to enhance exploratory data analysis with VCF files. With the help of features such as input selection, zooming, panning and tooltips, researchers can explore different aspects and dive deeper into specific areas of interest to uncover patterns, outliers and relationships that may not be apparent in static plots. All graphics made with the tool are supplemented with plot customization features to improve the effectiveness, clarity and visual impact of the data visualizations. As points of interest from the displayed interactive plots can be customized and saved locally, users can extract publication-ready visualizations from SCI-VCF.

Use case for SCI-VCF application

To demonstrate its utility, we performed a case study only using SCI-VCF. Figure 2 displays the visualizations taken directly from SCI-VCF at various stages of this analysis. The input for the study contained variants captured by the BWA-GATK pipeline in the whole-genome sequence (WGS) of the GIAB sample HG002. A total of ∼4.78 million variants were present in the file. Figure 2A shows the summary of the entries present in the file. Nearly 80% of the variants in the file were SNPs, with a Ts/Tv of 2.02, which was the expected value for the human WGS variants. By examining the INDEL size distribution plot, it was evident that no structural variants were

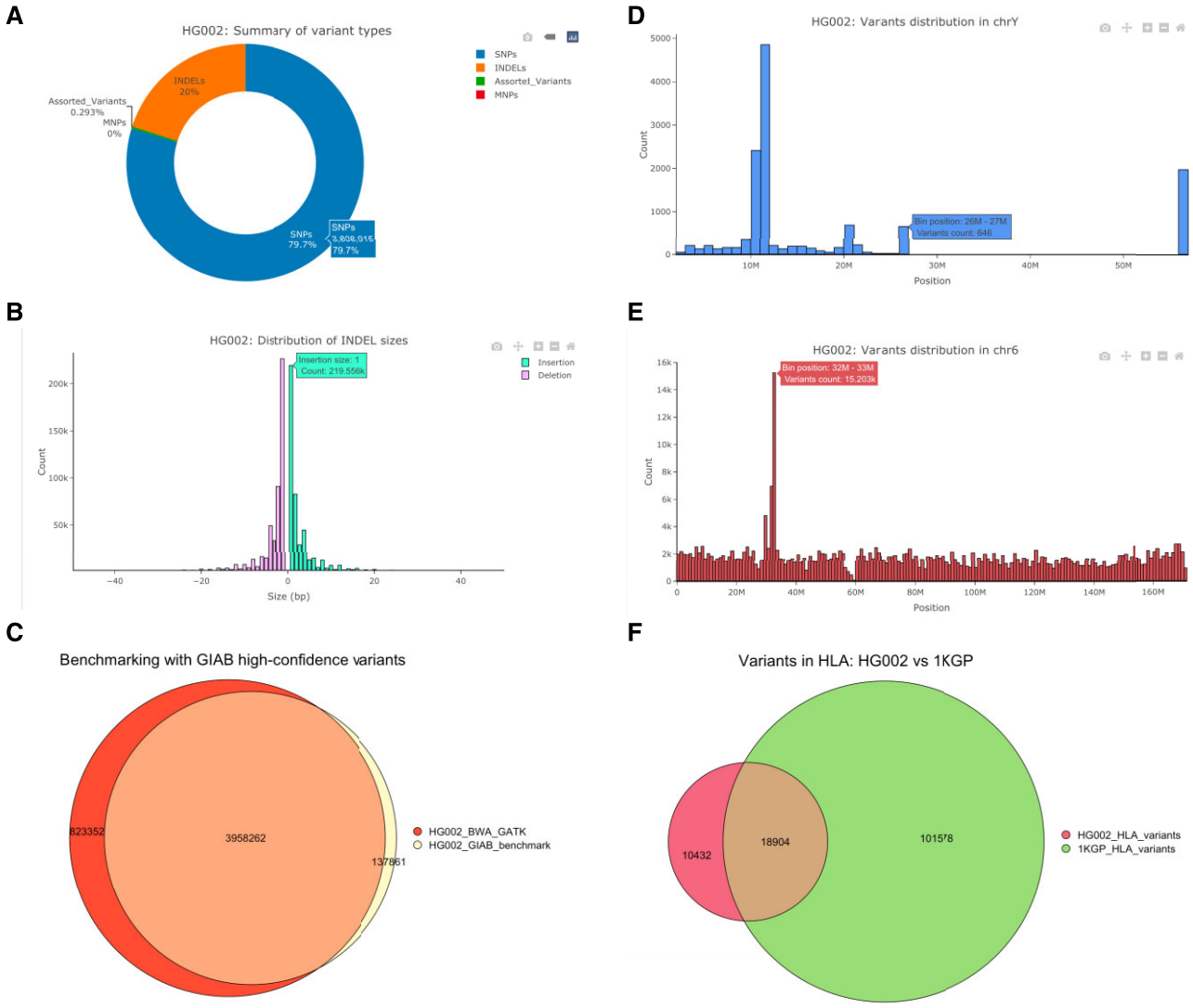

Figure 2. Analyses of variants in HG002 using SCI-VCF. (**A**) Distribution of variant types; (**B**) size distribution of INDELs; (**C**) comparison of HG002 variants with the GIAB high-confidence variants; (**D**) distribution of variants in chrY; (**E**) distribution of variants in chr6; (**F**) comparison of variants in the HLA region with the same from the 1000 genomes project.

present in the file. This result was anticipated as the variant calling workflow was not explicitly designed to capture structural variants. The maximum size of an insertion was 456 bp, and deletion was 339 bp, with most INDELs biased towards smaller sizes. The functionalities to interact with the visualizations were seamlessly integrated within SCI-VCF, and the same was used to zoom in on the INDEL size distribution plot to limit the sizes to 50 bp on both axes in Figure 2B.

To validate the variant calling pipeline used to create the VCF file, we benchmarked its output with the high-confidence VCF file provided by the GIAB consortium. For this task, we used the 'Compare' workflow, and the results in the form of a Venn diagram are depicted in Figure 2C. The pipeline captured nearly 96.6% of the high-confidence variants released by GIAB, ensuring its credibility. Further study of the position-level distribution of variants revealed that chrY had a large genomic region, nearly 30 Mbp, with no variants. Meanwhile, the 32–33 Mbp genomic locus in chr6 contained >15 000 variants, the maximum observed value for any 1 Mbp genomic window. Figure 2D and E depicted the distribution of variants in chrY and chr6, respectively. Upon cross-referencing with the literature, we found that the outlier 1 Mbp window in chr6 lay in the human leukocyte antigen (HLA), a super-locus responsible for regulating the immune system, previously known to be highly polymorphic (18).

The HLA region in the human genome spanned from genomic coordinates 29602228 to 33410226 on chr6 of the human genome reference assembly GRCh38. Using the 'Interface' module, we confirmed from the meta-information of the VCF file that the variants were called concerning GRCh38 and proceeded to filter the variants in the HLA region. Summarizing the filtered variants, we observed that 29 336 variants were present in the HLA region. We analysed further to identify if these variants were previously known by comparing the filtered VCF file with the 1KGP variants. To this end, we filtered the HLA variants from the entries in chr6 of the 1KGP variants and compared them with the HLA variants in HG002. Of the 4.86 million 1KGP variants reported in chr6, 120 482 variants corresponded to the HLA region. From Figure 2F, we observe that 35.6% of the variants called in the HLA region in HG002 were novel to the 1KGP variants. Further analyses using other bioinformatics methods and

Table 1. Comparison of features of VCF analysis tools

Features	SCI-VCF	VIVA	vcflib	vcfR	CuteVariant	re-Searcher	VCF-Miner	VCFtools	GEMINI
TECHNICAL									
Language	R	Julia	C++	R	Python	Python	Java	C++, Perl	Python
Environment (OS)	Windows, Mac, Linux	Windows, Mac, Linux	Windows, Mac, Linux	Windows, Mac, Linux	Windows, Mac, Linux	Windows, Mac, Linux	Windows	Windows, Mac, Linux	Windows, Mac, Linux
Graphical user interface	✓				✓	✓	✓		
Online version	✓					✓			✓
SUMMARIZE									
Variant level summary	✓		✓	✓*				✓	
Sample level summary		✓	✓	✓*				✓*	
COMPARE									
Variant level comparison	✓		✓	✓*				✓	
Unique/ intersection set summary	✓		✓*	✓*				✓*	
INSPECT									
Variant-based filtering	✓	✓	✓	✓	✓	✓	✓	✓	✓
Sample-based filtering		✓	✓	✓	✓	✓	✓	✓	✓
VISUALIZE									
Interactive plots	✓	✓		✓*					
Visualize genomic regions	✓	✓		✓*					
OUTPUT									
Publication quality graphics	✓	✓	✓	✓*					
Export results as tabular data	✓	✓	✓*	✓*	✓	✓*	✓	✓	✓

*Multistep process involving other tools/libraries

tools are required to understand the significance of these novel variants.

SCI-VCF adds features to the existing suite of open-source tools

We compared the main features in SCI-VCF with the existing open-source tools VIVA (10), vcfR (11), vcflib (12), CuteVariant (19), re-Searcher (20), VCF-Miner (21), VCFtools (9) and GEMINI (22). The results are detailed in Table 1. Apart from SCI-VCF, CuteVariant, VCF-Miner and re-Searcher provided GUIs, while only re-Searcher and GEMINI provided installation-free online access. Visualization of genetic variants could be handled effectively by only SCI-VCF and VIVA. SCI-VCF is the only tool equipped with a GUI that can summarize, compare and visualize genetic variants in a single platform. We have designed SCI-VCF to make genomic data analyses more open and accessible to all researchers, irrespective of their programming expertise, by simplifying the essential components for VCF-based genetic variant studies. In tasks such as sample filtering and variant annotations where SCI-VCF cannot perform optimally, other existing tools execute well, and the resulting VCF files are back-compatible, making their analysis within SCI-VCF feasible. So, SCI-VCF is a valuable augmentation to the current suite of tools rather than a substi-

tute. It complements and enhances existing tools, making genomics more accessible and beneficial for non-programmers.

Discussion

We have developed SCI-VCF as a cross-platform application to summarize, compare, inspect and visualize the genetic variants from the widely accepted VCF files. Researchers and clinicians can use the guided GUI setting of SCI-VCF to perform exploratory genomic data analysis on VCF files, irrespective of their programming expertise. The user-friendly and intuitive design of SCI-VCF will increase the approachability of genomics to newcomers and introduce genomic data analysis expeditiously. We show a case study to illustrate the utility of SCI-VCF. While the use cases pertained to the human genome, the tool is not specific and can be generalized to variants in any organism. We also compared the features in SCI-VCF with other existing VCF analysis software.

Ongoing developments are in progress to enhance the utility of the current version of SCI-VCF. While SCI-VCF offers a wide range of functionalities, we seek to refine its capacity for advanced variant annotation analyses and allele frequency-based filtering. Despite its capability to seamlessly handle VCF files with multiple samples, SCI-VCF does not incorporate a merge function to combine variants from multiple samples. These advanced tasks are resource-intensive, and their incorporation would make maintaining the online version of SCI-VCF expensive. In future versions of SCI-VCF, we aim to expand the framework of SCI-VCF to support the resource-efficient pre-processing of VCF files to add these advanced features successfully. We also intend to parse VCF files parallelly instead of sequentially to handle exceptionally large VCF files efficiently.

Software and code availability

The versatility of SCI-VCF makes it suitable for local installation, allowing personal use and server deployment permitting communal use. SCI-VCF can be used without any installation as an online tool at https://ibse.shinyapps.io/sci-vcf-online/. It can also be installed in diverse ways according to the user's requirements. As the online version of the tool has resource constraints, the maximum file upload size is limited to 10 MB. A local installation is preferable to work with larger VCF files. With R (version $\geq$ 4.2.1) installed, local installation of SCI-VCF is practicable as the compatible dependencies are downloaded automatically upon first use. By default, the offline version can handle files up to 1 GB, but this limit can be altered if required.

For better package management, a Conda virtual environment with the dependencies of SCI-VCF was created, which can be reproduced easily for improved environment handling. A containerized form of the application is available as a Docker image that helps run SCI-VCF irrespective of platforms, making it straightforward to port the tool to remote HPC clusters. Users can download and install SCI-VCF using the code available on Zenodo (https://doi.org/10.5281/zenodo.11453080) and GitHub (https://github.com/HimanshuLab/SCI-VCF). A well-documented guide is available at https://himanshulab.github.io/SCI-VCF-docs/ for improved user support and understanding of the system.

All the analysis in the case study section of the manuscript was done using a system with an Apple M1 processor and 8 GB of RAM. The VCF files used in this section are available at https://doi.org/10.5281/zenodo.10780916. The variants were called in the GIAB HG002 sample using the BWA-GATK pipeline at https://github.com/IBSE-IITM/genome-analysis-pipeline.

Acknowledgements

We acknowledge Ayam Gupta for creating the VCF file used in the case study section. We thank Veerendra Gadekar and Preetha Ravi for suggesting features in the tool and proofreading the manuscript. We are grateful to the Centre for Integrative Biology and Systems Medicine (IBSE) and Robert Bosch Centre for Data Science and Artificial Intelligence (RBCDSAI) members for discussions and comments.

Funding

The Department of Biotechnology, Govt. of India [BT/GenomeIndia/2018] and the Centre for Integrative Biology and Systems Medicine, IIT Madras [BIO/18-19/304/ALUM/KARH] to HS.

Conflict of interest statement

None declared.

References

1. Hudson,K.L. (2011) Genomics, health care, and society. *N. Engl. J. Med.*, **365**, 1033–1041.
2. Wang,W., Cao,X.H., Miclăus,M., Xu,J. and Xiong,W. (2017) The promise of agriculture genomics. *Int. J. Genomics*, **2017**, e9743749.
3. Padmanabhan,R., Mishra,A.K., Raoult,D. and Fournier,P.-E. (2013) Genomics and metagenomics in medical microbiology. *J. Microbiol. Methods*, **95**, 415–424.
4. Klaper,R. and Thomas,M.A. (2004) At the crossroads of genomics and ecology: the promise of a canary on a chip. *Bioscience*, **54**, 403–412.
5. Rokas,A. and Abbot,P. (2009) Harnessing genomics for evolutionary insights. *Trends Ecol. Evol.*, **24**, 192–200.
6. Hoehe,M.R. and Morris-Rosendahl,D.J. (2018) The role of genetics and genomics in clinical psychiatry. *Dialogues Clin. Neurosci.*, **20**, 169–177.
7. Benn Torres,J. (2020) Anthropological perspectives on genomic data, genetic ancestry, and race. *Am. J. Phys. Anthropol.*, **171**, 74–86.
8. Navarro,F.C.P., Mohsen,H., Yan,C., Li,S., Gu,M., Meyerson,W. and Gerstein,M. (2019) Genomics and data science: an application within an umbrella. *Genome Biol.*, **20**, 109.
9. Danecek,P., Auton,A., Abecasis,G., Albers,C.A., Banks,E., DePristo,M.A., Handsaker,R.E., Lunter,G., Marth,G.T., Sherry,S.T., *et al.* (2011) The variant call format and VCFtools. *Bioinformatics*, **27**, 2156–2158.
10. Tollefson,G.A., Schuster,J., Gelin,F., Agudelo,A., Ragavendran,A., Restrepo,I., Stey,P., Padbury,J. and Uzun,A. (2019) VIVA (VIsualization of VAriants): a VCF file visualization tool. *Sci. Rep.*, **9**, 12648.
11. Knaus,B.J. and Grünwald,N.J. (2017) vcfr: a package to manipulate and visualize variant call format data in R. *Mol. Ecol. Resour.*, **17**, 44–53.
12. Garrison,E., Kronenberg,Z.N., Dawson,E.T., Pedersen,B.S. and Prins,P. (2022) A spectrum of free software tools for processing the VCF variant call format: vcflib, bio-vcf, cyvcf2, hts-nim and slivar. *PLoS Comput. Biol.*, **18**, e1009123.

13. Wickham,H. (2011) ggplot2. *WIREs Comput. Stat.*, **3**, 180–185.
14. Wang,J., Raskin,L., Samuels,D.C., Shyr,Y. and Guo,Y. (2015) Genome measures used for quality control are dependent on gene function and ancestry. *Bioinformatics*, **31**, 318–323.
15. Auton,A., Abecasis,G.R., Altshuler,D.M., Durbin,R.M., Abecasis,G.R., Bentley,D.R., Chakravarti,A., Clark,A.G., Donnelly,P., Eichler,E.E., *et al.* (2015) A global reference for human genetic variation. *Nature*, **526**, 68–74.
16. Karczewski,K.J., Francioli,L.C., Tiao,G., Cummings,B.B., Alföldi,J., Wang,Q., Collins,R.L., Laricchia,K.M., Ganna,A., Birnbaum,D.P., *et al.* (2020) The mutational constraint spectrum quantified from variation in 141,456 humans. *Nature*, **581**, 434–443.
17. Zook,J.M., Catoe,D., McDaniel,J., Vang,L., Spies,N., Sidow,A., Weng,Z., Liu,Y., Mason,C.E., Alexander,N., *et al.* (2016) Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Sci. Data*, **3**, 160025.
18. Kulski,J.K., Suzuki,S. and Shiina,T. (2022) Human leukocyte antigen super-locus: nexus of genomic supergenes, SNPs, indels, transcripts, and haplotypes. *Hum. Genome Var.*, **9**, 49.
19. Schutz,S., Monod-Broca,C., Bourneuf,L., Marijon,P. and Montier,T. (2022) Cutevariant: a standalone GUI-based desktop application to explore genetic variations from an annotated VCF file. *Bioinform. Adv.*, **2**, vbab028.
20. Karabayev,D., Molkenov,A., Yerulanuly,K., Kabimoldayev,I., Daniyarov,A., Sharip,A., Ashenova,A., Zhumadilov,Z. and Kairov,U. (2021) re-searcher: gUI-based bioinformatics tool for simplified genomics data mining of VCF files. *PeerJ*, **9**, e11333.
21. Hart,S.N., Duffy,P., Quest,D.J., Hossain,A., Meiners,M.A. and Kocher,J.-P. (2016) VCF-Miner: gUI-based application for mining variants and annotations stored in VCF files. *Brief. Bioinform.*, **17**, 346–351.
22. Paila,U., Chapman,B.A., Kirchner,R. and Quinlan,A.R. (2013) GEMINI: integrative exploration of genetic variation and genome annotations. *PLoS Comput. Biol.*, **9**, e1003153.

Received: March 7, 2024. **Revised:** June 3, 2024. **Editorial Decision:** June 24, 2024. **Accepted:** July 1, 2024