SCI-VCF: A cross-platform GUI solution to Summarise, Compare, Inspect, and Visualise the Variant Call Format

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

[1]Centre for Integrative Biology and Systems Medicine (IBSE), IIT Madras, Chennai, India

[2]Robert Bosch Centre for Data Science and Artificial Intelligence (RBCDSAI), IIT Madras, Chennai, India

[3]Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, IIT Madras, Chennai, India

[*]Corresponding authors

Keywords:

Genetic Variants, Variant Call Format, Genomic Data Analysis, Visualisation, Online/Offline Tool

Subject Areas:

Bioinformatics, Genomics, Molecular Biology, Medical Genetics, Software Development

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 summarise, compare, inspect, analyse, and design interactive visualisations of the genetic variants from the variant call format. With an intuitive GUI, SCI-VCF aims to bridge the approachability gap in genomics that arises from the existing predominantly command-line utilities. SCI-VCF is written with R and is freely available at https://github.com/HimanshuLab/SCI-VCF. For installation-free access, users can avail 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 getting sequenced, enabling the applications of genomics to proliferate into diverse disciplines such as medicine (Hudson, 2011), agriculture (Wang et al., 2017), microbiology (Padmanabhan et al., 2013), ecology (Klaper and Thomas, 2004), evolutionary biology (Rokas and Abbot, 2009), psychology (Hoehe and Morris-Rosendahl, 2018), and anthropology (Benn Torres, 2020). This is accelerated further by combining genomics with advancements in data science and artificial intelligence (Navarro et al., 2019). 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 like single nucleotide polymorphisms (SNP), insertions, deletions (INDEL), structural variants (SV), and other assorted variants (Danecek et al., 2011). 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 visualisation is a ubiquitous method for quickly understanding complex information, but tools that visualise a VCF file are scarce. VIVA (Tollefson et al., 2019), vcfR (Knaus and Grünwald, 2017), and the vcflib (Garrison et al., 2022) are some existing frameworks that allow the visualisation of VCF. However, effectively using these variant visualisation tools also requires proficiency in programming on the user's end.

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 summarise, interpret, and compare genomic variants from VCF files. It also equips users to design interactive visualisations of the VCF 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 infeasible for a single GUI-based tool to support the wide-spanning analytic capabilities of the VCF, SCI-VCF provides a well-founded framework that simplifies the core components of VCF analyses, thus increasing the approachability of genomics to novices.

METHODS

SCI-VCF is written in R, a free and open-source programming language for statistical computing, data analysis, and visualisation. 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 visualisations of the VCF files are developed with *ggplot2* (Wickham, 2011), which operates on a grammatical theory of graphics. The static plots are then transformed into interactive visualisations 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: Summarise, 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 is also added to provide a tabular variant-level analysis system. Every plot generated can be customised to equip the user to change various aspects of the visualisation. All the data and visualisations generated by the tool are downloadable for further examination. Sample VCF files are provided to get started with the tool from the ground up.

RESULTS AND DISCUSSION

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 normalised VCF file uniquely. Normalising 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 Multi Nucleotide Polymorphisms, multi-allelic 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, like the ratio of transitions to transversions, are generally used as quality control metrics to assess the confidence in the variants captured by the variant calling algorithm (Wang et al., 2015). The occurrence of variants throughout the genome is presented as an interactive histogram to recognise 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 VCFs 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 summarised and visualised in multiple ways to advance further investigation by the user. Comparing VCFs and recapitulating

110 the resulting variant sets enables the comprehension of the genetic diversity across individuals
111 and populations and aids in gaining crucial insights about them. For example, novel variants with
112 potential clinical and biological significance could be uncovered by comparing the VCFs of an
113 individual or population with a reference database like the 1000 Genomes project (1KGP) (Auton
114 et al., 2015) or the gnomAD dataset (Karczewski et al., 2020). Variant calling pipelines could be
115 validated for consistency and accuracy by comparing their results with available high-confidence
116 variant sets for thoroughly studied samples like the Genome In A Bottle (GIAB) benchmark sets
117 (Zook et al., 2016).

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

126 **SCI-VCF provides in-depth visualisation of the contents of a VCF file**

127 SCI-VCF offers a framework to view, search, sort, identify, and filter the contents of a VCF file.
128 Entries in a file are filterable in terms of keywords, variant sites, quality scores and variant types.
129 Variant sorting and quality filtering are standard analyses when dealing with VCF files, quality
130 filtering aiding in removing the low-quality variants that might have resulted from sequencing
131 errors or mapping artefacts. The variant site and type filtering enable the prioritisation of variants
132 and help focus the analysis on specific regions of interest, such as genes, exons, or regulatory
133 elements. The keyword search was extended to each VCF column, enabling sophisticated
134 filtration capabilities, including annotation-based variant extraction. These variant inspections
135 can be beneficial when studying variants in candidate genes or genomic regions associated with
136 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, like 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 summarised and compared using the respective modules in SCI-VCF, the results downloaded in CSV file type can be inspected in-depth in the 'View CSV Files' module.

All data visualisations in SCI-VCF are presented as interactive plots to enhance exploratory data analysis with VCF files. With the help of features like 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 customisation features to improve the effectiveness, clarity, and visual impact of the data visualisations. As points of interest from the displayed interactive plots can be customised and saved locally, users can extract publication-ready visualisations 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 visualisations 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 about 4.78M 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 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 visualisations 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 bps 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 over 15,000 variants, the maximum observed value for any 1 Mbp genomic window. Figures 2D and 2E 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 the regulation of the immune system, previously known to be highly polymorphic (Kulski et al., 2022).

The HLA region in the human genome GRCh38 spanned from genomic coordinates 29,602,228 to 33,410,226 on chr6 of the human genome reference assembly GRCh38. Using the 'Interface' module, we confirmed from the meta-information of the VCF that the variants were called concerning GRCh38 and proceeded to filter the variants in the HLA region. Summarising 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 4.86M 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 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 (Tollefson et al., 2019), vcfR (Knaus and Grünwald, 2017), vcflib (Garrison et al., 2022), CuteVariant (Schutz et al., 2022), re-Searcher (Schutz et al., 2022), VCF-Miner (Hart et al., 2016), VCFtools (Danecek et al., 2011), and GEMINI (Paila et al., 2013). The results are detailed in Table 1. Apart from SCI-

193 VCF, CuteVariant, VCF-Miner, and re-Searcher provided GUIs, while only re-Searcher and GEMINI
194 provided installation-free online access. Visualisation of genetic variants could be handled
195 effectively by only SCI-VCF and VIVA. SCI-VCF is the only tool equipped with a GUI that can
196 summarise, compare and visualise genetic variants in a single platform. We have designed SCI-
197 VCF to make genomic data analyses more open and accessible to all researchers, irrespective of
198 their programming expertise, by simplifying the essential components for VCF-based genetic
199 variant studies. In tasks like sample filtering and variant annotations where SCI-VCF cannot
200 perform optimally, other existing tools execute well, and the resulting VCF files are back-
201 compatible, making their analysis within SCI-VCF feasible. So, SCI-VCF is a valuable augmentation
202 to the current suite of tools rather than a substitute. It complements and enhances existing tools,
203 making genomics more accessible and beneficial for non-programmers.

204 **CONCLUSION**

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

214 Ongoing developments are in progress to enhance the utility of the current version of SCI-VCF.
215 While SCI-VCF offers a wide range of functionalities, we seek to refine its capacity for advanced
216 variant annotation analyses and allele frequency-based filtering. Despite its capability to
217 seamlessly handle VCF files with multiple samples, SCI-VCF does not incorporate a merge function
218 to combine variants from multiple samples. These advanced tasks are resource-intensive, and
219 their incorporation would make maintaining the online version of SCI-VCF expensive. In future
220 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 per the user's requirements. With R installed, local installation of SCI-VCF is practicable as the dependencies are downloaded automatically upon first use. To work with larger VCF files, it is preferable to use a local installation, as the online version contains resource constraints.

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 containerised 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 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 system understanding.

CONFLICT OF INTEREST STATEMENT

The authors declare no competing interests.

SUPPLEMENTARY INFORMATION

All the analysis in the case study section of the manuscript was done using a system with an M1 processor and 8GB 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 containerised at https://hub.docker.com/r/ibse/genome-india.

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 INFORMATION

The work was supported by 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 H.S.

REFERENCES

Auton, A., Abecasis, G.R., Altshuler, D.M., Durbin, R.M., Abecasis, G.R., Bentley, D.R., et al., 2015. A global reference for human genetic variation. Nature 526, 68–74. https://doi.org/10.1038/nature15393

Benn Torres, J., 2020. Anthropological perspectives on genomic data, genetic ancestry, and race. Am. J. Phys. Anthropol. 171, 74–86. https://doi.org/10.1002/ajpa.23979

Danecek, P., Auton, A., Abecasis, G., Albers, C.A., Banks, E., DePristo, M.A., et al., 1000 Genomes Project Analysis Group, 2011. The variant call format and VCFtools. Bioinformatics 27, 2156–2158. https://doi.org/10.1093/bioinformatics/btr330

Garrison, E., Kronenberg, Z.N., Dawson, E.T., Pedersen, B.S., 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. https://doi.org/10.1371/journal.pcbi.1009123

Hoehe, M.R., Morris-Rosendahl, D.J., 2018. The role of genetics and genomics in clinical psychiatry. Dialogues Clin. Neurosci. 20, 169–177.

Hudson, K.L., 2011. Genomics, Health Care, and Society. N. Engl. J. Med. 365, 1033–1041. https://doi.org/10.1056/NEJMra1010517

Karczewski, K.J., Francioli, L.C., Tiao, G., Cummings, B.B., Alföldi, J., Wang, Q., et al., 2020. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature 581, 434–443. https://doi.org/10.1038/s41586-020-2308-7

273 Klaper, R., Thomas, M.A., 2004. At the Crossroads of Genomics and Ecology: The Promise of a Canary on
274 a Chip. BioScience 54, 403–412. https://doi.org/10.1641/0006-
275 3568(2004)054[0403:ATCOGA]2.0.CO;2
276 Knaus, B.J., Grünwald, N.J., 2017. vcfr: a package to manipulate and visualize variant call format data in R.
277 Mol. Ecol. Resour. 17, 44–53. https://doi.org/10.1111/1755-0998.12549
278 Kulski, J.K., Suzuki, S., Shiina, T., 2022. Human leukocyte antigen super-locus: nexus of genomic
279 supergenes, SNPs, indels, transcripts, and haplotypes. Hum. Genome Var. 9, 1–15.
280 https://doi.org/10.1038/s41439-022-00226-5
281 Navarro, F.C.P., Mohsen, H., Yan, C., Li, S., Gu, M., Meyerson, W., Gerstein, M., 2019. Genomics and data
282 science: an application within an umbrella. Genome Biol. 20, 109.
283 https://doi.org/10.1186/s13059-019-1724-1
284 Padmanabhan, R., Mishra, A.K., Raoult, D., Fournier, P.-E., 2013. Genomics and metagenomics in medical
285 microbiology. J. Microbiol. Methods 95, 415–424. https://doi.org/10.1016/j.mimet.2013.10.006
286 Paila, U., Chapman, B.A., Kirchner, R., Quinlan, A.R., 2013. GEMINI: Integrative Exploration of Genetic
287 Variation and Genome Annotations. PLOS Comput. Biol. 9, e1003153.
288 https://doi.org/10.1371/journal.pcbi.1003153
289 Rokas, A., Abbot, P., 2009. Harnessing genomics for evolutionary insights. Trends Ecol. Evol. 24, 192–200.
290 https://doi.org/10.1016/j.tree.2008.11.004
291 Schutz, S., Monod-Broca, C., Bourneuf, L., Marijon, P., Montier, T., 2022. Cutevariant: a standalone GUI-
292 based desktop application to explore genetic variations from an annotated VCF file. Bioinforma.
293 Adv. 2, vbab028. https://doi.org/10.1093/bioadv/vbab028
294 Tollefson, G.A., Schuster, J., Gelin, F., Agudelo, A., Ragavendran, A., Restrepo, I., et al., 2019. VIVA
295 (VIsualization of VAriants): A VCF File Visualization Tool. Sci. Rep. 9, 12648.
296 https://doi.org/10.1038/s41598-019-49114-z
297 Wang, J., Raskin, L., Samuels, D.C., Shyr, Y., Guo, Y., 2015. Genome measures used for quality control are
298 dependent on gene function and ancestry. Bioinformatics 31, 318–323.
299 https://doi.org/10.1093/bioinformatics/btu668
300 Wang, W., Cao, X.H., Miclăuș, M., Xu, J., Xiong, W., 2017. The Promise of Agriculture Genomics. Int. J.
301 Genomics 2017, e9743749. https://doi.org/10.1155/2017/9743749
302 Wickham, H., 2011. ggplot2. WIREs Comput. Stat. 3, 180–185. https://doi.org/10.1002/wics.147
303 Zook, J.M., Catoe, D., McDaniel, J., Vang, L., Spies, N., Sidow, A., et al., 2016. Extensive sequencing of seven
304 human genomes to characterize benchmark reference materials. Sci. Data 3, 160025.

https://doi.org/10.1038/sdata.2016.25

307 **FIGURE LEGENDS**

308

309 **Figure 1: VCF analysis workflow in SCI-VCF**

310 **A)** Composition of the standard VCF. Metadata and the information of variants in the first eight
311 columns are mandatory. Genotype information is optional and can be present across multiple
312 columns. **B)** Workflow methodology of SCI-VCF describing the core functionalities offered by the
313 tool. Created with Biorender.com

314

315 **Figure 2: Analyses of variants in HG002 using SCI-VCF**

316 **A)** Distribution of variant types; **B)** Size distribution of INDELs; **C)** Comparison of HG002 variants
317 with the GIAB high-confidence variants; **D)** Distribution of variants in chrY; **E)** Distribution of
318 variants in chr6; **F)** Comparison of variants in the HLA region with the same from the 1000
319 genomes project.

320

321 **TABLES**

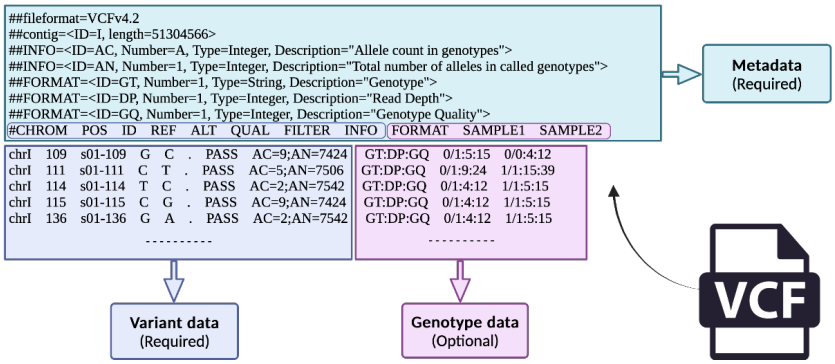
322 **Table 1**: Comparison of features of VCF analysis tools.

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	**Yes**	No	No	No	Yes	Yes	Yes	No	No
Online version	**Yes**	No	No	No	No	Yes	No	No	Yes
Summarise									
Variant level summary	**Yes**	No	Yes	Yes*	No	No	No	Yes	No
Sample level summary	No	Yes	Yes	Yes*	No	No	No	Yes*	No
Compare									
Variant level comparison	**Yes**	No	Yes	Yes*	No	No	No	Yes	No
Unique/intersection set summary	**Yes**	No	Yes*	Yes*	No	No	No	Yes*	No

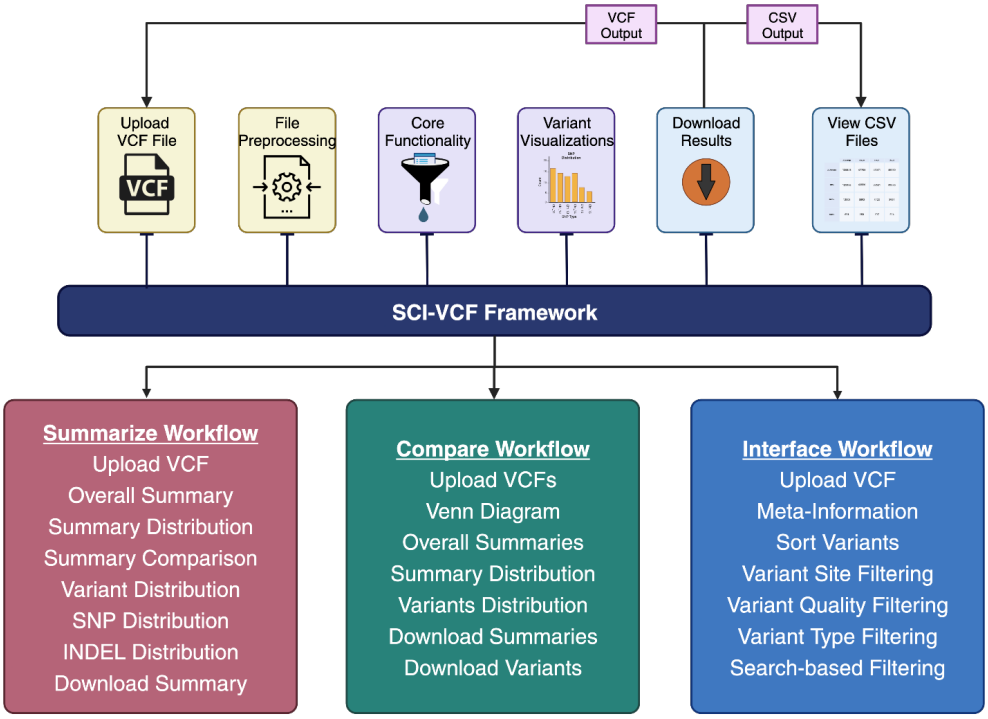
Inspect									
Variant-based filtering	**Yes**	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Sample-based filtering	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Visualise									
Interactive plots	**Yes**	Yes	No	Yes*	No	No	No	No	No
Visualise Genomic regions	**Yes**	Yes	No	Yes*	No	No	No	No	No
Output									
Publication quality graphics	**Yes**	Yes	Yes	Yes*	No	No	No	No	No
Export results as tabular data	**Yes**	Yes	Yes*	Yes*	Yes	Yes*	Yes	Yes	Yes
* Multi-step process involving other tools/libraries									

Figure 1.

A) Composition of the Variant Call Format

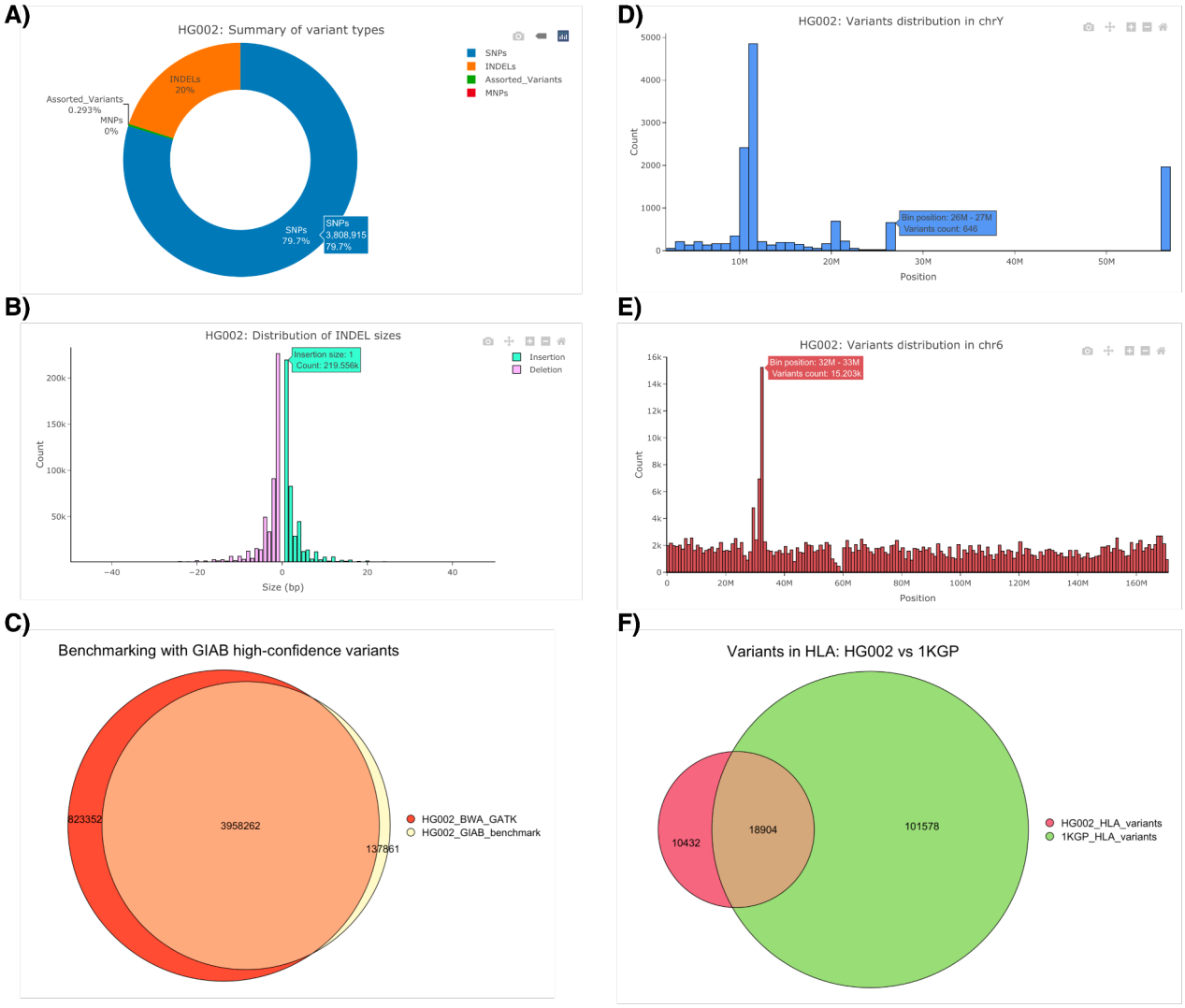


B) Workflow Methodology of SCI-VCF



327

Figure 2.


A)
HG002: Summary of variant types
SNPs
INDELs
Assorted_Variants
MNPs
INDELs 20%
Assorted_Variants 0.293%
MNPs 0%
SNPs 79.7%
SNPs 3,808,915 79.7%
B)
HG002: Distribution of INDEL sizes
Insertion size: 1 Count: 219.556k
Insertion
Deletion
Count
Size (bp)
C)
Benchmarking with GIAB high-confidence variants
823352
3958262
137861
HG002_BWA_GATK
HG002_GIAB_benchmark
D)
HG002: Variants distribution in chrY
Bin position: 26M - 27M Variants count: 646
Count
Position
E)
HG002: Variants distribution in chr6
Bin position: 32M - 33M Variants count: 15.203k
Count
Position
F)
Variants in HLA: HG002 vs 1KGP
10432
18904
101578
HG002_HLA_variants
1KGP_HLA_variants