A systematic benchmark of copy number variation detection tools for high density SNP genotyping arrays

M.N. van Baardwijk, L.S.E.M. Heijnen, H. Zhao, M. Baudis, A.P. Stubbs

PII: S0888-7543(24)00183-6

DOI: https://doi.org/10.1016/j.ygeno.2024.110962

Reference: YGENO 110962

To appear in: *Genomics*

Received date: 17 July 2024

Revised date: 20 October 2024

Accepted date: 9 November 2024

Please cite this article as: M.N. van Baardwijk, L.S.E.M. Heijnen, H. Zhao, et al., A systematic benchmark of copy number variation detection tools for high density SNP genotyping arrays, *Genomics* (2024), https://doi.org/10.1016/j.ygeno.2024.110962

A Systematic Benchmark of Copy Number Variation Detection Tools for High Density SNP Genotyping Arrays

M.N. van Baardwijk[1,2], L.S.E.M. Heijnen[1], H. Zhao[3,4], M. Baudis[3,4], A.P. Stubbs[1]

1. Department of Pathology and Clinical Bioinformatics, Erasmus MC, University Medical Center Rotterdam, Rotterdam, the Netherlands.

2. Department of Surgery, Division of HPB & Transplant Surgery, Erasmus MC, University Medical Center Rotterdam, Rotterdam, the Netherlands.

3. Department of Molecular Life Sciences, University of Zurich, Switzerland.

4. Swiss Institute of Bioinformatics, Switzerland.

Highlights

- Novel tools such as EnsembleCNV, which integrates calls from multiple methods, do not yet outperform PennCNV in balancing precision and recall.
- The significant lack of consensus among CNV calling results highlights the need for careful interpretation and integration of high-confidence calls.
- We present reproducible CNV calling workflows, which can uniform CNV detection across studies, and benchmarking performance analyses, which can be applied to other CNV detection tools.

Abstract

Copy Number Variations (CNVs) are crucial in various diseases, especially cancer, but detecting them accurately from SNP genotyping arrays remains challenging. Therefore, this study benchmarked five CNV detection tools—PennCNV, QuantiSNP, iPattern, EnsembleCNV, and R-GADA—using SNP array and WGS data from 2,002 individuals of the DRAGEN re-analysis of the 1000 Genomes project. Results showed significant variability in tool performance. R-GADA had the highest recall but low precision, while PennCNV was the most reliable in terms of precision and F1 score. EnsembleCNV improved recall by combining multiple callers but increased false positives. Overall, current tools, including new methods, do not outperform PennCNV in precise CNV detection. Improved reference data and consensus on true positive CNV calls are necessary. This study provides valuable insights and scalable workflows for researchers selecting CNV detection methods in future studies.

Keywords: Copy Number Variation, SNP arrays, variant calling, benchmarking, reference datasets

Introduction

Genetic variations within the human genome have gathered significant attention in recent years, ranging from Single Nucleotide Variants (SNVs) to large (>50 bp) structural alterations and rearrangements in the genome. Among these, Copy Number Variations (CNVs) are a critical type of structural variation in which a segment of DNA is deleted, duplicated, or amplified, greater than 50 bp in length (Zarrei et al., 2015) though other operational definitions may be applied (MacDonald et al., 2014; Redon et al., 2006; F. Zhang et al., 2009). These CNVs can influence phenotype through several mechanisms, such as alteration of dose-sensitive genes, change of gene structure, regulatory effects and position effects (Gamazon & Stranger, 2015). Consequently, CNVs are a rich source for diagnostic and prognostic models, particularly in oncogenomic research (Gunturu et al., 2013; Hu et al., 2021).

CNVs can be detected using either array-based or Next Generation Sequencing (NGS) technologies. While the two approaches differ in resolution and NGS is rapidly advancing, array-based platforms continue to offer significant advantages, such as lower costs and reduced resource requirements. This makes them particularly useful for CNV detection in many research and clinical environments, particularly where resources are limited or large datasets have already been generated using these technologies. For instance, a recent study by Gan et al. developed and validated a genetic testing workflow for known pharmacogenomic genes using the Illumina Global Screen array (Gan et al., 2024). Similarly, SNP array data was used alongside Whole Genome Sequencing in the All of Us Research Program to complement genomic analyses for over 200,000 individuals (Bick et al., 2024). Regardless of the selected platform for CNV detection, both sequencing and array-based techniques rely heavily on the utilization of Bioinformatic algorithms. As there are many different algorithms publicly available (Colella et al., 2007; Pique-Regi et al., 2010; K. Wang et al., 2007) which are often inconsistent in the resulting CNV calls (Nutsua et al., 2015), selecting the most suitable tool can easily become a daunting task. Benchmark studies are essential to providing objective evaluations of multiple available tools under various conditions (Weber et al., 2019).

Several different benchmark studies have been conducted for array-based CNV callers (Marenne et al., 2011; Xu et al., 2013). The most recent study by Nutsua et al. found that Hidden Markov Model (HMM)-based methods, in particular PennCNV, had the highest prediction accuracy (Nutsua et al., 2015). However, recent advancements have introduced new methods, including more advanced ensemble and deep learning methods (Eghbal-Zadeh et al., 2019; Z. Zhang et al., 2019). Furthermore, the previous studies did not determine the performance of tools given different conditions such as the interest in a single

type of CNV or the sample size. Finally, tool installation and application is often complex without a computational background. To conclude, there is a need for stable CNV detection workflows that can be easily implemented and adapted with objective performance assessments.

To address these gaps, the current study systematically benchmarks state-of-the-art CNV detection tools for Whole-Genome SNP genotyping arrays. Utilizing publicly available data from the 1000 Genomes project as a gold standard (Auton et al., 2015), we aim to provide reproducible and adaptable workflows through the use of Docker images and Nextflow pipelines. Our evaluation will include comparison of the resulting CNV calls to the DRAGEN re-analysis of the 1000 Genomes dataset, taking into account various characteristics such as CNV type and size. Ultimately, we seek to offer recommendations on the most suitable tools for different research needs.

Methods

Data collection and preprocessing

1000 Genomes HD Genotype Data

From the 1000 Genomes phase 3 dataset, dense genotyping data originating from the Illumina HumanOmni2.5.Quad v1.0B SNP array was retrieved (Auton et al., 2015). This dataset consists of 2,141 samples that passed a call rate threshold of 97% and were concordant to their provided gender, as defined by the internal Quality Control (QC) pipeline. The array data was retrieved in raw .idat format. Unfortunately, the manifest file including the probe sequences and corresponding genomic locations for this array provided by Illumina have not been updated beyond reference genome GRCh36. Therefore, the raw data was processed using GenomeStudio v2.0 using the standard map of probe sequences for reference genome GRCh36.

1000 Genomes DRAGEN Re-analysis

The recent reanalysis of the 1000 Genomes phase 3 Dataset executed with the Illumina DRAGEN (Dynamic Read Analysis for GENomics) 3.5 Bio-IT platform (Olson et al., 2022) was selected as the gold standard due to its high quality CNV calls. A recent preprint validated the DRAGEN algorithm against the Genome in a Bottle SV benchmark set and reported improved performances for small CNVs (1-10kbp) and similar performances for larger CNVs (>10kbp) compared to its competitors (Behera et al., 2024). The CNV calls were accessed on 27-07-2022 from https://registry.opendata.aws/ilmn-dragen-1kgp. This dataset includes 2,504 unrelated samples from Phase 3 of the 1000 Genomes project (Auton et al., 2015), as well as 698 additional related samples

funded by the National Human Genome Research Institute (NHGRI). These samples were previously sequenced by the Illumina NovaSeq 6000 system at >30x coverage and with 150bp paired reads. The CNV calls from the 2,002 samples that were identical to the raw array data available were selected.

Selection of tools

The five tools selected for this benchmark study – PennCNV (K. Wang et al., 2007), QuantiSNP (Colella et al., 2007), iPattern (Pinto et al., 2010), EnsembleCNV (Z. Zhang et al., 2019) and R-GADA (Pique-Regi et al., 2010) – were primarily chosen based on their established relevance and widespread use in the field of CNV detection from SNP arrays. This is further justified by their citation frequency in existing literature, indicating their importance in both clinical and research contexts. These tools also represent a diverse range methodologies, including Hidden Markov Models (HMMs), Bayesian approaches and ensemble methods, providing a comprehensive evaluation across different algorithms. EnsembleCNV was specifically included due to the novelty of the method and the reported improvements in performance over tools like PennCNV, QuantiSNP and iPattern, as highlighted in its original publication. Table 1 provides an overview of the general characteristics of each method, highlighting the methodological differences and key features for CNV detection.

Tool	Methodology	Version	Features	Joint vs Individual	URL	Language	Year of publication	# Citations*
PennCNV	Hidden Markov Model	1.0.5	Gap fraction between adjacent calls (fraction) & the minimum number of SNPs for an individual call (numsnp).	Individual or joint-calling for families	https://penncnv.openbioinformatics.org/	Perl	2007	1,999

QuantiSNP	Hidden Markov Model	2	The characteristic length used to calculate transition probabilities, the number of iterations used for the EM algorithm during learning, number of mixture components used in noise model (nComp), degrees of freedom of student t-distribution (v), shape parameters of Beta prior on outlier rate (nu_alpha & nu_beta), scale parameters of Dirichlet prior on genotype/mixture proportion (w_alpha & q_alpha), concentration parameter of Normal-Wishart prior (tau), scale parameters of Wishart prior on covariance matrix (S_alpha & S_alpha_homdel) & characteristic length for normal state	Individual	https://sites.google.com/site/quantisnp/	MATLAB	2007	769

			(longChromosome).					
iPattern	Gausian Mixture Model	0.58	The minimum number of SNPs for an individual call (winSize), maximum distance between adjacent SNPs for an individual call (maxProbeDistance), value for density estimation (bandWidth) & parameter for identifying density clusters within	Joint-calling	http://www.tcag.ca/tools/index.html	R & Python	2010	N.A.**

			windows (peakSeparation).					
EnsembleCNV	Ensemble method	N.A.	Frequency cut-off for selecting CNVRs with common CNV genotype that will be subjected to boundary refinement (freq) & GQ score threshold for filtering the final result (gqscore).	Joint-calling	https://github.com/HaoKeLab/ensembleCNV	Perl & R	2019	21
R-GADA	Segmentation	2.0.1	The sparseness hyper parameter of the sparse Bayesian learning step (aAlpha), the array noise level (sigma^2), the critical value of the t-statistic of segment breakpoints as computed by backwards elimination (t) & the minimum number of SNPs for an		https://github.com/isglobal-brge/R-GADA	R	2011	64

			individual call (MinSegLen).					

***Table 1**: The CNV detection methods tested in this benchmark study together with their characteristics. *The number of citations was determined on the 3rd of June 2024 using Google Scholar. **iPattern had no citations as this method was never officially published.*

QuantiSNP

QuantiSNP is an Objective Bayes Hidden-Markov Model (OB-HMM) inferring copy number state based on the log R ratio (LRR) and the B-allele frequency (BAF) of SNPs (Colella et al., 2007). The LRR and BAF refer to the total fluorescent intensity and the relative ratio of this fluorescent intensity between the two probes representing the two alleles at each SNP, respectively. QuantiSNP defines six hidden states representing different CNV events, as well as the normal state. The Objective Bayes paradigm is utilized to give probabilities to the called copy number states.

PennCNV

PennCNV is a Hidden Markov Model (HMM) based method as well (K. Wang et al., 2007), that has been widely applied in previous research (Marshall et al., 2016; K. Wang et al., 2014). Besides LRR and BAF values it utilizes the population allele frequency and distance between neighboring SNPs. This method differs from QuantiSNP in the application of state-specific and distance-dependent transition probabilities in the HMM. The first is based on the fact that certain copy number state transitions are more common than others, while the latter is based on that SNPs in close proximity are more likely to have the same state compared to those with large distances between them. In addition, PennCNV is the first method that can also incorporate family information for the joint-calling of CNVs when this information is available. However, as the data in this study originate from unrelated individuals, this property will not influence the current performance.

iPattern

iPattern is a method that can perform joint-calling of CNVs for both related and unrelated individuals (Pinto et al., 2010). This tool takes advantage of consensus in signals across samples resulting in better detection of copy number polymorphisms (CNPs) with higher allele frequencies. Instead of

LRR and BAF values, it utilizes the signal intensities after normalization with GenomeStudio. The iPattern pipeline includes balancing of the X and Y channels, quantile normalization, intensity rescaling and variance normalization. Subsequently, a sliding-window method is used to identify potential CNV regions which is followed by boundary refinement resulting in the final CNV calls.

EnsembleCNV

Recently, Zhang et al. introduced a novel ensemble method called EnsembleCNV (Z. Zhang et al., 2019). This method takes the CNV calls from three other tools, PennCNV, QuantiSNP and iPattern, and combines them using a heuristic algorithm. In their study, Zhang et al. showed that their algorithm outperformed simple combining of these callers with either the 'intersection' or 'union' strategy in terms of concordance rate and stability of technical duplicates (Z. Zhang et al., 2019). EnsembleCNV consists of four steps; 1) identification of batch effects based on pre-generated sample summary statistics and LRR values; 2) identification of CNV regions (CNVRs) by combining individual CNV calls using a forward-screening and backward-pruning approach; 3) re-genotyping of CNVRs using locally fitted likelihood models; 4) boundary refinement of CNVRs using local correlation matrices. Additionally, they defined a genotype quality (GQ) score that can be used to filter out low-quality CNVs.

R-GADA

R-GADA is an R-based implementation of the segmentation algorithm called Genome Alteration Detection Analysis (GADA) (Pique-Regi et al., 2010). First, GADA fits a sparse Bayesian Learning (SBL) model to identify the most likely candidate breakpoints for each copy number state. Second, GADA implements a backward elimination (BE) procedure to remove false positive breakpoints based on a user defined cut-off to control the false discovery rate (FDR). In this study, the sparseness hyperparameter α was set as 0.8 and the critical value for BE as 5, as this was recommended for achieving a lower FDR, although in trade-off with an expected lower recall.

Data post processing

As each CNV calling method uses their own format for representing the resulting CNV calls these were first converted into a general tab separated format including the sample identifier, chromosome, start coordinate, end coordinate, length and CNV type. The array-based identifiers were mapped to individual identifiers supplied by the 1000Genomes project, which accord with identifiers used in the gold standard VCF file. As the gold standard CNV calls were established using GRCh38 as a reference, the LiftOver tool from UCSC was used to map the start and end coordinates of the CNVs called by each method to GRCh38 as well. In addition, gold standard CNVs were filtered

on overlap with a minimum of five probes on the HumanOmni2.5.Quad v1.0B SNP array to include only CNVs detectable by this array.

Implementation of tools and benchmarking

An overview of the implementation of the tools and the execution of the benchmark analysis can be seen in Figure 1. To provide stable and transferable software packages, all software and corresponding dependencies were installed within individual Docker containers for each of the tools (Merkel, 2014). Subsequently, stable and reproducible pipelines were written in a Nextflow workflow (DI Tommaso et al., 2017). All tools were run with default parameters according to their original publications, which can be viewed in Supplementary Table 1. The versions of the tools and their dependencies are provided in Supplementary Table 2. The full workflow, including all individual pipelines, can be accessed through GitHub (https://github.com/mbaardwijk/aCNVbench). To ensure the reproducibility of our analysis, we provide the following example of how to run the CNV detection and benchmarking workflow using Nextflow:

```
nextflow run main.nf --inputSheet {path to tab-separated SNP array input file} --goldstandardFile {path to gold standard file in VCF or tab-separated format} --genome 'hg18'
```

All tool specific parameters are preconfigured in the nextflow.config file, which can be easily modified to adjust the settings for each tool. The input sheet is a tab-separated file including the following:

- The Illumina final report file, containing the columns Sample ID, SNP Name, Chr, Position, Allele 1 – Forward, Allele 2 – Forward, X, Y, B allele frequency and Log R ratio.
- A sample map, with at least the columns Name and Gender.
- A probe map, specifying the chromosome and position for every probe on the array.

The workflow allows for flexibility by enabling users to exclude specific tools from the analysis. For instance, the parameter '—skipPennCNV' can be added to remove PennCNV from the pipeline execution.

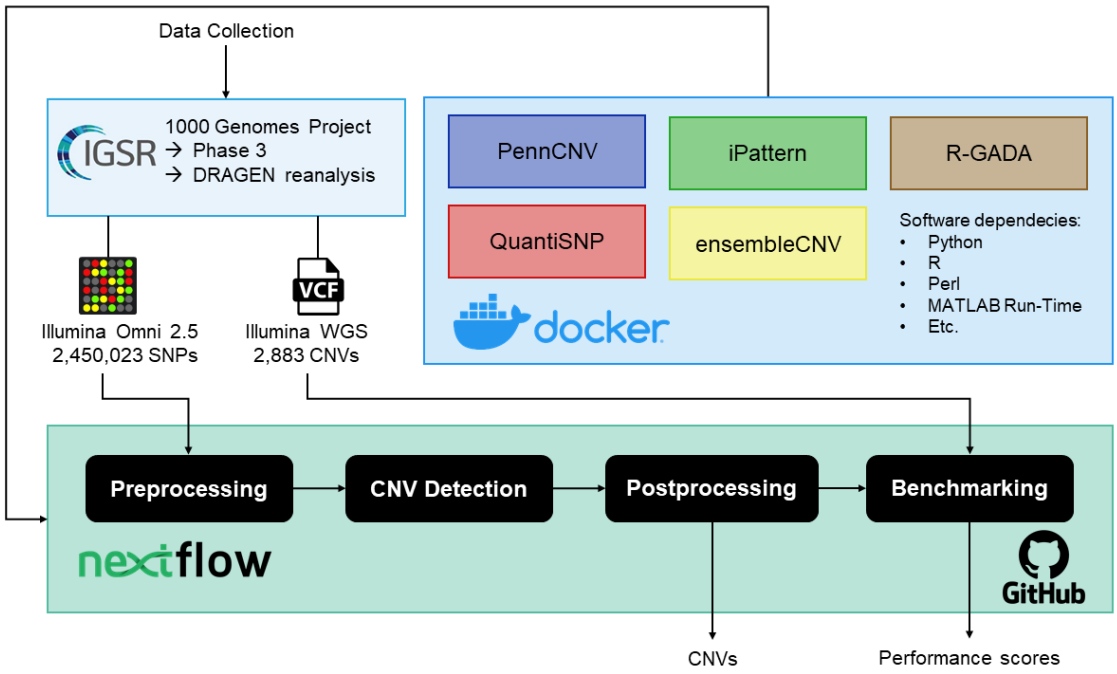

***Figure 1**: Schematic overview of the data collection and implementation of CNV calling methods. Data was collected from the 1000 Genomes Project. Individual CNV callers and their software dependencies were wrapped in Docker images and are available through DockerHub (https://hub.docker.com/u/mbaardwijk). Nextflow workflows were written for each CNV calling method, making use of these Docker images, including preprocessing, CNV detection, post processing and the final benchmarking and are available through GitHub (https://github.com/mbaardwijk/aCNVbench).*

Performance assessment of CNV calls

First, the definition of true positive (TPs), false positive (FP) and false negative (FN) calls needs to be established to facilitate the calculation of different performance metrics of the resulting CNV calls. Haraksingh et al. defined true positive calls as those with 50% reciprocal overlap with an individual or set of CNVs in the gold standard dataset or being in a set of CNVs with 50% reciprocal overlap (Haraksingh et al., 2017). Subsequently, false positive calls are defined as CNV calls with no or in-sufficient reciprocal overlap with the gold standard dataset. Finally, false negative calls are defined as CNVs in the gold standard dataset with no or in-sufficient reciprocal overlap with the CNV calls. In this study, we adapted this approach by using a range of different thresholds for the percentage of reciprocal overlap starting at a single basepair of overlap until 50% of reciprocal overlap. This allowed us to assess the impact of using different thresholds and to determine the maximum achievable performance for each CNV calling method. Examples of the classification methodology using the

threshold of 50% reciprocal overlap can be seen in Figure 2A-C.

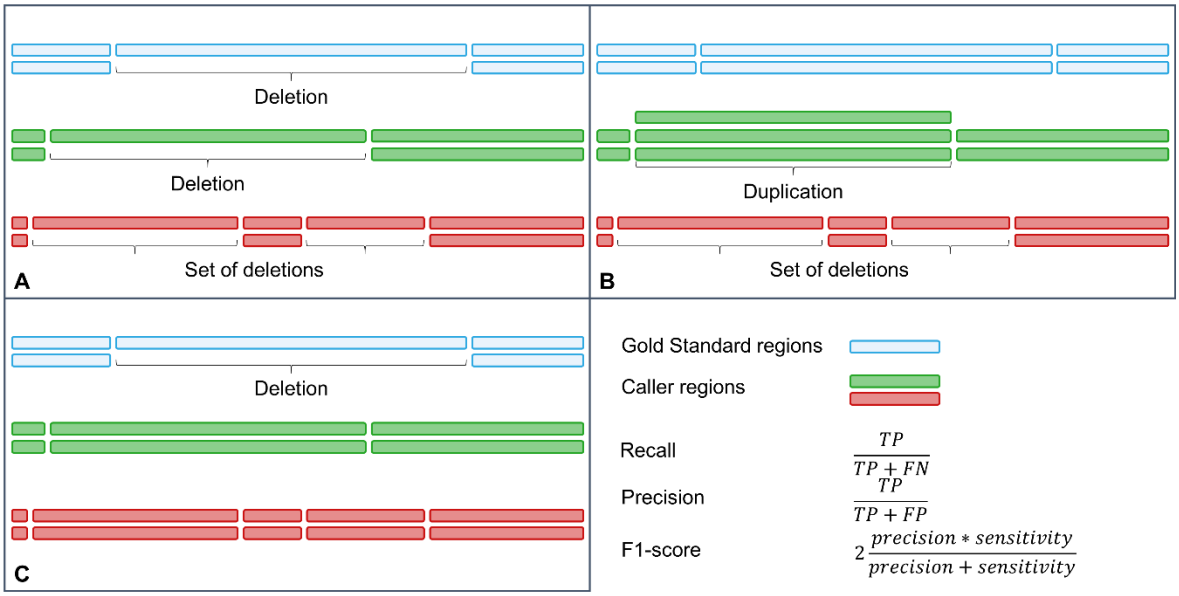

Figure 2*: An example of the classification scheme for performance assessment with a threshold of 50% overlap. Gold standard regions are visualized in blue, while the examples of two call sets are provided in green and red. Formulas for how to calculate the recall, precision and F1-score using the resulting TPs, FPs and FNs are also provided.* ***A****: The green calling method shows a deletion with more than 50% reciprocal overlap with a deletion in the gold standard data, while the red calling method shows a set of deletions that combined also show more than 50% reciprocal overlap. For both methods, these will be defined as a single TP.* ***B****: The green calling method shows a duplication, while the red calling method shows a set of deletions. In the gold standard data, no duplication or deletion was observed so these will be defined as one FP for the green method and two FPs for the red method.* ***C****: No CNVs were discovered in the green or red calling method, while a deletion was observed in the gold standard data. For both methods, this will be defined as a single FN.*

For each sample, the calling method and threshold of overlap, the number of true positives, false positives and false negatives was established using the aforementioned classification. Using the sum of these values, recall, precision and F1 scores were calculated for each calling method, summarizing the overall performance. Recall, also referred to as sensitivity, represents the proportion of true CNV calls of the gold standard dataset that are identified by the calling method. Precision measures the proportion of CNVs detected by the calling method that are true CNV calls. Finally, the F1-score represents the balance of the recall and precision scores. The formulas for calculating the recall, precision and F1-score are provided in Figure 2. These calculations were repeated for each threshold of (reciprocal) overlap to assess how overlap stringency affected the

performance of the detection methods. To determine the most suitable tool given different conditions, different types of copy number variations were defined, namely deletions and duplications.

Results

We processed 2,002 HumanOmni2.5.Quad SNP array samples using five CNV detection tools (QuantiSNP, PennCNV, iPattern, R-GADA, EnsembleCNV), which are summarized in Table 1. First, the characteristics of the called CNVs will be described. Second, the overlap in called CNVs between methods will be compared. Third, the called CNVs will be compared to a reference database and known functional genomic regions such as exons, introns and promoters. Fourth, the benchmark performance will be assessed in terms of recall, precision and F1 score. Fifth, the memory and CPU requirements of the different CNV callers will be evaluated. Finally, multiple tool recommendations will be made based on different aspects of the CNV callers.

Distribution of CNV count, length and type

For each method, CNVs were called as described in the methods section. The genomic coordinates of the resulting CNVs were converted from GRCh36 to GRCh38 using the UCSC liftOver tool. After this conversion, 0.1-7.2% of the segments were lost due to mapping issues (Supplementary Table 3). The 57,826 unique WGS based CNV calls were filtered on overlap with a minimum of 5 probes on the SNP array, resulting in 32,258 unique CNVs being included in the benchmark. The average number of CNVs called by the different algorithms varied greatly between methods, as can be seen in Figure 3A. However, all tools have in common that they called more deletions than duplications for most samples, as shown by Figure 3B-C. Compared to the gold standard dataset, PennCNV, QuantiSNP and iPattern called on average less CNVs per sample, while R-GADA called more CNVs per sample and EnsembleCNV called a similar number of CNVs per sample.

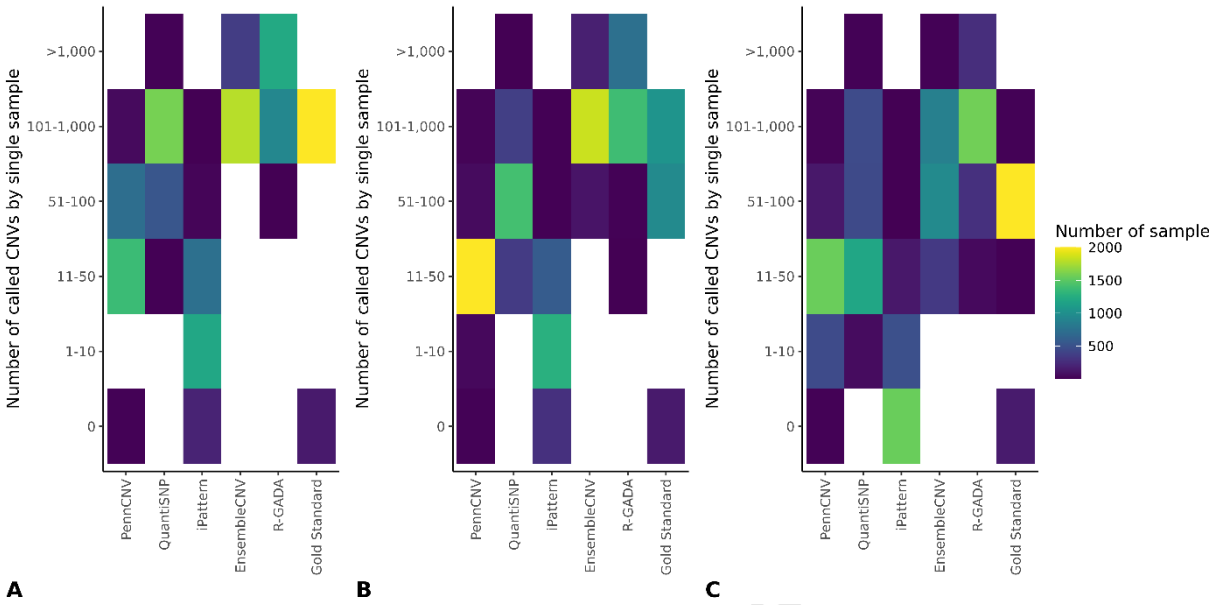

***Figure 3**: A heatmap showing the number of called CNVs for each CNV detection method and the gold standard dataset per individual sample. The number of called CNVs were divided into bins of different CNV count ranges, which were subsequently counted for all individual samples (**A**), for all individuals including only deletions (**B**) and for all individuals including only duplications (**C**). Tiles were left blank if no individual sample had a CNV count within that range for that method.*

On top of large differences in average CNV count, the CNV calling algorithms also differed in the size of the called CNVs, as is shown in Figure 4. Especially the R-GADA algorithm resulted in larger CNVs. Another interesting observation is that most CNV calling algorithms (PennCNV, QuantiSNP, iPattern and EnsembleCNV) also called CNVs with smaller sizes as compared to the gold standard dataset, even though WGS is reported to be more sensitive in detecting small CNVs due to its base-pair level resolution. This might indicate that with the right algorithms smaller CNVs can be detected within SNP genotyping array when there are enough probes to cover that region. However, it could also be that these regions are artifacts and that these algorithms are more sensitive to noise.

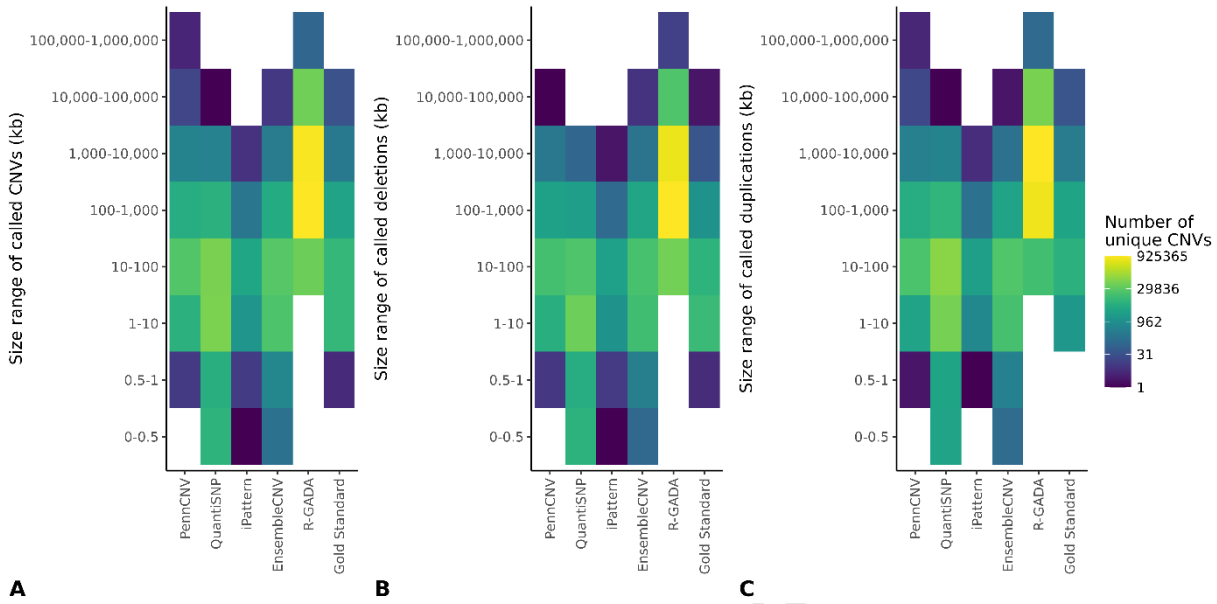

***Figure 4**: A heatmap showing the size distributions of unique CNVs called by the different CNV detection methods in comparison to the gold standard. The unique CNVs were divided into bins of different size ranges, which were subsequently counted, and log-transformed for all called CNVs (**A**), for all deletions (**B**) and for all duplications (**C**). Tiles were left blank if no unique CNV with a size within that range was detected for that method.*

Consensus between different CNV callers and reference databases

To establish whether there was consensus between the different callers, all unique CNVs were split into segments in which no sample changed state. Subsequently, each segment was tested for overlap with any of the unique CNVs called by each different method. Figure 5 shows the overlap between methods by means of an UpSet plot (Lex et al., 2014). From Figure 5, it can be established that most CNV segments can be assigned to R-GADA, followed by EnsembleCNV, while iPattern shows the smallest number of CNV segments. With only 9,752 out of 2,878,941 (0.34%) segments being shared by all five methods, the consensus between all callers is limited. R-GADA shows the most overlap with CNV segments from other individual callers, although this is likely due to chance by the large CNVs called by R-GADA. Most notably, the most similar methodologies, PennCNV and QuantiSNP, share more CNV segments with other callers than each other. To determine whether there was any bias in the size of the separated segments, the size distribution of all sets was included within Figure 5 as well. From this it can be observed that the majority of segments was in between 1 and 10 kb, and this was approximately the same for all different sets.

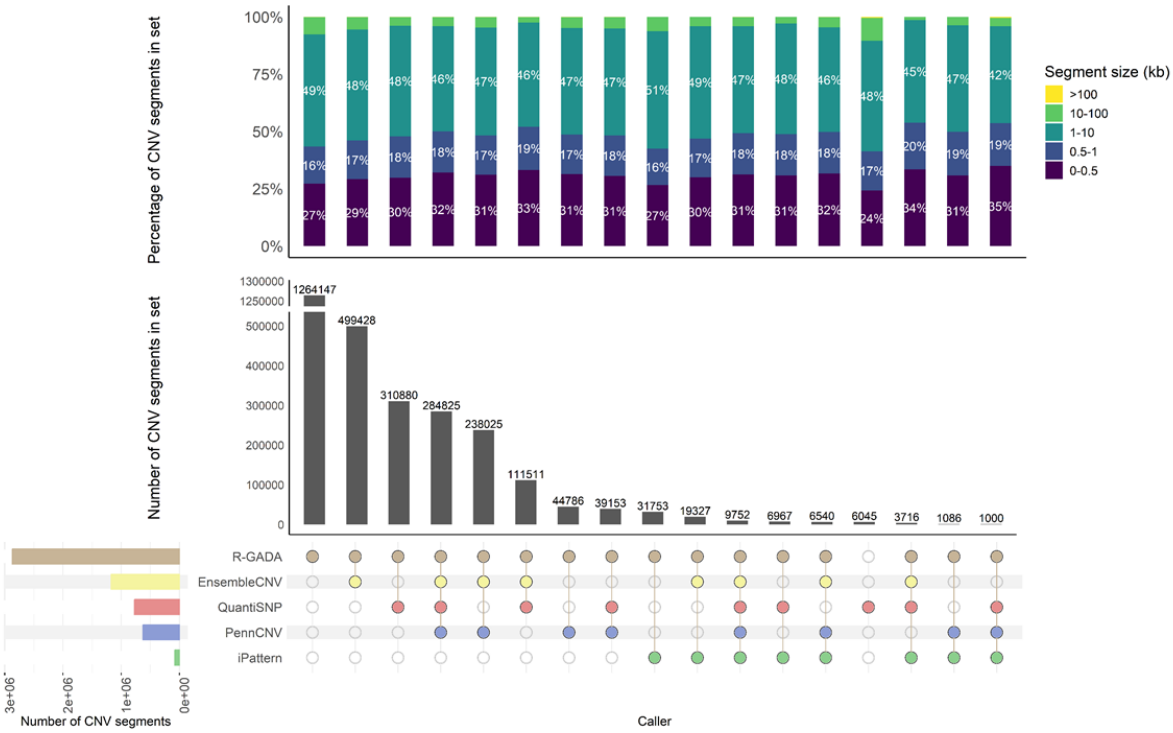

***Figure 5**: An UpSet plot showing the overlapping CNV segments called by the different methods. The bottom left graph shows the number of CNV segments identified by each individual caller, while the bottom right graph indicates which callers are represented within each set. The middle graph shows how many CNV segments are represented within each set. The top graph shows the distribution of the segment size of the CNV segments represented within each set.*

The called CNVs as well as the gold standard CNVs were also compared to reference databases of known variation and relevant regions within the genome. Figure 6A shows the fraction of CNVs called by each method that show more than 50% reciprocal overlap with CNVs from CNVs within the Database of Genomic Variants (DGV) (MacDonald et al., 2014). From Figure 6A it can be observed that the majority of CNVs in the gold standard dataset are likely previously established CNVs, since 79.0% of the DGV CNVs were rediscovered by the Gold Standard dataset. For the array-based tools, PennCNV showed the most overlap with previously established CNVs, followed by QuantiSNP, although this was limited to 31.4% of the called CNVs. Therefore, it is likely that there is a bias within DGV for variants originating from the sequencing based 1000Genomes dataset. Figure 6B shows the fraction of CNVs called by each method that show overlap with exon, intron and promoter regions originating from the UCSC knownGene hg38 dataset (Team BC & Maintainer BP, 2019). When CNVs overlapped multiple regions they were classified by the maximum overlap and when CNVs did not overlap any of these regions, they were classified as intergenic. For all CNV call sets, most CNVs were discovered in intergenic regions, while the least amount of CNVs overlapped with exons.

Notably, the CNVs called by R-GADA, besides intergenic regions, only showed overlap with some intronic regions.

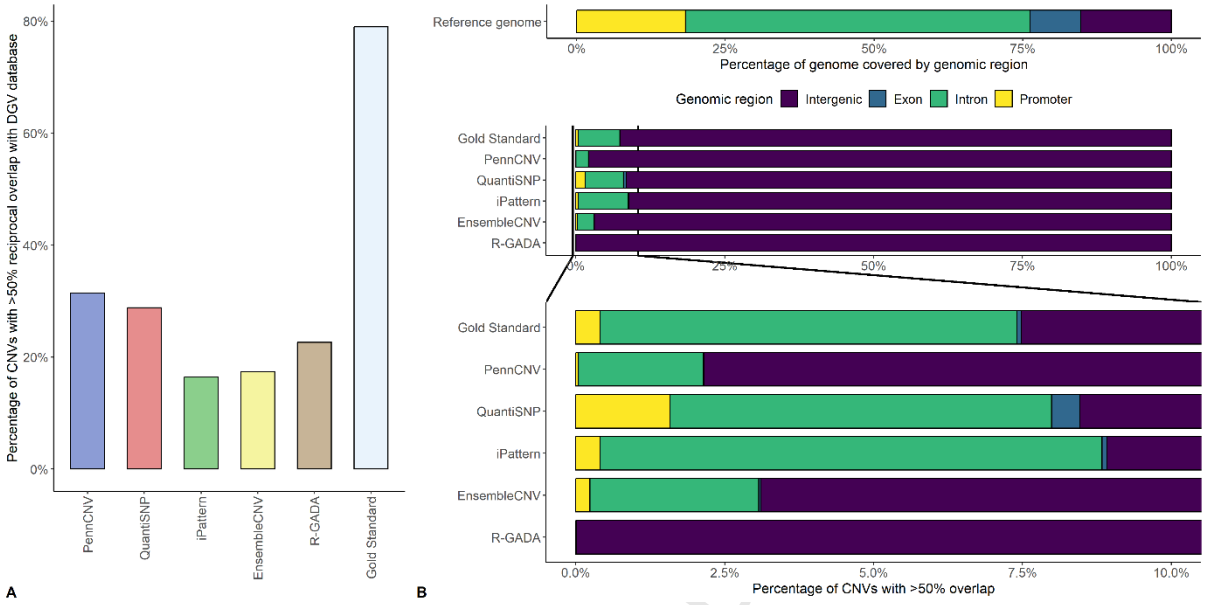

Figure 6*: The overlap of CNVs call sets including the gold standard set with known reference databases.* ***A****: The fraction of CNVs with >50% reciprocal overlap with CNVs from the Database of Genomic Variants (DGV) release 2020-02-25.* ***B****: The fraction of CNVs with >50% overlap with exon, intron and promoter regions from UCSC knownGene based on reference genome hg38.*

Benchmark performances

The performance of each individual method was established by determining the total number of true positive, false positive and false negative CNV calls for each of the 2,002 samples for which both SNP array data and WGS based CNV calls were available as explained in the methods section. This process was repeated over different thresholds for reciprocal overlap, ranging from 1-50%, as well as 1bp of overlap. Figure 7A-C show the recall, precision and F1 scores respectively. From Figure 7A, it can be observed that R-GADA shows the highest recall for both deletions and duplications and each threshold of overlap, ranging from 0.83 to 0.97. Additionally, it can be observed that all methods show higher recall for duplications compared to deletions. Figure 7B shows that PennCNV attained the highest precision for both deletions and duplications and each threshold of overlap, ranging from 0.14 to 0.47. While PennCNV and QuantiSNP showed a higher precision for deletions, the other three methods showed a higher precision for duplications. As can be seen from Figure 7C, PennCNV also achieved the highest F1 score for both deletions and duplications and each threshold of overlap, ranging from 0.20 to 0.47. While the different thresholds did not change the best performing method per score, they do provide some information on the highest obtainable score for each method. Most

noticeably, the scores for duplications called by EnsembleCNV drop severely between a threshold of 40% and 50% reciprocal overlap.

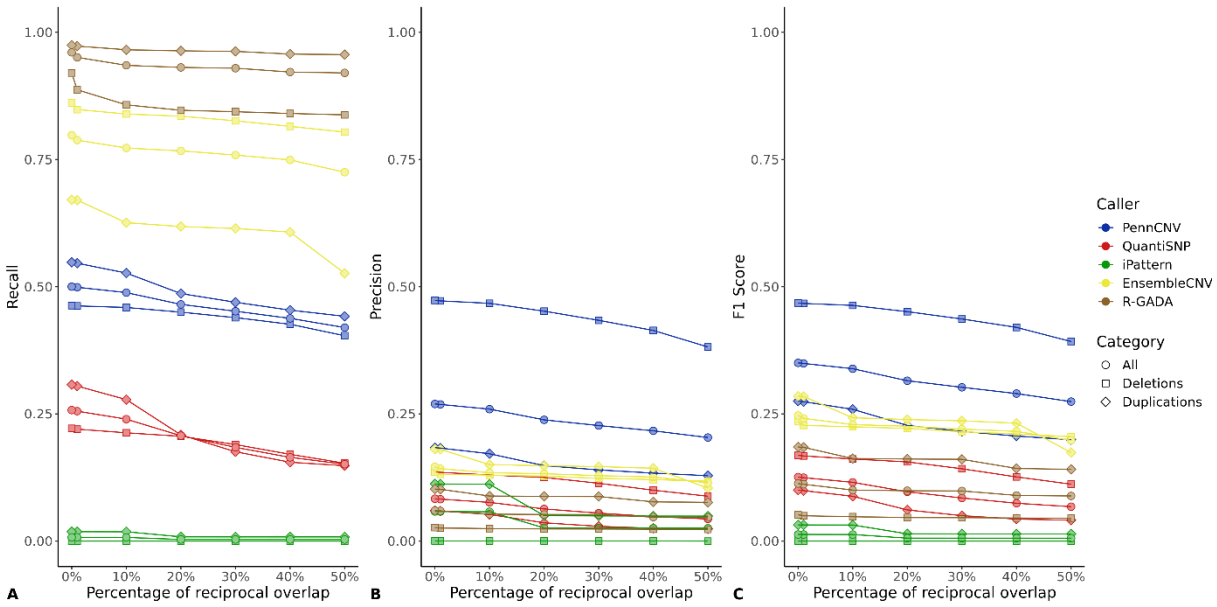

Figure 7*: The overall benchmark performances for the different CNV detection methods and CNV type categories for different thresholds of reciprocal overlap.* ***A-B****: Recall and precision scores were established from the sum of all per-sample true positives, false positives and false negatives, using the formulas described in the methods section.* ***C****: F1-score was calculated over the recall and precision scores in A-B, using the formula described in the methods section.*

Computational requirements for CNV callers

The CPU and memory requirements were obtained from the Nextflow tracing file after running the pipeline using a maximum of 48 CPUs and 192 GB of memory. Figure 8 shows the average processing time per sample, as well as the peak of real memory. From Figure 8A it can be observed that QuantiSNP and iPattern required a significantly higher amount of computational time compared to PennCNV and R-GADA. It should also be noted that because EnsembleCNV requires input from PennCNV, QuantiSNP and iPattern, the actual processing time per sample is highest for EnsembleCNV. Figure 8B shows that EnsembleCNV required a significantly higher amount of memory compared to all others.

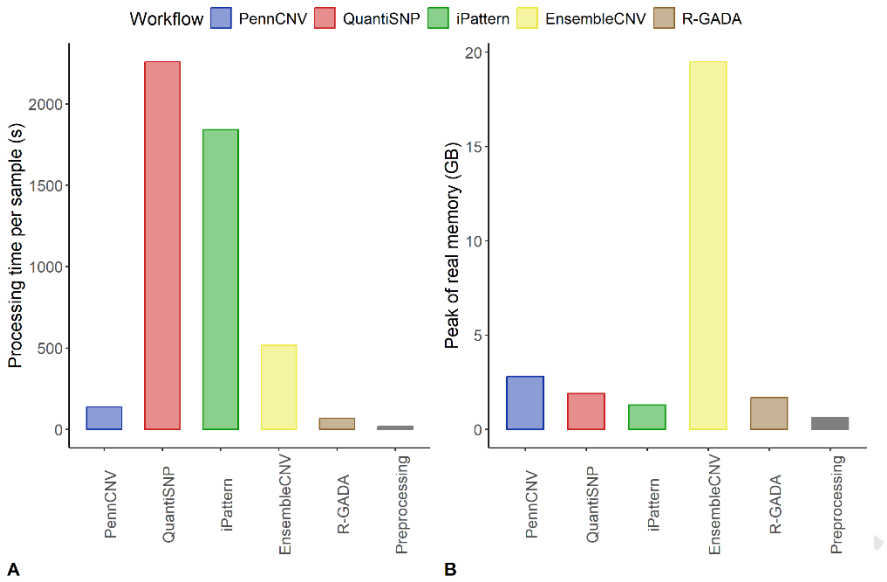

***Figure 8**: Computational requirements for the different callers, as well as preprocessing. **A**: Total CPU time to process a single sample for each tool, calculated by dividing the overal processing time by the numbers of samples. **B**: The maximum of real memory usage for each individual CNV detection method measured in gigabytes. The workflows were executed on a CPU server running Ubuntu 22.04.5 LTS, utilizing up to 48 CPUs and 192GB of memory.*

Recommendations

Based on all the previous findings, several recommendations can be made. As no differences were observed in the tools' highest recall, precision or F1 score between deletions, duplications and the combination of both, the CNV type of interest should not influence the choice of method. When the goal is to discover as many true positive calls as possible, regardless of a high false discovery rate, R-GADA is the best performing tool. For those aiming to still detect numerous true CNVs while maintaining a better balance between true and false discoveries, EnsembleCNV is a viable option. However, even though it discovers fewer CNV call, PennCNV should still be regarded as the most reliable method based on its precision and F1 score. If processing time is limited, QuantiSNP, iPattern or EnsembleCNV should be avoided. Similarly, if computational resources are limited, EnsembleCNV should not be considered.

Discussion

Algorithms for the detection of Copy Number Variations (CNVs) for SNP genotyping array data have been around for more than a decade (Olshen et al., 2004). Recent advances in algorithm development using ensemble or deep learning methods have resulted in novel methods for this purpose. This study aimed to benchmark five state-of-the-art CNV detection tools for SNP genotyping array data. Unlike previous benchmark studies (Marenne et al., 2011; Nutsua et al., 2015; Winchester et al., 2009), this study has determined performance for different thresholds of reciprocal overlap, offering a more nuanced comparison. We also introduced EnsembleCNV, a novel ensemble method reported to outperform established tools such as PennCNV, QuantiSNP and iPattern. Additionally, we assessed the speed and memory usage for each tool to establish which tools are suitable if computational resources are limited. Finally, based on our findings, we provide practical recommendations on which tools are best suited for specific research conditions.

Overall, the performance of CNVs detection was limited for all tools. Our findings suggest that PennCNV is the most precise CNV caller in all categories, while R-GADA is the most sensitive one. While there were differences observed in the performance for the different types of calls made by the same algorithm, unexpectedly, the most sensitive and precise CNV callers remained the same across categories. This indicates that a CNV caller should be selected based on the preference of better recall or precision instead of a CNV type of interest. Alternatively, the F1-Score can be chosen as a measure of balanced recall and precision, based on which PennCNV was found to be the best performing CNV calling method for all categories. Surprisingly, the novel ensembling method EnsembleCNV detected many false positive CNV calls. An explanation for this observation might be that the developers assessed their method solely on the recall and concordance between technical duplicates only (Z. Zhang et al., 2019). Another explanation is that ensembling methods are limited by the performance of the other methods that they draw from. EnsembleCNV takes CNV calls from PennCNV, QuantiSNP and iPattern as starting point, of which the latter two methods lacked precision in this study.

Previous studies have applied other strategies to evaluate the performance of CNV detection tools but found similar results. Marenne et al. compared CNV calls made by PennCNV, QuantiSNP and CNVPartition to a reference dataset generated using Multiplex Ligation-dependent Probe Amplification (MLPA) (Marenne et al., 2011). Similar to most methods included in our study, they found that precision was usually higher than recall. Marenne et al. concluded that while PennCNV was their best performing method based on reliability, all algorithms were limited in recall. Given the current results, it is unfortunate that novel methods like EnsembleCNV have still not solved these limitations in CNV detection from SNP arrays.

Another strategy for benchmarking CNVs, namely using SNP array trio data, was executed by Nutsua et al. (Nutsua et al., 2015). This strategy relies on the fact that over 99% of the CNVs found in an individual are inherited by one of the parents. They compared PennCNV, QuantiSNP, R-GADA, GLAD, VEGA and APT, and found that the Hidden Markov Model (HMM)-based methods like PennCNV and QuantiSNP achieved the highest precision. However, in line with the current study, the authors noted that array-based CNV detection methods fall short in limiting false positive calls and show little concordance in CNV calls.

Even though NGS technologies have become popular in recent years, SNP genotyping arrays remain an essential technique in genomic research. Due to their affordability and scalability, they are more suitable for large-scale populations studies compared to WGS. SNP arrays allow for efficient detection of CNVs and continue to provide valuable insights in various research contexts. Therefore, the benchmarking results and workflows presented in this study are highly relevant for researchers and clinicians in this field.

While this study provides valuable insights into the performance of various CNV detection tools, it is important to acknowledge its limitations. First, the benchmark was based on a single dataset and may therefore not fully represent the genomic diversity present in broader populations. To mitigate this, we also compared the detected CNVs against reference CNVs in the Database of Genomic Variants (DGV). Additionally, recent validation of the DRAGEN algorithm against the Genome in a Bottle reference (Behera et al., 2024) has demonstrated the reliability of the DRAGEN re-analysis dataset used in this study, mitigating concerns about potential false positives and supporting the robustness of our gold standard. Second, the use of WGS-based CNVs as a gold standard introduces challenges due to fundamental differences between WGS and SNP-array technologies. WGS provides base-pair level resolution, allowing for the detection of small CNVs, whereas SNP arrays are dependent on how the probes are distributed across the genome, limiting the resolution and coverage. To account for this, we filtered the gold standard CNVs to only include those spanning at least five probes on the SNP array. Despite this adjustment, it is important to note that WGS can still produce false positive and negative results, especially in regions with complex genome structures such as repetitive regions. SNP arrays, on the other hand, may produce false positives due to signal noise caused by factors such as poor DNA quality, batch effects or inherent limitations of CNV detection algorithms. The lack of an optimal gold standard dataset is a major challenge in the development and evaluation of CNV detection tools. In the end, a true "Gold Standard" dataset can only be generated through the production of high quality telomere-to-telomere pangenomes by combining different read technologies across numerous genomes from various ancestral backgrounds (T. Wang et al., 2022).

Conclusions

In this thorough benchmark study, we evaluated the performance of five popular SNP-array-based CNV calling methods: PennCNV, QuantiSNP, iPattern, EnsembleCNV and R-GADA, using data from the DRAGEN re-analysis of the 1000 Genomes project as a gold standard. The analysis focused on key metrics such as recall, precision, F1 score, and categorized CNVs based on CNV type and size.

The results demonstrate significant variability in the performance of different CNV calling methods. PennCNV consistently achieved the highest precision and F1 score across various thresholds, indicating its reliability in CNV detection. R-GADA, while showing the highest recall, tended to call many false positive CNVs, indicating a higher sensitivity to noise. EnsembleCNV, leveraging a combination of methods, presented a balanced performance but did not surpass PennCNV in precision or R-GADA in recall.

Our findings also highlight the challenges associated with CNV detection in SNP array data. The low consensus between different callers, as pointed out by the minimal overlap in detected CNV segments, underscores the necessity for cautious interpretation of results and need for potential integration of multiple tools for comprehensive analysis.

In addition, reproducible CNV calling and benchmark workflows were developed in Nextflow, accompanied by Docker containers, making the application of these tools as well as execution of the benchmark highly scalable for future studies. In conclusion, while no single tool outperformed others across all metrics, PennCNV is deemed to be the most reliable tool for CNV calling based on the precision and F1 scores. As all methods are either limited in recall or precision, novel methods for CNV calling or for combining high-quality calls of multiple tools are necessary.

Abbreviations

BAF = B-Allele Frequency

BE = Backwards Elimination

CNP = Copy Number Polymorphisms

CNV = Copy Number Variation

CNVR = Copy Number Variation Region

DGV = Database of Genomic Variants

DRAGEN = Dynamic Read Analysis for GENomics

FDR = False Discovery Rate

FN = False Negative

FP = False Positive

GADA = Genome Alteration Detection Analysis

GQ = Genotype Quality

HMM = Hidden Markov Model

kb = Kilo base pair

LRR = Log R Ratio

MLPA = Multiplex Ligation-dependent Probe Amplification

NHGRI = National Human Genome Research Institute

NGS = Next Generation Sequencing

OB-HMM = Objective Bayes Hidden-Markov Model

SBL = Sparse Bayesian Learning

SNP = Single Nucleotide Polymorphism

TP = True Positive

QC = Quality Control

WGS = Whole Genome Sequencing

Funding sources

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Auton, A., Abecasis, G. R., Altshuler, D. M., Durbin, R. M., Bentley, D. R., Chakravarti, A., Clark, A. G., Donnelly, P., Eichler, E. E., Flicek, P., Gabriel, S. B., Gibbs, R. A., Green, E. D., Hurles, M. E., Knoppers, B. M., Korbel, J. O., Lander, E. S., Lee, C., Lehrach, H., … Schloss, J. A. (2015). A global reference for human genetic variation. *Nature 2015 526:7571*, *526*(7571), 68–74. https://doi.org/10.1038/nature15393

Behera, S., Catreux, S., Rossi, M., Truong, S., Huang, Z., Ruehle, M., Visvanath, A., Parnaby, G., Roddey, C., Onuchic, V., Cameron, D. L., English, A., Mehtalia, S., Han, J., Mehio, R., & Sedlazeck, F. J. (2024). Comprehensive and accurate genome analysis at scale using DRAGEN accelerated algorithms. *BioRxiv*, 2024.01.02.573821. https://doi.org/10.1101/2024.01.02.573821

Bick, A. G., Metcalf, G. A., Mayo, K. R., Lichtenstein, L., Rura, S., Carroll, R. J., Musick, A., Linder, J. E., Jordan, I. K., Nagar, S. D., Sharma, S., Meller, R., Basford, M., Boerwinkle, E., Cicek, M. S., Doheny, K. F., Eichler, E. E., Gabriel, S., Gibbs, R. A., … Denny, J. C. (2024). Genomic data in the All of Us Research Program. *Nature 2024 627:8003*, *627*(8003), 340–346. https://doi.org/10.1038/s41586-023-06957-x

Chen, K., Wallis, J. W., McLellan, M. D., Larson, D. E., Kalicki, J. M., Pohl, C. S., McGrath, S. D., Wendl, M. C., Zhang, Q., Locke, D. P., Shi, X., Fulton, R. S., Ley, T. J., Wilson, R. K., Ding, L., & Mardis, E. R. (2009). BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nature Methods 2009 6:9*, *6*(9), 677–681. https://doi.org/10.1038/nmeth.1363

Colella, S., Yau, C., Taylor, J. M., Mirza, G., Butler, H., Clouston, P., Bassett, A. S., Seller, A., Holmes, C.

C., & Ragoussis, J. (2007). QuantiSNP: an Objective Bayes Hidden-Markov Model to detect and accurately map copy number variation using SNP genotyping data. *Nucleic Acids Research*, *35*(6), 2013. https://doi.org/10.1093/NAR/GKM076

DI Tommaso, P., Chatzou, M., Floden, E. W., Barja, P. P., Palumbo, E., & Notredame, C. (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology 2017 35:4*, *35*(4), 316–319. https://doi.org/10.1038/nbt.3820

Eghbal-Zadeh, H., Fischer, L., Popitsch, N., Kromp, F., Taschner-Mandl, S., Gerber, T., Bozsaky, E., Ambros, P. F., Ambros, I. M., Widmer, G., & Moser, B. A. (2019). DeepSNP: An End-to-End Deep Neural Network with Attention-Based Localization for Breakpoint Detection in Single-Nucleotide Polymorphism Array Genomic Data. *Journal of Computational Biology*, *26*(6), 572–596. https://doi.org/10.1089/cmb.2018.0172

Gamazon, E. R., & Stranger, B. E. (2015). The impact of human copy number variation on gene expression. *Briefings in Functional Genomics*, *14*(5), 352. https://doi.org/10.1093/BFGP/ELV017

Gan, P., Hajis, M. I. Bin, Yumna, M., Haruman, J., Matoha, H. K., Wahyudi, D. T., Silalahi, S., Oktariani, D. R., Dela, F., Annisa, T., Pitaloka, T. D. A., Adhiwijaya, P. K., Pauzi, R. Y., Hertanto, R., Kumaheri, M. A., Sani, L., Irwanto, A., Pradipta, A., Chomchopbun, K., & Gonzalez-Porta, M. (2024). Development and validation of a pharmacogenomics reporting workflow based on the illumina global screening array chip. *Frontiers in Pharmacology*, *15*, 1349203. https://doi.org/10.3389/FPHAR.2024.1349203/BIBTEX

Gunturu, K. S., Yao, X., Cong, X., Thumar, J. R., Hochster, H. S., Stein, S. M., & Lacy, J. (2013). FOLFIRINOX for locally advanced and metastatic pancreatic cancer: Single institution retrospective review of efficacy and toxicity. *Medical Oncology*, *30*(1), 1–7. https://doi.org/10.1007/S12032-012-0361-2/TABLES/4

Haraksingh, R. R., Abyzov, A., & Urban, A. E. (2017). Comprehensive performance comparison of high-resolution array platforms for genome-wide Copy Number Variation (CNV) analysis in humans. *BMC Genomics*, *18*(1). https://doi.org/10.1186/S12864-017-3658-X

Hu, W., Li, M., Zhang, Q., Liu, C., Wang, X., Li, J., Qiu, S., & Li, L. (2021). Establishment of a novel CNV-related prognostic signature predicting prognosis in patients with breast cancer. *Journal of Ovarian Research*, *14*(1). https://doi.org/10.1186/S13048-021-00823-Y

Koboldt, D. C., Zhang, Q., Larson, D. E., Shen, D., McLellan, M. D., Lin, L., Miller, C. A., Mardis, E. R., Ding, L., & Wilson, R. K. (2012). VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Research*, *22*(3), 568–576. https://doi.org/10.1101/GR.129684.111

Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R., & Pfister, H. (2014). UpSet: Visualization of Intersecting Sets. *IEEE Transactions on Visualization and Computer Graphics*, *20*(12), 1983. https://doi.org/10.1109/TVCG.2014.2346248

MacDonald, J. R., Ziman, R., Yuen, R. K. C., Feuk, L., & Scherer, S. W. (2014). The Database of Genomic Variants: a curated collection of structural variation in the human genome. *Nucleic Acids Research*, *42*(Database issue). https://doi.org/10.1093/NAR/GKT958

Marenne, G., Rodríguez-Santiago, B., Closas, M. G., Pérez-Jurado, L., Rothman, N., Rico, D., Pita, G., Pisano, D. G., Kogevinas, M., Silverman, D. T., Valencia, A., Real, F. X., Chanock, S. J., Génin, E., & Malats, N. (2011). Assessment of Copy Number Variation Using the Illumina Infinium 1M SNP-Array: A Comparison of Methodological Approaches in the Spanish Bladder Cancer/EPICURO Study. *Human Mutation*, *32*(2), 240. https://doi.org/10.1002/HUMU.21398

Marshall, C. R., Howrigan, D. P., Merico, D., Thiruvahindrapuram, B., Wu, W., Greer, D. S., Antaki, D.,

Shetty, A., Holmans, P. A., Pinto, D., Gujral, M., Brandler, W. M., Malhotra, D., Wang, Z., Fuentes Fajarado, K. V., Maile, M. S., Ripke, S., Agartz, I., Albus, M., … Sebat, J. (2016). Contribution of copy number variants to schizophrenia from a genome-wide study of 41,321 subjects. *Nature Genetics 2016 49:1*, *49*(1), 27–35. https://doi.org/10.1038/ng.3725

Merkel, D. (2014). Docker: lightweight Linux containers for consistent development and deployment. *Linux Journal*, *239*, 2. https://doi.org/10.5555/2600239.2600241

Nowakowska, B. (2017). Clinical interpretation of copy number variants in the human genome. *Journal of Applied Genetics*, *58*(4), 449–457. https://doi.org/10.1007/S13353-017-0407-4

Nutsua, M. E., Fischer, A., Nebel, A., Hofmann, S., Schreiber, S., Krawczak, M., & Nothnagel, M. (2015). Family-Based Benchmarking of Copy Number Variation Detection Software. *PLOS ONE*, *10*(7), e0133465. https://doi.org/10.1371/JOURNAL.PONE.0133465

Olshen, A. B., Venkatraman, E. S., Lucito, R., & Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics (Oxford, England)*, *5*(4), 557–572. https://doi.org/10.1093/BIOSTATISTICS/KXH008

Olson, N. D., Wagner, J., McDaniel, J., Stephens, S. H., Westreich, S. T., Prasanna, A. G., Johanson, E., Boja, E., Maier, E. J., Serang, O., Jáspez, D., Lorenzo-Salazar, J. M., Muñoz-Barrera, A., Rubio-Rodríguez, L. A., Flores, C., Kyriakidis, K., Malousi, A., Shafin, K., Pesout, T., … Zook, J. M. (2022). PrecisionFDA Truth Challenge V2: Calling variants from short and long reads in difficult-to-map regions. *Cell Genomics*, *2*(5), 100129. https://doi.org/10.1016/J.XGEN.2022.100129

Pinto, D., Pagnamenta, A. T., Klei, L., Anney, R., Merico, D., Regan, R., Conroy, J., Magalhaes, T. R., Correia, C., Abrahams, B. S., Almeida, J., Bacchelli, E., Bader, G. D., Bailey, A. J., Baird, G., Battaglia, A., Berney, T., Bolshakova, N., Bölte, S., … Betancur, C. (2010). Functional Impact of Global Rare Copy Number Variation in Autism Spectrum Disorder. *Nature*, *466*(7304), 368. https://doi.org/10.1038/NATURE09146

Pique-Regi, R., Cáceres, A., & González, J. R. (2010). R-Gada: A fast and flexible pipeline for copy number analysis in association studies. *BMC Bioinformatics*, *11*(1), 1–12. https://doi.org/10.1186/1471-2105-11-380/FIGURES/10

Redon, R., Ishikawa, S., Fitch, K. R., Feuk, L., Perry, G. H., Andrews, T. D., Fiegler, H., Shapero, M. H., Carson, A. R., Chen, W., Cho, E. K., Dallaire, S., Freeman, J. L., González, J. R., Gratacòs, M., Huang, J., Kalaitzopoulos, D., Komura, D., MacDonald, J. R., … Hurles, M. E. (2006). Global variation in copy number in the human genome. *Nature 2006 444:7118*, *444*(7118), 444–454. https://doi.org/10.1038/nature05329

Team BC, & Maintainer BP. (2019). TxDb.Hsapiens.UCSC.hg38.knownGene: Annotation package for TxDb object(s). In *R package version 3.4.6* (R package version 3.2.2).

Wang, K., Li, M., Hadley, D., Liu, R., Glessner, J., Grant, S. F. A., Hakonarson, H., & Bucan, M. (2007). PennCNV: An integrated hidden Markov model designed for high-resolution copy number variation detection in whole-genome SNP genotyping data. *Genome Research*, *17*(11), 1665. https://doi.org/10.1101/GR.6861907

Wang, K., Yuen, S. T., Xu, J., Lee, S. P., Yan, H. H. N., Shi, S. T., Siu, H. C., Deng, S., Chu, K. M., Law, S., Chan, K. H., Chan, A. S. Y., Tsui, W. Y., Ho, S. L., Chan, A. K. W., Man, J. L. K., Foglizzo, V., Ng, M. K., Chan, A. S., … Leung, S. Y. (2014). Whole-genome sequencing and comprehensive molecular profiling identify new driver mutations in gastric cancer. *Nature Genetics 2014 46:6*, *46*(6), 573–582. https://doi.org/10.1038/ng.2983

Wang, T., Antonacci-Fulton, L., Howe, K., Lawson, H. A., Lucas, J. K., Phillippy, A. M., Popejoy, A. B., Asri, M., Carson, C., Chaisson, M. J. P., Chang, X., Cook-Deegan, R., Felsenfeld, A. L., Fulton, R. S.,

Garrison, E. P., Garrison, N. A., Graves-Lindsay, T. A., Ji, H., Kenny, E. E., … Haussler, D. (2022). The Human Pangenome Project: a global resource to map genomic diversity. *Nature 2022 604:7906*, *604*(7906), 437–446. https://doi.org/10.1038/s41586-022-04601-8

Weber, L. M., Saelens, W., Cannoodt, R., Soneson, C., Hapfelmeier, A., Gardner, P. P., Boulesteix, A. L., Saeys, Y., & Robinson, M. D. (2019). Essential guidelines for computational method benchmarking. *Genome Biology*, *20*(1), 1–12. https://doi.org/10.1186/S13059-019-1738-8/FIGURES/1

Winchester, L., Yau, C., & Ragoussis, J. (2009). Comparing CNV detection methods for SNP arrays. *Briefings in Functional Genomics*, *8*(5), 353–366. https://doi.org/10.1093/BFGP/ELP017

Xu, L., Hou, Y., Bickhart, D. M., Song, J., & Liu, G. E. (2013). Comparative Analysis of CNV Calling Algorithms: Literature Survey and a Case Study Using Bovine High-Density SNP Data. *Microarrays*, *2*(3), 171. https://doi.org/10.3390/MICROARRAYS2030171

Ye, K., Schulz, M. H., Long, Q., Apweiler, R., & Ning, Z. (2009). Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*, *25*(21), 2865–2871. https://doi.org/10.1093/BIOINFORMATICS/BTP394

Zarrei, M., MacDonald, J. R., Merico, D., & Scherer, S. W. (2015). A copy number variation map of the human genome. *Nature Reviews Genetics 2015 16:3*, *16*(3), 172–183. https://doi.org/10.1038/nrg3871

Zhang, F., Gu, W., Hurles, M. E., & Lupski, J. R. (2009). Copy Number Variation in Human Health, Disease, and Evolution. *Annual Review of Genomics and Human Genetics*, *10*, 451. https://doi.org/10.1146/ANNUREV.GENOM.9.081307.164217

Zhang, Z., Cheng, H., Hong, X., Di Narzo, A. F., Franzen, O., Peng, S., Ruusalepp, A., Kovacic, J. C., Bjorkegren, J. L. M., Wang, X., & Hao, K. (2019). EnsembleCNV: an ensemble machine learning algorithm to identify and genotype copy number variation using SNP array data. *Nucleic Acids Research*, *47*(7), e39–e39. https://doi.org/10.1093/NAR/GKZ068