OXFORD

Briefings in Bioinformatics, 2022, **23(5)**, 1–9
https://doi.org/10.1093/bib/bbac375
Problem Solving Protocol

PEcnv: accurate and efficient detection of copy number variations of various lengths

Xuwen Wang, Ying Xu, Ruoyu Liu, Xin Lai, Yuqian Liu, Shenjie Wang, Xuanping Zhang and Jiayin Wang

Corresponding author: Jiayin Wang, Shaanxi Engineering Research Center of Medical and Health Big Data, School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China. E-mail: wangjiayin@mail.xjtu.edu.cn

Abstract

Copy number variation (CNV) is a class of key biomarkers in many complex traits and diseases. Detecting CNV from sequencing data is a substantial bioinformatics problem and a standard requirement in clinical practice. Although many proposed CNV detection approaches exist, the core statistical model at their foundation is weakened by two critical computational issues: (i) identifying the optimal setting on the sliding window and (ii) correcting for bias and noise. We designed a statistical process model to overcome these limitations by calculating regional read depths via an exponentially weighted moving average strategy. A one-run detection of CNVs of various lengths is then achieved by a dynamic sliding window, whose size is self-adopted according to the weighted averages. We also designed a novel bias/noise reduction model, accompanied by the moving average, which can handle complicated patterns and extend training data. This model, called PEcnv, accurately detects CNVs ranging from kb-scale to chromosome-arm level. The model performance was validated with simulation samples and real samples. Comparative analysis showed that PEcnv outperforms current popular approaches. Notably, PEcnv provided considerable advantages in detecting small CNVs (1 kb–1 Mb) in panel sequencing data. Thus, PEcnv fills the gap left by existing methods focusing on large CNVs. PEcnv may have broad applications in clinical testing where panel sequencing is the dominant strategy. Availability and implementation: Source code is freely available at https://github.com/Sherwin-xjtu/PEcnv

Keywords: genomics, sequencing data analysis, variant detection, copy number variation (CNV), exponentially weighted moving average, clinical panel sequencing

Introduction

Copy number variation (CNV) refers to the deletion and duplication of DNA fragments. Their sizes range from thousands to several million base pairs [1–3]. CNVs are common, comprising more than 12% of the human genome [4, 5]. CNVs play a crucial role in the diagnosis and treatment of various complex diseases [6], including cancers [7], neuropsychiatric illness [8] and Huntington's disease [6, 9]. Thus, detecting CNVs became routine in clinical laboratory practice. Compared to traditional technologies, such as fluorescence in situ hybridization (FISH) [10] and array comparative genomic hybridization (array CGH) [11], sequencing-based approaches are popular due to their higher resolution, better efficiency and lower cost [12–15]. The past decade has seen the development of several bioinformatics approaches for detecting CNVs in next-generation sequencing (NGS) data. Zare [16], Zhao [17] and others have comprehensively summarized the CNV detection approaches; here, we start from their conclusions. Most methods have been developed for whole genome sequencing (WGS) or whole-exome sequencing (WES) and several CNV tools for panel sequencing data have been developed (CONTRA [7], CoNVaDING [18]) as well. However, detecting CNVs of varying sizes is challenging, especially from panel sequencing data.

Why is the detection of CNVs of different sizes challenging for existing methods? Before better clarifying this question, we highlight some well-known methods (CONTRA [7], CNVKIT [19],

Xuwen Wang is studying for a Ph.D. in the School of Computer Science and Technology, Xi'an Jiaotong University, China. His research interests include bioinformatics and machine learning.
Ying Xu is an assistant professor of medical informatics at the School of Computer Science and Technology, Xi'an Jiaotong University. Her research interests include biomedical multi-omics data mining and machine learning.
Ruoyu Liu is studying for a Ph.D. in the School of Computer Science and Technology, Xi'an Jiaotong University, China. His research interests include bioinformatics and machine learning.
Xin Lai is an associate professor at the School of Computer Science and Technology, Xi'an Jiaotong University. His research interests include quality control on sequencing data, long-term survivor models, and risk and risk-adjusted monitoring approaches.
Yuqian Liu is an assistant professor at the School of Computer Science and Technology, Xi'an Jiaotong University. Her research interests are dynamics and control, adaptive and intelligent systems, and data mining.
Shenjie Wang is studying for a Ph.D. in the School of Computer Science and Technology, Xi'an Jiaotong University, China. His research interests include bioinformatics and machine learning.
Xuanping Zhang is a professor at the School of Computer Science and Technology, Xi'an Jiaotong University. His research interests include biological information processing, machine learning and data mining.
Jiayin Wang is a faculty member of the School of Computer Science and Technology, Xi'an Jiaotong University. His research interest includes the management issues in bioinformatics, computational biology and cancer genomics.
Received: April 23, 2022. **Revised:** June 19, 2022. **Accepted:** August 8, 2022

FACETS [20]) based on the read depth (RD) strategy [21, 22]. The RD strategy consists of (1) read depth preprocessing to control for bias and noise [23] and (2) locating copy number breakpoints by segmenting the sequenced regions and identifying those segments, which may harbor CNVs. A non-overlapping static sliding window is usually adopted here. The number of mapped reads is calculated in each window bin, and then $\log_2$ copy ratios (LogR) are computed as the read count versus the count from the control (paired) sample or reference for bin or region [16, 24]. LogR is the popular statistical indicator of CNVs [20]. Current approaches present various algorithms and statistical models according to the RD strategy. They primarily differ in their focus on either region segmentation or bias/noise correction [16, 17]. However, these two steps present challenges preventing the better application of RD strategies when handling panel sequencing data. The main challenge is that unsuitable window lengths destroy the signals for short-to-medium copy number variations.

As mentioned above, current methods adopt a static sliding window bin in which the average sequencing depth is calculated to obtain LogR. The size of the window bin directly affects the detection of differently sized CNVs. If the window bin is too large, short CNVs cannot be detected, leading to false-negative errors. A window that is too small yields false-positive errors. We can formulate the problem as an 'optimal window bin size' problem, explained in the following examples (see Supplementary 1).

Moreover, due to inter-individual variability (tumor heterogeneity in cancer sequencing scenario), the size distribution of CNVs carried by different patients is inconsistent [16, 17, 20]. Therefore, existing methods that use a static sliding window cannot precisely identify CNVs of varying sizes. Perhaps an exhaustive enumeration of all window sizes may solve this problem, but it dramatically increases the time required for analysis and introduces false-positive results due to bias. In addition, panel sequencing data yield complicated patterns of systematic errors, and it is difficult to detect CNVs that can vary in size and copy number in panel sequencing data (details in Supplementary 2).

We present a novel approach, PEcnv, to address the limitations of traditional methods. The key features of PEcnv are threefold.

(1) Adjusting base-level coverage. We pioneered a strategy to use base coverage information around the target base to correct its coverage by the exponentially weighted moving average. Considering base coverage around the target base can effectively solve the complex distribution problem of the read depth. The probability of consecutive low base coverage is not significant in real sequencing; thus, we significantly increase the number of control samples.
(2) Improved identification of varying sizes of CNVs by using a dynamic sliding window. We divide the genome into candidate and non-candidate CNV regions and set the dynamic sliding window bin sizes according to the different regions in the bias correction and segmentation steps. This novel strategy helps detect all CNVs of different sizes simultaneously.
(3) Our method applies to panel sequencing as well as WGS and WES. PEcnv can precisely identify variously sized CNVs simultaneously. We tested and validated PEcnv's performance detecting CNVs on simulation datasets and real sequenced samples. We also compared its performance to several existing methods. The results showed that PEcnv performs well with sensitivity, precision and f1-score, especially for detecting small CNVs (1 kb ~ 1 Mb) in panel sequencing data. Furthermore, our method can complement existing methods and easily be integrated into existing analysis pipelines for CNV.

Materials and methods

The input of PEcnv is a case sequencing mapped reads file (BAM format [25]), a control sequencing mapped reads file (BAM format), and a reference sequence file (FASTA format). The output is a CNV calling report file (CSV file). As an RD strategy-based approach, the key points of PEcnv include adjusting base coverage by accounting for reading depth information around the base, dividing the genome into the candidate and non-candidate CNV regions, and setting the dynamic sliding window bin sizes according to the different regions from the bias of correction and segmentation steps. Key components of PEcnv are presented here, while the whole pipeline is illustrated in Supplementary 3.

Adjusting base-level coverage

One of the challenges of CNV detection for panel sequencing data is the uneven RD distribution. We take several steps to reduce systematic biases to correct RD. First, we use the reads from intergenic (off-target) reads and, usually, intronic regions (target reads) shown in previous studies [19, 26]. Second, we use the coverage information of the bases around the targeted base to correct its coverage. We correct the coverage of the bases in both the case and control samples. The coverage of the surrounding bases corrects the coverage of the target base. We adopt an exponentially weighted moving average strategy to adjust the coverage of the targeted base. Each coverage value is weighted exponentially, decreasing with distance from the targeted base b, with the closer base weighted more heavily. Still, a more distant base also adds some weight. Base-level coverage is then computed for each targeted base.

$$d_b = \lambda \overline{X_b} + (1-\lambda)\, d_{b-1}, 0 < \lambda < 1, d_0 = \mu_0 \tag{1}$$

$$\overline{X_b} = \frac{\left(\sum_{i=1}^{N_1} x_{fi} + \sum_{j=1}^{N_2} x_{rj}\right)}{N}, N \geq N_1 + N_2 \tag{2}$$

where d_b is the adjusted coverage of the targeted base b, d_{b-1} is the adjusted coverage of the targeted base 5′ of b. We assume that the $x_{fi} \in x_f = \{x_{f1}, x_{f2}, \cdots, x_{fN_1}\}$ is the coverage of i-th base 5′ to the targeted base b and the $x_{rj} \in x_r = \{x_{r1}, x_{r2}, \cdots, x_{rN_2}\}$ is the coverage of j-th base 3′ to the targeted base b. $\overline{X_b}$ is the raw coverage around the targeted base. N is the number of bases around the targeted base, N_1 is the number of bases 5′ to the targeted base b, and N_2 is the number of bases 3′ to the targeted base b. The term $\lambda \in [0, 1]$ is a constant. μ_0 is the centerline or the average value of all genome coverage. Supplementary 3 describes some of the model parameter settings.

We correct the coverage of the targeted base using Equations (1) and (2). This effectively solves the complex distribution problem of read depth and creates a robust baseline.

Identifying candidate CNV regions via a dynamic statistical process

Identifying candidate and non-candidate CNV regions

To detect different sizes of CNV simultaneously, we designed a novel two-stage strategy of dynamic sliding windows. The first stage divides the genome into the candidate and non-candidate CNV regions (Figure 1). It is well established that the LogR of the abnormal region on the genome is likely to be a CNV. We

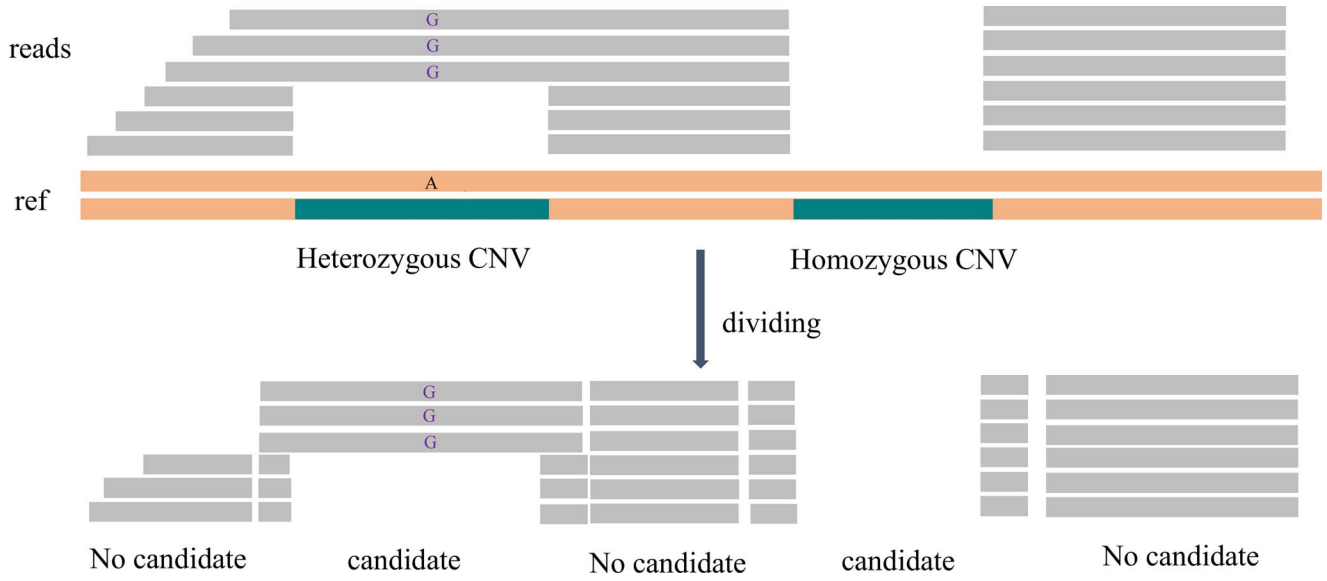

Figure 1. The workflow for identifying the candidate and non-candidate CNV regions.

use control charts to find these exceptions [27] via a statistical process. The system performs a statistical analysis of the adjusted coverage of each base and then finds the abnormal region in the shortest possible fragment. The adjusted coverage of each base is a statistic derived from the raw coverage of the target base and the raw coverages of the bases around the target base, as described in Equation (1).

Unlike the exhaustive method, our method does not require all past values to be saved, significantly reducing computational effort and the spatial complexity of processing massive amounts of sequencing data. The other benefit of the control limits is that they are not significantly affected when a small or large value is added to the calculation, thus helping to reduce the effect of noise. On the other hand, finding abnormal regions in the shortest possible fragment is similar to comparing whether the LogR of the current interval is significantly different from that of the previous interval. The length of the current and previous intervals change depending on the statistical characteristics of the control charts. We thus use the control limits to divide the genome sequences into candidate and non-candidate CNV regions [27]. The region between the control limits is considered a non-candidate CNV region, and the region outside the control limits is regarded as a candidate CNV region. The control limits are as follows (Equation (3)) [27].

$$\begin{cases} UCL = \mu_0 + L\sigma_{d_b}, \\ LCL = \mu_0 - L\sigma_{d_b}, \end{cases} \tag{3}$$

$$\mu_0 = E\left(d_b\right) = E\left(\overline{X_b}\right) \tag{4}$$

$$\sigma_{d_b} = \sqrt{\left(\frac{\lambda}{1-\lambda}\right)\left[1-(1-\lambda)^{2b}\right] * Var\left(\overline{X_b}\right)} \tag{5}$$

The expected d_b value μ_0 and standard deviation σ_{d_b} can be calculated as Equations (4) and (5), respectively [27, 28]. Therefore, the control limits can also be calculated (Equations (6)).

$$\begin{cases} UCL = \mu_0 + L\sigma\sqrt{\frac{\lambda}{2-\lambda}\left[1-(1-\lambda)^{2b}\right]}, \sigma = \sqrt{Var\left(\overline{X_b}\right)}, \\ LCL = \mu_0 - L\sigma\sqrt{\frac{\lambda}{2-\lambda}\left[1-(1-\lambda)^{2b}\right]}, \sigma = \sqrt{Var\left(\overline{X_b}\right)}, \end{cases} \tag{6}$$

where μ_0 is the centerline or the average value of all genome coverage and σ is the standard deviation of the raw coverage around the targeted base. The σ_{d_b} is the expected value of the standard deviation of the adjusted coverage around the targeted base. L is the parameter that needs to be selected. UCL is the upper limit of the 'alarm', and LCL is the lower limit of the 'notice'. An alarm is triggered when $d_b <$ LCL, the average value of the process drifts down. Thus, there may be a deletion of the CNV region. However, this method cannot accurately identify the boundaries of the CNV region, so we define this genome region as a candidate CNV region for deletion. An alarm is triggered when $d_b >$ UCL, the average value of the process drifts upward. As a result, there may be a duplication of the CNV region, and we define this region as the candidate CNV region of duplication. When LCL $< d_b <$ UCL, there may be neither deletion nor duplication of the CNV region; we define this as a non-candidate CNV region. Figure 2 depicts the dividing method workflow. We denote the candidate CNV regions $CR = \{R_1, R_2, \cdots, R_M\}$ and the non-candidate CNV regions as $NCR = \{R_1, R_2, \cdots, R_P\}$, R_i belongs to CR or NCR. It can be expressed as $R_i = (s, d)$, where s and d are the start and end positions in the genome region.

Setting the dynamic sliding window bin

This step refines the window bins to fit dynamic regions. We propose adopting a dynamic sliding window bin, distinct from existing methods. Based on candidate CNV and non-candidate CNV regions, we can set the different sizes of window bins. The size of the bin is dependent on the size of the region and can be calculated as:

$$w_i = l_i * s \tag{9}$$

where w_i is the size of the sliding window bin of ith region. l_i is the length of the region and s is the smoothing coefficient for all regions. We calculate LogR across the window bin w_i for each region (non-candidate or candidate CNV). This dynamic process is helpful for later bias correction and region segmentation and helps accurately detect different sizes of CNVs simultaneously. In Supplementary 3, we further describe the parameter settings.

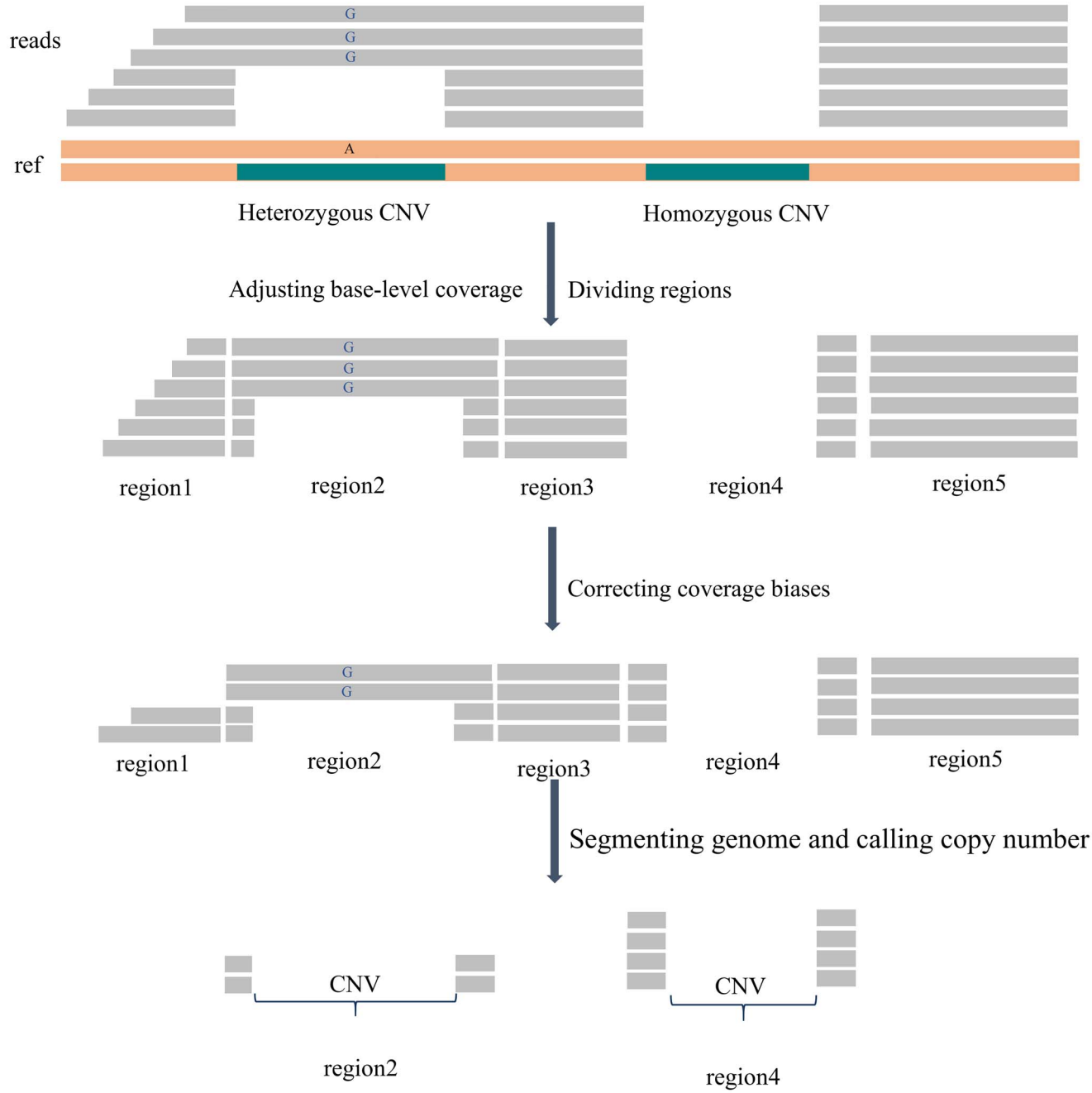

Figure 2. The PEcnv workflow.

Generating sequencing data

We produced simulation data by GSDcreator [29] (details in Supplementary 4). These experiments were performed as follows: we simulated paired samples for panel and WES with the read length being 75 bp. We also simulated various gradients of coverage depth, tumor purity, CNV type, CNV size for every panel and WES samples. We were thus able to analyze the model's performance from different perspectives precisely. For panel samples, we simulated sequencing data with coverage depth from 500× to 2000×, tumor purity from 0.2 to 0.8, CNV absolute copy number from 0 to 6, and CNV size from 1 kb to 10 Mb. For WES samples, we simulated sequencing data with a coverage depth of 60× with tumor purity of 0.67, CNV absolute copy number from 0 to 6 and CNV size from 1 kb to 10 Mb.

A copy number variant is an element that may be present in variable numbers of copies in the genome. Therefore, we defined more or fewer copies of one element here as copy number variants. For example, a deletion, duplication or unbalanced translocation is considered a copy number variant [30, 31]. The final copy number status of chromosomal segments is determined by correctly adjusting template sequence numbers. Specifically, we enlarge or decrease the number of template sequences in the copy number variant region according to the preset copy multiple. As the number of copies increases, the number of templates will increase according to the amplification ratio. According to the

reduction ratio, if the number of copies reduces, the number of templates will be reduced [29].

Samples from the 1000 genomes project

Twenty-three human individuals studied in both the HapMap and the 1000 Genomes project were selected to evaluate the model's performance. The study group comprised Asian, African and European individuals. The Asian samples IDs are NA18537, NA18542, NA18547, NA18552, NA18564, NA18566, NA18570, NA18582, NA18592, NA18942, NA18947, NA18969, NA18997, NA18951, NA18972, NA18968 and NA18973. We also evaluated European samples, NA10851, NA11893, NA12413, NA12775, NA12878 and African samples NA19240. The gold standard CNV calls were obtained from http://dgv.tcag.ca/dgv/app/downloads, https://www.ncbi.nlm.nih.gov/genome, and the HapMap website. The exome sequencing data (bam files) were downloaded from the 1000Genomes project website.

Results

We developed PEcnv, a novel method for the simultaneous detection of CNVs of different sizes based on read depth. PEcnv is a dynamic statistical process model based on the exponentially weighted moving average strategy. A dynamic sliding window achieves a one-run detection of varied-length CNVs, the size of which is self-adopted according to the weighted averages. We also defined a novel bias/noise reduction model, accompanied by the moving average, allowing the model to accommodate complicated patterns and extend training data. We analyzed the performance of PEcnv in the context of CNV size and RD based on the simulation data and real sample data. Other CNV caller tools, including CNVKIT, CONTRA and FACETS, were used for comparison. To assess the ability of our approach to correct coverage bias, we constructed kitPEcnv, a model only for segmentation, without the bias reduction step. We compared the sensitivity, precision and F1-score of PEcnv and other methods, using the default parameters for each model. We obtained CNV results from each caller and defined calls with at least 50% overlap as matches [19].

Evaluating CNV models with simulation samples

We compared the performance of PEcnv, CNVKIT, CONTRA and FACETS with a simulated dataset. The 180-panel sequencing case series had CNV sizes of 1 kbp to 10 Mbp, depth from 500× to 2000×, and tumor purity of 0.2 to 0.8. Each case contains 171 CNVs. We also simulated a matched control for each case. Here, we only evaluated the performance of each method in detecting different sizes of CNVs and the different RDs. Supplementary 4 describes other evaluations and method comparisons with various absolute copy number calls and read lengths.

Performance on detecting different sizes of CNVs

We compared the performance of PEcnv with CNVKIT, CONTRA and FACETS with simulated panel sequencing cases with CNVs ranging from 1 kbp to 10 Mbp. Overall, the sensitivity of all methods increased with increasing CNV size (Table 1). PEcnv provided the highest sensitivity for detecting small CNVs (1–10 kb, 10–100 kb; 0.86, 0.88) as compared with 0.81 and 0.73 for CONTRA, the second-best performer (Figure 3A). PEcnv also had the highest sensitivity (0.89, 0.90) for detecting large CNVs (100–1000 kb, 1–1 Mb) compared with FACETS, the second-best performer, which had sensitivities of 0.69 and 0.79. The kitPEcnv model also performs very well, maintaining a detection sensitivity above 0.67 for different sizes of CNVs. In general, for detecting CNVs of various sizes, PEcnv had greater sensitivity than all other comparator methods. CONTRA is more sensitive to small CNVs than other methods (except our model), and FACETS is more sensitive to large CNVs than other methods (except our model). To further demonstrate our method's effectiveness, we compared it with others in terms of precision and F1-scores (Supplement 4). In some cases, our method was slightly less precise than other methods, but overall, not much worse (Supplemental Figure 2A). In general, for detecting CNV of various sizes, the F1-score of PEcnv is always higher (0.03 ~ 0.13) than that of other methods (Supplemental Figure 2B).

We also compared our method with other methods using simulated WES data. We simulated sequencing data with a sequencing coverage depth of 60×, tumor purity of 0.67, CNV absolute copy number from 0 to 6, and CNV size from 1 kb to 10 Mb. Our approach is more sensitive for detecting small CNVs, detecting 373/392 versus 246/392 for the second-best performer, CNVKIT. In contrast, the detection sensitivity for large CNVs remains unchanged, with 296/308 for PEcnv versus 248/308 for FACETS, the second-best performer (Figure 3B). The results show that the sensitivity of PEcnv is 0.957 versus 0.757 for the second-best performer, CNVKIT (Table 2). The precision of our method is slightly lower than other methods. The F1-score of PEcnv is 0.886 versus 0.784 obtained by the second-best performer, CNVKIT. These findings suggest that our method works significantly better on WES data than existing methods. All tools produced good results with depth coverage of 60×, except for CONTRA.

Performance on different read depths

To evaluate the performance of PEcnv on different read depths, we compared PEcnv with CNVKIT, CONTRA and FACETS on simulated panel sequencing samples with depths from 500× to 2000× and tumor purity 0.2 to 0.8. The results show that the average sensitivity of PEcnv is 0.88 versus 0.71 obtained by CONTRA, the second-best performer (Figure 3C). When the tumor purity is greater than about 0.44, the sensitivity of all methods increases with increasing read depth. When the tumor purity is less than about 0.44, the sensitivity of all methods decreases slowly with decreasing tumor purity, even if the depth increases. These show that the effect of tumor purity is more significant than read depth in CNV detection, similar to results published elsewhere [32]. Experimental results also showed that CONTRA is less affected by tumor purity than the other algorithms tested. To further demonstrate the effectiveness of our method, we compared it to other methods in terms of precision (Supplemental Figure 3A) and F1-score (Supplemental Figure 3B). The results show that the average precision of PEcnv is 0.86 versus 0.89 obtained by CONTRA, the best performer and the average F1-score of PEcnv is 0.87 versus 0.78 obtained by CONTRA, the second-best performer.

Evaluating CNV models based on real samples with known CNV

We applied CNV models to panel sequencing data from 23 healthy human individuals that have been studied in both the International HapMap Project (www.hapmap.org) and the 1000 Genomes Project (www.1000genomes.org). Previous studies have tested the performance of CNV detection tools on datasets from these sources [7, 33, 34]. However, there is no benchmark for the panel sequencing data from the 1000 Genomes and HapMap projects. To consider bias in sequencing, we chose WES data instead of WGS to generate panel sequencing data. In a way, WES can be regarded as an extended version of panel sequencing, as the biases and errors caused during sequencing are similar

Table 1. Performance of various tools in detecting CNVs of differing sizes in simulated panel sequencing data

	CNV size	CNVKIT	kitPEcnv	PEcnv	CONTRA	FACETS
Sensitivity	1–10 k	0.44	0.67	**0.86**	0.81	0.21
	10–100 k	0.48	0.76	**0.88**	0.73	0.53
	100–1000 k	0.52	0.77	**0.90**	0.61	0.74
	1–10 M	0.70	0.78	**0.90**	0.71	0.78
Precision	1–10 k	0.77	0.82	**0.86**	0.86	0.77
	10–100 k	0.80	0.82	**0.85**	0.90	0.87
	100–1000 k	0.75	0.78	**0.88**	0.88	0.78
	1–10 M	0.83	0.83	**0.85**	0.92	0.93
F1-score	1–10 k	0.56	0.74	**0.86**	0.83	0.33
	10–100 k	0.60	0.79	**0.86**	0.80	0.66
	100–1000 k	0.61	0.78	**0.89**	0.72	0.76
	1–10 M	0.76	0.81	**0.88**	0.80	0.85

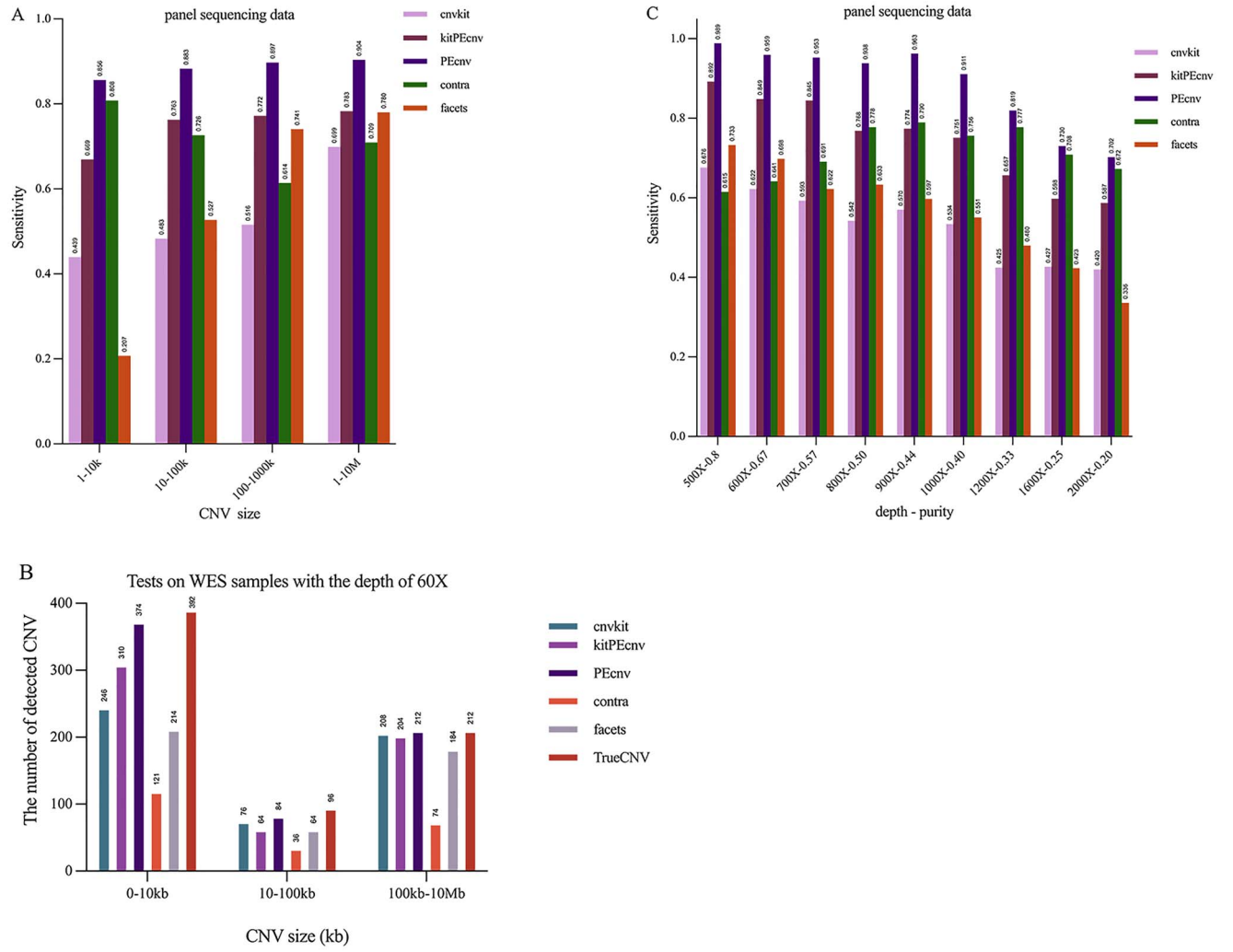

Figure 3. (**A**) The sensitivity of each method (CNVKIT, kitPEcnv, PEcnv, CONTRA, FACETS) for detecting different sizes of CNVs on the 180-panel samples. The tested result of each tool on the sequencing coverage depth from 500× to 2000×, tumor purity from 0.2 to 0.8 with the CNV size from 1 kb to 10 Mb. (**B**) The number of detected true CNV of each tool on WES samples (with the sequencing coverage depth 60×, tumor purity from 0.67), with the CNV size from 1 kb to 10 Mb. (**C**) The sensitivity of each method (CNVKIT, kitPEcnv, PEcnv, CONTRA, FACETS) for CNVs with simulated panel sequencing sample data with depth and tumor purity of 500× to 2000× and 0.2 to 0.8, respectively.

between panel sequencing and WES. The datasets from both sources were limited to those with WES data with coverage from 10× to 80×. We then randomly selected previously verified CNVs by IGV [35] and used these CNV regions to generate the panel bed and sequencing data files. For each sample, a region was considered real CNV if its HapMap copy is not two, and at least four of the remaining 23 samples have a copy number equal to 2 for that region [7, 34]. These 'known truth' CNVs ranged from 1 kb to 10 Mb. Previous studies used NA10851 as the control sample and the rest as case samples [7, 16]. It is worth mentioning that,

Table 2. The current method as it compares to other methods with simulated WES data

Methods	True CNV	TP	FP	Sensitivity	Specificity	F1-score
CNVKIT	700	530	122	0.757	0.812	0.784
kitPEcnv	700	578	132	0.826	0.814	0.820
PEcnv	**700**	**670**	**142**	**0.957**	**0.825**	**0.886**
CONTRA	700	231	3	0.33	0.987	0.494
FACETS	700	462	38	0.66	0.924	0.770

We simulated sequencing data with coverage depth of 60×, tumor purity is 0.67, CNV absolute copy number from 0 to 6 except 2 and CNV size from 1 kb to 10 Mb. Abbreviations: CNV: copy number variation. WES: whole-exome sequencing. TP: true positive copy number variation. FP: false-positive copy number variation. cn: absolute copy number.

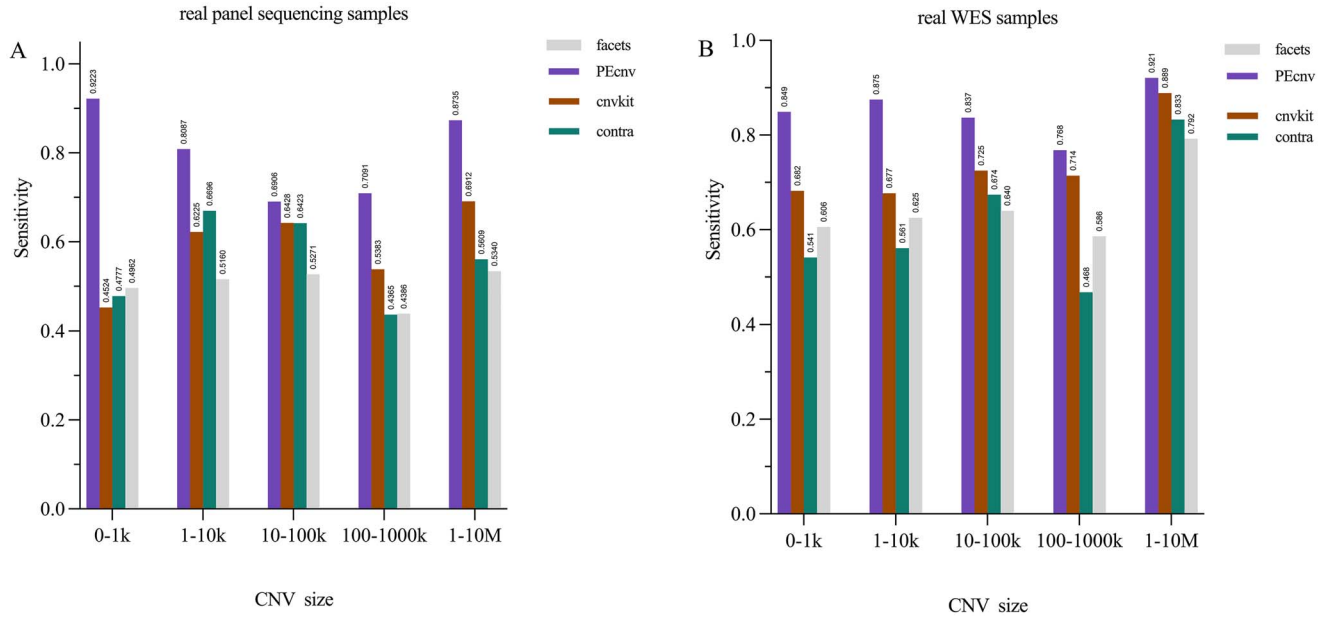

Figure 4. The average sensitivity of each method (CNVKIT, contra, PEcnv, FACETS) for detecting CNVs of differing sizes in real samples with (**A**) panel sequencing sample data and (**B**) sensitivity with WES sample data.

for the sake of fairness, we have chosen NA18051 as the control sample and the default parameters for all test tools.

We counted and analyzed the detection results of each CNV detection tool on 22 real samples for different sizes of CNV. The results show that the average detection sensitivity for different sizes of CNVs (0–1 kb, 1–10 kb, 10–100 kb, 100 kb–1 Mb, 1–10 Mb) was 0.92, 0.80, 0.69, 071 and 0.87 for PEcnv versus 0.50, 0.67, 0.64, 0.54 and 0.69 for the second-best performer (Figure 4A). We compared our method with other methods in terms of precision (Supplemental Figure 4A) and F1-score (Supplemental Figure 4B). The results show that the average precision of PEcnv is 0.98 versus 0.94 obtained by CNVKIT, the second-best performer and the average F1-score of PEcnv is 0.88 versus 0.72 obtained by CNVKIT, the second-best performer. We also evaluated each tool's performance on WES data. The results show that the average detection sensitivity for CNVs of varying sizes (0–1 kb, 1–10 kb, 10–100 kb, 100 kb–1 Mb, 1–10 Mb) was 0.88, 0.87, 0.83, 076, 0.92 for PEcnv versus 0.68, 0.68, 0.73, 0.71 and 0.89 for the second-best performer (Figure 4B). These results, presented in Figure 4, are similar to those in previously published studies [16, 34]. To further demonstrate the effectiveness of our method, we compared our method with other methods in terms of precision (Supplemental Figure 4C) and F1-score (Supplemental Figure 4D). PEcnv achieved an average precision of 0.95 and F1-score of 0.89. CNVKIT, the second-best performer achieved precision 0.96 and F1-score 0.83.

Discussion and conclusion

This paper presents PEcnv, a novel approach to detecting various sizes of CNVs based on WGS, WES and panel sequencing. The main task is detecting CNVs that can vary in size for panel sequencing data. During segmentation and CNV prediction, we divide the genome into candidate and non-candidate CNV regions and set the dynamic sliding window bin sizes according to the different regions in bias correction and segmentation steps. This method is more helpful than the non-overlapping static sliding window strategy. Our approach provides improved performance compared to other CNV detection processes and can be used to improve existing algorithms. PEcnv can be easily incorporated into existing CNV detection algorithms, and we believe it can help improve the detection accuracy of CNVs of different sizes. In addition, current strategies do not reduce the negative impact of errors in detecting CNV in panel sequencing data. PEcnv takes full advantage of the coverage information from the region surrounding the targeted base to correct for coverage, while other methods do not. Our model has some parameters that users can define according to their needs (Supplementary 3).

Through extensive simulated and real sequencing data analyses, we have demonstrated that PEcnv can precisely detect various sizes of CNVs, especially with panel sequencing data. The test results show that it is suitable for the simultaneous detection of CNVs of different sizes. We used the CNV detection tools for comparison support for single and multiple control samples to

build the reference set. However, some studies suggest there may be problems if multiple control samples are used, as they may increase the risk of false-negative or false-positive results [18, 30]. To avoid these problems, we used a single control sample to build the reference set for each CNV detection tool. Some methods were less effective in detecting CNV in our experiments may be due to the fact that we used one control sample instead of a strong sample pool, and most of our tests were for small CNVs. Still, our test results to evaluate CNV callers were similar to other tests, such as those published by Iria Roca et al. [36] and Talevich [19].

In the future, we will pursue two experimental aims. First, we found that the existing methods are not very effective when tumor purity is low, so we will continue to improve the detection accuracy of CNV in samples of low purity. Second, we will expand our method to detect genomic scars. A growing number of studies have demonstrated that genomic scars play an essential role in cancer research and that CNV detection is vital for accurately identifying genomic scars [37].

Author Contributions

J.W. and X.W. conceived and designed this research; X.W., X.L., R.L designed the model; X.W., S.W. implemented the program and performed the experiments; X.W. analyzed the 1000G data; X.W., Y.X., Y.L., X.Z. wrote the manuscript. X.W. and J.W conducted the revision. All authors have read and agreed to the latest version of the manuscript.

Key Points

- PEcnv is a novel approach to detect copy number variations that vary in size and copy number.
- PEcnv enables accurate detection of copy number variants with short-to-medium length, which are hard to identify with existing approaches but have been emphasized recently in cancer research and treatment development.
- PEcnv is among the first approaches to incorporate the genomic bases' features around a target base to correct the bias and noise on the read depth. This solves the lack of training data for clinical panel sequencing data.
- PEcnv applies to panel sequencing data and also works for whole genome sequencing and whole-exome sequencing data.

Supplementary data

Supplementary data are available online at https://academic.oup.com/bib.

Acknowledgements

We thank all the editors and reviewers for the valuable suggestions. We also thank the faculty members and graduate students who discussed the issues with us.

Funding

This work was supported by Shaanxi's Natural Science Basic Research Program, grant number 2020JC-01 (also to A.P.C).

Conflict of Interest

The authors declare no conflict of interest. The founding sponsors had no role in the study's design or in data collection, analyses, or interpretation, in the writing of the manuscript, or in the decision to publish the results.

References

1. Liu P, Carvalho C, Hastings PJ, *et al.* Mechanisms for recurrent and complex human genomic rearrangements. *Curr Opin Genet Dev* 2012;**22**(3):211–20.
2. Fromer M, Purcell SM. Using XHMM software to detect copy number variation in whole-exome sequencing data. *Curr Protoc Hum Genet* 2014;**81**(1):7.23.21–27.23.21.
3. Freeman J, Perry GH, Feuk L, *et al.* Copy number variation: new insights in genome diversity. *Genome Res* 2006;**16**(8):949–61.
4. Albertson DG, Collins C, Mccormick F, *et al.* Chromosome aberrations in solid tumors. *Recent Results Cancer Res* 2003;**34**(4):369–76.
5. Redon R, Ishikawa S, Fitch KR, *et al.* Global variation in copy number in the human genome. *Nature* 2006;**444**(7118):444–54.
6. Lee JA, Lupski JR. Genomic rearrangements and gene copy-number alterations as a cause of nervous system disorders. *Neuron* 2006;**52**(1):103–21.
7. Li J, Lupat R, Amarasinghe KC, *et al.* CONTRA: copy number analysis for targeted resequencing. *Bioinformatics* 2012;**28**(10):1307–13.
8. Cid RD, Riveira-Munoz E, Zeeuwen PLJM, *et al.* Deletion of the late cornified envelope LCE3B and LCE3C genes as a susceptibility factor for psoriasis. *Nat Genet* 2009;**41**(2):211–5.
9. Stahl EA, Raychaudhuri S, Remmers EF, *et al.* Genome-wide association study meta-analysis identifies seven new rheumatoid arthritis risk loci. *Nat Genet* 2010;**42**(6):508–14.
10. Buysse K, Chiaie BD, Coster RV, *et al.* Challenges for CNV interpretation in clinical molecular karyotyping: lessons learned from a 1001 sample experience. *Eur J Med Genet* 2009;**52**(6):398–403.
11. Carter NP. Methods and strategies for analyzing copy number variation using DNA microarrays. *Nat Genet* 2007;**39**(7):S16.
12. Meyerson M, Gabriel S, Getz G. Advances in understanding cancer genomes through second-generation sequencing. *Nat Rev Genet* 2010;**11**(10):685.
13. Klambauer G, Schwarzbauer K, Mayr A, *et al.* Cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. *Nucleic Acids Res* 2012;**40**(9):e69–9.
14. Ansorge WJ. Next generation DNA sequencing techniques and applications. *N Biotechnol* 2010;**27**(1):S3.
15. Crowgey EL, Stabley DL, Chen C, *et al.* An integrated approach for analyzing clinical genomic variant data from next-generation sequencing. *J Biomol Tech* 2015;**26**:19–28.
16. Zare F, Dow M, Monteleone N, *et al.* An evaluation of copy number variation detection tools for cancer using whole exome sequencing data. *BMC Bioinformatics* 2017;**18**(1):286.
17. Zhao L, Liu H, Yuan X, *et al.* Comparative study of whole exome sequencing-based copy number variation detection tools. *BMC Bioinformatics* 2020;**21**(1):97.
18. Johansson LF, Dijk FV, de Boer EN, *et al.* CoNVaDING: single exon variation detection in targeted NGS data. *Hum Mutat* 2016;**37**(5):457–64.
19. Talevich E, Shain AH, Botton T, *et al.* CNVkit: genome-wide copy number detection and visualization from targeted DNA sequencing. *PLoS Comput Biol* 2016;**12**(4):e1004873.

20. Shen R, Seshan VE. FACETS: allele-specific copy number and clonal heterogeneity analysis tool for high-throughput DNA sequencing. *Nucleic Acids Res* 2016;**44**(16):e131–1.
21. Zhao M, Wang Q, Wang Q, *et al.* Computational tools for copy number variation (CNV) detection using next-generation sequencing data: features and perspectives. *BMC Bioinformatics* 2013;**14**(S11):S1.
22. Friedrich S, Barbulescu R, Helleday T, *et al.* MetaCNV-a consensus approach to infer accurate copy numbers from low coverage data. *BMC Med Genomics* 2020;**13**(1):76.
23. Dohm JC, Lottaz C, Borodina T, *et al.* Substantial biases in ultra-short read data sets from high-throughput DNA sequencing. *Nucleic Acids Res* 2008;**36**(16):e105.
24. Jiang Y, Oldridge D, Diskin SJ, *et al.* CODEX: a normalization and copy number variation detection method for whole exome sequencing. *Nucleic Acids Res* 2015;**43**(6):e39.
25. Li H, Handsaker B, Wysoker A, *et al.* The sequence alignment/map format and SAMtools. *Bioinformatics* 2009;**25**(16):2078–9.
26. Bellos E, LJM C. cnvOffSeq: detecting intergenic copy number variation using off-target exome sequencing data. *Bioinformatics* 2014;**30**:i639–45.
27. Roberts SW. Control chart tests based on geometric moving averages. *Dent Tech* 1959;**1**(3):239–50.
28. Fatahi AA, Noorossana R, Dokouhaki P, *et al.* Zero inflated poisson ewma control chart for monitoring rare health-related events. *Journal of Mechanics in Medicine and Biology* 2012;**12**(04):1250065.
29. Wang S, Wang J, Xiao X, *et al.* GSDcreator: an efficient and comprehensive simulator for genarating ngs data with population genetic information. In: *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*: 2019.
30. Rapti M, Zouaghi Y, Meylan J, *et al.* CoverageMaster: comprehensive CNV detection and visualization from NGS short reads for genetic medicine applications. *Brief Bioinform* 2022;**23**(2):1–8.
31. Qin M, Liu B, Conroy JM. SCNVSim: somatic copy number variation and structure variation simulator. *BMC Bioinformatics* 2015;**16**(1):66.
32. Xiao W, Ren L, Chen Z, *et al.* Toward best practice in cancer mutation detection with whole-genome and whole-exome sequencing. *Nat Biotechnol* 2021;**39**(9):1141–50.
33. Conrad DF, Pinto D, Redon R, *et al.* Origins and functional impact of copy number variation in the human genome. *Nature* 2010;**464**(7289):704–12.
34. Park H, Kim JI, Ju YS, *et al.* Discovery of common Asian copy number variants using integrated high-resolution array CGH and massively parallel DNA sequencing. *Nat Genet* 2012;**42**(5):400–5.
35. Robinson JT, Thorvaldsdóttir H, Wenger AM, *et al.* Variant review with the integrative genomics viewer. *Cancer Res* 2017;**77**(21):e31–4.
36. Roca I, González-Castro L, Fernández H, *et al.* Free-access copy-number variant detection tools for targeted next-generation sequencing data. *Mutat Res Rev Mutat Res* 2019;**779**:114–25.
37. Miller RE, Leary A, Scott CL, *et al.* ESMO recommendations on predictive biomarker testing for homologous recombination deficiency and PARP inhibitor benefit in ovarian cancer. *Ann Oncol* 2020;**31**(12):1606–22.