Nano-GLADIATOR: real-time detection of copy number alterations from nanopore sequencing data.

Alberto Magi[1,*,†], Davide Bolognini[2,†], Niccoló Bartalucci[3,†], Alessandra Mingrino[2], Roberto Semeraro[2], Luna Giovannini[2], Stefania Bonifacio[4], Daniela Parrini[4], Elisabetta Pelo[4], Francesco Mannelli[3], Paola Guglielmelli[3], Alessandro Maria Vannucchi[3].

[1]Department of Information Engineering, University of Florence, Florence, Italy, [2]Department of Experimental and Clinical Medicine, University of Florence, Florence, Italy, [3]Department of Experimental and Clinical Medicine, CRIMM, Center Research and Innovation of Myeloproliferative Neoplasms, Azienda Ospedaliera Universitaria Careggi, University of Florence, Florence, Italy, [4]Department of Laboratory Diagnosis, Genetic Diagnosis Service, Careggi Teaching Hospital, Florence, Italy. †These authors contributed equally to this work.

1 RC bias distribution and normalization

To evaluate if RC data follow Poisson or negative binomial distributions, we calculated the Kolmogorov-Smirnov statistics D that measures the distance between two empirical distribution function and the smaller is D the closer are the two distributions.

To mitigate the effect of GC content % and mappability biases we used a bias removal procedure based on the median normalization approach that we introduced in (1) and in (2). For each GC percentage (0, 1, 2,..,100%) and each bin of mappability score (0, 0.1, 0.2,...,1) we calculated the deviation of RC or DOC from the window average and then corrected each RC according to the following formula:

$$\overline{RC_i} = RC_i \cdot \frac{m}{m_{\mathrm{X}}}, \tag{1}$$

where RC_i are the window mean read counts of the i-th window, m_{X} is the median RC of all the windows that have the same X value (where X=[GC content, mappability score]) as the i-th window, and m is the overall median of all the windows.

For each window of the reference genome, GC content % was calculated with the nucBed command of bedtools 3 while the mappability score was estimated by using the gem-mappability module of the Genome Multitool (GEM) mapper (4).

2 Shifting level model (SLM) algorithm

In 2010 we introduced a powerful segmentation algorithm, based on shifting level models (SLM), for analyzing log_2Ratio genomic profiles from array-CGH. SLMs (5) model noisy sequential processes with sudden shifts in the mean $x = (x_1, .., x_i, .., x_N)$ as the sum of two independent stochastic processes:

$$x_i = m_i + \epsilon_i, \tag{2}$$

$$m_i = (1 - z_{i-1}) \cdot m_{i-1} + z_{i-1} \cdot (\mu + \delta_i). \tag{3}$$

where m_i is the unobserved mean level that follows a normal distribution with mean μ and variance σ_μ^2 ($m_i \sim N(\mu, \sigma_\mu^2)$) and ϵ_i is a normally distributed white noise with variance σ_ϵ^2 ($\epsilon_i \sim N(0, \sigma_\epsilon^2)$, Figure 1.a).

The process m_i changes its value independently of m_{i-1} and is controlled by the process z_i : when $z_{i-1} = 0$, m_i is the same as m_{i-1} and when $z_{i-1} = 1$, m_i is incremented by the normal random variable δ_i ($\delta_i \sim N(0, \sigma_\mu^2)$). $z_1, z_2, ...$ are independent and identically distributed random variables taking the values 0,1 with probabilities $\eta = Pr(z_i = 1)$, $1 - \eta = Pr(z_i = 0)$.

It has been demonstrated that SLM is a special class of hidden markov models (HMM) and for this reason we can use classical HMM algorithms, such as Baum and Welch and Viterbi algorithms to estimate its parameters (5).

Nano-GLADIATOR tool

In "On-line" mode Nano-GLADIATOR must be launched simultaneously to a nanopore experiment and, monitoring the data flow generated by the sequencing process, starts the analysis to perform real time molecular karyotype when the number of reads produced by the device reaches a predefined threshold. On the other hand, the "Off-line" modality allows to analyze multiple "finished" WGS experiment in parallel by setting the number of processor.
"Off-line" analysis can be performed with two different experimental/computational strategies: "nocontrol" and "paired". In "nocontrol" mode each sample is analyzed individually without the need of a matched control, the RCs are normalized, rescaled to two copies, log-transformed, segmented with the SLM algorithm and allelic fraction is estimated with FractionPred by using Poisson distribution. In "paired" mode (well suited for the detection of somatic CNAs) each test sample needs a matched normal sample, the RCs of test sample are compared with the RCs of control sample, log-transformed, segmented with SLM and allelic fraction is estimated with FractionPred by using Skellam distribution. At present, "On-line" modality is limited to analyze only one sample at time in "nocontrol" mode.
The output of the Nano-GLADIATOR tool is a tab-delimeted file containing information for each segmented region and an interactive html file that shows $log_2 RC_{Norm}$, segmentation results and allelic fraction prediction for all analyzed chromosomes. The graphical interface of the html file allows to browse each segmented region of the genome and to filter alterations as a function of predicted allelic fraction. Moreover, for each segmented region a link to UCSC genome browser is provided.
Nano-GLADIATOR can run on any unix system (desktop and servers) and in a desktop computer with a 2.5 GHz CPU and 8 GB of RAM, by using four cores, it takes 5 minutes to analyze the data generated by a 12 hours MinION run. Nano-GLADIATOR is freely available at `https://sourceforge.net/projects/nanogladiator/`.

3 NA12878 nanopore data

The nanopore WGS consortium (6) (`https://github.com/nanopore-wgs-consortium`) sequenced the CEPH1463 (NA12878/GM12878, Ceph/Utah pedigree) human genome on the ONT MinION by using 39 R9/R9.4 flow cells generating 4,183,584 base-called reads containing 91,240,120,433 bases with a read N50 of 10,589 bp. Reads in FastQ format for all 39 runs were downloaded from `https://github.com/nanopore-wgs-consortium`, aligned against the human reference genome (hg19) by using minimap2 (7) with $-ont$ option and converted to bam format with samtools (8). Bam files were then downsampled to generate datasets made of N reads (with N=10.000, 20.000, 50.000, 100.000, 200.000, 500.000 and 1.000.000 sequences) that mimic the ONT sequencing process at different time points (5 m, 10 m, 30 m, 1 h, 2 h, 6 h and 12 hours). Downsampled datasets were then elaborated with Xome-Blender to generate synthetic datasets exploited in results section. Xome-Blender is a software tool that allows to generate synthetic genomes with user-defined features such as the number of subclones, subclones allele fraction, the number of somatic variants and the presence of CNVs-CNAs, without the addition of any synthetic element. For "copy number detection" section we used Xome-Blender to generate synthetic genomes with duplications and deletions of different length that range from 50 kb to 20 Mb (50 kb, 100 kb, 200 kb, 500 kb, 1 Mb, 2 Mb, 5 Mb, 10 Mb, 20 Mb). For "allele fraction prediction" section Xome-Blender was used to simulate duplications and deletions of different size (1 Mb, 2 Mb, 5 Mb, 10 Mb, 20 Mb) and allele fraction from 10% to 90%.

3.1 Euskirchen et al. dataset

Euskirchen et al. (9) used low pass WGS experiments obtained from MinION device to study genomic alterations of brain tumors. To this end, they sequenced 28 Brain tumors Gliomas with six hours runs of the MinION Mk 1B device and R9 or R9.4 flow-cells obtaining an average of 50k reads for each sample (min 15k max 80k). For our analyses we used seven samples that were previously characterized with Illumina SNP array data. Nanopore data were downloaded in Fast5 formats from `https://ega-archive.org/studies/EGAS00001002213`. For each sequencing run Fast5 files were converted to FastQ with poretools (10) and aligned to the human reference genome (hg19) with minimap2. SNP-array data for the seven samples were downloaded from ArrayExpress database under accession codes E-MTAB-3905(Illumina Human610 Quad), E-MTAB-3907(Illumina HumanCNV370), E-MTAB-3896(Illumina HumanCore) and E-MTAB-3902(Illumina HumanOmniExpress). Normalized log_2Ratio profiles were segmented by using the CBS algorithm and altered regions were called with FastCall.

4 Algorithms Comparison

We compared our tool to other three previously published software tools for CNAs detection: CNANorm (11), HMMCopy (12) and BICSeq2 (13). We downloaded the CNANorm R package version 1.26.0 at `https://bioconductor.org/packages/release/bioc/html/CNAnorm.htmll`, HMMCopy R package version 1.22.0 at `https://bioconductor.org/packages/release/bioc/html/HMMcopy.html` and BIC-Seq2 at `http://compbio.med.harvard.edu/BIC-seq/`. We ran the three methods with default parameter setting by following author suggestions: for the seven Gliomas (Euskirchen et al.) and the four "On-line" analyses at 100.000 reads we used a window size of 1 Mb, while for the two "On-line" analyses at 500.000 and 1.000.000 reads we used 100 kb window size. Since CNANorm, HMMCopy and BICSeq2 only allow to perform paired analysis, to properly run the three tools, we used as matched normal controls downsampled dataset of the NA12878 sequenced by the nanopore WGS consortium. In particular, for the Euskirchen et al. dataset we generated a WGS dataset with 50.000 reads, while for the hematologic malignancies datasets we downsampled WGS at 100.000, 500.000 and 1.000.000 of reads. Downsampled WGS experiments were also used for the analyses performed by Nano-GLADIATOR in paired mode.
Before calculating precision and recall for the three tools we compared the CNA sets with the CNVs identified by HapMap and 1000 Genome Project consortia for the NA12878 individual. We did not find any overlap. This was expected since the the largest CNVs detected by the HapMap for NA12878 was an homozygous deletion of around 300 Kb, while the largest event detected by the 1000 Genomes Project was a deletion of around 140 Kb. The size of these events is below the resolution of the the three tools with window sizes of 100 Kb, 500 Kb and 1 Mb.

4.1 Hematologic malignancies dataset

MinION runs were performed on six samples with hematologic malignancies selected from our sample bank with signed informed consent. Two samples were from patients diagnosed with Acute Myeloid Leukemia and four samples from Myeloproliferative Neoplasm patients. For MinION sequencing experiments, genomic DNA was extracted using Maxwell automated extraction system (Promega, WI, US) from granulocyte previously purified by density gradient from peripheral blood. Nucleic acids were quantified by Qubit 2.0 Fluorometer (Life Technologies, MA, US) while purity and DNA fragmentation were assessed respectively by Nanodrop One (Thermo Fisher Scientific, MA, US) and Agilent Bioanalizer (Agilent, CA, US). 1.5 ug of recovered genomic DNA from each sample was used to prepare libraries following 1D SQK-LSK108 Oxford Nanopore protocol. According to protocol, DNA was sheared by g-TUBE (Covaris, MA, US) centrifugation, end-repaired with NEBNext end repair module (New England Biolabs, MA, US) and subsequently dA-tailed by using NEBNext dA-tailing module (New England Biolabs, MA, US) before Nanopore-specific adapters ligation. Each step was followed by purification with Agencourt AMPure XP beads (Beckman Coulter, CA, US). After the platform quality control, the sequencing mixes were loaded in 9.4.1 chemistry flow-cells. For each MinION run, while the sequencing process was started, we simultaneously launched the Nano-GLADIATOR tool in "On-line" modality to perform real time detection of CNAs. Analyzed MinION data in FastQ format are available at the European Nucleotide Archive under accession PRJEB27516. CGH-arrays were performed using the SurePrint G3 ISCA CGH+SNP 4x180K Microarray Kit (Agilent Technologies, Santa Clara, CA, USA), according to the manufacturer's protocols. The Agilent Feature Extraction software has been used to perform image analysis, while CNVs analysis were performed with the CytoGenomics Software.

References

[1] Magi, A., Tattini, L., Pippucci, T., Torricelli, F., Benelli, M. (2012). Read count approach for DNA copy number variants detection. Bioinformatics, 28(4), 470-478.

[2] Magi, A., Tattini, L., Cifola, I., D'Aurizio, R., Benelli, M., Mangano, E. et al. (2013). EXCAVATOR: detecting copy number variants from whole-exome sequencing data. Genome biology, 14(10), R120.

[3] Quinlan, A. R., Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics, 26(6), 841-842.

[4] Derrien, T., Estell, J., Sola, S. M., Knowles, D. G., Raineri, E., Guig, R., Ribeca, P. (2012). Fast computation and applications of genome mappability. PloS one, 7(1), e30377.

[5] Magi, A., Benelli, M., Marseglia, G., Nannetti, G., Scordo, M. R., Torricelli, F. (2010). A shifting level model algorithm that identifies aberrations in array-CGH data. Biostatistics, 11(2), 265-280.

[6] Jain, M., Koren, S., Miga, K. H., Quick, J., Rand, A. C., Sasani, T. A. et al. (2018). Nanopore sequencing and assembly of a human genome with ultra-long reads. Nature biotechnology, 36(4), 338.

[7] Li, H. (2018). Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics, 1, 7.

[8] Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N. et al. (2009). The sequence alignment/map format and SAMtools. Bioinformatics, 25(16), 2078-2079.

[9] Euskirchen, P., Bielle, F., Labreche, K., Kloosterman, W. P., Rosenberg, S., Daniau, M. et al. (2017). Same-day genomic and epigenomic diagnosis of brain tumors using real-time nanopore sequencing. Acta neuropathologica, 134(5), 691-703.

[10] Loman, N. J., Quinlan, A. R. (2014). Poretools: a toolkit for analyzing nanopore sequence data. Bioinformatics, 30(23), 3399-3401.

[11] Boeva, V., Popova, T., Bleakley, K., Chiche, P., Cappo, J., Schleiermacher, G. et al. (2011). Control-FREEC: a tool for assessing copy number and allelic content using next-generation sequencing data. Bioinformatics, 28(3), 423-425.

[12] Gusnanto, A., Wood, H. M., Pawitan, Y., Rabbitts, P., Berri, S. (2011). Correcting for cancer genome size and tumour cell content enables better estimation of copy number alterations from next-generation sequence data. Bioinformatics, 28(1), 40-47.

[13] Ha, G., Roth, A., Lai, D., Bashashati, A., Ding, J., Goya, R., et al. (2012). Integrative analysis of genome-wide loss of heterozygosity and monoallelic expression at nucleotide resolution reveals disrupted pathways in triple-negative breast cancer. Genome research. 22(10), 1995-2007.

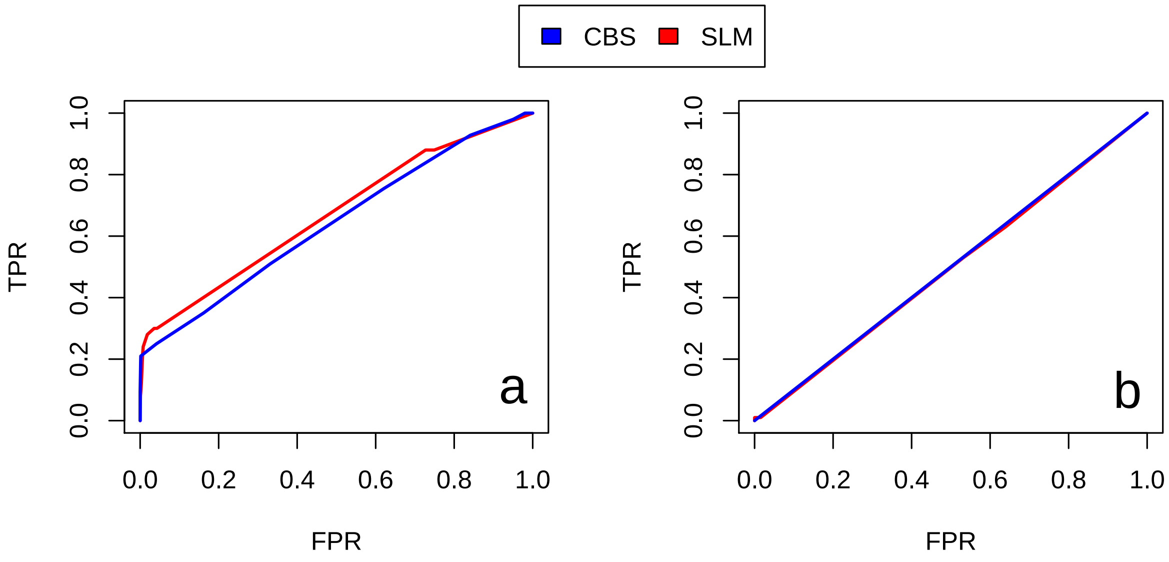

Figure 1: ROC curves for SLM and CBS on RC profiles from 10K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 10K reads analyzed with 1 Mb window size.

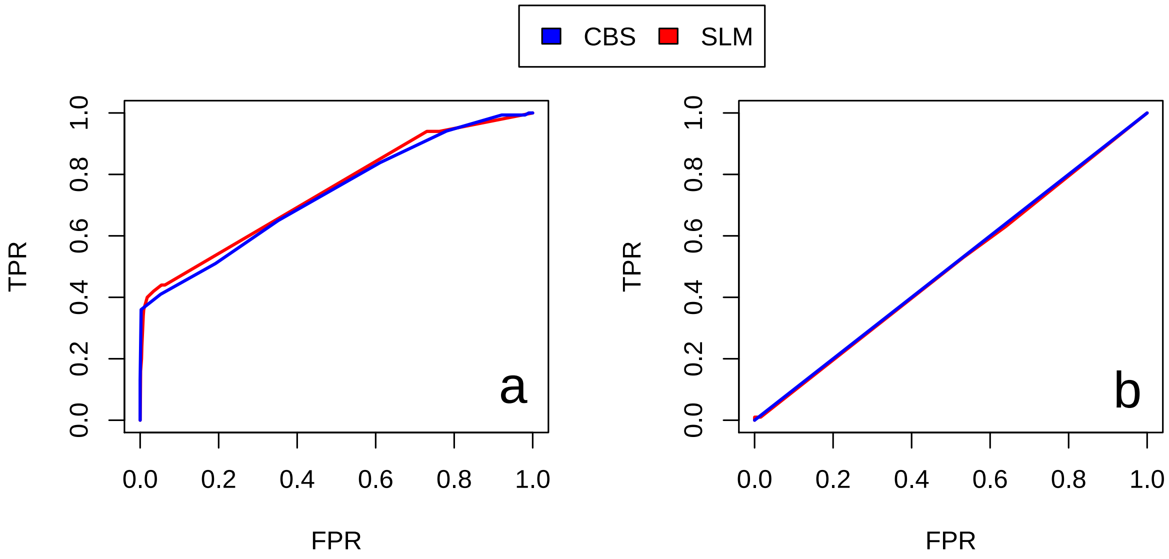

Figure 2: ROC curves for SLM and CBS on RC profiles from 10K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 10K reads analyzed with 2 Mb window size.

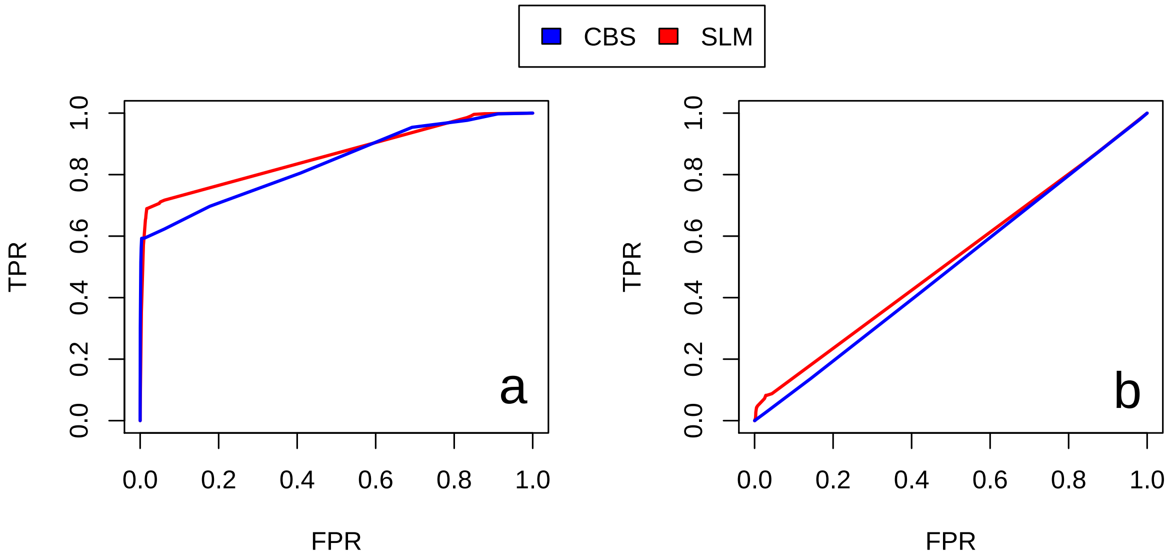

Figure 3: ROC curves for SLM and CBS on RC profiles from 20K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 20K reads analyzed with 1 Mb window size.

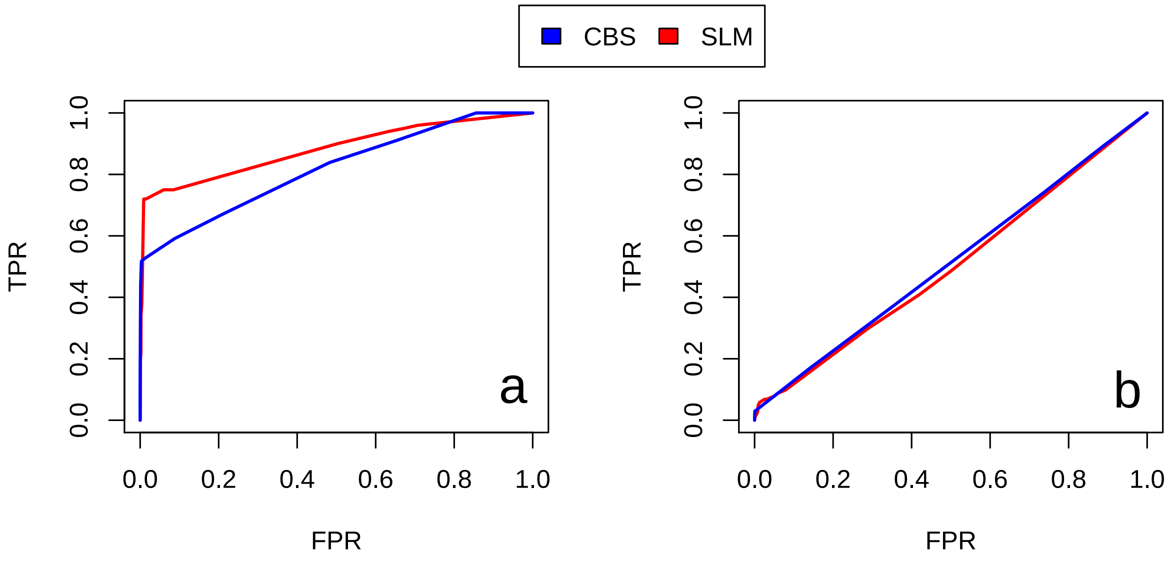

Figure 4: ROC curves for SLM and CBS on RC profiles from 20K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 20K reads analyzed with 2 Mb window size.

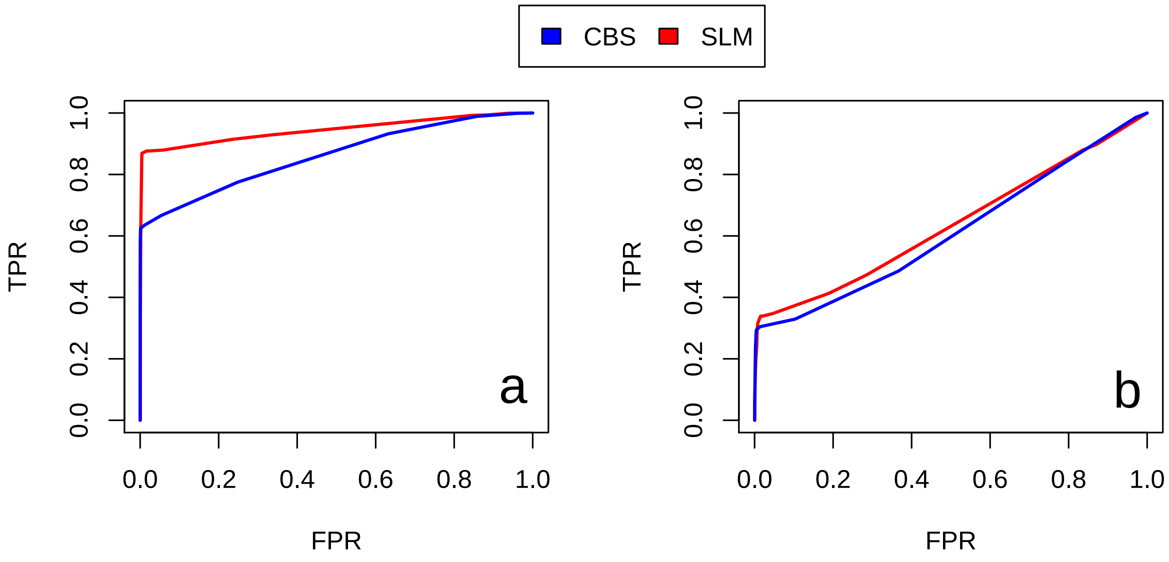

Figure 5: ROC curves for SLM and CBS on RC profiles from 50K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 50K reads analyzed with 1 Mb window size.

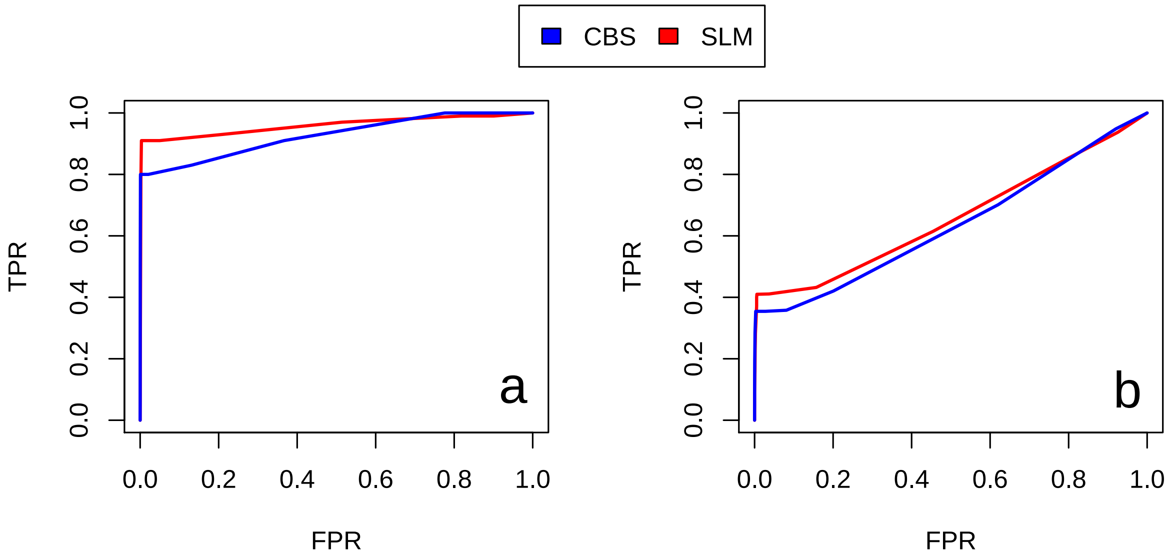

Figure 6: ROC curves for SLM and CBS on RC profiles from 50K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 50K reads analyzed with 2 Mb window size.

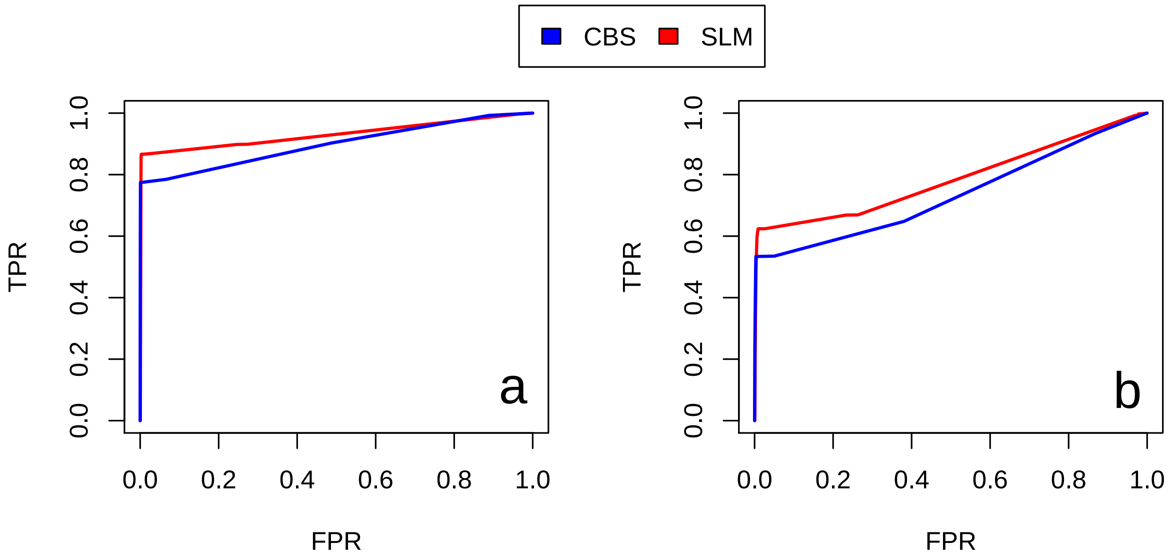

Figure 7: ROC curves for SLM and CBS on RC profiles from 100K reads analyzed with 500 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 100K reads analyzed with 2 Mb window size.

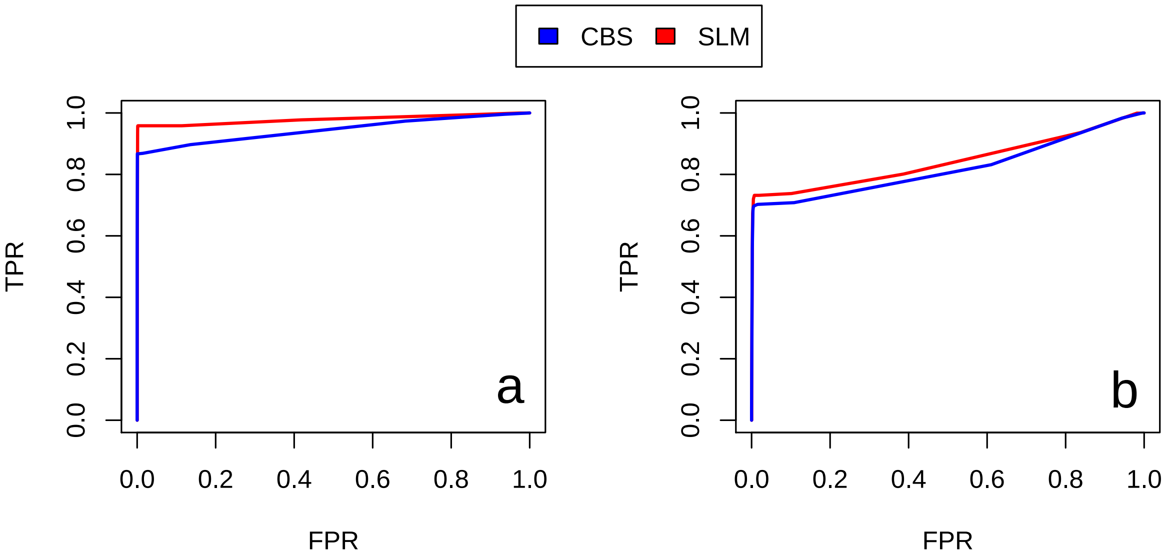

Figure 8: ROC curves for SLM and CBS on RC profiles from 100K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 100K reads analyzed with 1 Mb window size.

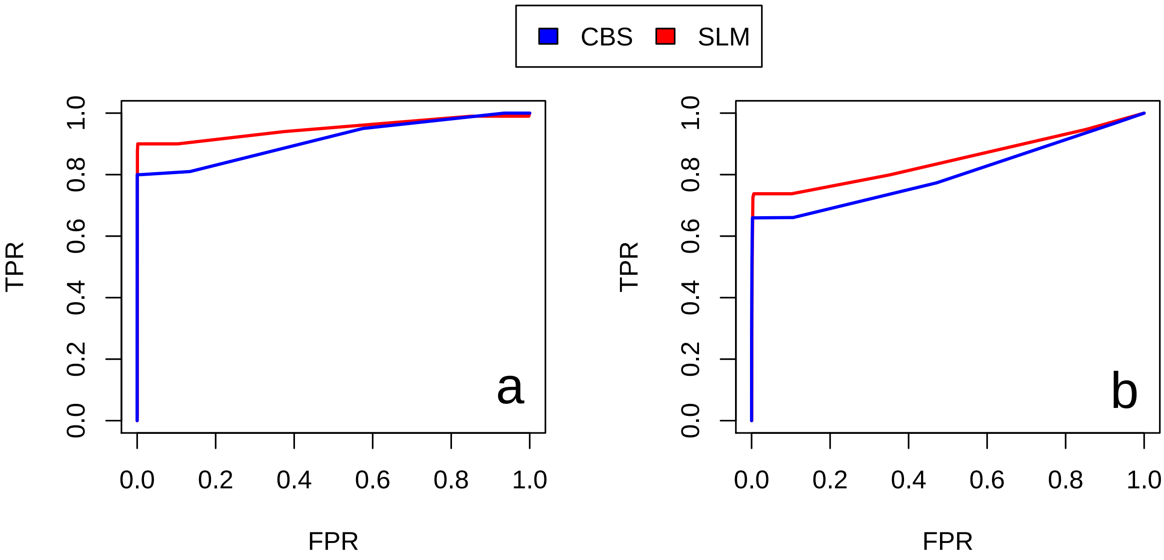

Figure 9: ROC curves for SLM and CBS on RC profiles from 100K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 100K reads analyzed with 2 Mb window size.

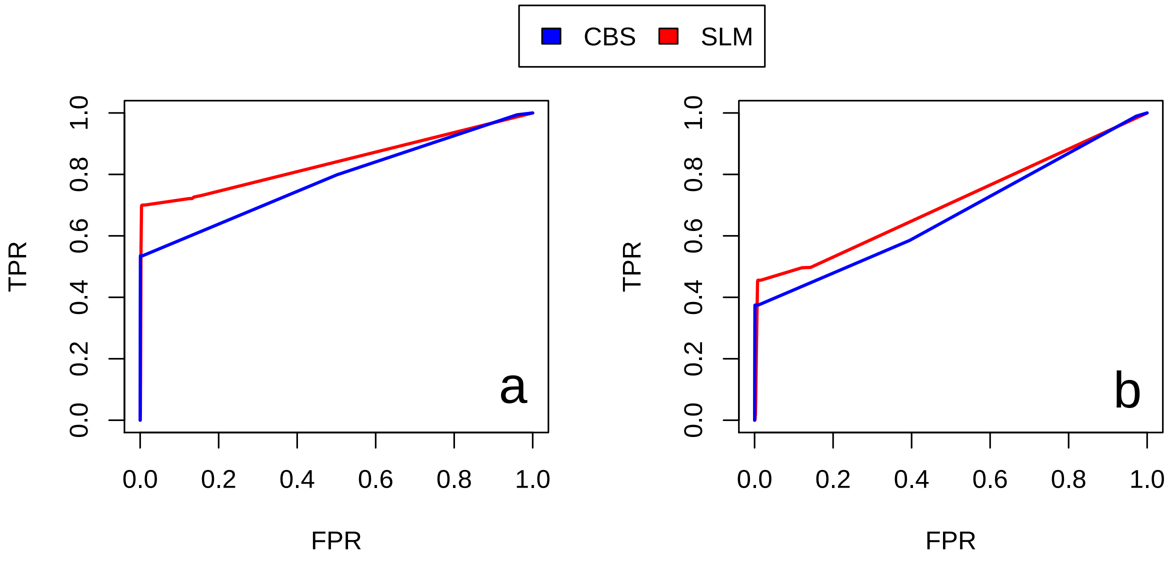

Figure 10: ROC curves for SLM and CBS on RC profiles from 200K reads analyzed with 100 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 200K reads analyzed with 100 Kb window size.

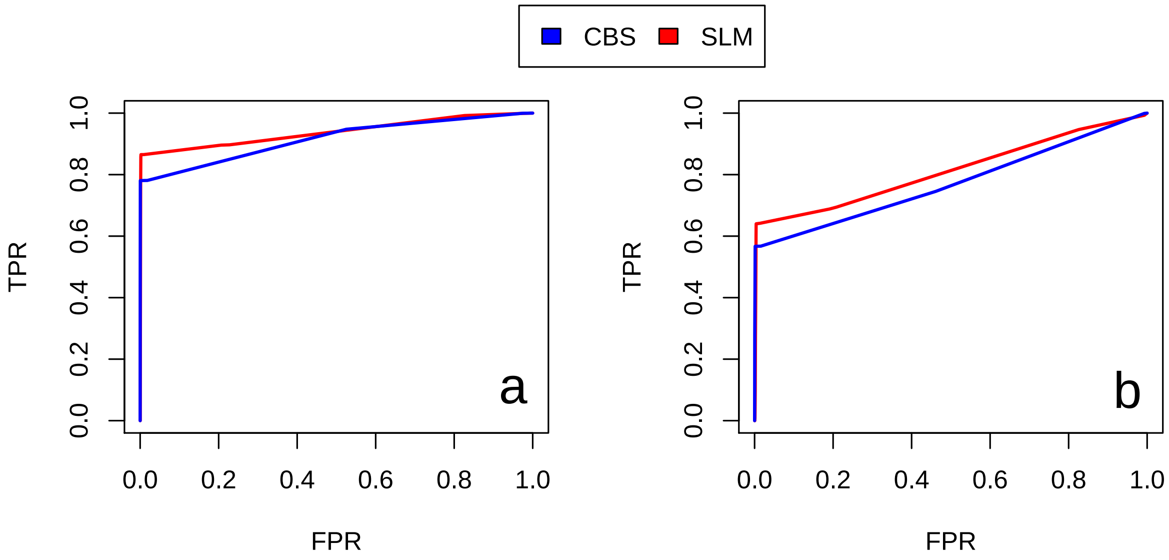

Figure 11: ROC curves for SLM and CBS on RC profiles from 200K reads analyzed with 200 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 200K reads analyzed with 200 Kb window size.

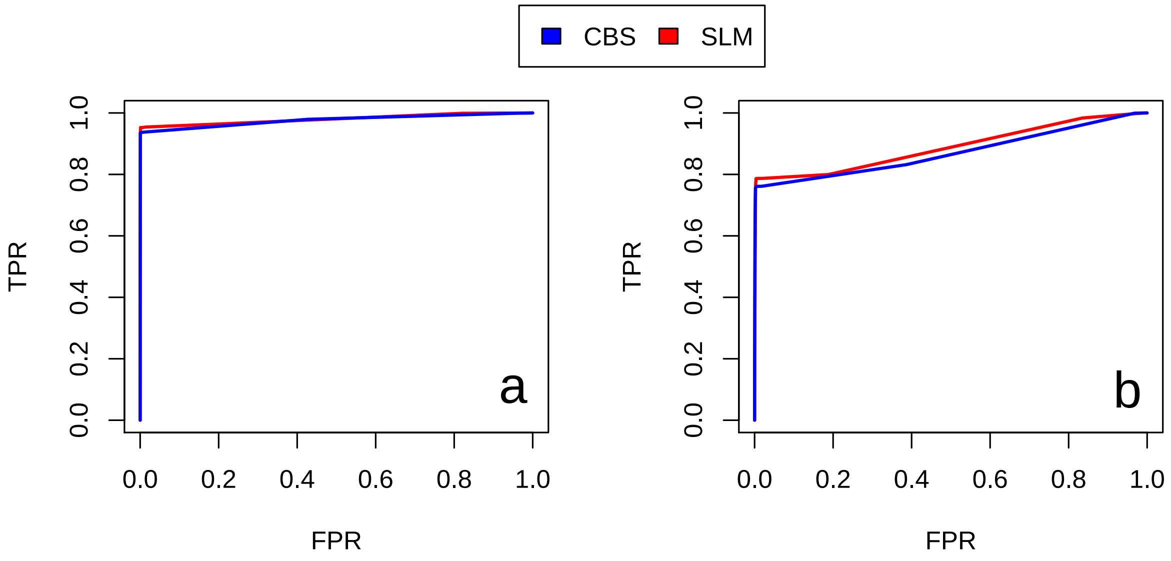

Figure 12: ROC curves for SLM and CBS on RC profiles from 200K reads analyzed with 500 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 200K reads analyzed with 500 Kb window size.

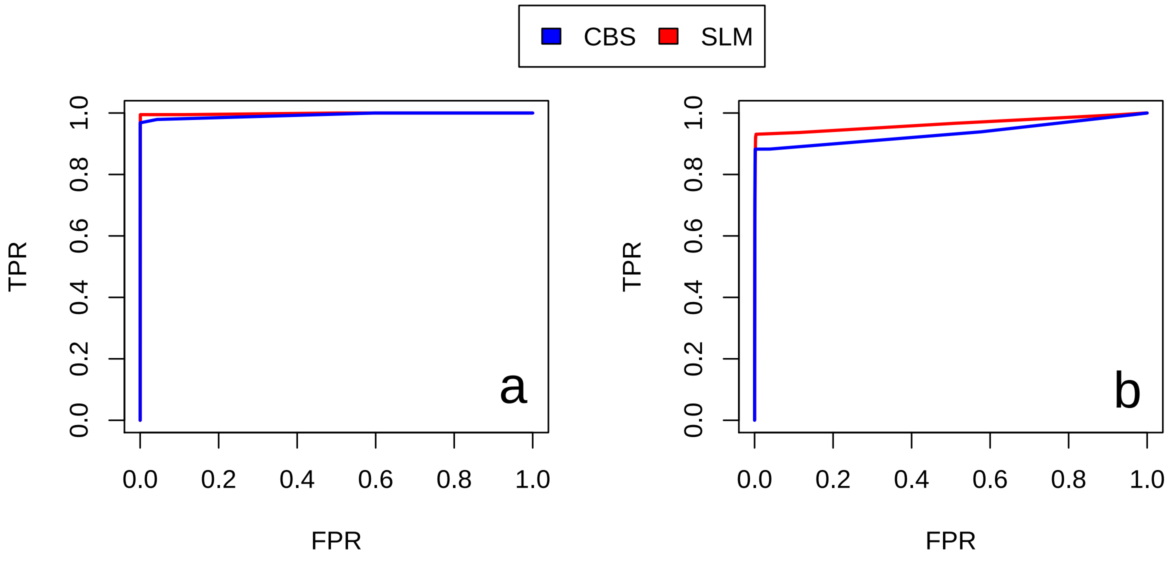

Figure 13: ROC curves for SLM and CBS on RC profiles from 200K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 200K reads analyzed with 2 Mb window size.

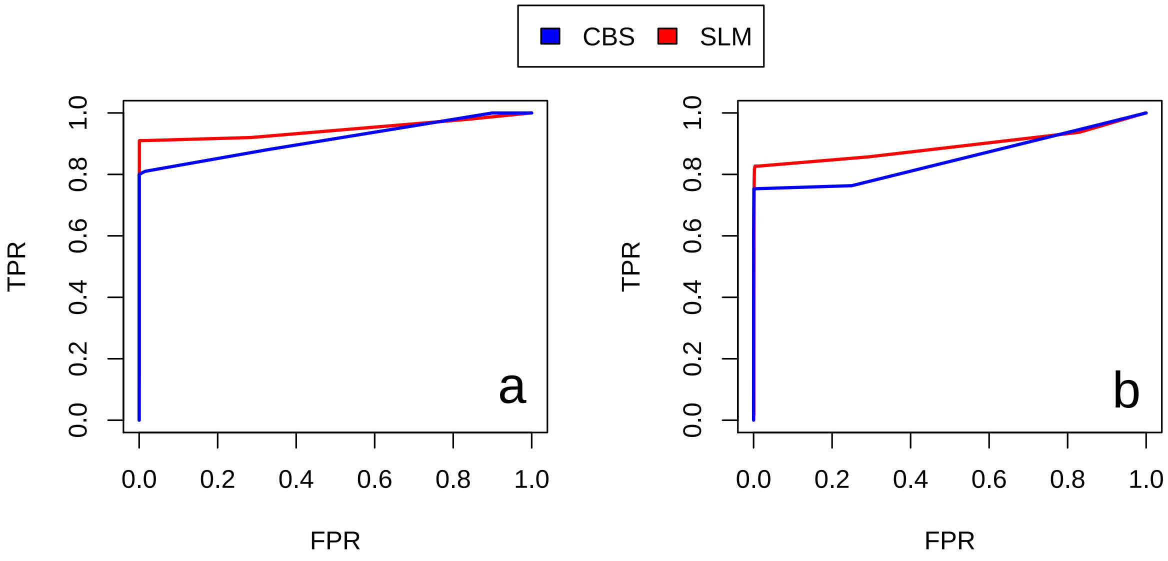

Figure 14: ROC curves for SLM and CBS on RC profiles from 200K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 200K reads analyzed with 2 Mb window size.

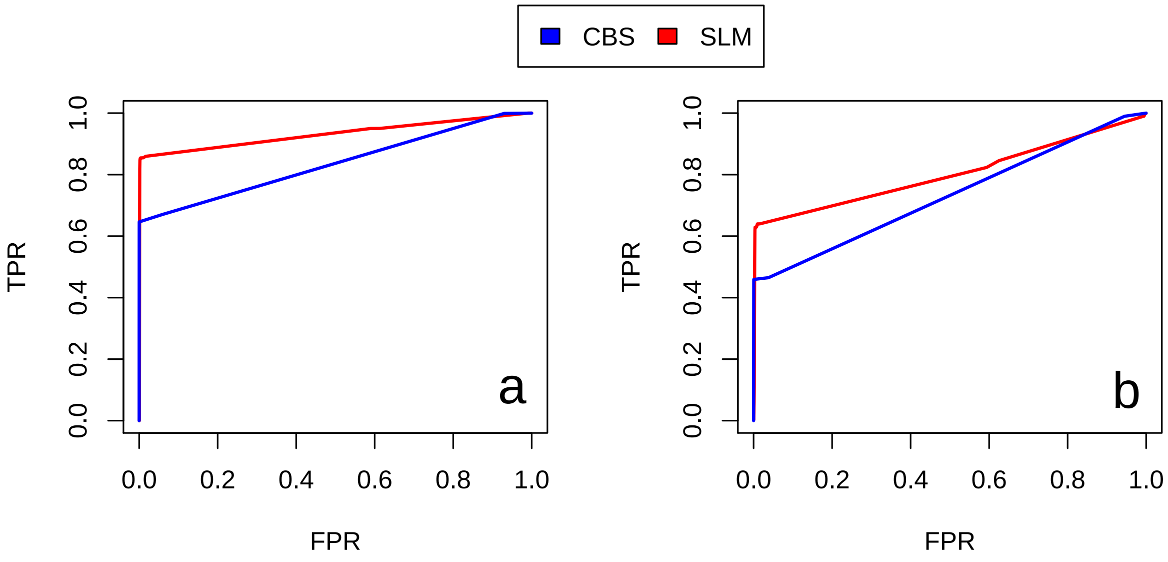

Figure 15: ROC curves for SLM and CBS on RC profiles from 500K reads analyzed with 100 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 500K reads analyzed with 100 Kb window size.

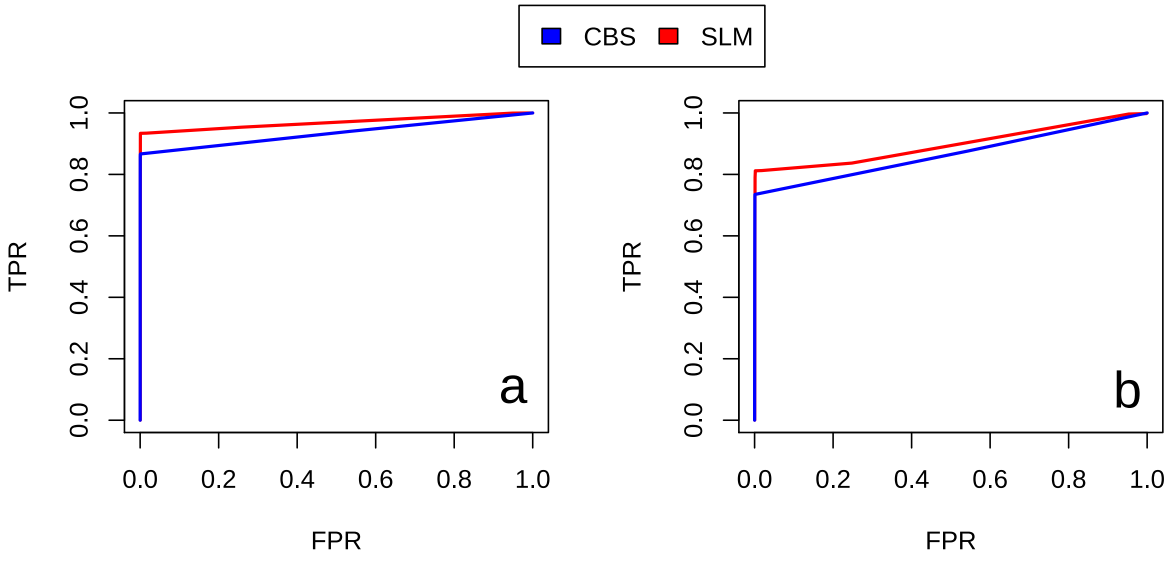

Figure 16: ROC curves for SLM and CBS on RC profiles from 500K reads analyzed with 200 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 500K reads analyzed with 200 Kb window size.

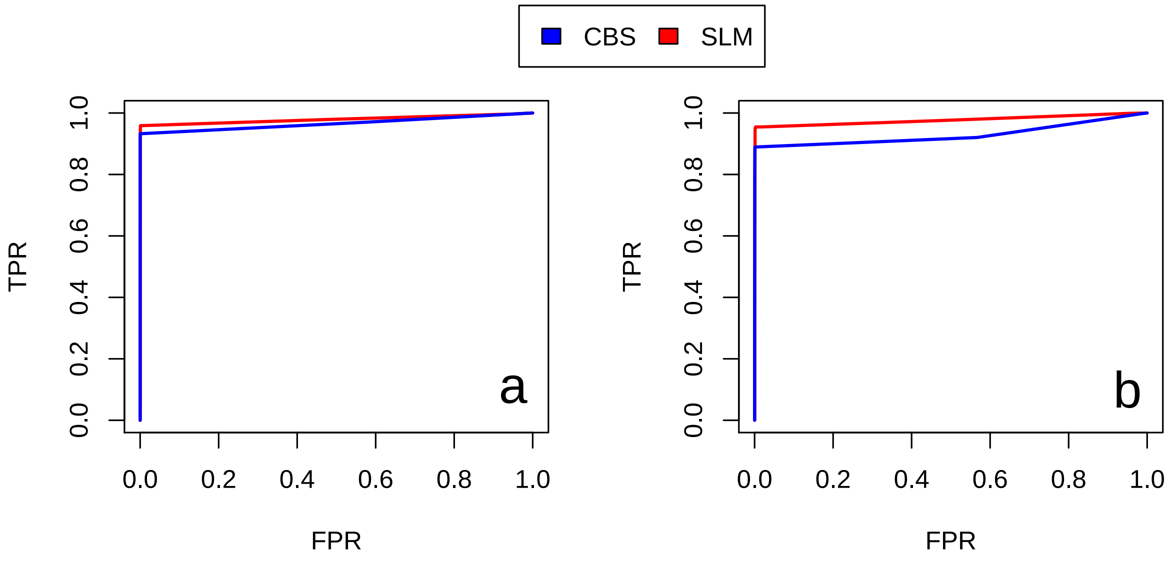

Figure 17: ROC curves for SLM and CBS on RC profiles from 500K reads analyzed with 500 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 500K reads analyzed with 500 Kb window size.

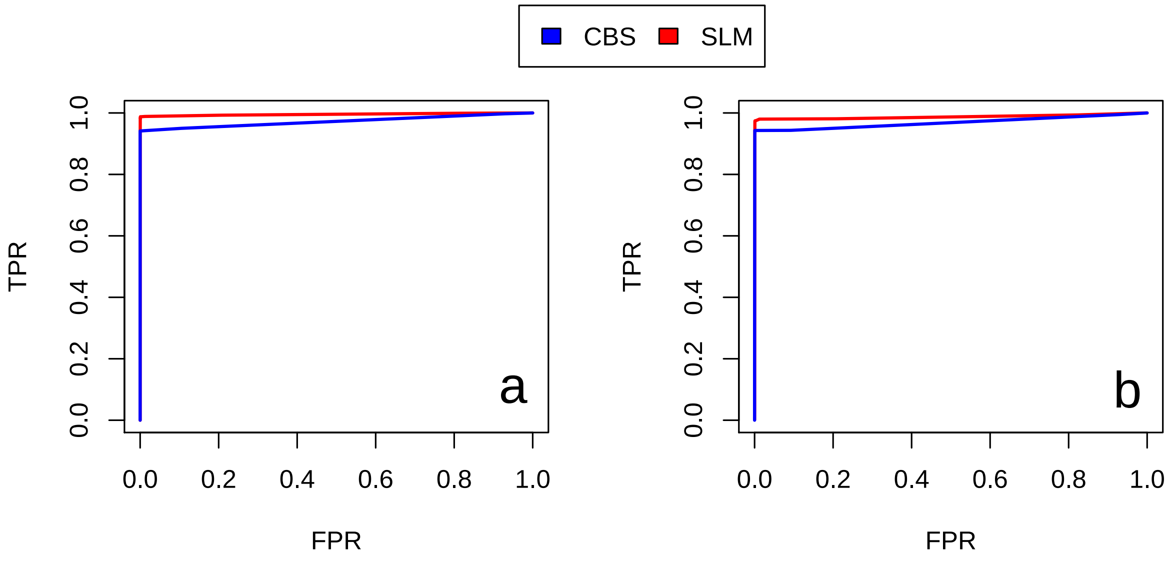

Figure 18: ROC curves for SLM and CBS on RC profiles from 500K reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 500K reads analyzed with 1 Mb window size.

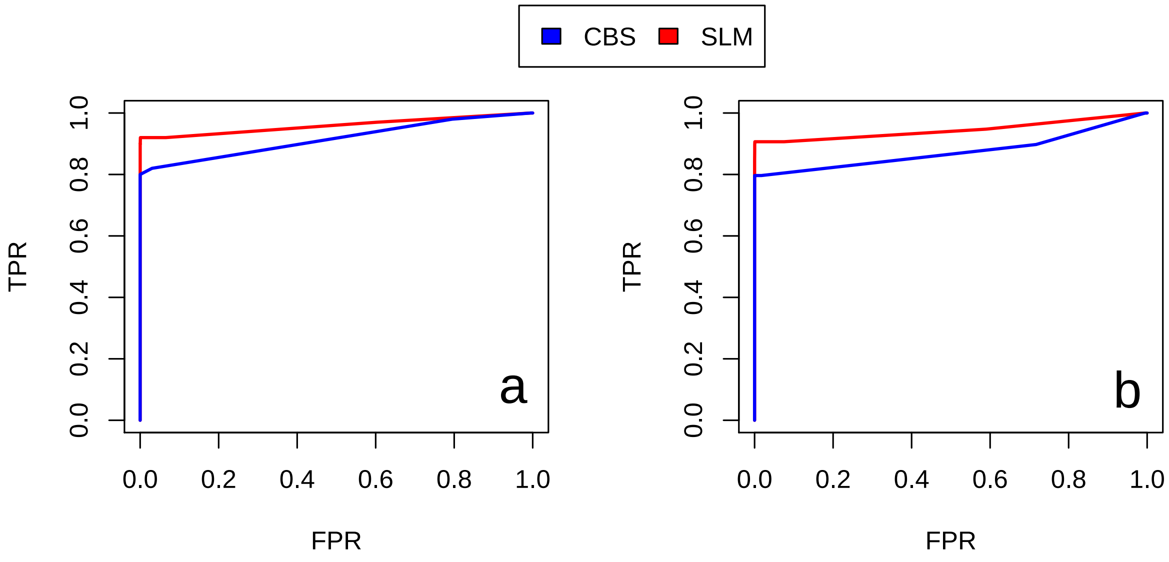

Figure 19: ROC curves for SLM and CBS on RC profiles from 500K reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 500K reads analyzed with 2 Mb window size.

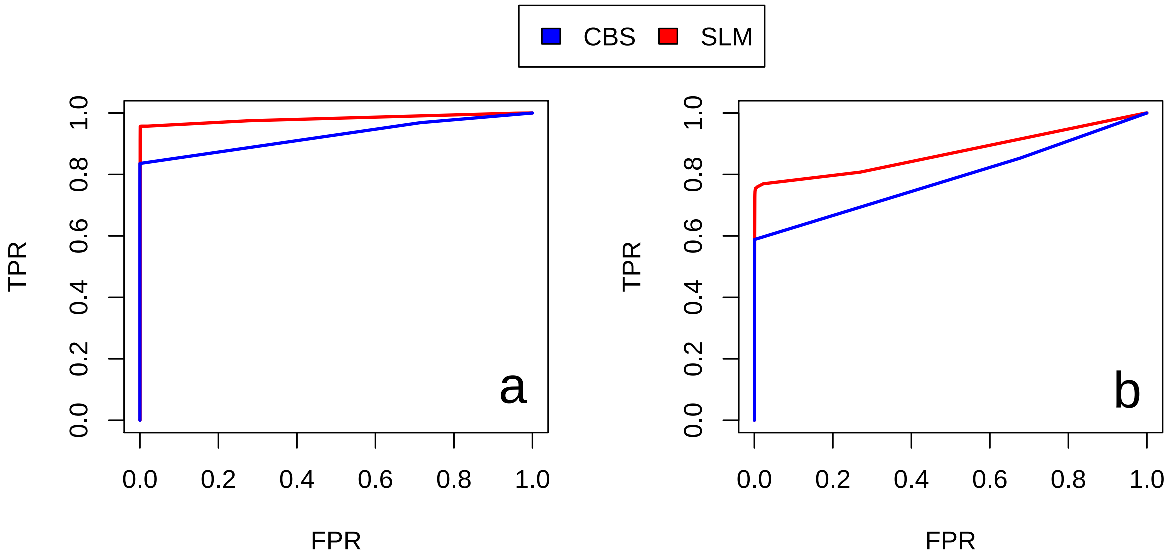

Figure 20: ROC curves for SLM and CBS on RC profiles from 1M reads analyzed with 100 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 1M reads analyzed with 100 Kb window size.

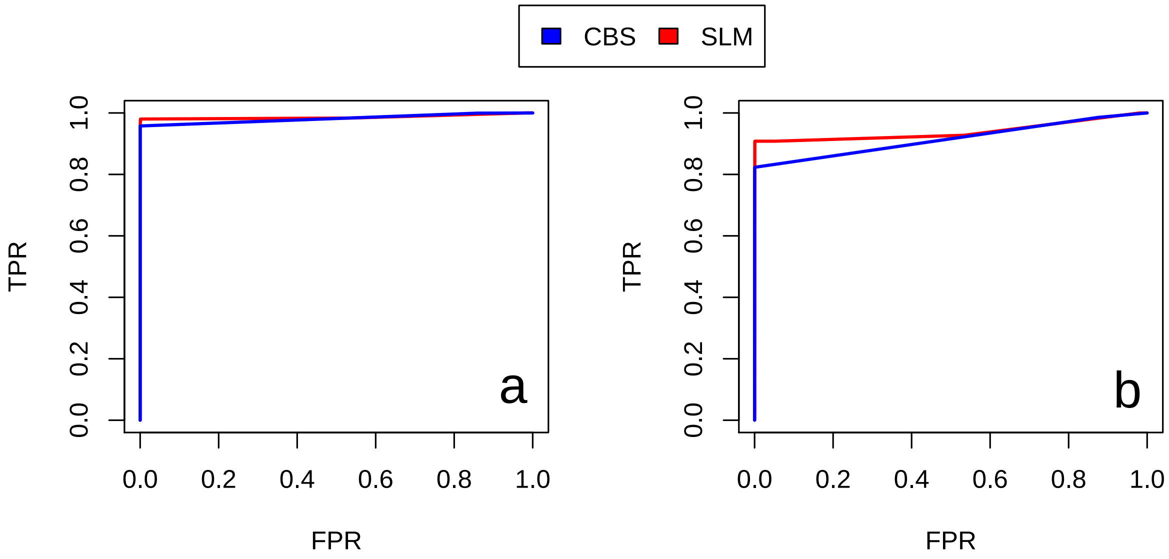

Figure 21: ROC curves for SLM and CBS on RC profiles from 1M reads analyzed with 200 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 1M reads analyzed with 200 Kb window size.

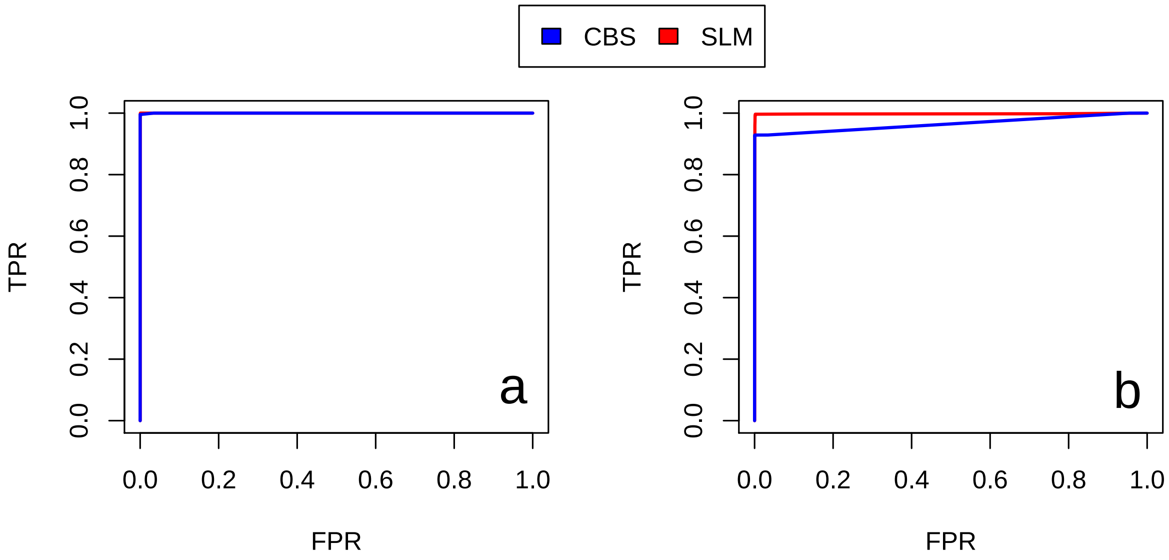

Figure 22: ROC curves for SLM and CBS on RC profiles from 1M reads analyzed with 500 Kb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 1M reads analyzed with 500 Kb window size.

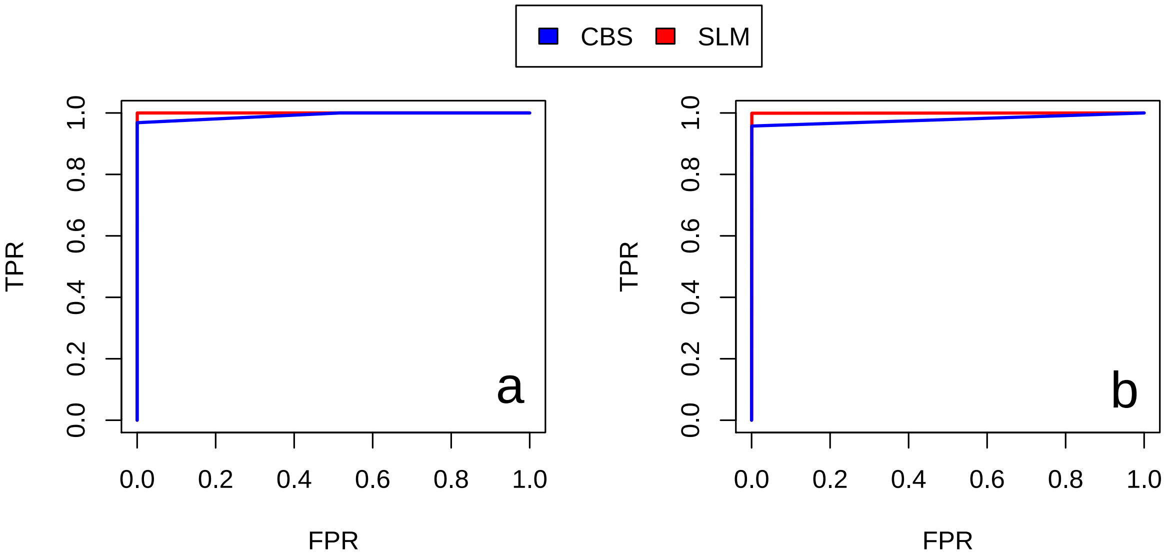

Figure 23: ROC curves for SLM and CBS on RC profiles from 1M reads analyzed with 1 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 1M reads analyzed with 1 Mb window size.

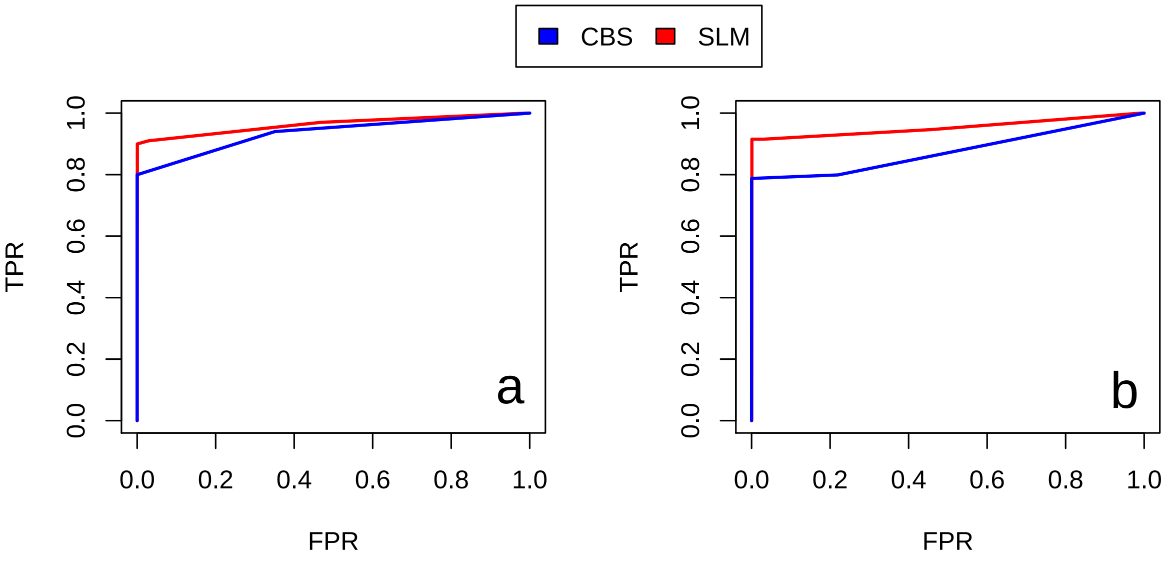

Figure 24: ROC curves for SLM and CBS on RC profiles from 1M reads analyzed with 2 Mb window size. Panels a and b report ROC curves to compare sensitivity and specificity of SLM and CBS algorithms in the detection of deletions (a) and duplications (b) on RC profiles from 1M reads analyzed with 2 Mb window size.

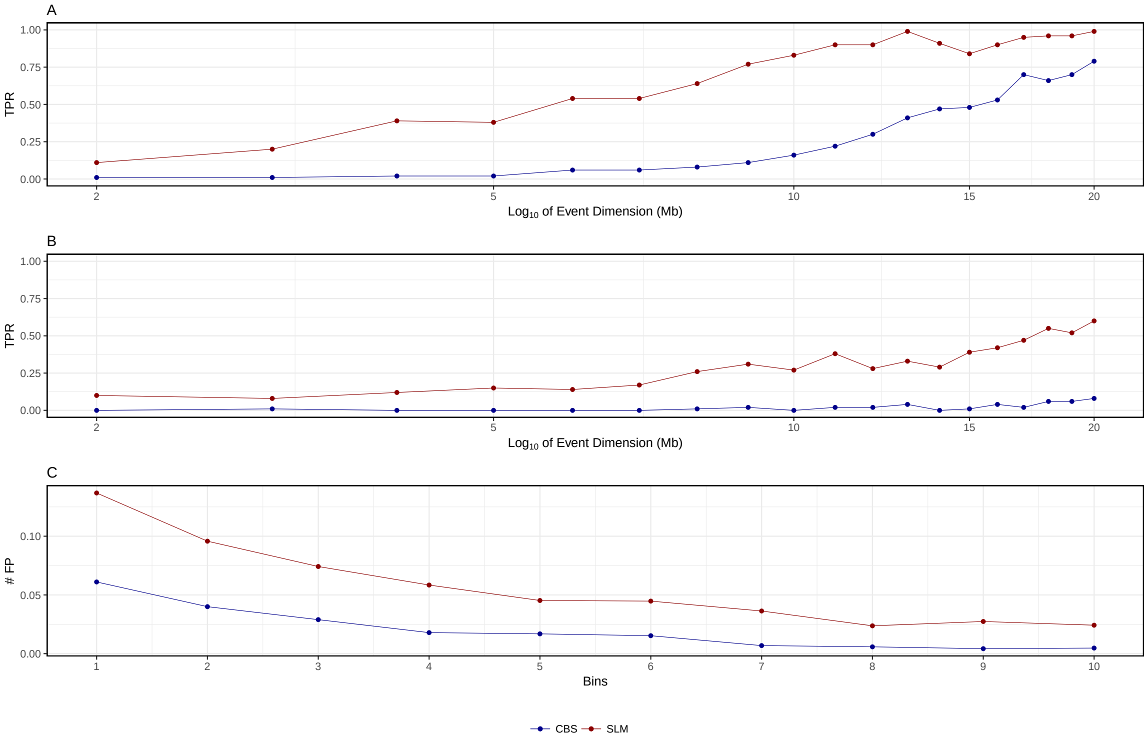

Figure 25: TPR and # FP for SLM and CBS on RC profiles from 10K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 1 Mb windows size obtained from bam file with 10k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

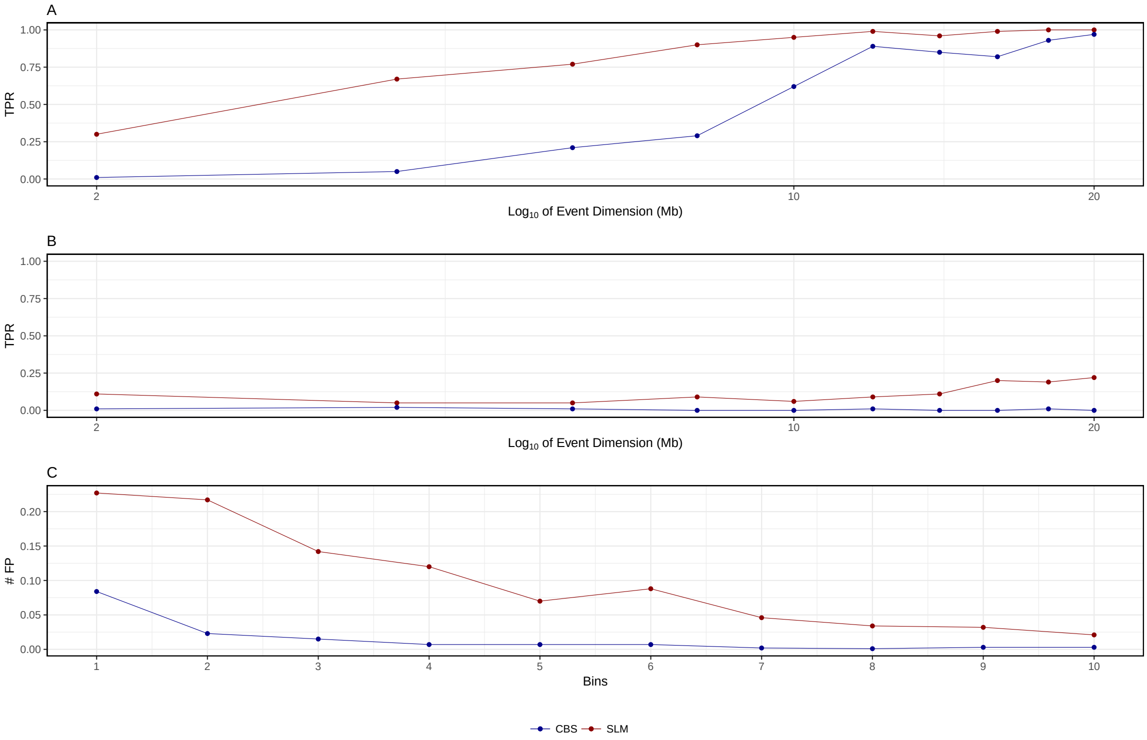

Figure 26: TPR and # FP for SLM and CBS on RC profiles from 10K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 2 Mb windows size obtained from bam file with 10k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

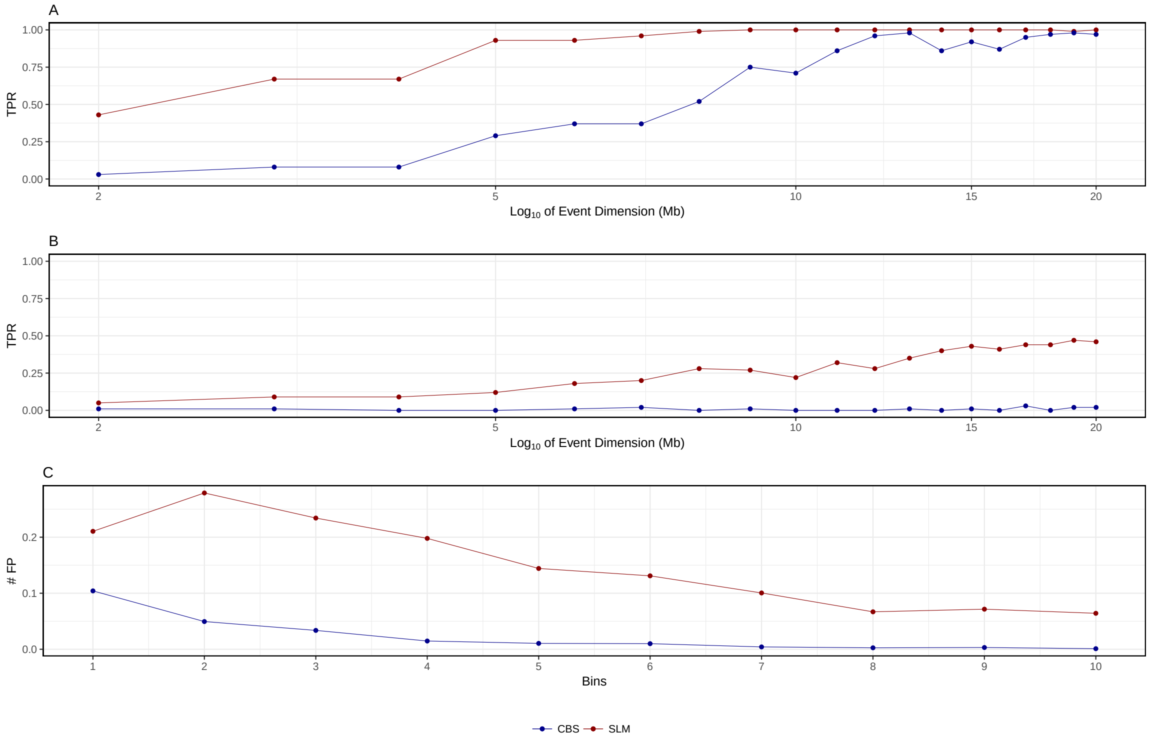

Figure 27: TPR and # FP for SLM and CBS on RC profiles from 20K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 1 Mb windows size obtained from bam file with 20k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

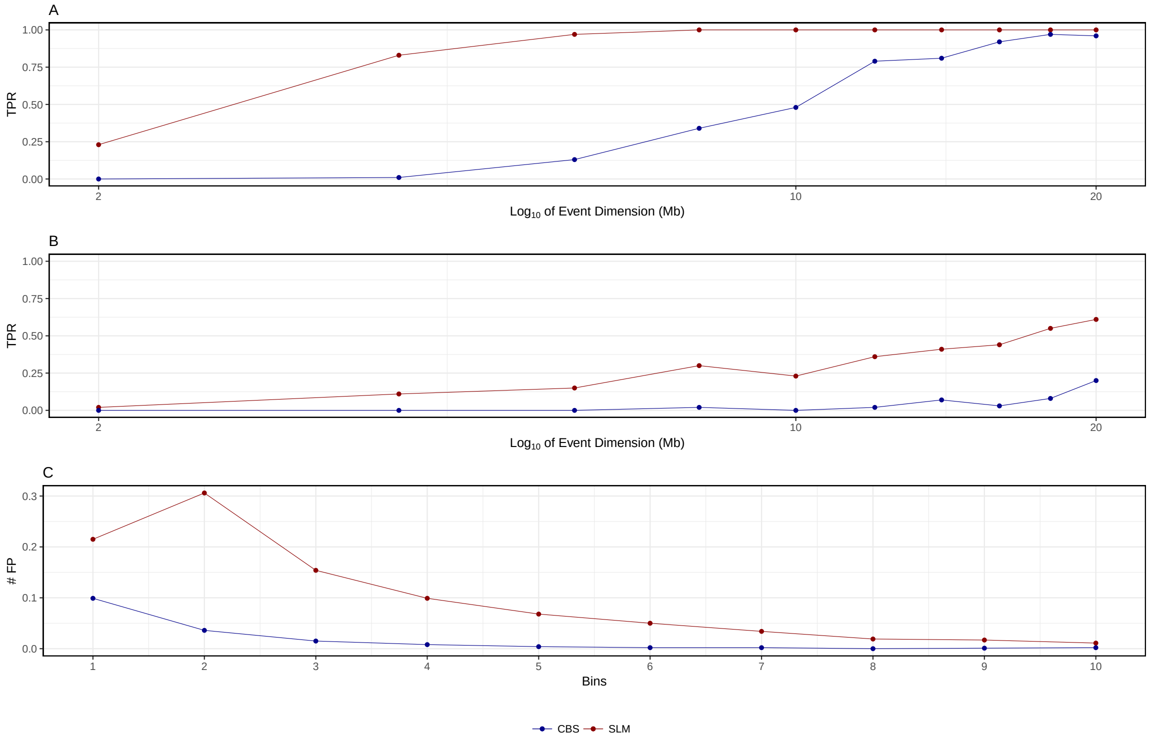

Figure 28: TPR and # FP for SLM and CBS on RC profiles from 20K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 2 Mb windows size obtained from bam file with 20k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

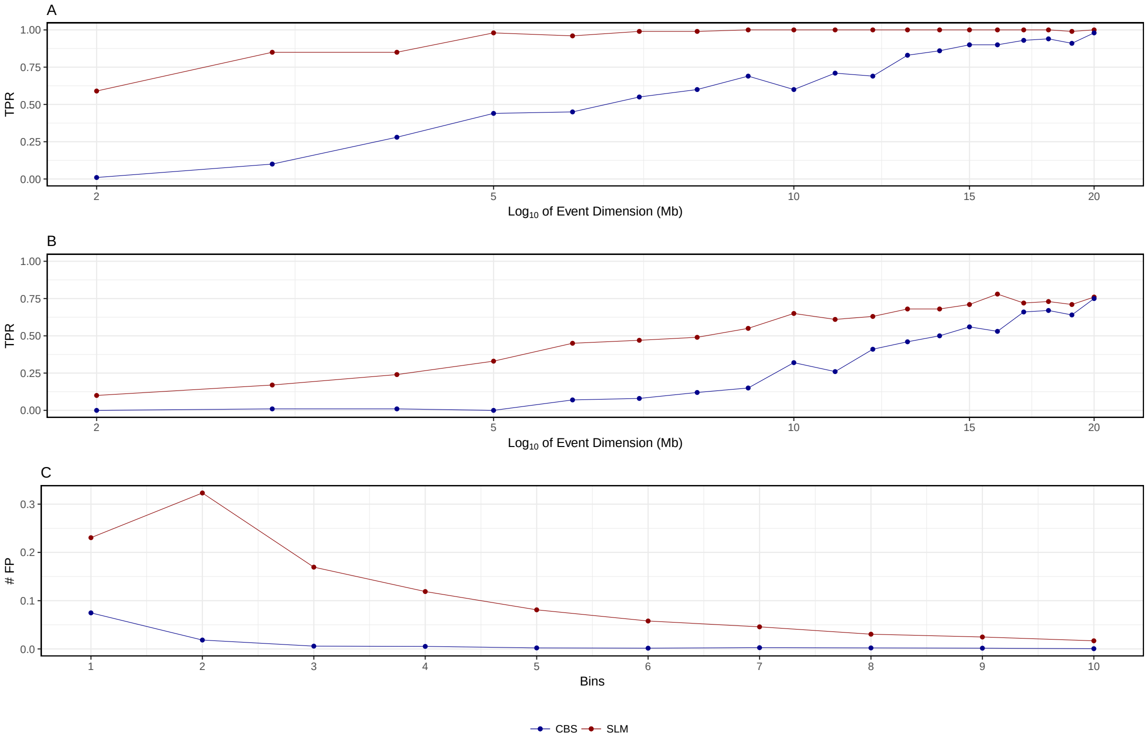

Figure 29: TPR and # FP for SLM and CBS on RC profiles from 50K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 1 Mb windows size obtained from bam file with 50k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

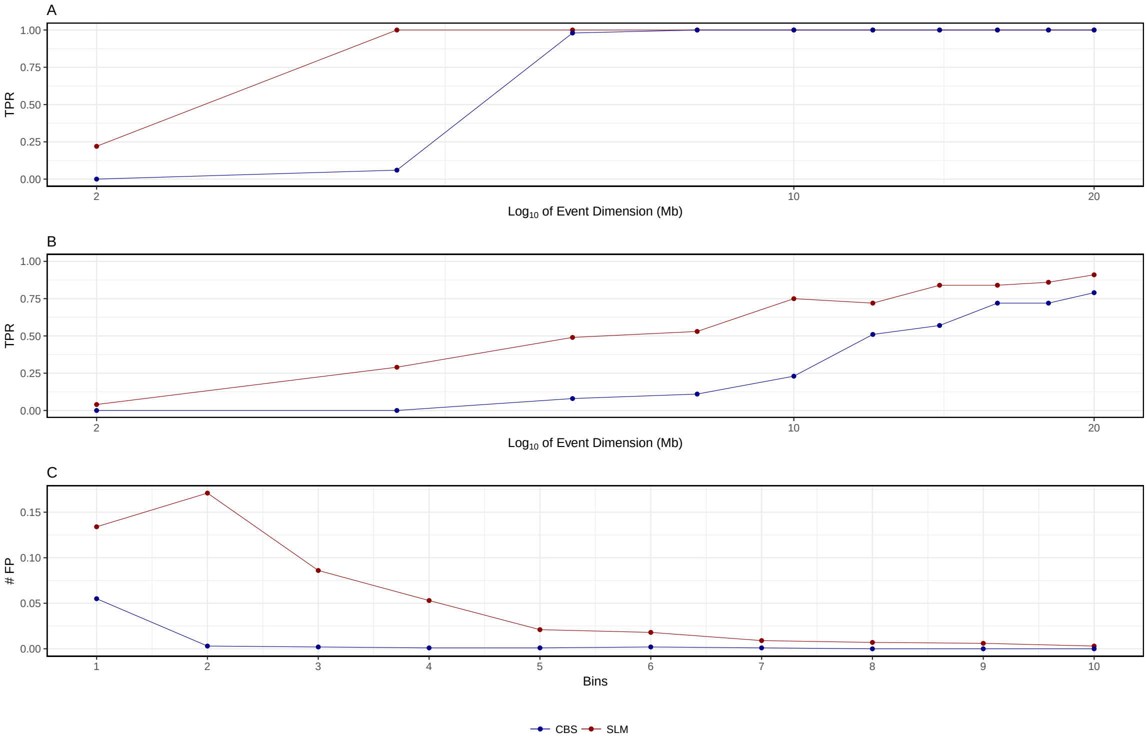

Figure 30: TPR and # FP for SLM and CBS on RC profiles from 50K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 2 Mb windows size obtained from bam file with 50k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

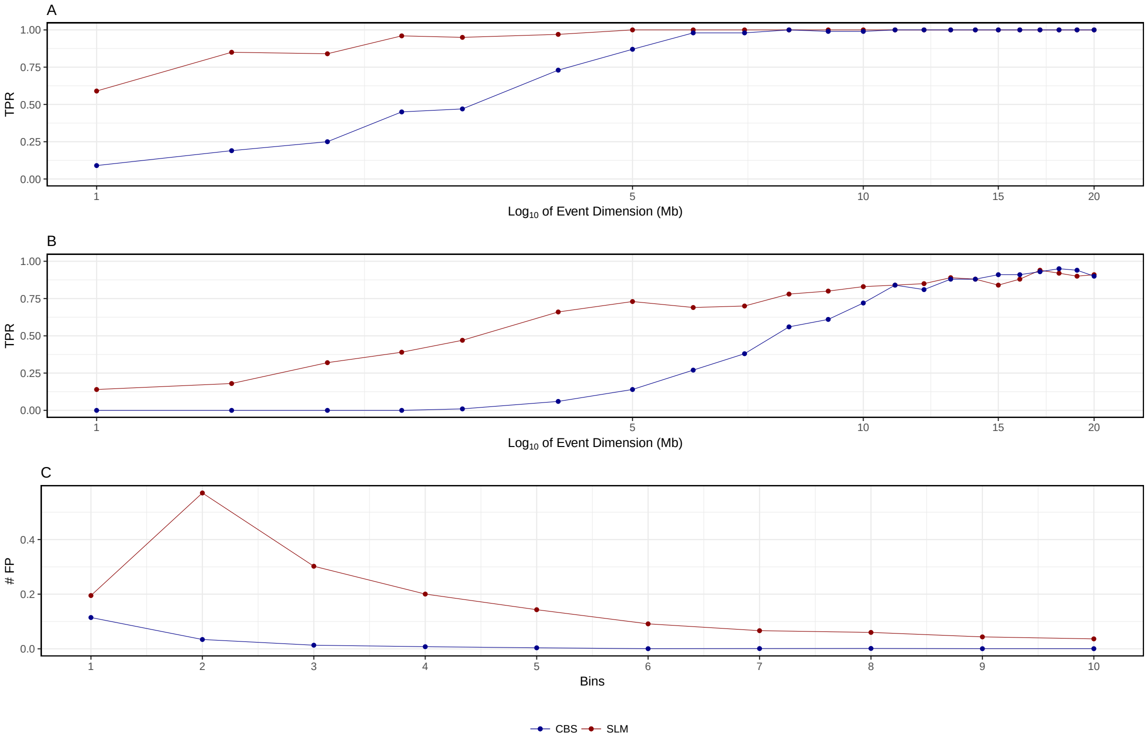

Figure 31: TPR and # FP for SLM and CBS on RC profiles from 100K reads analyzed with 500 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 500 Kb windows size obtained from bam file with 100k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

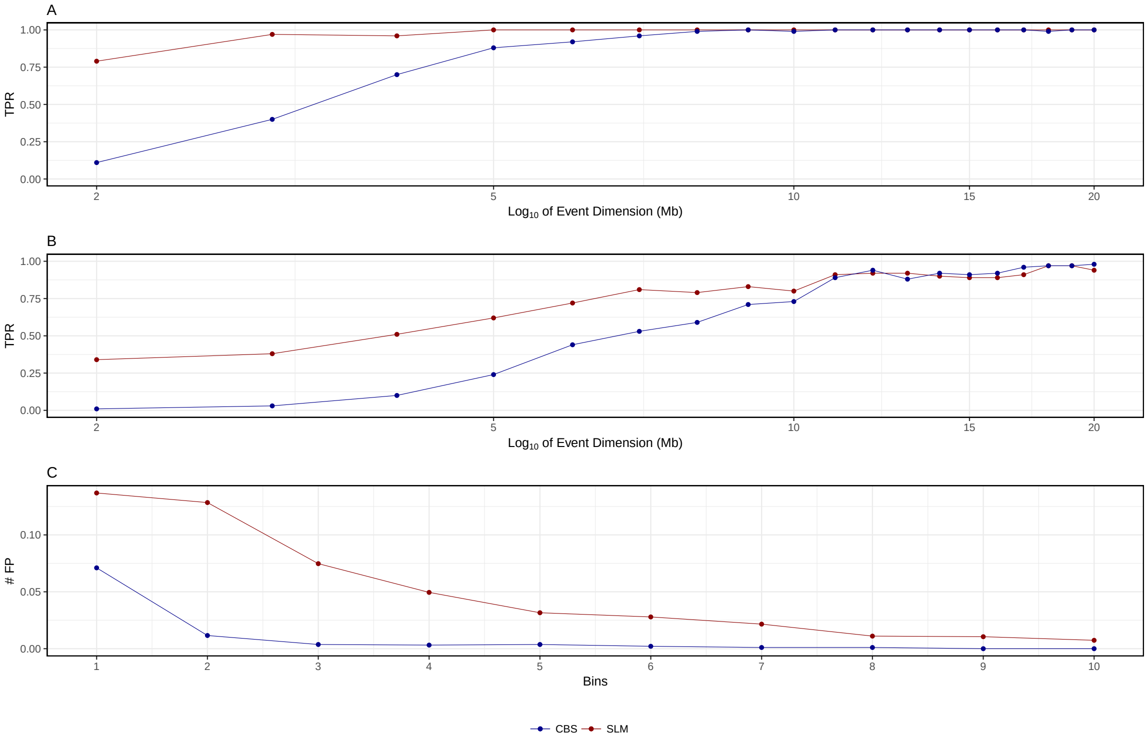

Figure 32: TPR and # FP for SLM and CBS on RC profiles from 100K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 1 Mb windows size obtained from bam file with 100k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

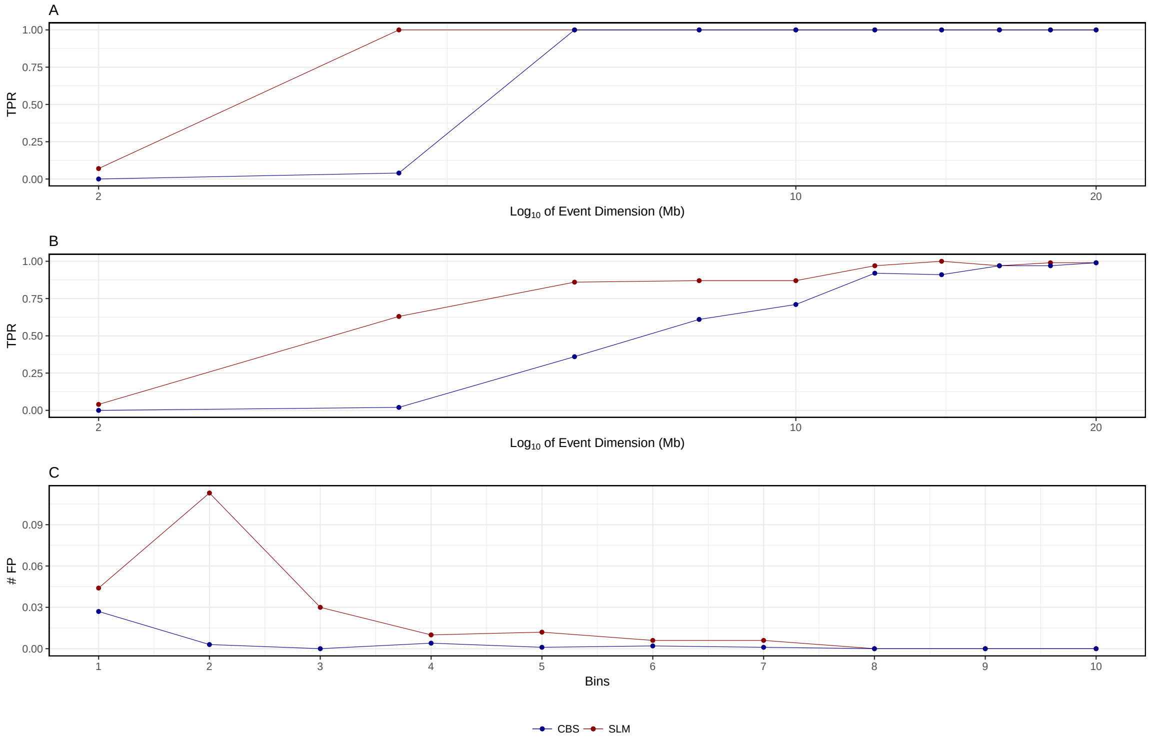

Figure 33: TPR and # FP for SLM and CBS on RC profiles from 100K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 2 Mb windows size obtained from bam file with 100k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

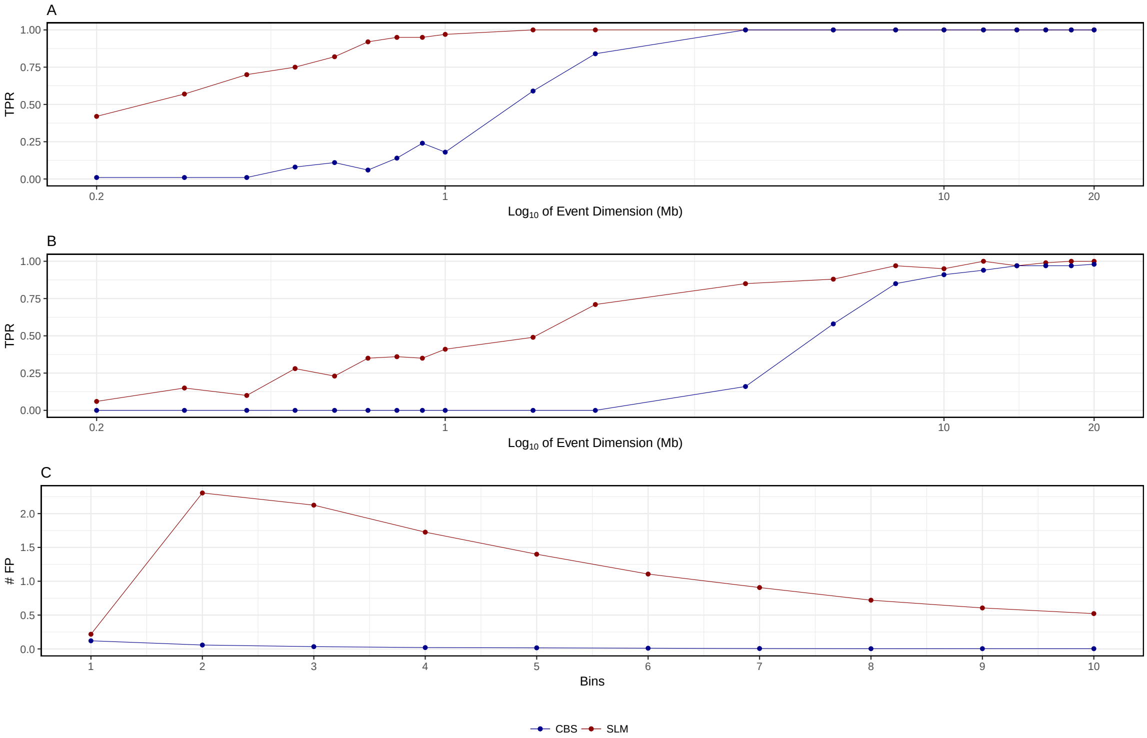

Figure 34: TPR and # FP for SLM and CBS on RC profiles from 200K reads analyzed with 100 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 100 Kb windows size obtained from bam file with 200k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

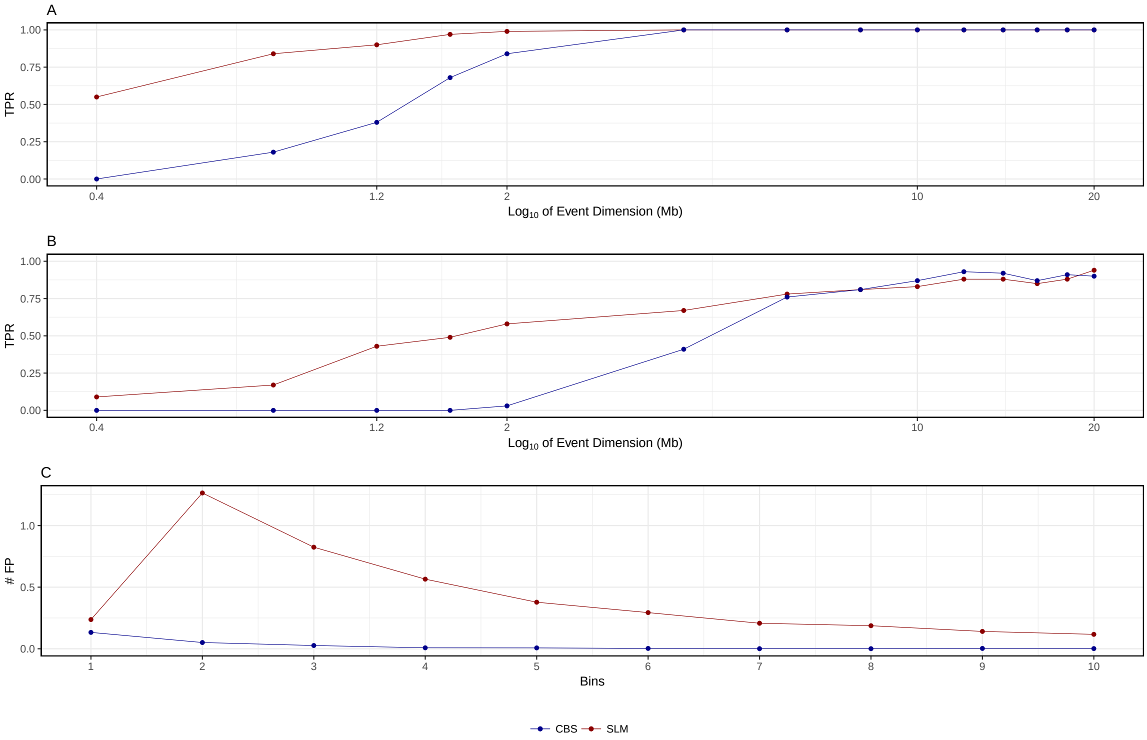

Figure 35: TPR and # FP for SLM and CBS on RC profiles from 200K reads analyzed with 200 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 200 Kb windows size obtained from bam file with 200k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

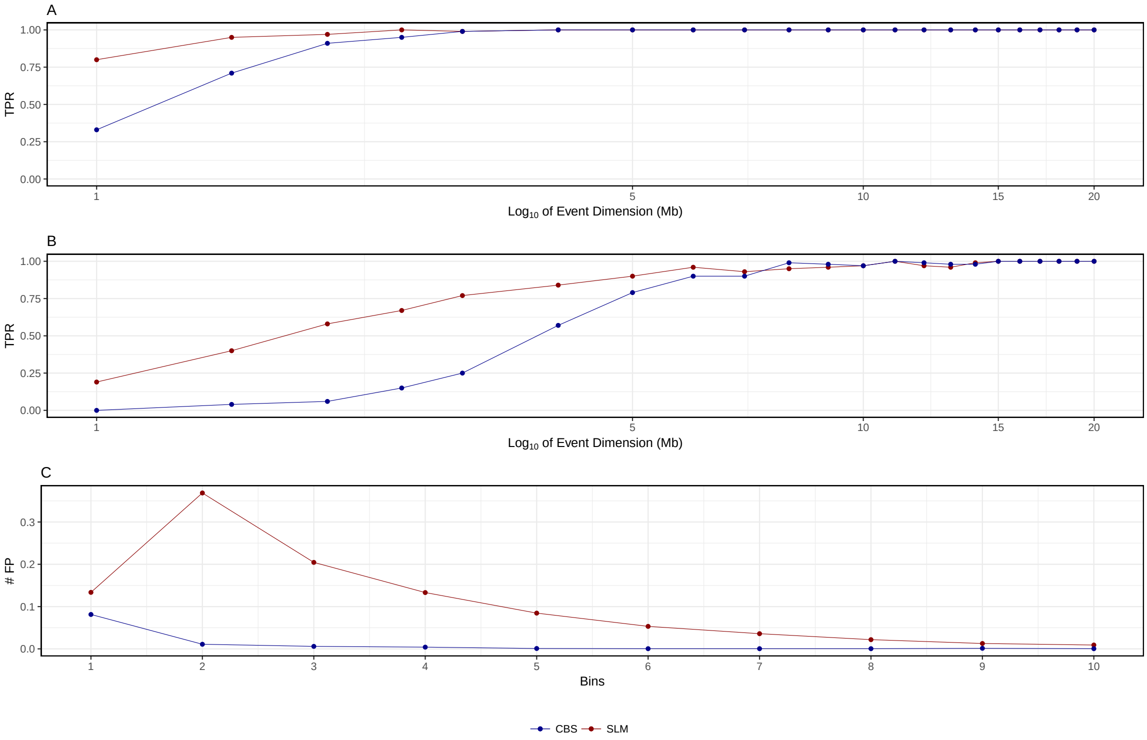

Figure 36: TPR and # FP for SLM and CBS on RC profiles from 200K reads analyzed with 500 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 500 Kb windows size obtained from bam file with 200k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

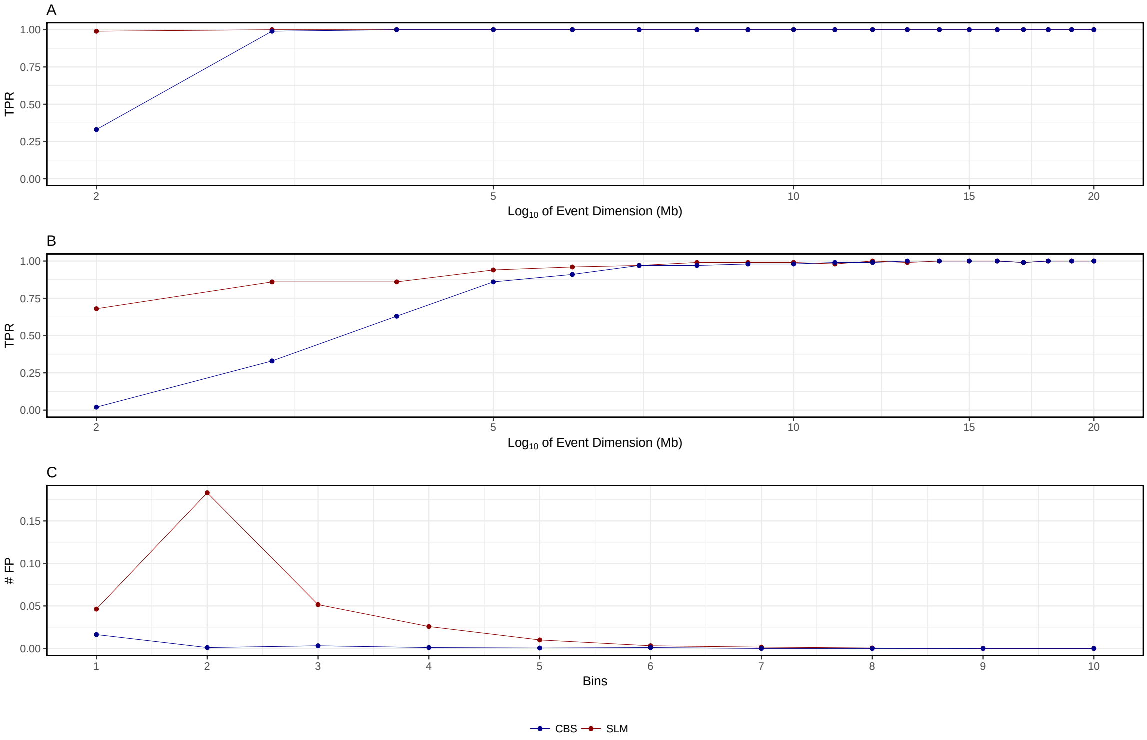

Figure 37: TPR and # FP for SLM and CBS on RC profiles from 200K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 1 Mb windows size obtained from bam file with 200k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

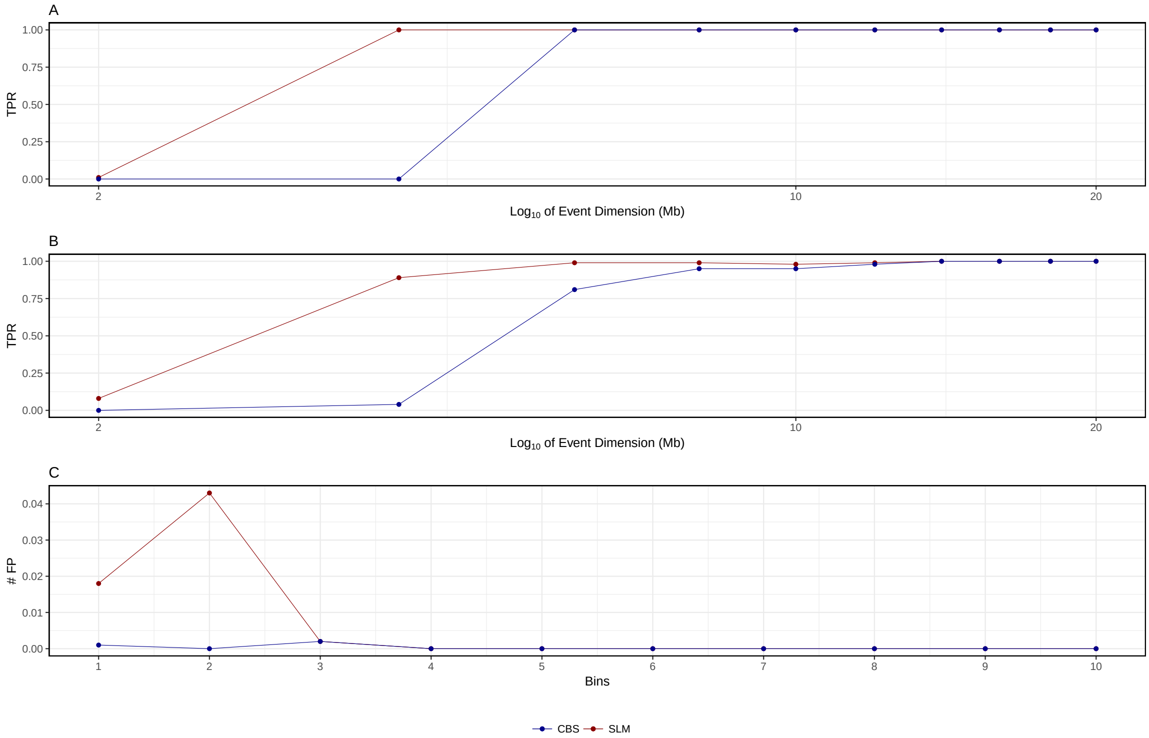

Figure 38: TPR and # FP for SLM and CBS on RC profiles from 200K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 2 Mb windows size obtained from bam file with 200k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

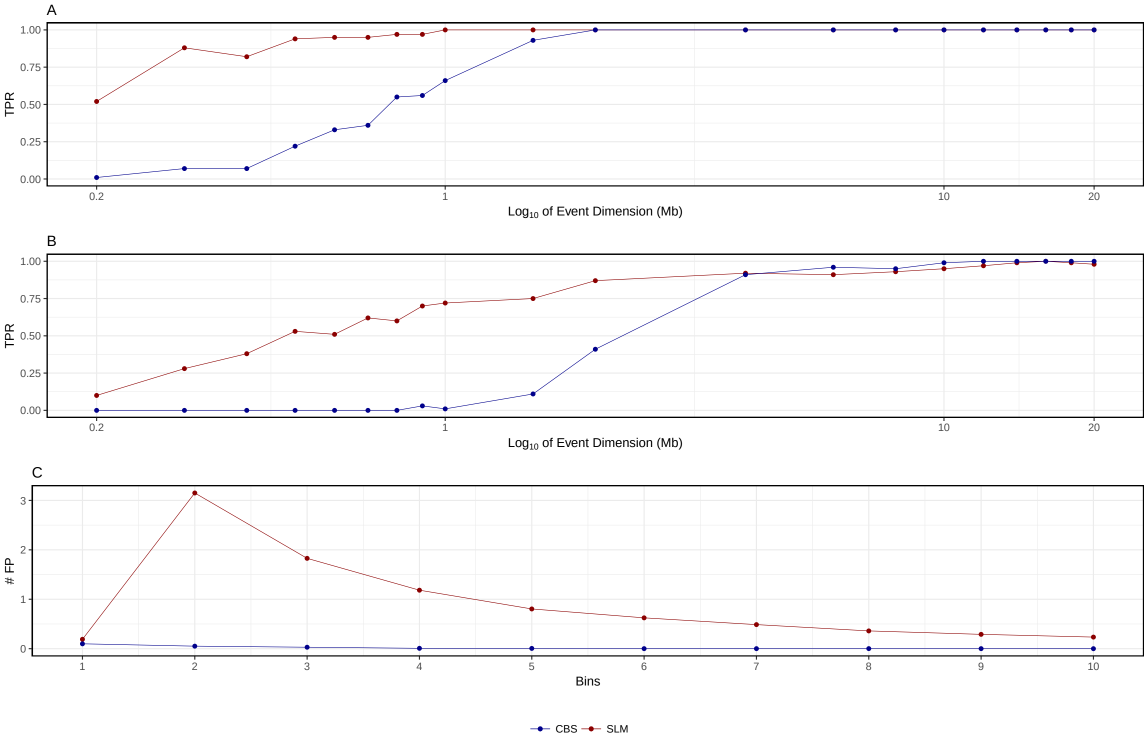

Figure 39: TPR and # FP for SLM and CBS on RC profiles from 500K reads analyzed with 100 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 100 Kb windows size obtained from bam file with 500k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

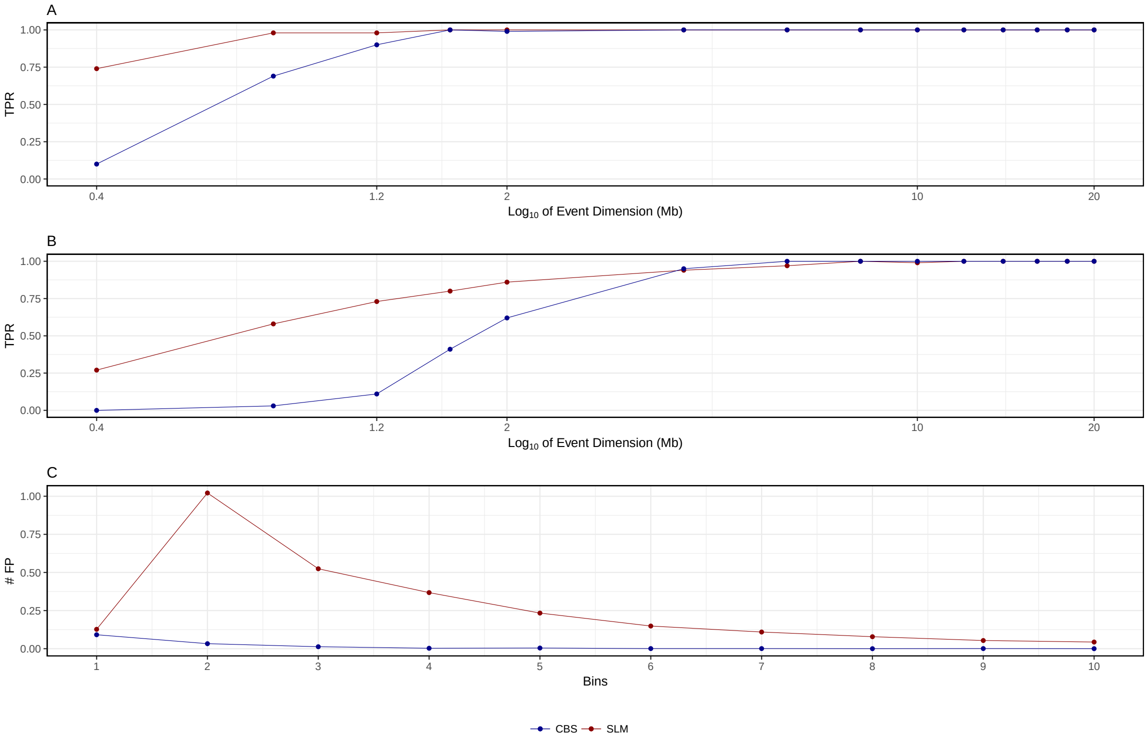

Figure 40: TPR and # FP for SLM and CBS on RC profiles from 500K reads analyzed with 200 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 200 Kb windows size obtained from bam file with 500k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

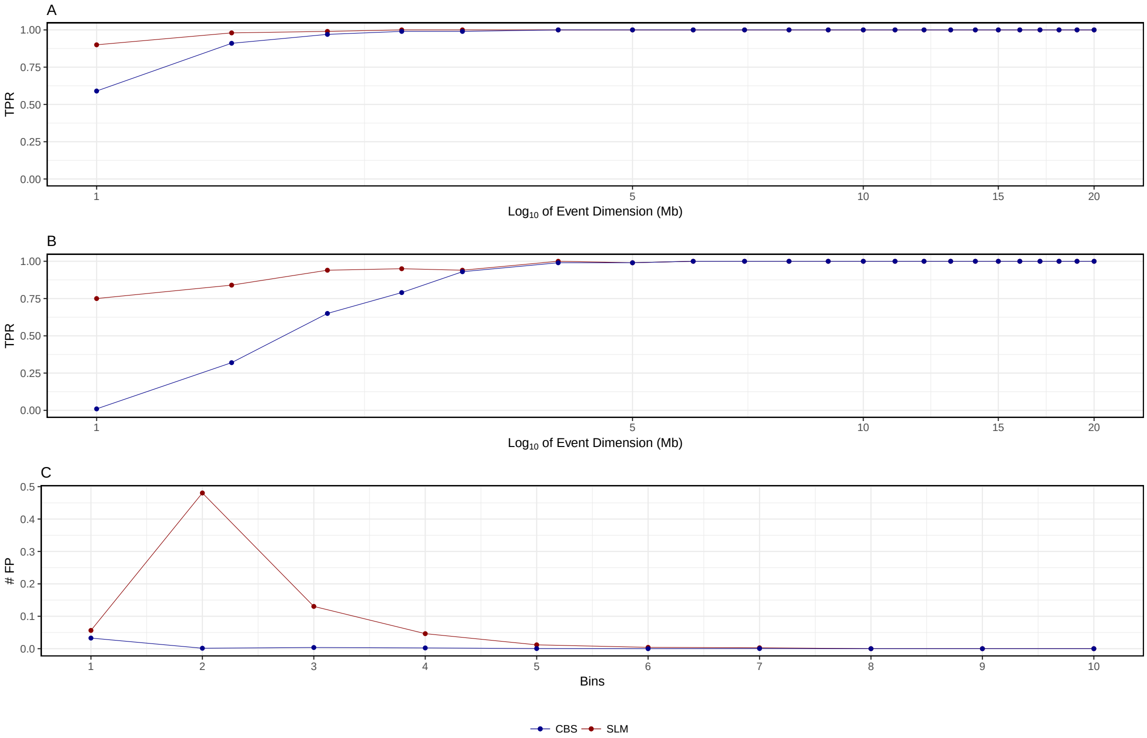

Figure 41: TPR and # FP for SLM and CBS on RC profiles from 500K reads analyzed with 500 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 500 Kb windows size obtained from bam file with 500k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

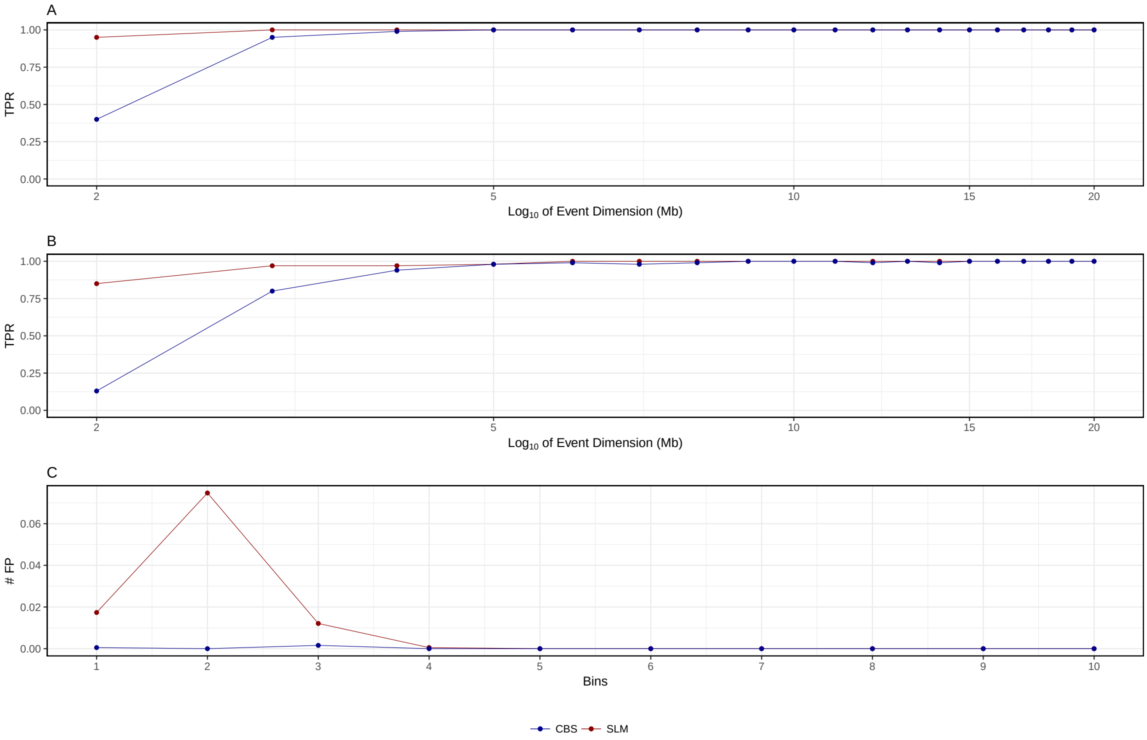

Figure 42: TPR and # FP for SLM and CBS on RC profiles from 500K reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 1 Mb windows size obtained from bam file with 500k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

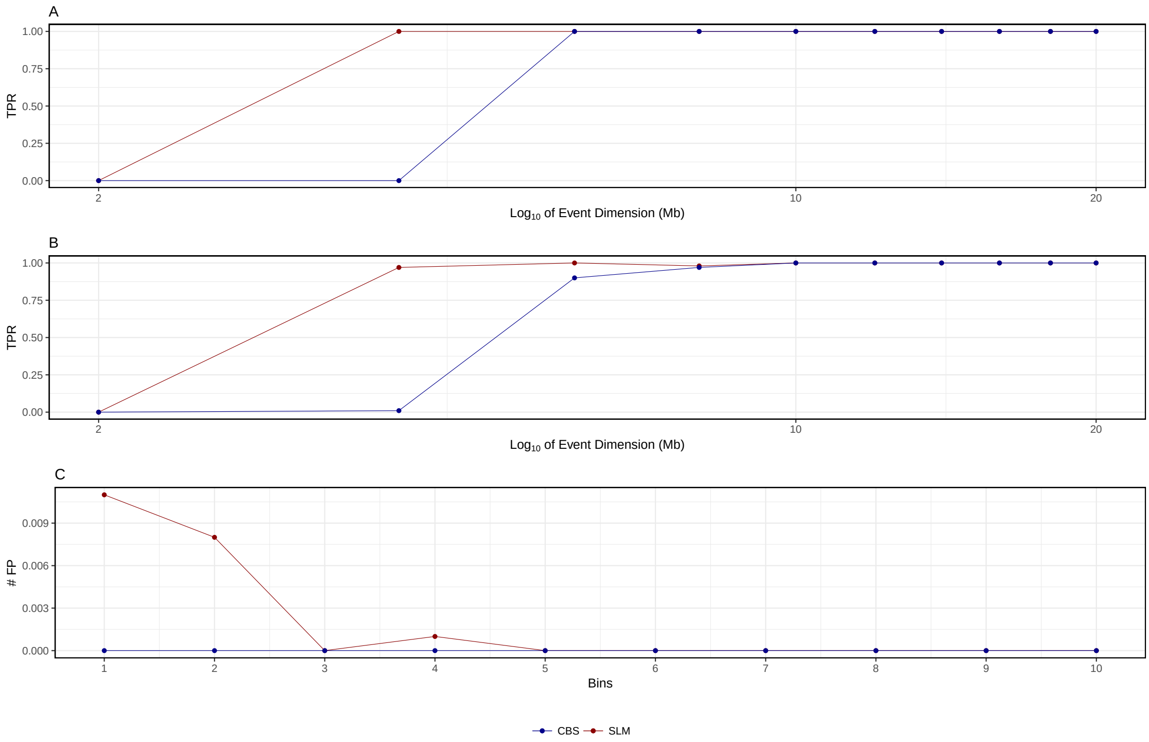

Figure 43: TPR and # FP for SLM and CBS on RC profiles from 500K reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 2 Mb windows size obtained from bam file with 500k reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

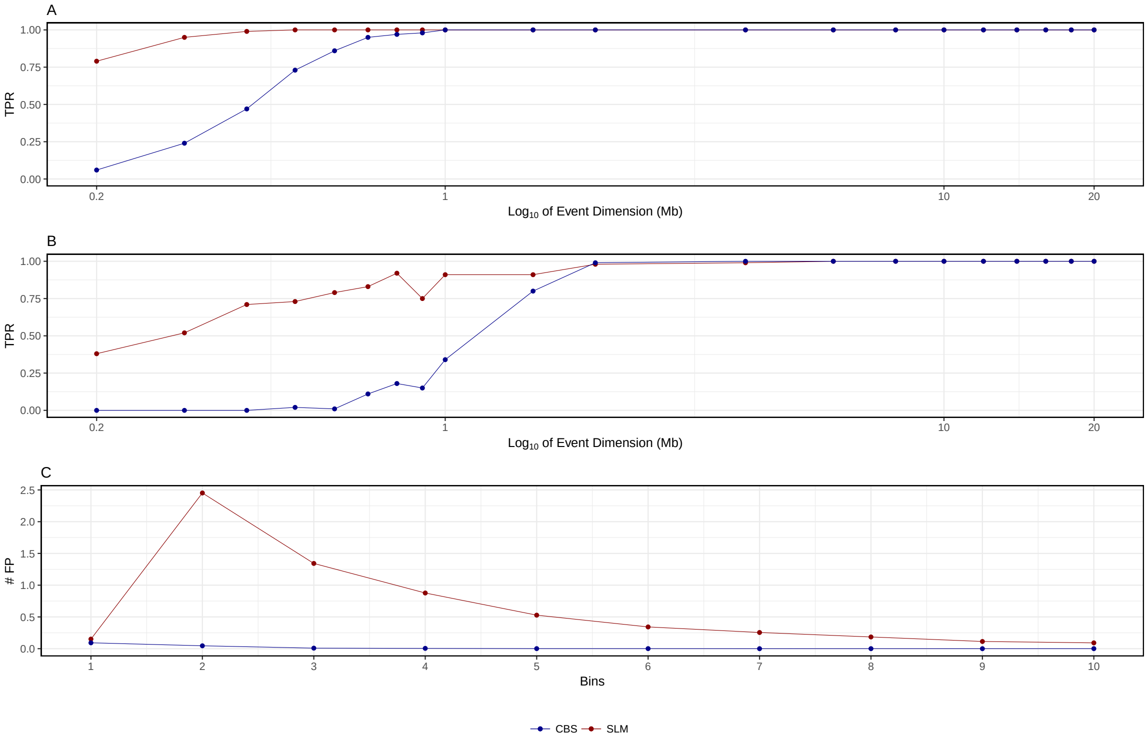

Figure 44: TPR and # FP for SLM and CBS on RC profiles from 1M reads analyzed with 100 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of log_2RC_{Norm} profiles with 100 Kb windows size obtained from bam file with 1M reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

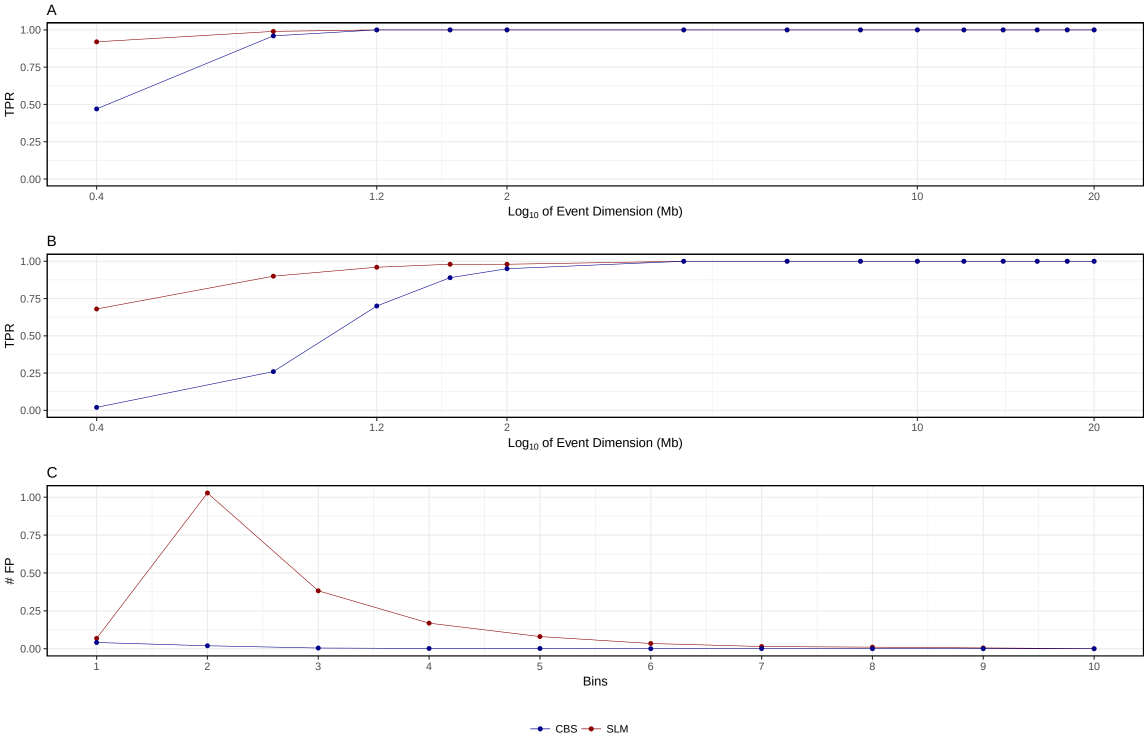

Figure 45: TPR and # FP for SLM and CBS on RC profiles from 1M reads analyzed with 200 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 200 Kb windows size obtained from bam file with 1M reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

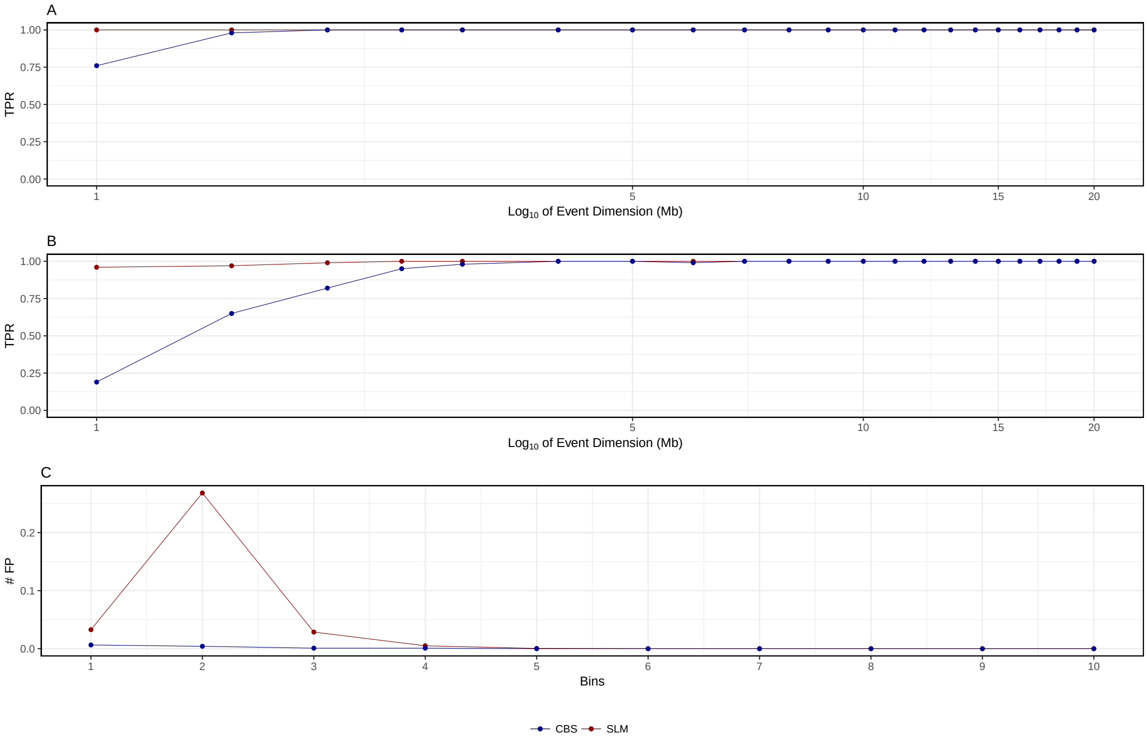

Figure 46: TPR and # FP for SLM and CBS on RC profiles from 1M reads analyzed with 500 Kb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 500 Kb windows size obtained from bam file with 1M reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

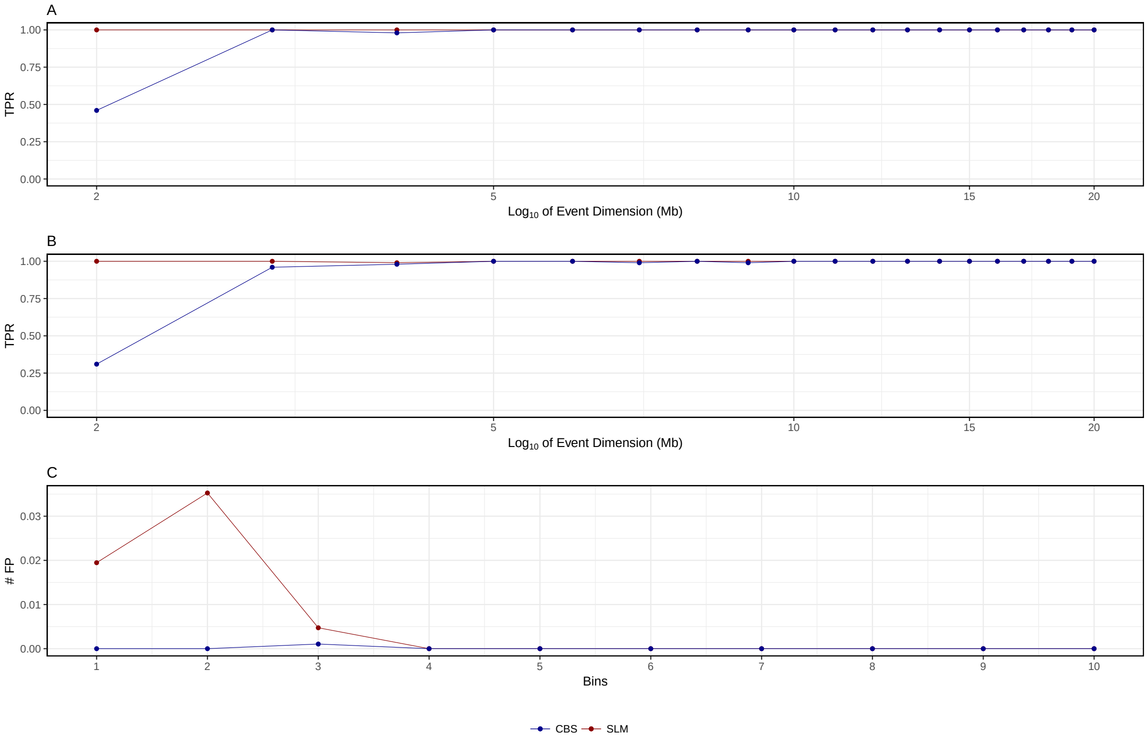

Figure 47: TPR and # FP for SLM and CBS on RC profiles from 1M reads analyzed with 1 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 1 Mb windows size obtained from bam file with 1M reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

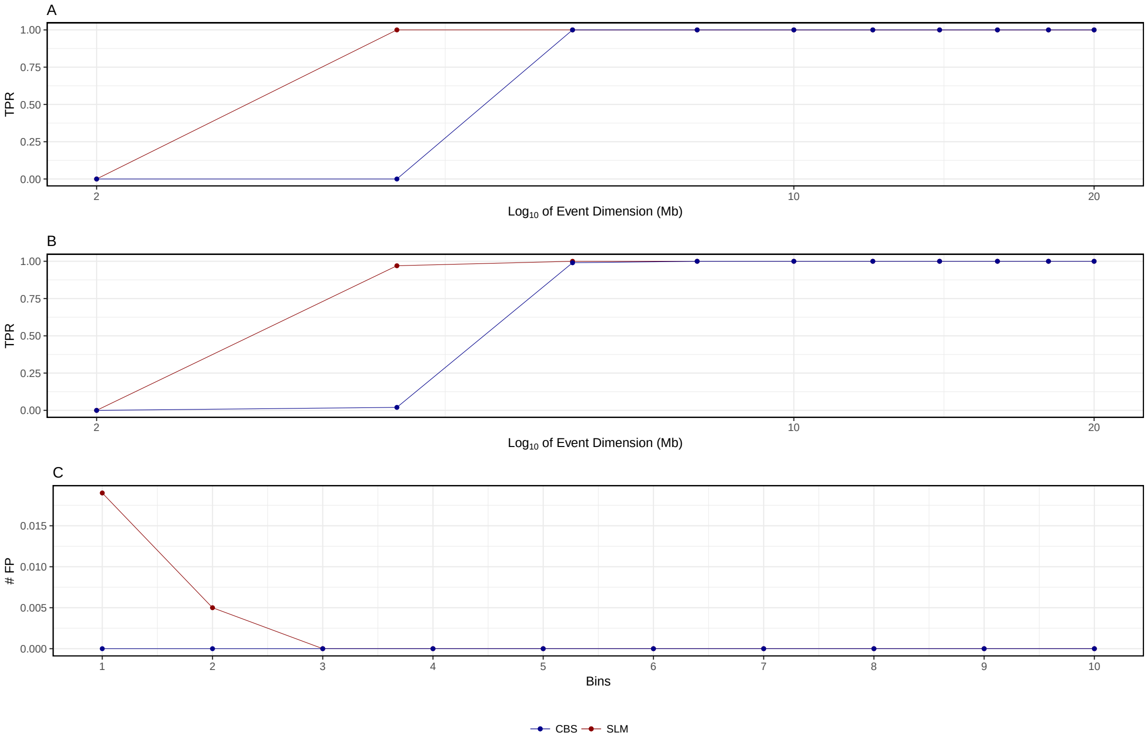

Figure 48: TPR and # FP for SLM and CBS on RC profiles from 1M reads analyzed with 2 Mb window size. Panels report the TPR and number of FP detected by the two segmentation algorithm on the analysis of $log_2 RC_{Norm}$ profiles with 2 Mb windows size obtained from bam file with 1M reads generated by Xome-Blender. Panels A and B report TPR for duplications (A) and deletions (B) as a function of event size. Panel C reports number of FP detected by the two segmentation algorithms.

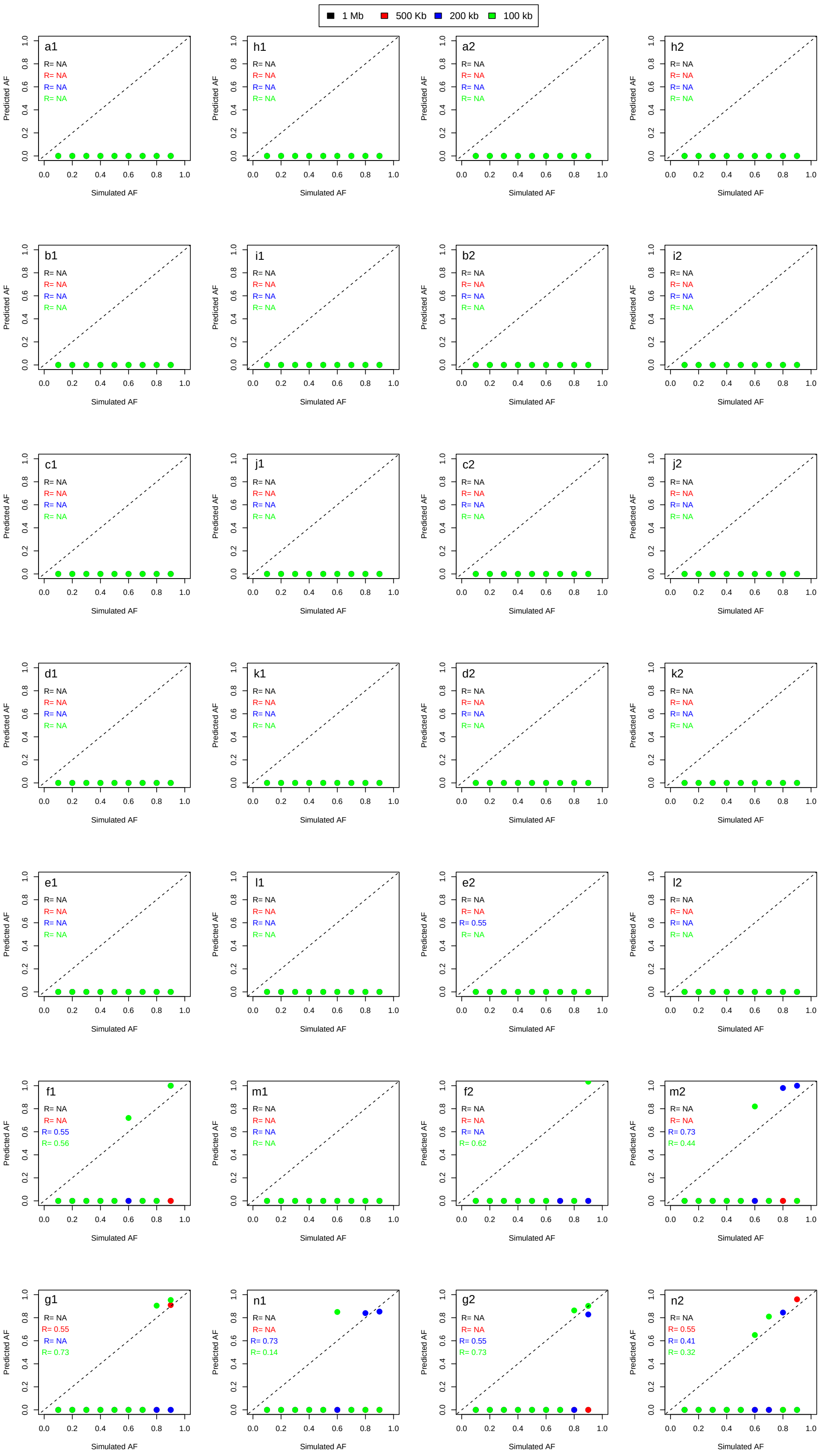

Figure 49: Allelic fraction prediction results for for duplications and deletions of 1 Mb. In panels are reported the correlations between the allelic fractions simulated by Xome-blender and the allelic fraction predicted by our novel probabilistic method FractionPred for different window sizes (100 Kb, 200 Kb, 500 Kb and 1 Mb). Panels (a1-n1) report the results for single sample analysis, (a1-g1) for deletions and (h1-n1) for duplications, while panels (a2-n2) report the results for paired sample analysis, (a2-g2) for deletions and (h2-n2) for duplications. Each panel row report prediction results for different number of simulated reads: N=10.000 (a1,h1,a2,h2), N=20.000 (b1,i1,b2,i2), N=50.000 (c1,j1,c2,j2), N=100.000 (d1,k1,d2,k2), N=200.000 (e1,l1,e2,l2), N=500.000 (f1,m1,f2,m2), N=1.00.000 (g1,m1,g2,m2). Colors reported in legend represent the four window sizes used in the analyses.

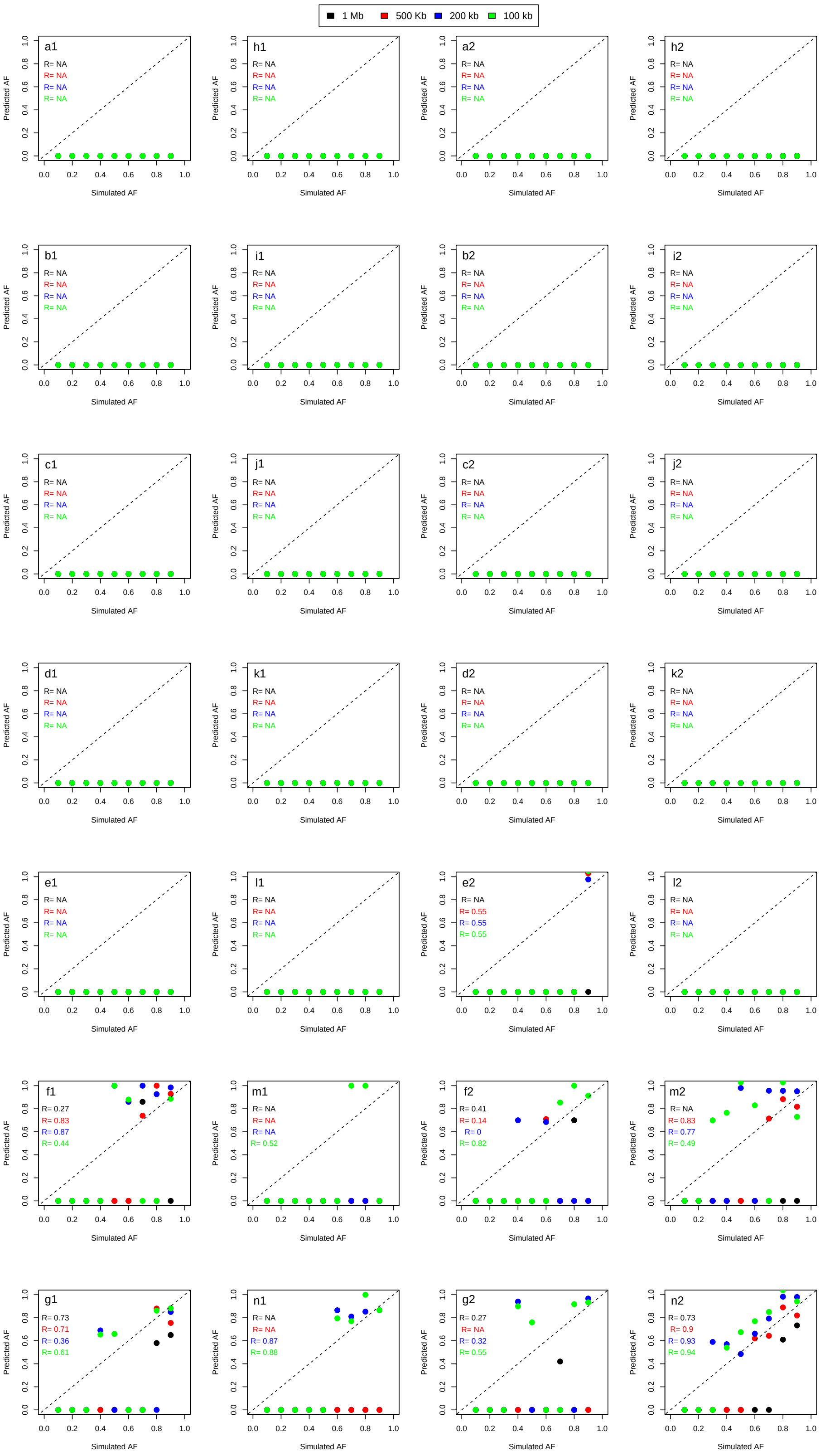

Figure 50: Allelic fraction prediction results for for duplications and deletions of 2 Mb. In panels are reported the correlations between the allelic fractions simulated by Xome-blender and the allelic fraction predicted by our novel probabilistic method FractionPred for different window sizes (100 Kb, 200 Kb, 500 Kb and 1 Mb). Panels (a1-n1) report the results for single sample analysis, (a1-g1) for deletions and (h1-n1) for duplications, while panels (a2-n2) report the results for paired sample analysis, (a2-g2) for deletions and (h2-n2) for duplications. Each panel row report prediction results for different number of simulated reads: N=10.000 (a1,h1,a2,h2), N=20.000 (b1,i1,b2,i2), N=50.000 (c1,j1,c2,j2), N=100.000 (d1,k1,d2,k2), N=200.000 (e1,l1,e2,l2), N=500.000 (f1,m1,f2,m2), N=1.00.000 (g1,m1,g2,m2). Colors reported in legend represent the four window sizes used in the analyses.

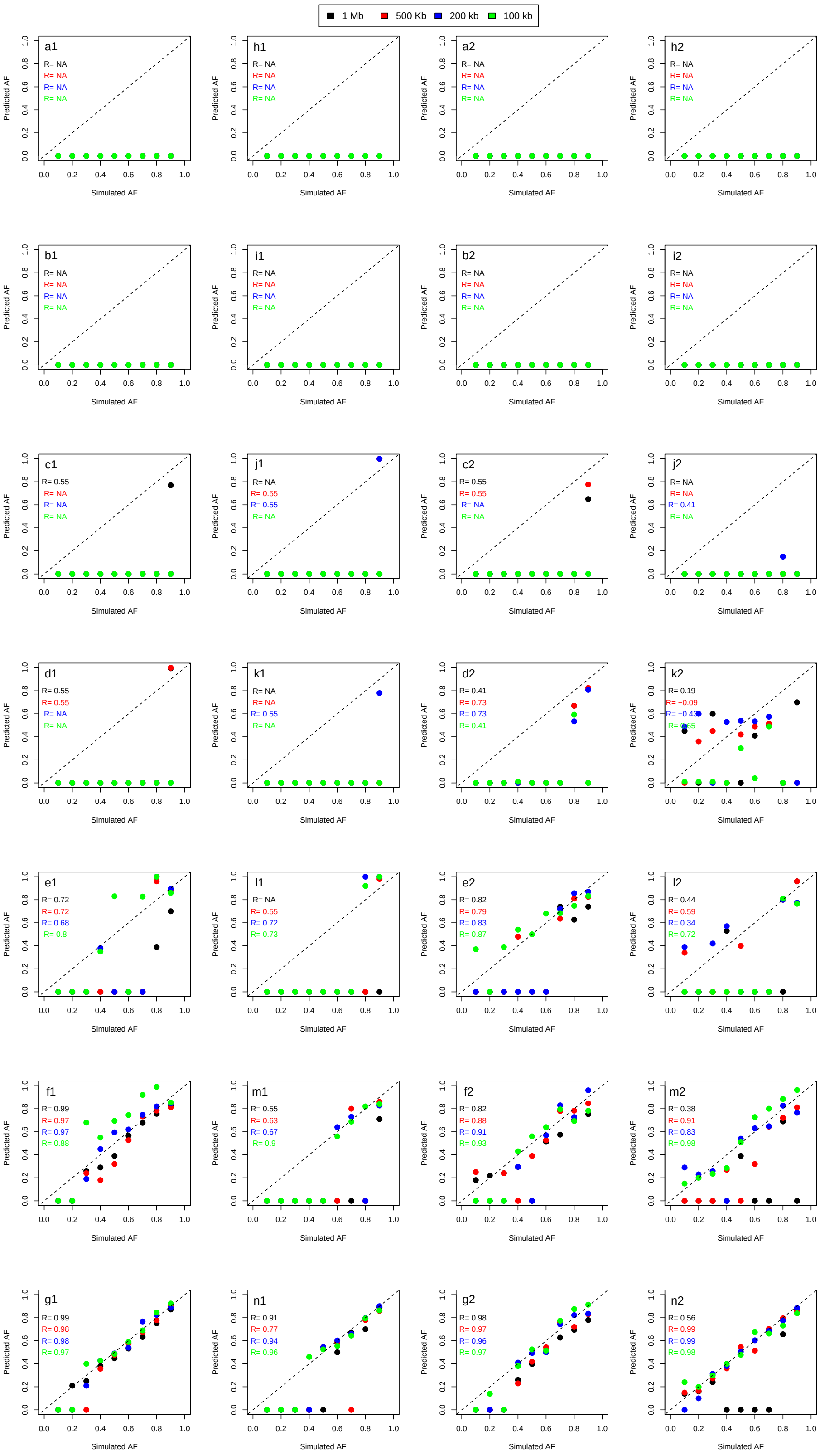

Figure 51: Allelic fraction prediction results for duplications and deletions of 5 Mb. In panels are reported the correlations between the allelic fractions simulated by Xome-blender and the allelic fraction predicted by our novel probabilistic method FractionPred for different window sizes (100 Kb, 200 Kb, 500 Kb and 1 Mb). Panels (a1-n1) report the results for single sample analysis, (a1-g1) for deletions and (h1-n1) for duplications, while panels (a2-n2) report the results for paired sample analysis, (a2-g2) for deletions and (h2-n2) for duplications. Each panel row report prediction results for different number of simulated reads: N=10.000 (a1,h1,a2,h2), N=20.000 (b1,i1,b2,i2), N=50.000 (c1,j1,c2,j2), N=100.000 (d1,k1,d2,k2), N=200.000 (e1,l1,e2,l2), N=500.000 (f1,m1,f2,m2), N=1.00.000 (g1,m1,g2,m2). Colors reported in legend represent the four window sizes used in the analyses.

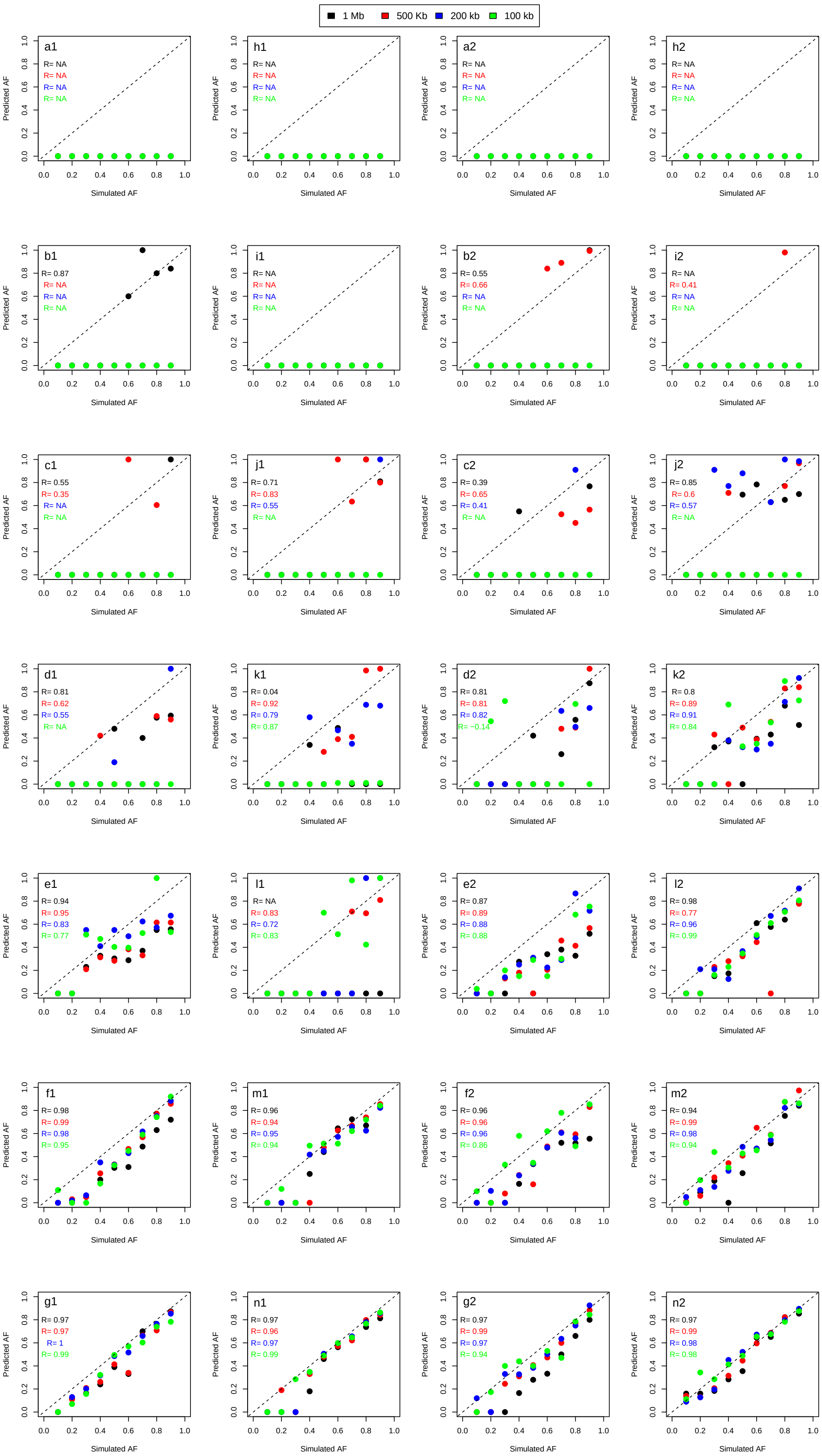

Figure 52: Allelic fraction prediction results for duplications and deletions of 10 Mb. In panels are reported the correlations between the allelic fractions simulated by Xome-blender and the allelic fraction predicted by our novel probabilistic method FractionPred for different window sizes (100 Kb, 200 Kb, 500 Kb and 1 Mb). Panels (a1-n1) report the results for single sample analysis, (a1-g1) for deletions and (h1-n1) for duplications, while panels (a2-n2) report the results for paired sample analysis, (a2-g2) for deletions and (h2-n2) for duplications. Each panel row report prediction results for different number of simulated reads: N=10.000 (a1,h1,a2,h2), N=20.000 (b1,i1,b2,i2), N=50.000 (c1,j1,c2,j2), N=100.000 (d1,k1,d2,k2), N=200.000 (e1,l1,e2,l2), N=500.000 (f1,m1,f2,m2), N=1.00.000 (g1,m1,g2,m2). Colors reported in legend represent the four window sizes used in the analyses.

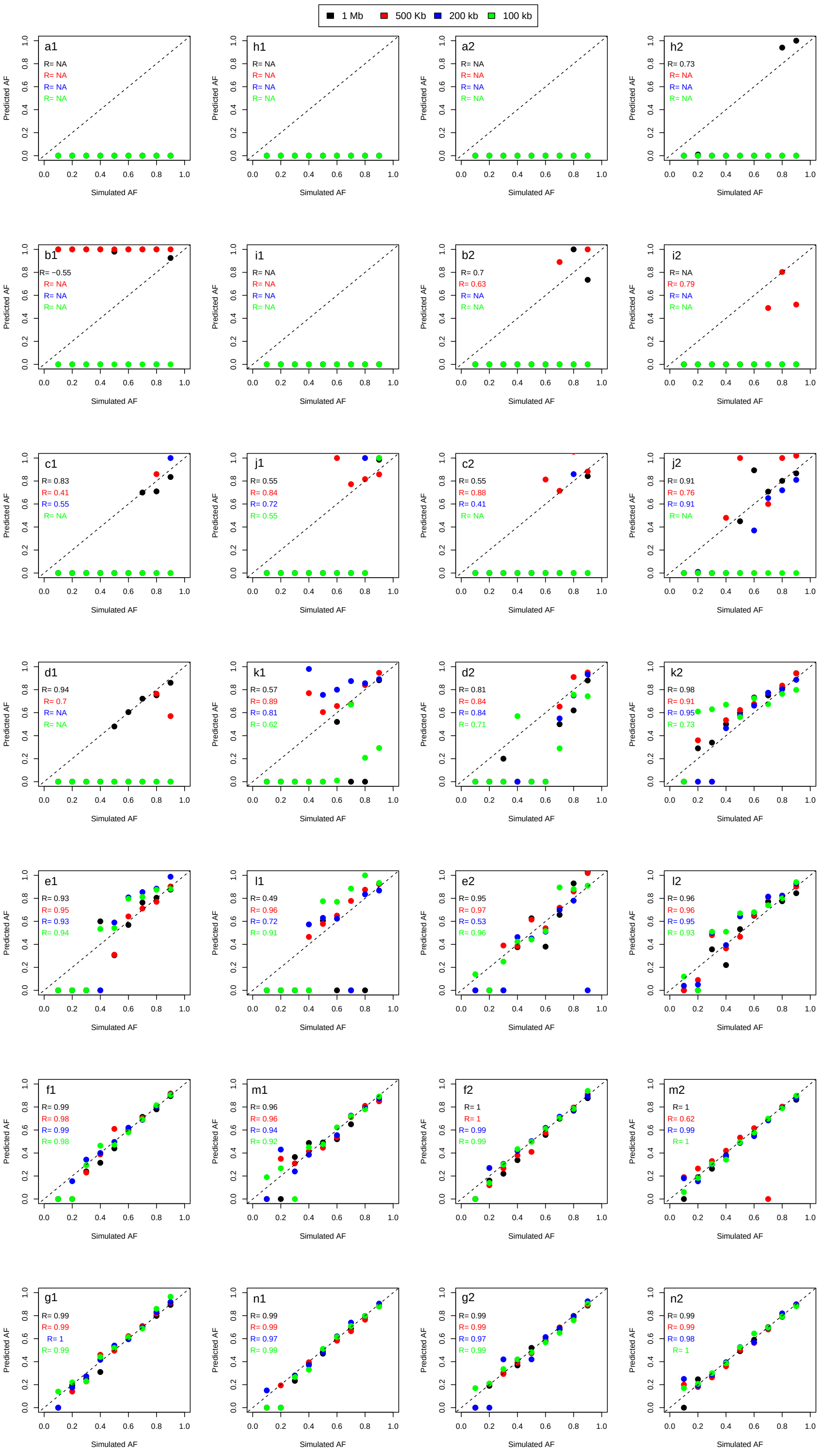

Figure 53: Allelic fraction prediction results for for duplications and deletions of 20 Mb. In panels are reported the correlations between the allelic fractions simulated by Xome-blender and the allelic fraction predicted by our novel probabilistic method FractionPred for different window sizes (100 Kb, 200 Kb, 500 Kb and 1 Mb). Panels (a1-n1) report the results for single sample analysis, (a1-g1) for deletions and (h1-n1) for duplications, while panels (a2-n2) report the results for paired sample analysis, (a2-g2) for deletions and (h2-n2) for duplications. Each panel row report prediction results for different number of simulated reads: N=10.000 (a1,h1,a2,h2), N=20.000 (b1,i1,b2,i2), N=50.000 (c1,j1,c2,j2), N=100.000 (d1,k1,d2,k2), N=200.000 (e1,l1,e2,l2), N=500.000 (f1,m1,f2,m2), N=1.00.000 (g1,m1,g2,m2). Colors reported in legend represent the four window sizes used in the analyses.

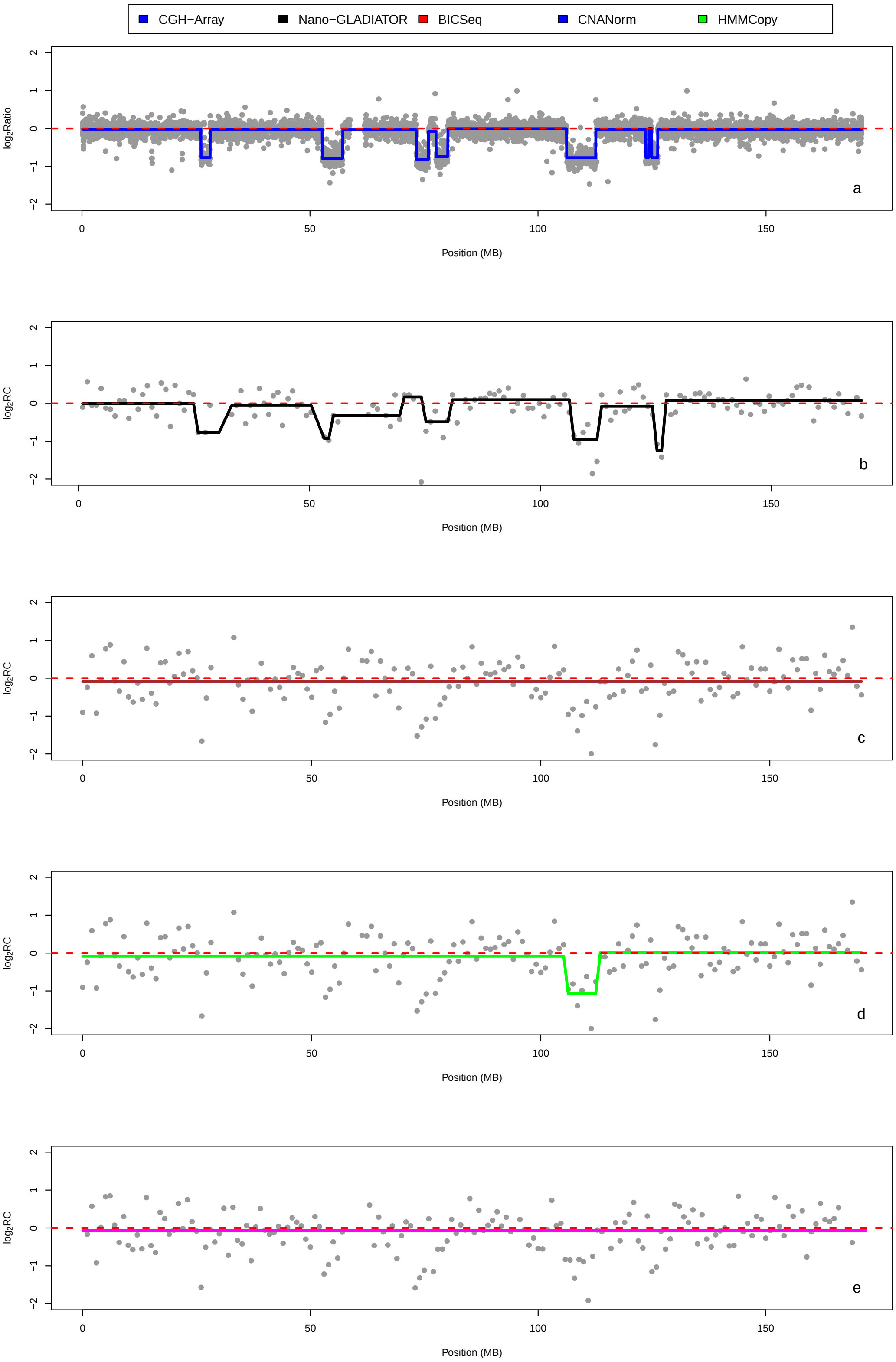

Figure 54: Genomic profiles for chromosome 6 of Sample 1. Panels report the log_2Ratio and log_2RC_{Norm} profiles obtained by CGH-array and nanopore data for chromosome 6 of sample 1. Panel (a) show the CGH-array log_2Ratio segmented by CBS. Panels (b-e) show the log_2RC_{Norm} from nanopore data analyzed by Nano-GLADIATOR (b), BICSeq (c), CNANorm (d) and HMMCopy (e).