Bioinformatics, 32(7), 2016, 984–992
doi: 10.1093/bioinformatics/btv751
Advance Access Publication Date: 6 January 2016
Original Paper

OXFORD

Genome analysis

SV-Bay: structural variant detection in cancer genomes using a Bayesian approach with correction for GC-content and read mappability

Daria Iakovishina[1], Isabelle Janoueix-Lerosey[2,3], Emmanuel Barillot[2,4,5,6], Mireille Regnier[1] and Valentina Boeva[2,4,5,6]*

[1]INRIA Projet AMIB, Ecole Polytechnique, Palaiseau, France, [2]Institut Curie, Centre De Recherche, Paris, [3]Inserm, U830, Department Genetics and Biology of Cancers, Paris, France, [4]Inserm, Department of Bioinformatics, Biostatistics, Epidemiology and Computational Systems Biology of Cancer, U900, Paris, France, [5]Mines ParisTech, Centre for Computational Biology, Fontainebleau, France and [6]PSL Research University, Paris, France

*To whom correspondence should be addressed.
Associate Editor: Benjamin Raphael

Received on 19 March 2015; revised on 10 November 2015; accepted on 20 December 2016

Abstract

Motivation: Whole genome sequencing of paired-end reads can be applied to characterize the landscape of large somatic rearrangements of cancer genomes. Several methods for detecting structural variants with whole genome sequencing data have been developed. So far, none of these methods has combined information about abnormally mapped read pairs connecting rearranged regions and associated global copy number changes automatically inferred from the same sequencing data file. Our aim was to create a computational method that could use both types of information, i.e. normal and abnormal reads, and demonstrate that by doing so we can highly improve both sensitivity and specificity rates of structural variant prediction.

Results: We developed a computational method, SV-Bay, to detect structural variants from whole genome sequencing mate-pair or paired-end data using a probabilistic Bayesian approach. This approach takes into account depth of coverage by normal reads and abnormalities in read pair mappings. To estimate the model likelihood, SV-Bay considers GC-content and read mappability of the genome, thus making important corrections to the expected read count. For the detection of somatic variants, SV-Bay makes use of a matched normal sample when it is available. We validated SV-Bay on simulated datasets and an experimental mate-pair dataset for the CLB-GA neuroblastoma cell line. The comparison of SV-Bay with several other methods for structural variant detection demonstrated that SV-Bay has better prediction accuracy both in terms of sensitivity and false-positive detection rate.

Availability and implementation: https://github.com/InstitutCurie/SV-Bay

Contact: valentina.boeva@inserm.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Whole genome sequencing (WGS) has become routine for detection of both small and large somatic mutations, i.e. point mutations, small indels and structural variants (SVs) in cancer genomes. Paired-end sequencing of mate-pair libraries is often employed when the aim of the study is the detection of large SVs, i.e. variants of greater length than the read length (Boeva *et al.*, 2013; Pleasance *et al.*, 2010; Stephens *et al.*, 2009, 2011). A long insert size of mate-pair libraries (usually 3–4 kb) allows for high physical coverage of SV junctions.

Each type of large SVs (translocation, duplication, deletion, inversion, etc.) corresponds to a particular paired-end mapping signature (PEM signature) (Zeitouni *et al.*, 2010). As such, deletions are characterized by a larger than expected distance between mapped paired reads (insert size), while insertions have an insert size shorter than expected (Supplementary Fig. S1). Additionally, SVs often result in a change of copy number status around the breakpoint junction, which is reflected in changes in read depth of coverage (DOC). For instance, deleted regions have a relatively low DOC, whereas duplicated regions are characterized by high DOC (Boeva *et al.*, 2011). Thus, differences in DOC and abnormal positioning of mapped reads often indicate the same genomic abnormality (e.g., a deletion or a tandem duplication). However, there has been no effort made to combine these two types of information into one unified computational approach.

Most of the current SV detection approaches can be classified into three categories: methods based on (i) PEM signatures, (ii) DOC and (iii) split read mappings (Medvedev *et al.*, 2009). Each of these approaches has its limits in terms of the type and size of SVs that it is able to detect.

PEM-based algorithms can be grouped into two categories: those based on read clustering, and those based on fragment length distribution. The former identify discordant PEMs as PEMs with unexpected orientation or insert size, cluster them and apply statistical tests to validate candidate clusters (Hormozdiari *et al.*, 2009, 2010; Korbel *et al.*, 2009; Sindi *et al.*, 2009; Zeitouni *et al.*, 2010), whereas the latter compare the observed insert-size distribution of all read pairs in a given window versus the expected distribution. Windows with a significant proportion of read pairs having unexpected insert-sizes are annotated as containing SVs (Lee *et al.*, 2009). In some cases, the same package, e.g. BreakDancer (Chen *et al.*, 2009), provides two complementary methods for SV detection: clustering-based (BreakDancerMax) and distribution-based (BreakDancerMini) to detect large and small size SVs, respectively.

DOC-based methods detect regions of gain and loss in the genome using DOC normalized for GC-content bias (Boeva *et al.*, 2011, 2012; Yoon *et al.*, 2009). A deviation from the expected DOC suggests putative gain or loss of genomic material. DOC-based methods do not provide information about the adjacency of DNA regions involved in copy number changes. Thus, such methods are not able to indicate the type of SV (e.g. tandem duplication, fragment reinsertion, translocation) causing genomic loss or gain. Additionally, the resolution of such methods rather is low for low DOC datasets: a 30× coverage dataset allows approximately a resolution of 1 kb for rearrangement breakpoints.

Split-read-based methods use partial read alignments for SV detection (Schröder *et al.*, 2014; Trappe *et al.*, 2014; Wang *et al.*, 2011). Although such methods may be efficient for data with high read coverage, they may fail to identify SVs with breakpoints located in repetitive elements of the genome. Ideally, these types of approach should be combined with paired-end signatures; this idea was implemented in SVMerge (Wong *et al.*, 2010), PRISM (Jiang *et al.*, 2012), Meerkat (Yang *et al.*, 2013), SMUFIN (Moncunill *et al.*, 2014) and DELLY (Rausch *et al.*, 2012).

Combining information about discordant PEMs with changes in DOC is a promising solution for the SV detection problem. Probabilistic models integrating both the DOC signal and PEM signatures provide higher specificity at equal or greater sensitivity than tools that simply use paired-end signatures (Escaramís *et al.*, 2013; Handsaker *et al.*, 2011; Layer *et al.*, 2014; Oesper *et al.*, 2012; Qi and Zhao, 2011; Sindi *et al.*, 2012). However, most of these methods do not take into account certain important parameters that affect read count for both normal and abnormal mappings: GC-content and read mappability. Another general drawback of the majority of these methods is their lack of ability to detect all possible types of SV that can be present in cancer data including co-amplifications, tandem duplications with inversions, linking insertions, etc.

Here, we propose a Bayesian framework for SV detection using paired-end or mate-pair libraries, implemented as the software SV-Bay. In this framework, we combine both PEM signatures and information about changes in DOC in regions flanking each candidate rearrangement. Our method takes into account GC-content and mappability. The use of a Bayesian framework based on both PEM and DOC information allows us to significantly decrease the level of false positive predictions while retaining high sensitivity. Additionally, SV-Bay infers 15 different types of structural variant from the detected novel genomic adjacencies (Supplementary Fig. S1).

2 Methods

SV-Bay uses a Bayesian formulation to assess the likelihood of an SV. To this end, SV-Bay combines information about abnormal PEM signature and DOC. SV-Bay separately analyzes whether each cluster of abnormally mapped paired reads is a part of a true rearrangement (novel genomic adjacency) and then combines closely located genomic adjacencies into complex SVs. The method includes several pre-processing and post-processing steps (Fig. 1). The Bayesian model is applied at the level of detection of novel genomic adjacencies (explained in detail in Section 2.5). Sections 2.1–2.4 provide elements necessary to the understanding of the constructed Bayesian model. The procedure for selecting the likeliest breakpoint

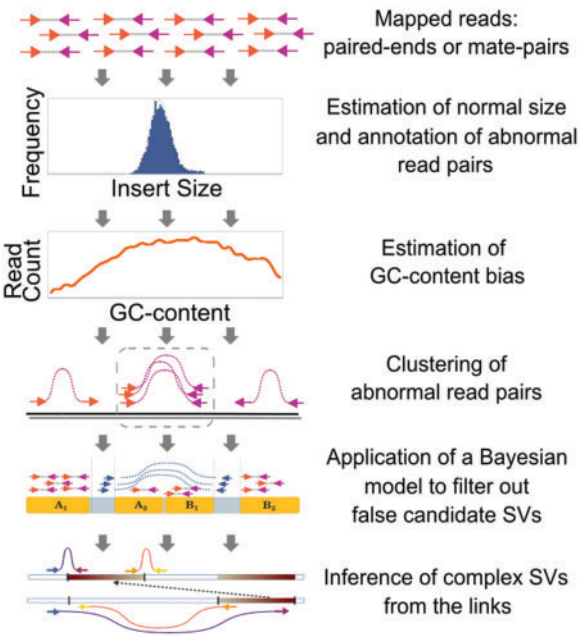

Fig. 1. Outline of SV-Bay. SV-Bay evaluates the insert size distribution and identifies the default orientation of reads in pairs. It applies this distribution to annotate read pairs as normal and abnormal. The latter may correspond to novel SVs in the genome. By calculating the density of normal reads along the genome (in copy neutral regions), SV-Bay evaluates the GC-content dependency bias that will be used in the Bayesian model

position without performing a local read assembly, the order in which candidate genomic adjacencies are tested for being false positives, and the way we use a normal control sample are explained in Supplementary Methods. Section 2.6 describes how complex rearrangements are inferred using closely located novel genomic adjacencies.

2.1 Definition of normal insert size of paired reads and annotation of normal and abnormal read pairs

We define normal paired reads as reads with the expected orientation and insert size. The expected orientation is defined by the technology used to generate and sequence the DNA fragment library: inwards for paired-end libraries, outwards for Illumina mate-pair libraries; this orientation is automatically detected by SV-Bay. To get the shape of the fragment length distribution, we analyze insert size (the distance between the leftmost position of the left read and the rightmost position of the right read) of read pairs with expected read orientation. By default, SV-Bay annotates as 'normal pairs' read pairs with both reads mapped to the same chromosome with insert size within the 99% of insert size distribution with the expected read orientation. The remaining read pairs are annotated as 'abnormal'. We also define μ (the median insert size) and σ (standard deviation of insert size). We denote the minimal and maximal insert sizes of a normal read pair as $l_{\min}$ and $l_{\max}$, respectively. We also discard PCR duplicates as read pairs with identical up to k bp start and end positions (k specified by the user).

2.2 Estimation of the expected number of read pairs per position and genomic region

For calculation of likelihood probabilities used in the Bayesian model, we need to estimate the expected number of reads per region given the GC-content and mappability. SV-Bay accepts read alignment BAM files generated by BWA option 'aln' (Li and Durbin, 2009). For each read pair, BWA provides information about the mapping quality for a pair and the uniqueness of mapping of each read within the pair.

2.2.1. Calculation of the expected number of read pairs per position and region for normal reads

For normal reads, we assign a mappability of 1 to a given position i ($M_i = 1$) if the k-mer starting at position i in the reference genome is unique up to m mismatches in the reference genome (m defined by the user, default value $m = 2$). Otherwise, we assign mappability of 0 ($M_i = 0$). The number of mapped reads per position and per region may highly depend on GC-content (Benjamini and Speed, 2012). Taking this into account, we can define the expected number of reads per genomic region [a, b] ($E_{[a,b]}$) as follows (see Supplementary Methods):

$$E_{[a,b],\alpha} = \sum_{i=a}^{b} M_i \cdot \alpha \cdot \lambda(\mathrm{GC}(i)), \tag{1}$$

where α is the number of copies of region [a, b] ($\alpha = 2$ for autosomal positions in diploid genomes), $GC(i)$ the GC-content for position i of the reference genome (i.e. the fraction of C and G nucleotides in a window of size μ starting at i) and $\lambda(x)$ the average number of read pairs starting per mappable position with a given GC-content x in a region present in one copy. The GC-content of a given position is evaluated in a window of length μ following the observation of Benjamini and Speed (2012) that the best window size for GC-content bias correction corresponds to the average fragment length.

Parameters $\lambda(x)$ can be empirically evaluated as follows:

$$\hat{\lambda}(X) = \frac{\sum_{i=1}^{L} M_i \cdot O_i/\alpha_i \cdot I\{\mathrm{GC}(i) = x\}}{\sum_{i=1}^{L} M_i \cdot I\{\mathrm{GC}(i) = x\}}, \tag{2}$$

where L is the genome length, α_i the number of genomic copies for position i, O_i the observed number of normal read pairs mapped to position i and $I\{GC(i) = x\}$ the indicator that GC-content at position i is equal to x. In practice, to evaluate $\lambda(x)$, we do not consider all genomic positions. Instead, we use a large enough random subset so that we can evaluate $\lambda(x)$ up to the third decimal place (Supplementary Methods). For the tumor genome, we select positions coming from copy neutral regions, i.e. regions with copy number α_i equal to the main ploidy of the tumor genome. The selection of these regions is based on the output of Control-FREEC (Boeva *et al.*, 2011, 2012), included in the SV-Bay pipeline.

To account for possibly mismapped reads in homozygous deletion regions, we modify formula (1) for $\alpha = 0$:

$$E_{[a,b],\alpha=0} = (b - a) \cdot N_{\mathrm{abnormal}}/L, \tag{3}$$

where L is the genome length, and N_{abnormal} the total number of abnormal read pairs, which approximates the number of incorrectly mapped read pairs in a given experiment.

2.2.2. Calculation of the expected number of read pairs per position and breakpoint for abnormal reads

To calculate the expected number of read pairs per position for abnormal reads, we make some adjustments to formula (1). Here, we consider only read pairs with unique mapping of both reads. The mappability value per position is now defined as

$$\overline{\overline{M_i}} = M_i \cdot \left(\sum\nolimits_{i=i+\mu-c}^{i+\mu+c} M_i\right)/(2c), \tag{4}$$

where $[\mu - c, \mu + c]$ is the region in which we expect to map the right-most mate of the left-most read in a pair; c is defined as $c = \sqrt{3}\sigma$ (Supplementary Fig. S2 and Supplementary Methods). Using the redefined mappability $\overline{\overline{M_i}}$, we reevaluate the expected number of read pairs per position $\bar{\bar{\lambda}}(x)$:

$$\hat{\bar{\bar{\lambda}}}(x) = \frac{\sum_{i=1}^{L} \overline{\overline{M_i}} \cdot \overline{\overline{O_i}}/\alpha_i \cdot I\{\mathrm{GC}(i) = x\}}{\sum_{i=1}^{L} \overline{\overline{M_i}} \cdot I\{\mathrm{GC}(i) = x\}}, \tag{5}$$

where $\overline{\overline{O_i}}$ is the observed number of read pairs mapped to position i with the left-most read such that both reads in the pair are uniquely mappable.

For a breakpoint junction connecting chromosomes A and B, we can now evaluate the expected number of abnormal fragments spanning breakpoints $x_{\mathrm{A}}^{\mathrm{break}}$ and $x_{\mathrm{B}}^{\mathrm{break}}$ on chromosomes A and B, respectively. Without loss of generality, we assume that the junction connects a region upstream to $x_{\mathrm{A}}^{\mathrm{break}}$ to a region downstream of $x_{\mathrm{B}}^{\mathrm{break}}$. Then, the expected number of abnormal fragments spanning the breakpoints $E\left(x_{\mathrm{A}}^{\mathrm{break},-}, x_{\mathrm{B}}^{\mathrm{break},+}\right)$ can be calculated as:

$$\begin{aligned} E_{x_A^{\mathrm{break},-}, x_B^{\mathrm{break},+}, \gamma>0} &= \sum\nolimits_{i=x_A^{\mathrm{break}}-l_{\max}+r}^{x_A^{\mathrm{break}}-r} \overline{\overline{M_i}}(x_A^{\mathrm{break},-}, x_B^{\mathrm{break},+}) \cdot \gamma \cdot \\ &\cdot \bar{\bar{\lambda}}(GC(i, x_A^{\mathrm{break},-}, x_B^{\mathrm{break},+})) \cdot p(\text{Insert Size} \geq x_A^{\mathrm{break}} - i + r), \end{aligned} \tag{6}$$

where $\overline{\overline{M_i}}(x_{\mathrm{A}}^{\mathrm{break},-}, x_{\mathrm{B}}^{\mathrm{break},+})$ is calculated similarly to $\overline{\overline{M_i}}$ and $GC(i, x_{\mathrm{A}}^{\mathrm{break},-}, x_{\mathrm{B}}^{\mathrm{break},+})$ similarly to $GC(i)$ with the exception that instead of the continuous genomic region starting at i, we use stitched

regions upstream x_A^{break} and downstream x_B^{break}. Here, γ is the number of alleles involved in the SV, and r the read length.

Similarly to (3), we approximate the expected number of read pairs located at a distance less than l_{max} to a breakpoint due to misalignment or artefacts in library preparation as $E_{x_A^{break,-}, x_B^{break,+}, \gamma=0}$:

$$E_{x_A^{break,-}, x_B^{break,+}, \gamma=0} = l_{max} \cdot N_{abnormal}/L. \quad (7)$$

2.3 Clustering of abnormal reads to detect candidate novel genomic adjacencies

In order to detect candidate SVs, we group abnormal read pairs in clusters potentially corresponding to simple SVs or novel genomic adjacencies. Read pairs corresponding to the same novel genomic adjacency have similar insert size and identical orientation. Clusters of such read pairs are called links. We cluster read pairs in a way that all read pairs corresponding to a genomic adjacency of any type (inversion, deletion, inverted duplication, etc.) are clustered together in one link. The clustering is based on two parameters: the difference in insert sizes, I, and the difference in the read coordinates D, where read coordinates are characterized with respect to the midpoint between read starts. Details of the clustering procedure are provided in Supplementary Methods.

2.4 Definition of flanking regions

Each unbalanced SV will change the copy number (and DOC) in regions flanking the SV breakpoints. To include the DOC in these regions in the Bayesian model, we formally define four flanking regions for each link.

Any novel genomic adjacency has two breakpoints in the reference genome, with the exception of small insertions and mirror duplications (Supplementary Fig. S1). Without loss of generality, we further assume that there are two breakpoints per link. We define two flanking regions for each breakpoint: one upstream and one downstream of the breakpoint. We denote these (A_1, A_2) and (B_1, B_2) for the leftmost and rightmost breakpoints, respectively (Supplementary Fig. S3). Each flanking region should not overlap any SV that could affect the number of normal read pairs within this region and should not include the interval around the breakpoint itself, where we expect to observe a gap in normal DOC (Sindi *et al.*, 2012). We call these small regions around the two breakpoints 'safety intervals': S_x and S_y (see Supplementary Methods for formal definitions).

The flanking regions A_1 and B_1 are defined as the largest regions upstream of the safety intervals S_x and S_y that do not contain any safety regions of other links. Similarly, the flanking regions A_2 and B_2 are defined as the largest regions downstream of the safety intervals S_x and S_y that do not contain any other safety intervals.

Closely located links may have overlapping safety intervals. In this case, we keep the corresponding flanking regions empty. Also, we do not allow flanking regions to span centromeric regions and long unassembled poly-N regions.

2.5 Bayesian model

We aim to determine the likelihood of a given link to be a part of a real SV and estimate the number of alleles (γ) involved in the given genomic adjacency. To this end, for each link representing a candidate genomic adjacency, we calculate the probability of a model $M^{\alpha_1,\alpha_2,\beta_1,\beta_2,\gamma}$, where $\alpha_1, \alpha_2, \beta_1, \beta_2$ are copy numbers in flanking regions A_1, A_2, B_1, B_2. The parameters of the model have to satisfy to the following constraints:

$$\alpha_1 = \alpha_2 \pm \gamma; \beta_1 = \beta_2 \pm \gamma, \quad (8)$$

where the sign before γ depends on the orientation of reads in the corresponding link. In cases where some flanking regions are empty or A_2 and B_1 coincide (e.g. in case of short deletions), the number of parameters in the model is reduced.

The aim of our Bayesian approach is to match observations, i.e. the number of abnormal read pairs in the link and the number of normal pairs in the flanking regions, with a model $M^{\alpha_1,\alpha_2,\beta_1,\beta_2,\gamma}$. According to Bayes' rule the probability of model M' given observed data Δ is:

$$P(M'|\Delta) = \frac{P(\Delta|M')P(M')}{\sum_M P(\Delta|M)P(M)}. \quad (9)$$

Observations Δ are formalized as $\Delta = \{n_{A_1}, n_{A_2}, n_{B_1}, n_{B_2}, n_\Gamma\}$, where $n_{A_1}, n_{A_2}, n_{B_1}, n_{B_2}$ are the number of mapped read pairs in flanking regions A_1, A_2, B_1, B_2, respectively, and n_Γ the number of abnormal read pairs in the link. In the current version of the algorithm, we assume probabilities $P(M^{\gamma>0})$ of every model M where $\gamma > 0$ to be identical. We expect $P(M^{\gamma>0})$ to be much lower than $P(M^{\gamma=0})$, as we suppose that there are much less links corresponding to real SVs than to read mismappings and artefacts in library preparation. To include this intuition, we introduce a user-defined parameter for the expected number of true SVs in the dataset (E_{SV}, default value 1000). Then, the probabilities $P(M^{\gamma>0})$ and $P(M^{\gamma=0})$ are assigned as follows: $P(M^{\gamma>0}) = \min(E_{SV}/N_{links}, 1)$ and $P(M^{\gamma=0}) = 1 - P(M^{\gamma>0})$, where N_{links} is the total number of links.

In the general case (A_1, A_2, B_1, B_2 are not empty and do not overlap), the conditional probability $P(\Delta|M)$ can be factorized:

$$P(\Delta|M^{\alpha_1,\alpha_2,\beta_1,\beta_2,\gamma}) = P(n_{A_1}|\alpha_1) \cdot P(n_{A_2}|\alpha_2) \cdot P(n_{B_1}|\beta_1) \cdot P(n_{B_2}|\beta_2) \cdot P(n_\Gamma|\gamma). \quad (10)$$

To calculate these probabilities, we assume that the read count follows a Poisson distribution with the mean equal to the expected number of reads per region, calculated by formulas (1) and (3) for normal read pairs in flanking regions A_1, A_2, B_1, B_2, and formulas (6) and (7) for abnormal read pairs in the link:

$$P(n_{A_1}|\alpha_1) = \text{Pois}(E_{A_1,\alpha_1}; n_{A_1}), P(n_{A_2}|\alpha_2) = \text{Pois}(E_{A_2,\alpha_2}; n_{A_2}), \quad (11)$$

$$P(n_{B_1}|\beta_1) = \text{Pois}(E_{B_1,\beta_1}; n_{B_1}), P(n_{B_2}|\beta_2) = \text{Pois}(E_{B_2,\beta_2}; n_{B_2}), \quad (12)$$

$$\text{and } P(n_\Gamma|\gamma) = \text{Pois}(E_{x_A^{break}, x_B^{break}, \gamma}; n_\Gamma), \quad (13)$$

where $\text{Pois}(\lambda; k) = \lambda^k e^{-\lambda}/k!$ is the probability density function for the Poisson distribution with mean λ. In (13), we assume that we know the exact breakpoint position. In practice, we check several sets of breakpoints and keep the one providing the highest likelihood (Supplementary Methods). Since the total number of possible models to test is infinite, we limit ourselves to a set of the most plausible models (Supplementary Methods).

At the end of this step, for each link, SV-Bay will detect the most likely model to explain the observed read counts. When the evaluated number of alleles involved in the candidate adjacency is zero, we remove the link as a false positive candidate.

2.6 Combination of novel genomic adjacencies into complex SVs

When we have a list of novel genomic adjacencies validated by the Bayesian approach, we combine them into simple and complex SVs (Supplementary Fig. S1). For each link i, we define regions C_i^A and C_i^B around breakpoints $x_A^{break,i}$ and $x_B^{break,i}$: $C_i^A = [x_A^{break,i} - 2l_{max}, x_A^{break,i} + 2l_{max}]$ and $C_i^B = [x_B^{break,i} - 2l_{max}, x_B^{break,i} + 2l_{max}]$. If C_i^A and

C_i^B do not contain breakpoints of others links, we annotate link i as a simple SV: deletion, insertion, tandem duplication, inverted duplication, unbalanced translocation (Supplementary Fig. S1A). Otherwise, we add all links with breakpoints occurring in C_i^A or C_i^B to the set of links Ω_i. Then, we search for other links with breakpoints located within $2l_{max}$ distance from any breakpoint of any link in Ω_i. When we cannot add more links to Ω_i, we annotate Ω_i as a complex structural variant (Supplementary Fig. S1B): inversion, fragment re-insertion, balanced translocation, etc. The biological relevance of many of the SV types that we have included in SV-Bay has been demonstrated by several cancer studies (inverted duplications (Boeva *et al.*, 2013), amplifications and co-amplifications (Vogelstein and Kinzler, 2002), re-insertions and complex deletions (Yang *et al.*, 2013).

Amplifications of oncogenes (*MYC*, *MYCN*, *ERBB2*, *KRAS*, etc.) is a common phenomenon in cancer (Vogelstein and Kinzler, 2002). Such SVs result in the creation of dozens of copies of a given genomic region (Supplementary Fig. S1B). In SV-Bay, we define the default value for the minimal number of copies per amplification (or co-amplification) as 10. SV-Bay separately infers genomic amplifications and other SVs. In this way, closely located amplifications and other SVs cannot be mistakenly grouped together.

3 Results

We applied SV-Bay to simulated and experimental datasets and were able to demonstrate that it had a higher prediction accuracy than four other published methods: GASVPro (Sindi *et al.*, 2012), Lumpy (Layer *et al.*, 2014) and the two most popular methods for SV detection, BreakDancer (BreakDancerMax) (Chen *et al.*, 2009) and DELLY (Rausch *et al.*, 2012). We have selected GASVPro and Lumpy, since like SV-Bay, these tools take into consideration information about DOC changes in the proximity of candidate breakpoints. However, unlike SV-Bay, they consider only short proximity of breakpoints while SV-Bay calculates read counts in maximally large regions around the breakpoint, which increases the statistical power of its probabilistic approach. In addition, GASVPro, Lumpy and DELLY integrate split-reads into the analysis.

The version of each software and parameters used are summarized in Supplementary Table S1. Lumpy was applied to paired-end data only as it does not provide an option to run on mate-pair datasets.

In the analysis below, we assumed that an SV calling method correctly detected a given SV when (i) predicted breakpoints laid within a certain distance d from the correct breakpoints, and (ii) there was a match in read orientation. Distance d corresponded to the maximal fragment length, i.e. 5179 bp for simulated mate-pair data, 446 bp for simulated paired-end data, and 3542 bp for the experimental mate-pair data.

3.1 SV-bay performance on simulated paired-end and mate-pair datasets

To simulate tumor datasets, we used a combination of TGSim, software we developed to simulate a tumor genome (https://github.com/InstitutCurie/TGSim), and read simulation software PIRS (Hu *et al.*, 2012) (code.google.com/p/pirs/). To create a nucleotide sequence corresponding to a normal control diploid genome, we modified the reference human genome GRCh38/hg38 by adding 3 million heterozygous SNPs, 315 000 small indels and 972 small inversions. To simulate a matched tumor genome, we applied a sequence of genomic rearrangements to the nucleotide sequence of the diploid normal genome using TGSim. TGSim inserted into the normal genome 'standard' SVs (deletions, tandem duplications, insertions of random sequences, inversions, translocations) but also "complex" SVs, which are usually missed or incorrectly assembled by SV calling tools (co-amplifications, tandem duplications with an inversion of the duplicated unit, linking insertions and linking re-insertions, Fig. 2). We created two simulated tumor genomes: a near-diploid genome (T2) and a near-tetraploid genome (T4); the latter contained approximatively 4 copies of each chromosome but had less genomic rearrangements than the near-diploid genome (44 versus 114 SVs corresponding to 62 and 147 novel genomic adjacencies, respectively, Supplementary Tables S2 and S3). For each simulated tumor genome, we performed two read simulation experiments: paired-end (PE) and mate-pair (MP) read simulations denoted T2_PE, T2_MP, T4_PE and T4_MP (Supplementary Table S4). The read simulation

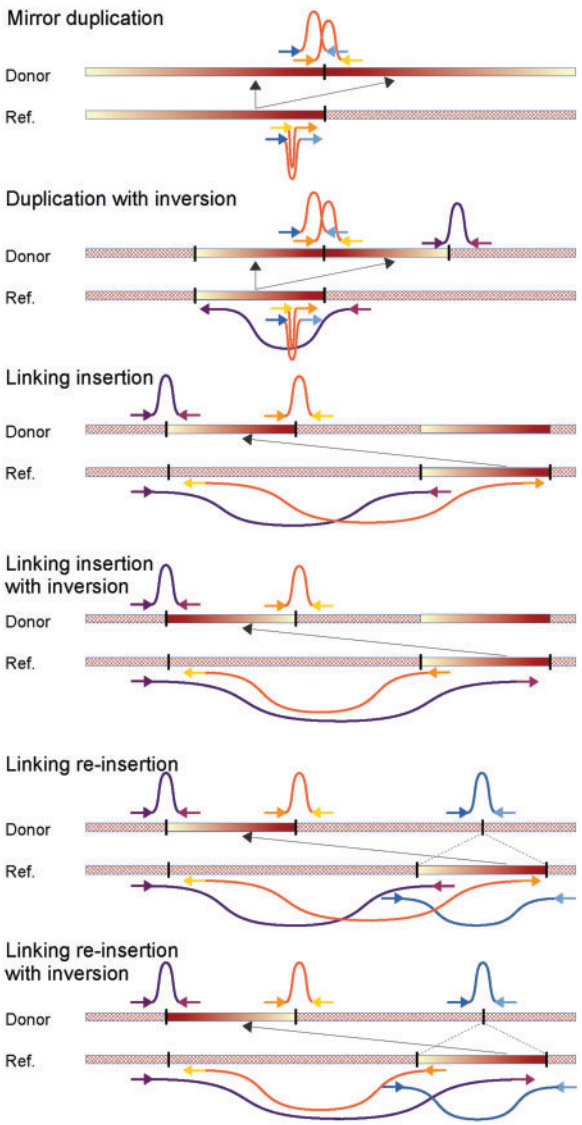

Fig. 2. Visualization of several complex types of SV that can be recognized by SV-Bay: mirror duplication, tandem duplication with inversion, linking inserting, linking insertion with inversion, linking re-insertion, and linking re-insertion with inversion. In addition to the represented SVs, SV-Bay is able to recognize balanced and unbalanced translocations (with and without inversion), amplifications and co-amplifications, simple inversions, direct tandem duplications, and so on. (Supplementary Fig. S1)

algorithm allowed us to take into account GC-content bias and experimental read error profile along the reads (Hu *et al.*, 2012). The corresponding parameters were taken from experimental neuroblastoma WGS datasets (Boeva *et al.*, 2013). We also generated matched normal datasets (Supplementary Table S4).

In the simulations, we expected that the average number of abnormal read pairs required to confirm each novel genomic adjacency (physical coverage) would be higher for mate-pair than for paired-end data (Supplementary Table S4). Thus, despite the fact that the mate-pair library contained from 5 to 10 times less reads than the paired-end one, we observed that in all cases but one all methods tested were able to identify more correct SVs in the mate-pair dataset (Fig. 3, Supplementary Tables S2 and S3). This observation supports the common choice of mate-pairs for annotation of structural variants in tumor genomes, even though creating a mate-pair library requires a more elaborate protocol.

SV-Bay achieved maximal recall on all the four simulated datasets and detected 125/147, 129/147, 34/62 and 59/62 correct novel genomic adjacencies for T2_PE, T2_MP, T4_PE and T4_ME, respectively (Fig. 3, Supplementary Tables S2 and S3). Of note, to construct precision/recall curves, we run each method only once with the most relaxed parameters; then, we used a threshold on the quality of predicted SVs provided by each tool. We confirm that the choice of user-defined value for the expected number of true SVs in the dataset (E_{SV}) plays a limited role in the prediction accuracy by SV-Bay (Supplementary Fig. S4). However, the user should consider to change the value of this parameter to higher values in case of interest in germline events or when analyzing chromothripsis cases.

In many cases, the recall provided by BreakDancer and DELLY on simulated datasets was almost as high as the recall provided by SV-Bay (Supplementary Tables S2 and S3). However, both BreakDancer and DELLY gave a large number of false-positive predictions (Fig. 3). Overall, on simulated data SV-Bay demonstrated better prediction accuracy than other tools both in terms of detection precision and recall.

SB-Bay can recognize and infer from novel genomic adjacencies more types of complex SVs than other methods (Fig. 2). For instance, using mate-pair data for the T4 genome all tools we tested were able to identify novel genomic adjacencies corresponding to a co-amplification (50 times) of several regions on chromosome 13. However, only our method was able to group four different read clusters related to this co-amplification into one complex SV. SV-Bay was also the only method to group together clusters corresponding to linking insertions and re-insertions we added to the simulated tumor genome.

3.2 SV-Bay performance on a neuroblastoma mate-pair dataset

To investigate the performance of SV-Bay on an experimental dataset, we selected a mate-pair dataset from a neuroblastoma diploid cell line CLB-GA recently sequenced using a mate-pair protocol together with a corresponding normal control (Boeva *et al.*, 2013). For this dataset, we recently performed SNP6 array experiment to characterize genotype and copy number alterations independently from WGS data. Also, for this cancer cell line, we had a set of 11 SVs validated by PCR and Sanger sequencing (Supplementary Table S5). Most of the 11 validated SVs correspond each to two breakpoints in the SNP array copy number profile. The following SVs correspond to only one breakpoint: (i) the SV between the ALK gene (chromosome 2p, 29 Mb) and a repetitive peri-telomeric sequence; the exact position of the latter could not be defined; (ii) the SV between chromosomes 12q and 20q, as it corresponds to a more complex SV on chromosome 12q (Fig. 4), and (iii) the inverted duplication at chromosome 5q. Both validated SVs and 27 breakpoints obtained by the analysis of Affymetrix SNP6.0 datasets were further used to check the performance of SV-Bay and compare it with BreakDancer, DELLY and GASVPro. We excluded Lumpy from this test as it is not able to analyze mate-pair data.

The total number of SVs predicted on this dataset by DELLY, GASVPro and BreakDancer was significantly higher than the number of SVs predicted by SV-Bay (62822, 1648 and 5543 versus 765).

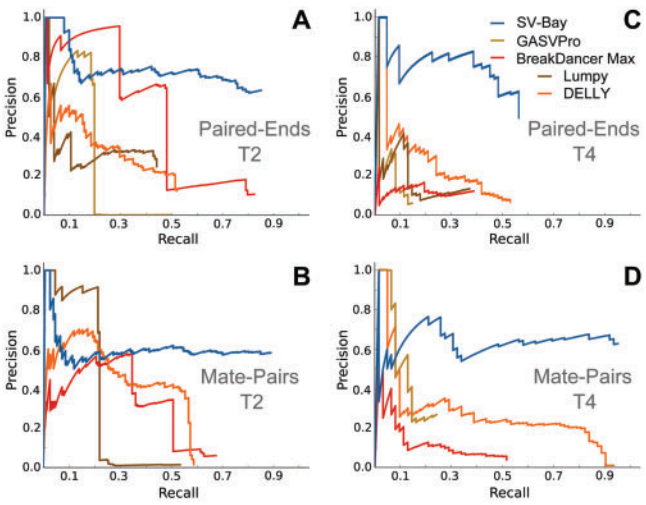

Fig. 3. Prediction accuracy on simulated data for BreakDancer, GASVPro, Lumpy, DELLY and SV-Bay. Precision/recall curves for simulated paired-end (PE) data, near-diploid genome T2 (**A**); mate-pair (MP) data, genome T2 (**B**); PE data, near-tetraploid genome T4 (**C**); MP data, genome T4 (**D**). The results for Lumpy are shown only for PE data. For all tools, we kept only SVs with average insert size larger than 100 bp and 500 bp for PE and MP data respectively. The total number of true genomic adjacencies: 147 for T2 and 62 for T4. The total number of predictions: BreakDancer 1177 (T2_PE), 1711 (T2_MP), 197 (T4_PE) and 1787 (T4_MP); GASVPro 49069 (T2_PE), 6268 (T2_MP), 153 (T4_PE) and 56 (T4_MP), Lumpy 272 (T2_PE) and 171 (T4_PE), DELLY 636 (T2_PE), 6289 (T2_MP), 479 (T4_PE) and 16675 (T4_MP), and SV-Bay 204 (T2_PE), 231 (T2_MP), 66 (T4_PE) and 85 (T4_MP). See Supplementary Tables S2 and S3 for more detail

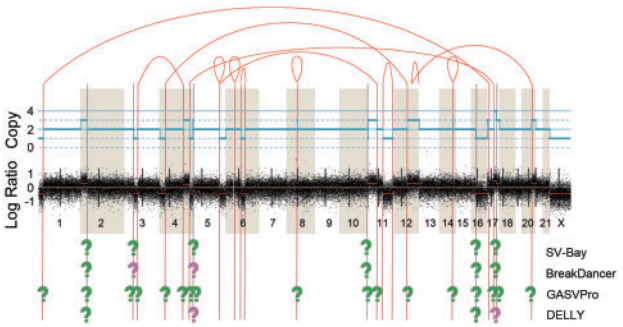

Fig. 4. Prediction sensitivity on experimental data (neuroblastoma cell line CLB-GA mate-pair dataset). The structure of SVs identified by SVBay (red links) explains well the change points detected in the Affymetrix SNP6.0 copy number profile (black profile, short vertical bars indicate centromeres); absolute copy numbers identified by GAP (Popova et al., 2009) are shown in blue. Change points in the copy number profile are shown with long vertical bars (explained with SVs predicted by SV-Bay: red, unexplained: grey). Green question marks indicate copy number changes unexplained by each tool tested. Purple question marks correspond to the cases where detected SVs are likely to correspond to false positive predictions

Only 88 SVs were predicted by all the methods (Supplementary Fig. S5). First, we correlated these SVs with breakpoints in the copy number profile calculated using an Affymetrix SNP6.0 array for CLB-GA using the genotyping software GAP (Popova *et al.*, 2009). GAP identified 27 breakpoints in the genome of this neuroblastoma cell line. Among them, 21 were explained by SVs predicted by SV-Bay (Fig. 4, Supplementary Table S6). The same 21 breakpoints were also explained by SVs discovered by BreakDancer and DELLY. Among the 62 thousand DELLY's predictions, there was also a translocation explaining the breakpoints on chromosomes 3 and 10 missed by other tools; it was tagged 'LowQual' and included 2 read pairs. In addition, DELLY predicted two SVs that could potentially explain the presence of breakpoints on chromosomes 5 and 17; these SVs were not detected by other tools. However, unless validated by PCR these SVs do not seem to be accurate: they are confirmed only by 2 read pairs, have a 'LowQual' tag in the output and their type is unbalanced translocation where the second ends are located in chromosomes 17q25 (79 Mb) and 18q12 (27 Mb). Both these regions do not show any copy number change point according to the SNP array analysis. Thus, the corresponding breakpoints are marked by purple question marks (possible false positives) for DELLY in Figure 4. GASVPro was able to identify SVs corresponding to only eight breakpoints in the SNP array copy number profile (Fig. 4, Supplementary Table S6).

Among the 11 experimentally validated SVs, 10 were successfully detected by SV-Bay, DELLY and BreakDancer (Supplementary Table S5). These three methods missed only one translocation between the *ALK* gene and a repetitive region in a telomere as the input data contained only one read-pair uniquely mapped to the corresponding peri-telomeric repetitive region. In the future, we plan to improve our approach by taking into account non-uniquely mapped reads. This is expected to improve the sensitivity of predictions. The GASVPro method was able to identify only five out of 11 validated SVs.

Although BreakDancer, DELLY and SV-Bay had equally good sensitivity on this experimental data, SV-Bay has a much better positive predictive value (or precision): the total number of predictions in the SV-Bay output was 8 times less than in the output of BreakDancer and 85 times less than in the output of DELLY. This is explained by the use of the Bayesian probabilistic model in SV-Bay in addition to the clustering of the abnormal reads employed in both methods. Among SV-Bay SV calls, 173 were not detected by any other method. The majority of them, 93%, corresponds to small size events (up to 5 kb), may represent false-positive discoveries and can be filtered out using a threshold on the SV size.

Even without using split reads, SV-Bay significantly outperformed BreakDancer in the identification of the exact breakpoint position (Supplementary Table S5): the average distance between validated and predicted breakpoints was 654 bp versus 1906 bp for SV-Bay and BreakDancer, respectively (Wilcoxon Rank Sum two-sided P-value < 0.01). However, GASVPro and DELLY were able to provide a better breakpoint resolution by taking into account split reads (average distance between validated and predicted breakpoints 314 and 373 bp, P-value for comparison with SV-Bay 0.2 and 0.01).

3.3 Basic features of SV-bay and execution time

The comparison of SV-Bay with other SV calling methods demonstrated clear advantages of SV-Bay: SV-Bay outputs more true SVs with lower false positive rate and is able to group links into more complex SVs such as fragment insertions or amplifications. The major differences between SV-Bay, BreakDancer, GASVPro, Lumpy and DELLY are summarized in Table 1. Unlike Lumpy, SV-Bay accepts both paired-end and mate-pair data. SV-Bay is the only method that corrects expected number of read pairs per link for GC-content and mappability, which are factors highly affecting read depth at a given region (Boeva *et al.*, 2011). Also, similarly to Lumpy and BreakDancer, SV-Bay can use BAM files generated from constitutive DNA in order to filter out read alignment artefacts and germline SVs.

SV-Bay is written in Python with the possibility to parallelize the analysis for different chromosomes. However, even without parallelization, SV-Bay demonstrated a very reasonable execution time (Table 2). BreakDancer and DELLY were the fastest tools among the five. For both paired-end and mate-pair simulated datasets, SV-Bay showed the third best execution time: less than 2–4 h for mate-pair and paired-end data, respectively. GASVPro took more than 12 days to analyze the mate-pair dataset. The reason for this may be the long insert size of mate-pair data (more than 4 kb in our case) and thus the range of all possible breakpoint positions per SV was extremely large and required a significant amount of time to be analyzed. SV-Bay also attempts to predict the most likely breakpoint position based on the data for each SV. However, in the case of large intervals in which the breakpoint can be possibly located, SV-Bay limits its analysis to only 10 equally spaced positions within the interval. Although this limits the breakpoint detection accuracy, it significantly speeds up the execution time for mate-pair libraries.

4 Discussion

We have proposed a new method SV-Bay for the detection of large SVs in cancer genomes. SV-Bay is based on a Bayesian probabilistic model. This allows it to be both sensitive and selective, discarding

Table 1. Outline of features of SV-Bay and other SV calling methods

	SV-Bay	BreakDancer	GASVPro	Lumpy	DELLY
Uses DOC information	+	−	+	±[a]	−
Uses split reads	−	−	+	+	+
Uses read mappability	+	−	±[b]	−	−
Uses GC-content	+	−	−	−	−
Detects complex SVs[c]	+	−	−	−	−
Uses normal controls	+	+	−	+	+
Processes PE libraries	+	+	+	+	+
Processes MP libraries	+	+	+	−	+

Abbreviations: paired-ends (PE); mate-pairs (MP).

[a]Lumpy removes regions with extremely high read coverage.

[b]GASVPro uses read mappability information only to estimate the number of abnormal read pairs spanning the breakpoint position, it does not use it to correct DOC in flanking regions.

[c]Complex SVs include co-amplifications, linking insertions, tandem duplications with inversion, etc.

Table 2. Execution time on simulated datasets

	SV-Bay	BreakDancer	GASVPro	Lumpy	DELLY
Mate-pair library	1 h 55 m	32 m	298 h 47 m[a]	N/A	45m
Paired-end library	3 h 58 m	19 m	4 h 39 m	4h02	1h44

[a]The extremely long execution time of GASVPro on mate-pair data is explained by the use of split reads to refine the breakpoint position. In mate-pair data, the range where the breakpoint can be located according to abnormal read mappings can be extremely large; testing all possibilities requires many hours.

many artefact clusters of mismapped read pairs. Indeed, in comparison with other methods, SV-Bay demonstrated a noticeably better SV detection accuracy both for simulated and experimental datasets. SV-Bay not only detects novel genomic adjacencies but also, where possible, groups them into more complex SVs such as co-amplifications, linking insertions, tandem duplications with inversion, etc. Overall, SV-Bay allows the user to skip such data post-processing steps like filtering out links with low number of fragments that do not correspond to copy number changes, filtering out events present both in the tumor and the matched normal control (artefacts and germline SVs), and performing manual inference of complex SV from the detected genomic adjacencies.

SV-Bay does not use split reads to improve the resolution of predicted breakpoints. There are two main reasons for this. First, the read coverage on breakpoints can be sufficiently high only for paired-end libraries, whereas we intended our method to also be applicable to mate-pair data. Second, structural variants in cancer often occur in low mappability repetitive regions or regions that have partial homology; additionally, there can be insertions of one of several genomic shards between two regions connected by a SV (Boeva *et al.*, 2013). These incidents reduce the capacity of read mappers to align correctly reads coming from SV junctions.

Like other methods, SV-Bay is tolerant to a certain degree to contamination of the tumor sample by normal cells. In the future, we intend to extend our model to handle both high normal cell contamination levels and be able to detect sub-clonal SVs when values of tumour purity and sub-clonal cellularity are provided.

The current version of SV-Bay is able to analyze only one tumor/normal pair at once. One of the interesting possible extensions to our method would be to add the ability to analyze several tumour datasets extracted from the same patient in order to increase the sensitivity of SV detection.

5 Conclusion

We have presented SV-Bay, a computational method and software to detect structural variants in cancer using whole genome sequencing data with or without matched normal control sample. SV-Bay does not only use information about abnormal read mappings but also assesses changes in the copy number profile and tries to associate these changes with candidate SVs. The likelihood of each novel genomic adjacency is evaluated using a Bayesian model. In its final step, SV-Bay annotates genomic adjacencies according to their type and, where possible, groups detected genomic adjacencies into complex SVs as balanced translocations, co-amplifications, and so on. A comparison of SV-Bay with BreakDancer, Lumpy, DELLY and GASVPro demonstrated its superior performance on both simulated and experimental datasets.

Funding

This work has been supported by the French program 'Investissement d'Avenir', action bioinformatique (ABS4NGS project), and by the French research cluster Digiteo.

Conflict of Interest: none declared.

References

Benjamini,Y. and Speed,T.P. (2012) Summarizing and correcting the GC content bias in high-throughput sequencing. *Nucl. Acids Res.*, **40**, e72–e72.

Boeva,V. *et al.* (2013) Breakpoint features of genomic rearrangements in neuroblastoma with unbalanced translocations and chromothripsis. *PLoS ONE*, **8**, e72182.

Boeva,V. *et al.* (2011) Control-free calling of copy number alterations in deep-sequencing data using GC-content normalization. *Bioinformatics*, **27**, 268–269.

Boeva,V. *et al.* (2012) Control-FREEC: a tool for assessing copy number and allelic content using next-generation sequencing data. *Bioinformatics*, **28**, 423–425.

Chen,K. *et al.* (2009) BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. *Nat. Methods*, **6**, 677–681.

Escaramís,G. *et al.* (2013) PeSV-Fisher: identification of somatic and non-somatic structural variants using next generation sequencing data. *PLoS ONE*, **8**, e63377.

Handsaker,R.E. *et al.* (2011) Discovery and genotyping of genome structural polymorphism by sequencing on a population scale. *Nat. Genet.*, **43**, 269–276.

Hormozdiari,F. *et al.* (2009) Combinatorial algorithms for structural variation detection in high-throughput sequenced genomes. *Genome Res.*, **19**, 1270–1278.

Hormozdiari,F. *et al.* (2010) Next-generation VariationHunter: combinatorial algorithms for transposon insertion discovery. *Bioinformatics*, **26**, i350–i357.

Hu,X. *et al.* (2012) pIRS: Profile-based illumina pair-end reads simulator. *Bioinformatics*, **28**, 1533–1535.

Jiang,Y. *et al.* (2012) PRISM: Pair-read informed split-read mapping for base-pair level detection of insertion, deletion and structural variants. *Bioinformatics*, **28**, 2576–2583.

Korbel,J.O. *et al.* (2009) PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biol.*, **10**, R23.

Layer,R.M. *et al.* (2014) LUMPY: a probabilistic framework for structural variant discovery. *Genome Biol.*, **15**, R84.

Lee,S. *et al.* (2009) MoDIL: detecting small indels from clone-end sequencing with mixtures of distributions. *Nat. Methods*, **6**, 473–474.

Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*, **25**, 1754–1760.

Medvedev,P. *et al.* (2009) Computational methods for discovering structural variation with next-generation sequencing. *Nat. Methods*, **6**, S13–S20.

Moncunill,V. *et al.* (2014) Comprehensive characterization of complex structural variations in cancer by directly comparing genome sequence reads. *Nat. Biotechnol.*, **32**, 1106–1112.

Oesper,L. *et al.* (2012) Reconstructing cancer genomes from paired-end sequencing data. *BMC Bioinformatics*, **13**, S10.

Pleasance,E.D. *et al.* (2010) A small-cell lung cancer genome with complex signatures of tobacco exposure. *Nature*, **463**, 184–190.

Popova,T. *et al.* (2009) Genome Alteration Print (GAP): a tool to visualize and mine complex cancer genomic profiles obtained by SNP arrays. *Genome Biol.*, **10**, R128.

Qi,J. and Zhao,F. (2011) inGAP-sv: a novel scheme to identify and visualize structural variation from paired end mapping data. *Nucleic Acids Res.*, **39**, W567–W575.

Rausch,T. *et al.* (2012) DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*, **28**, i333–i339.

Schröder,J. *et al.* (2014) Socrates: identification of genomic rearrangements in tumour genomes by re-aligning soft clipped reads. *Bioinformatics*, **30**, 1064–1072.

Sindi,S. *et al.* (2009) A geometric approach for classification and comparison of structural variants. *Bioinformatics*, **25**, i222–i230.

Sindi,S.S. *et al.* (2012) An integrative probabilistic model for identification of structural variation in sequencing data. *Genome Biol.*, **13**, R22.

Stephens,P.J. *et al.* (2009) Complex landscapes of somatic rearrangement in human breast cancer genomes. *Nature*, **462**, 1005–1010.

Stephens,P.J. *et al.* (2011) Massive genomic rearrangement acquired in a single catastrophic event during cancer development. *Cell*, **144**, 27–40.

Trappe,K. *et al.* (2014) Gustaf: detecting and correctly classifying SVs in the NGS twilight zone. *Bioinformatics*, btu431.

Vogelstein,B. and Kinzler,K.W. (2002) *The Genetic Basis of Human Cancer* 2 edn. McGraw-Hill Professional, New York.

Wang,J. *et al*. (2011) CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nat. Methods*, **8**, 652–654.

Wong,K. *et al*. (2010) Enhanced structural variant and breakpoint detection using SVMerge by integration of multiple detection methods and local assembly. *Genome Biol*., **11**, R128.

Yang,L. *et al*. (2013) Diverse mechanisms of somatic structural variations in human cancer genomes. *Cell*, **153**, 919–929.

Yoon,S. *et al*. (2009) Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Res*., **19**, 1586–1592.

Zeitouni,B. *et al*. (2010) SVDetect: a tool to identify genomic structural variations from paired-end and mate-pair sequencing data. *Bioinformatics*, **26**, 1895–1896.