Bioinformatics, 38(18), 2022, 4293–4300
https://doi.org/10.1093/bioinformatics/btac510
Advance Access Publication Date: 28 July 2022
Original Paper

OXFORD

Sequence analysis

SECEDO: SNV-based subclone detection using ultra-low coverage single-cell DNA sequencing

Hana Rozhoňová[1,2,3,†], Daniel Danciu[1,4,†], Stefan Stark [1,3,4], Gunnar Rätsch [1,3,4,5,‡], André Kahles [1,3,4,‡] and Kjong-Van Lehmann [1,3,4,6,7,*,‡]

[1]Biomedical Informatics Group, Department of Computer Science, ETH Zurich, Zurich, Switzerland, [2]Institute of Integrative Biology, ETH Zurich, Zurich, Switzerland, [3]Swiss Institute of Bioinformatics, Lausanne, Switzerland, [4]Biomedical Informatics Research, University Hospital Zurich, Zurich, Switzerland, [5]Department of Biology, ETH Zurich, Zurich, Switzerland, [6]Cancer Research Center Cologne Essen, University Hospital Cologne, Cologne, Germany and [7]Joint Research Center for Computational Biomedicine, University Hospital RWTH Aachen, Aachen, Germany

*To whom correspondence should be addressed.
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.
‡The authors wish it to be known that, in their opinion, the last three authors should be regarded as Joint Last Authors.
Associate Editor: Anthony Mathelier

Received on January 12, 2022; revised on July 4, 2022; editorial decision on July 8, 2022

Abstract

Motivation: Several recently developed single-cell DNA sequencing technologies enable whole-genome sequencing of thousands of cells. However, the ultra-low coverage of the sequenced data ($<0.05\times$ per cell) mostly limits their usage to the identification of copy number alterations in multi-megabase segments. Many tumors are not copy number-driven, and thus single-nucleotide variant (SNV)-based subclone detection may contribute to a more comprehensive view on intra-tumor heterogeneity. Due to the low coverage of the data, the identification of SNVs is only possible when superimposing the sequenced genomes of hundreds of genetically similar cells. Thus, we have developed a new approach to efficiently cluster tumor cells based on a Bayesian filtering approach of relevant loci and exploiting read overlap and phasing.

Results: We developed **S**ingl**e** **Ce**ll **D**ata Tum**o**r Clusterer (SECEDO, lat. 'to separate'), a new method to cluster tumor cells based solely on SNVs, inferred on ultra-low coverage single-cell DNA sequencing data. We applied SECEDO to a synthetic dataset simulating 7250 cells and eight tumor subclones from a single patient and were able to accurately reconstruct the clonal composition, detecting 92.11% of the somatic SNVs, with the smallest clusters representing only 6.9% of the total population. When applied to five real single-cell sequencing datasets from a breast cancer patient, each consisting of ≈2000 cells, SECEDO was able to recover the major clonal composition in each dataset at the original coverage of $0.03\times$, achieving an Adjusted Rand Index (ARI) score of ≈0.6. The current state-of-the-art SNV-based clustering method achieved an ARI score of ≈0, even after merging cells to create higher coverage data (factor 10 increase), and was only able to match SECEDOs performance when pooling data from all five datasets, in addition to artificially increasing the sequencing coverage by a factor of 7. Variant calling on the resulting clusters recovered more than twice as many SNVs as would have been detected if calling on all cells together. Further, the allelic ratio of the called SNVs on each subcluster was more than double relative to the allelic ratio of the SNVs called without clustering, thus demonstrating that calling variants on subclones, in addition to both increasing sensitivity of SNV detection and attaching SNVs to subclones, significantly increases the confidence of the called variants.

Availability and implementation: SECEDO is implemented in C++ and is publicly available at https://github.com/ratschlab/secedo. Instructions to download the data and the evaluation code to reproduce the findings in this paper are available at: https://github.com/ratschlab/secedo-evaluation. The code and data of the submitted version are archived at: https://doi.org/10.5281/zenodo.6516955.

Contact: kjlehmann@ukaachen.de

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

1 Introduction

Somatic single-nucleotide variants (SNVs) are commonly associated with cancer progression and growth (Stratton *et al.*, 2009). The recent development of single-cell DNA sequencing technologies (Gawad *et al.*, 2016) offers the ability to study somatic SNVs at a single-cell level, providing much more detailed information about tumor composition and phylogeny than traditional bulk sequencing (Kuipers *et al.*, 2017; Navin *et al.*, 2011). However, several technical obstacles decrease the interpretability of the data obtained using these technologies. In particular, most of the current single-cell DNA sequencing technologies require a whole-genome amplification step, which introduces artifacts such as DNA-amplification errors and imbalanced amplification of alleles (up to the complete dropout of alleles) (Gawad *et al.*, 2016). Several approaches (Bohrson *et al.*, 2019; Dong *et al.*, 2017; Hård *et al.*, 2019; Lähnemann *et al.*, 2021; Luquette *et al.*, 2019; Singer *et al.*, 2018; Zafar *et al.*, 2016) have been proposed to detect SNVs based on such data.

Approaches that do not require whole-genome amplification have been developed to overcome issues related to amplification (Laks *et al.*, 2019; Navin *et al.*, 2011). A prominent example of such technologies is 10X Genomics' Chromium Single Cell CNV Solution (https://www.10xgenomics.com/resources/datasets/). This technology allows the sequencing of hundreds to thousands of cells in parallel, albeit with only extremely low-sequencing coverage ($<0.05\times$ per cell). Hence, its use has been limited to the inference of copy number variations (CNVs) and alterations (CNAs) (https://bit.ly/37oZIPG) (Durante *et al.*, 2020; Velazquez-Villarreal *et al.*, 2020; Zaccaria and Raphael, 2021). The attempts to also use these data for the identification of tumor subclones based solely on SNVs have so far failed to provide a solution that would be able to recover the clonal composition at the original sequencing depth (Myers *et al.*, 2020); in particular, SBMClone, the algorithm of Myers *et al.* (2020), requires a minimum coverage of $\geq 0.2\times$ per cell, roughly four times more than what is currently achievable using the 10X Genomics technology (Velazquez-Villarreal *et al.*, 2020).

In this work, we propose SECEDO (**S**ingle **C**ell **D**ata Tumor Clusterer), a novel algorithm for clustering cells based on SNVs using single-cell sequencing data with ultra-low coverage. Using an extensive set of simulated data, as well as five real datasets, we show that SECEDO is able to correctly identify tumor subclones in datasets with per-cell coverage as low as $0.03\times$, improving the current state of the art by a factor of seven and thus rendering the algorithm applicable to currently available single-cell data. We also provide an efficient C++ implementation of SECEDO, which is able to quickly cluster sequencing data from thousands of cells while running on commodity machines.

2 Materials and methods

2.1 Overview

Due to the extremely low coverage of the data ($<0.05\times$ per cell), deciding whether two cells have identical or distinct genotypes is a difficult problem. Most loci are covered, if at all, by only one read (Supplementary Fig. S1). This makes it difficult, if not impossible, to interpret an observed mismatch when comparing data from two cells. The mismatch could be caused by an actual somatic SNV, by a sequencing error, or by a heterozygous locus that was sequenced in a different phase in the two cells. Hence, it is crucial to jointly leverage the information from all cells at the same time.

The pivotal blocks in the SECEDO pipeline (Fig. 1) are: (i) a Bayesian filtering strategy for efficient identification of relevant loci and (ii) derivation of a global cell-to-cell similarity matrix utilizing both the structure of reads and the haplotype phasing, which proves to be more informative than considering only one locus at a time.

SECEDO first performs a filtering step, in which it examines the pooled sequenced data for each locus and uses a Bayesian strategy to eliminate loci that are unlikely to carry a somatic SNV. The filtering step drastically increases the signal-to-noise ratio by reducing the number of loci by 3–4 orders of magnitude (depending on the coverage), while only eliminating approximately half of the loci that carry a somatic SNV. Moreover, the eliminated mutated loci typically have low coverage or high error rate and would not be very useful for clustering. In the second step, SECEDO builds a cell-to-cell similarity matrix based only on read-pairs containing the filtered loci, using a probabilistic model that takes into account the probability of sequencing errors, the frequency of SNVs, the filtering performance, and, crucially, the structure of the reads, i.e. the fact that the whole read was sampled from the same haplotype. In the third step of the pipeline, we use spectral clustering to divide the cells into two or more groups. At this point, we reduced the problem to an instance of the well-studied community detection problem (Porter *et al.*, 2009), so spectral clustering is a natural choice. Optionally, the results of spectral clustering can be further refined in a fourth step using the expectation–maximization (EM) algorithm (Dempster *et al.*, 1977). The whole pipeline is then repeated for each of the resulting subclusters. The process is stopped if (i) there is no evidence for the presence of at least two clusters in the similarity matrix, or (ii) the clusters are deemed too small. Downstream analysis, for instance, variant calling, can then be performed by pooling sequencing data from all cells in one cluster based on the results of SECEDO to create a pseudo-bulk sample.

2.2 Filtering uninformative loci

Consideration of all genomic loci is not desirable when performing the clustering and variant calling since most positions are not informative for clonal deconvolution. The most informative loci with respect to the clustering of the cells are the loci carrying somatic SNVs since they provide (i) information on the assignment of cells to clusters and (ii) information on haplotype phasing (due to loss/gain of heterozygosity). To a lesser extent, this is also true for germline heterozygous loci since they provide information on haplotype phasing. In other words, loci at which all the cells have the same homozygous genotype do not provide any information relevant to the task of dividing the cells into genetically homogeneous groups, so they can be excluded from downstream analysis.

Due to the low sequencing coverage, it is generally not possible to reliably assign genotypes to individual cells. However, we identify loci of interest by using the *pooled data* across all the cells to approximate posterior probabilities that the cells have the same genotype. Consider for example a specific locus at which all cells have genotype AA. Assuming sequencing errors happen independently with probability θ and are unbiased (i.e. all types of substitutions are equally probable), the fraction of As in the pooled data is in expectation $(1-\theta)$ and the fraction of all other bases is $\theta/3$. A locus with a significantly different proportion of observed bases indicates that there may be two (or more) different genotypes contributing to the observed data. In particular, we compute the posterior probability that all cells at the locus share the same homozygous genotype using an approximate Bayesian procedure. If this posterior is lower than a chosen threshold K, the locus is marked as 'informative'.

Formally, let C_1, C_2, C_3, C_4 be the bases sorted from the most to the least frequent in the pooled data at the given position, c_1, c_2, c_3, c_4 the corresponding counts ($c_1 \geq c_2 \geq c_3 \geq c_4$), c the total coverage ($c = c_1 + c_2 + c_3 + c_4$). Next, let M be an indicator random variable that is 1 if all cells in the sample have the same homozygous genotype and 0 otherwise. Applying Bayes rule, we can compute $P(M = 1|c_1, c_2, c_3, c_4)$ as:

$$P(M=1|c_1,c_2,c_3,c_4) = \frac{P(c_1,c_2,c_3,c_4|M=1)P(M=1)}{P(c_1,c_2,c_3,c_4)}. \quad (1)$$

We compute or approximate the individual terms as follows:

- $P(M=1)$ can be estimated from literature: the prevalence of somatic SNVs in cancer lies between 10^{-9} and 10^{-3} (Alexandrov *et al.*, 2013; Lawrence *et al.*, 2013); the frequency of heterozygous sites in a typical human genome lies between ca 0.04% and 0.11% (Bryc *et al.*, 2013; Meyer *et al.*, 2012). In order to be

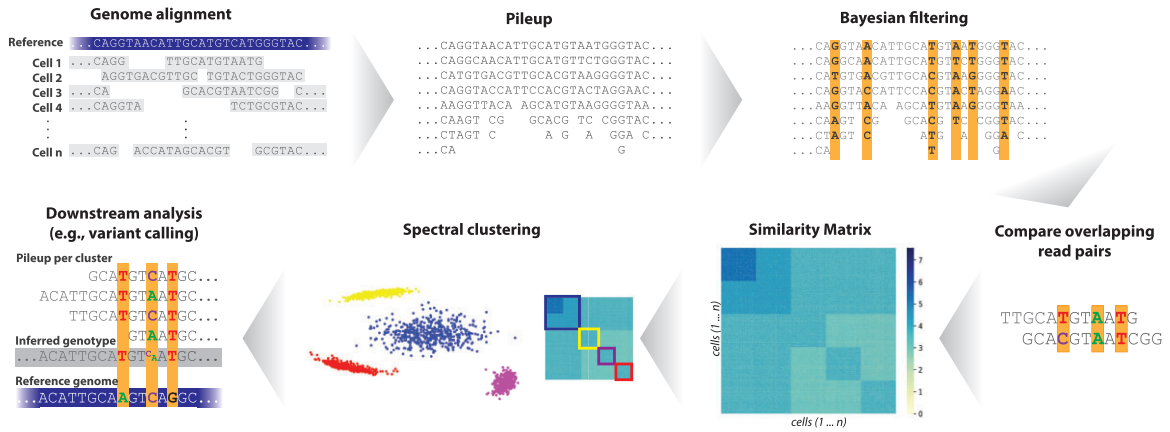

Fig. 1. The SECEDO pipeline. After sequencing, reads are piled up per locus and a Bayesian filter eliminates loci that are unlikely to carry a somatic SNV. For each pair of reads, SECEDO compares the filtered loci and updates the likelihoods of having the same genotype and of having different genotypes for the corresponding cells. The similarity matrix, computed as described in Section 2, is then used to cluster the cells into 2–4 groups (the number of groups depends on the data and is determined automatically by SECEDO) using spectral clustering. The algorithm is then recursively applied to each cluster until a termination criterion is reached

conservative, we choose the largest probability ($\approx 10^{-3}$) in both cases, resulting in $P(M=1) \approx 1 - 2 \times 10^{-3} = 0.998$.

- $P(c_1, c_2, c_3, c_4 | M = 1)$, is equal to:

$$P(c_1, c_2, c_3, c_4 | M = 1) = \sum_{g \in \mathcal{G}} \alpha_g P(g), \tag{2}$$

where $\alpha_g = P(c_1, c_2, c_3, c_4 | \text{genotype of all cells is } g)$ and $\mathcal{G} = \{AA, CC, GG, TT\}$ is the set of all possible homozygous genotypes.

The probability α_g of observing data (c_1, c_2, c_3, c_4) given that the genotype of all cells is g ($g = C_i C_i$) has a multinomial distribution with c trials and event probabilities equal to $\left(1-\theta, \frac{\theta}{3}, \frac{\theta}{3}, \frac{\theta}{3}\right)$:

$$\alpha_g = \frac{c!}{c_1! c_2! c_3! c_4!} (1-\theta)^{c_i} \left(\frac{\theta}{3}\right)^{c - c_i}.$$

Assuming the error rate θ is small, the result of the equation above is negligible for any c_i that is not close to c. As a consequence, if the prior $P(g)$ is approximately the same for all genotypes, we can approximate the sum in Equation (2) with the largest term:

$$P(c_1, c_2, c_3, c_4 | M = 1) \approx \max_{g \in \mathcal{G}} \alpha_g P(g). \tag{3}$$

- Computing $P(c_1, c_2, c_3, c_4)$ is intractable, as it would involve summing over all possible combinations of the cells' genotypes. We instead approximate the evidence by:

$$\begin{aligned} P(c_1, c_2, c_3, c_4) \approx \frac{c!}{c_1! c_2! c_3! c_4!} \Bigg[& p_{\text{hom}} (1-\theta)^{c_1} \left(\frac{\theta}{3}\right)^{c_2+c_3+c_4} \\ & + p_{\text{het}} \left(\frac{1}{2} - \frac{\theta}{3}\right)^{c_1+c_2} \left(\frac{\theta}{3}\right)^{c_3+c_4} \\ & + p_{\text{hom}} p_{\text{mut}} \left(\frac{3}{4} - \frac{2\theta}{3}\right)^{c_1} \left(\frac{1}{4}\right)^{c_2} \left(\frac{\theta}{3}\right)^{c_3+c_4} \\ & + p_{\text{het}} p_{\text{mut}} \frac{c!}{c_1! c_2! c_3! c_4!} \left(\frac{1}{2} - \frac{\theta}{3}\right)^{c_1} \left(\frac{1}{4}\right)^{c_2+c_3} \left(\frac{\theta}{3}\right)^{c_4} \Bigg] \end{aligned},$$

where $p_{\text{hom}}, p_{\text{het}}, p_{\text{mut}}$ represent the probability of a locus being homozygous, heterozygous and mutated, respectively. The first summation term estimates $P(c_1, c_2, c_3, c_4)$ for a homozygous locus, the second term assumes a heterozygous locus, the third term corresponds to a homozygous locus that suffered a somatic mutation, and the last term to a heterozygous locus with a somatic mutation (see Supplementary Material S1 for a more detailed derivation). In order to be consistent with the prior probability $P(M=1)$, we used $p_{\text{het}} = 10^{-3}$ (Bryc *et al.*, 2013; Meyer *et al.*, 2012), $p_{\text{mut}} = 10^{-3}$ (Alexandrov *et al.*, 2013; Lawrence *et al.*, 2013), and $p_{\text{hom}} = 1 - p_{\text{het}} - p_{\text{mut}}$.

We then include the locus into the subset of informative positions if $P(M = 1 | c_1, c_2, c_3, c_4) \leq K$ for a suitable constant K (see Supplementary Material S2 and Supplementary Table S1).

Filtering heterozygous loci is similar. Here, let $P(M' = 1 | c_1, c_2, c_3, c_4)$ be the probability that all cells have the same *heterozygous* genotype. The individual terms in Equation (1) are identical except that the event probabilities for the multinomial distribution are $\left(\frac{1}{2} - \frac{\theta}{3}, \frac{1}{2} - \frac{\theta}{3}, \frac{\theta}{3}, \frac{\theta}{3}\right)$. However, since heterozygous loci are three orders of magnitude fewer than homozygous loci (Bryc *et al.*, 2013; Meyer *et al.*, 2012) in addition to potentially being useful in haplotype phasing, we empirically determined that the following simpler and faster criteria works equally well in practice: denote the locus as informative if $c_1 > 1.5 \times c_2$, where c_1 and c_2 are the most frequent and the second most frequent bases at that locus, respectively (the expectation is that at a heterozygous locus c_1 and c_2 should not differ too much). In addition, we reject all loci for which $c_1 + c_2 + c_3 < 5$.

The final set of informative loci then includes those positions that were marked as informative by both filtering steps (i.e. filtering of both homozygous and heterozygous loci). In practice, sequencing artifacts may lead to loci with unusually high coverage. For this reason, we also eliminate any loci with coverage more than two SDs away from the expected coverage. In addition, we also eliminate loci where $c - c_1 < 5$ (see Supplementary Fig. S2).

2.3 Cell-to-cell similarities

We define the similarity $s(i, j)$ of cells i and j as the log-odds of the probability that cells i and j have the same genotype and the probability that they have different genotypes, given the corresponding sets of reads. Each of the two probabilities is then approximated as a product of probabilities of individual *overlaps* of two reads, one read from cell i and one read from cell j (Fig. 2). Formally:

$$\begin{aligned} s(i,j) &= \log \left(\frac{P[C(i) = C(j) | r_i, r_j, h, \epsilon]}{P[C(i) \neq C(j) | r_i, r_j, h, \epsilon]} \right) \\ &= \log \left(\frac{P[r_i, r_j | C(i) = C(j), h, \epsilon]}{P[r_i, r_j | C(i) \neq C(j), h, \epsilon]} \right) \end{aligned} \tag{4}$$

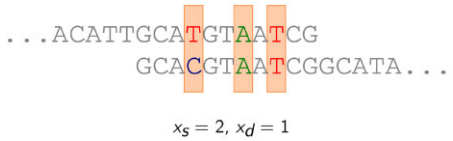

Fig. 2. Illustration of an overlap between two reads. The shaded positions are the positions chosen as informative. In this example, length of the overlap is 3, the number of positions where the bases are the same, x_s, is 2 and the number of positions where they are different, x_d, is 1. For our purposes, an overlap is fully described by the tuple (x_s, x_d)

$$\approx \sum_{k,l} \log\left(\frac{P[x_s(r_i^k, r_j^l), x_d(r_i^k, r_j^l) | C(i) = C(j), h, \epsilon]}{P[x_s(r_i^k, r_j^l), x_d(r_i^k, r_j^l) | C(i) \neq C(j), h, \epsilon]}\right), \tag{5}$$

where r_i is the set of reads from cell i, r_i^k is the k-th read from cell i, $x_s(p,q)$ and $x_d(p,q)$ the number of matches and mismatches, respectively, between reads p and q, $C(i)$ is the (true) cluster assignment of cell i, ϵ is the proportion of SNVs in the set of informative positions and h the proportion of homozygous loci in the set of informative positions (see below). In case the two cells have no overlapping reads, the similarity is by definition equal to 0 (i.e. we have no information on whether the two cells have equal or different genotypes). We assume that observing cells with the same genotype and with different genotypes has the same prior probability. (Notice that decomposing the probabilities in Equation (4) over pairs of reads is indeed only an approximation. In particular, the decomposition in Equation (5) would only be precise if no two reads coming from one cell were overlapping; in the opposite case, the probabilities of read pairs containing one of these overlapping reads are non-independent. However, since the per-cell coverage is so low (Supplementary Fig. S1), the number of such non-independent pairs is negligible.)

Notice that by decomposing the probabilities over the overlaps of reads we gain information not only on the number of matches and mismatches between the two reads (i.e. information on potential differences between the two cells), but also information on haplotype phasing. Moreover, it also allows us to put more weight on longer (and hence supposedly more informative) overlaps. For example, a long overlap with only matches is an indication that the two cells might have the same genotype. A long overlap with only mismatches, on the other hand, is not a strong indication towards the cells being from different clusters—another likely scenario is that the two reads were sampled from different haplotypes and we just observe a row of heterozygous loci in different phase. As a result, overlaps with a combination of matches and mismatches are the ones most strongly suggesting the 'different genotypes' case (Supplementary Fig. S3). We also show, using simulated data, that considering the number of matches and mismatches in the whole overlap of two reads provides strictly more information than considering each locus independently (Supplementary Fig. S4).

Below we give details on the computation of Equation (5), under the simplifying assumptions that (i) all cells are diploid, (ii) the somatic SNVs are with equal probability of type $AA{+}AB$ and $AB{+}AA$ (a homozygous site in cluster 1, heterozygous in cluster 2, or vice versa), and (iii) the prevalence of differences between any two subclones is μ (see Supplementary Material S3 for the full list of assumptions).

2.4 Parameters

The algorithm has three parameters: h, the fraction of the homozygous loci in the set of selected positions, ϵ, the fraction of the mutated loci in the set, and θ, the error rate. In our analyses, we used h=0.5, $\epsilon = 0.01$, and $\theta = 0.05$ (the θ parameter has higher value than the usually reported sequencing error rate, because the set of informative positions is enriched in positions carrying sequencing errors). See Supplementary Figure S5 for a justification of the given parameter choices and Supplementary Table S3 for an analysis of SECEDOs performance under various parameter combinations.

2.5 Computing the probabilities of overlaps

We define:

- $P_{s,s}$, the probability that sequencing of two bases of the same kind results again in two bases of the same kind: $P_{s,s} = (1-\theta)^2 + \frac{\theta^2}{3}$ (both bases are sequenced without error, or both are misread to the same base),
- $P_{s,d}$, the probability that sequencing of two bases of the same kind results in bases that differ from each other: $P_{s,d} = 1 - P_{s,s}$,
- $P_{d,s}$, the probability that two different bases are read as the same: $P_{d,s} = 2 \times (1-\theta) \times \frac{\theta}{3} + \frac{2\theta^2}{9}$ (one of the two bases is misread to the other one, or both are misread to the same base),
- $P_{d,d}$ the probability that two different bases are sequenced as different: $P_{d,d} = 1 - P_{d,s}$.

The probability of observing x_s matches and x_d mismatches in an overlap of length $x_s + x_d$, assuming cells i and j have the same genotype, is now:

$$P[x_s, x_d | C(i) = C(j), h, \epsilon] = \binom{x_s + x_d}{x_s} \sum_{k=0}^{x_s} \sum_{l=0}^{x_d} \binom{x_s}{k} \binom{x_d}{l} \times \overbrace{\left(1 - h - \frac{\epsilon}{2}\right)^{k+l} \left(\frac{1}{2}(P_{s,s}^k \times P_{s,d}^l + P_{d,s}^k \times P_{d,d}^l)\right)^{\delta(k+l)}}^{\text{heterozygous positions}} \times \underbrace{\left(h + \frac{\epsilon}{2}\right)^{(x_s + x_d - k - l)} \times P_{s,s}^{(x_s - k)} \times P_{s,d}^{(x_d - l)}}_{\text{homozygous positions}}, \tag{6}$$

where $\delta(x)$ is a function defined as 0, if $x = 0$, and 1, otherwise. In the formula we sum over all possible combinations of $(k+l)$ heterozygous loci and $(x_s + x_d - k - l)$ homozygous loci; k of the heterozygous loci result in a match, the remaining l in a mismatch.

The probability of observing x_s matches and x_d mismatches assuming cells i and j are in different clusters is:

$$\begin{aligned} P[x_s, x_d | C(i) \neq C(j), h, \epsilon] = & \binom{x_s + x_d}{x_s} \times \sum_{k=0}^{x_s} \sum_{p=0}^{x_s - k} \sum_{l=0}^{x_d} \sum_{q=0}^{x_d - l} \frac{x_s!}{k!p!(x_s - k - p)!} \cdot \frac{x_d!}{l!q!(x_d - l - q)!} \\ & \times \overbrace{(1 - h - \epsilon)^{k+l} \left(\frac{1}{2}(P_{s,s}^k \times P_{s,d}^l + P_{d,s}^k \times P_{d,d}^l)\right)^{\delta(k+l)}}^{\text{heterozygous positions}} \\ & \times \overbrace{\left(\frac{\epsilon}{2}\right)^{(x_s + x_d - k - l - p - q)} \times (P_{s,s} + P_{d,s})^{(x_s - k - p)}}^{\text{mutated positions}} \\ & \times \overbrace{(P_{d,d} + P_{s,d})^{(x_d - l - q)}}^{\text{mutated positions}} \times \overbrace{h^{(p+q)} \times P_{s,s}^p P_{s,d}^q}^{\text{homozygous positions}}. \end{aligned} \tag{7}$$

Here, k denotes the number of heterozygous positions giving rise to a match, l the number of heterozygous positions giving rise to a mismatch, p the number of positions with the same homozygous genotype in both types of cells that give rise to a match and q the number of these positions that result in a mismatch.

2.6 Clustering

We first normalize the computed similarity matrix by making sure all elements are positive: $S^* = -S + \min_{i,j} s(i,j)$. The cells are then clustered using a slight variation on spectral clustering (Ng *et al.*, 2001) as follows. We compute the symmetric normalized Laplacian

$\mathcal{L} = I - D^{-\frac{1}{2}} S^* D^{-\frac{1}{2}}$ and determine its first k (we used k=6 in all experiments in this paper) eigenvectors, corresponding to the k smallest eigenvalues. We then cluster into 1, 2, 3 or 4 clusters using k-means (Arthur and Vassilvitskii, 2006; Lloyd, 1982), computing the inertia values i_1, i_2, i_3, i_4 for each of the four options and the inertia gaps $g_k = i_k - i_{k-1}, k = 2, 3, 4$, and define $g_1 := 0$. The final number of clusters is $\max_{k=2,3,4}\{k | g_k > 0.75 g_{k-1}\}$.

An important feature of spectral clustering is that it leverages the information on similarities of all pairs of cells at the same time. Thus, even in case two cells would not have any overlapping reads (the probability of which is negligibly small, see Supplementary Material S4), they could still be clustered based on their similarities to other cells in the dataset.

Optionally, the results of the previous step are further refined using the EM algorithm (Dempster *et al.*, 1977) (Supplementary Material S5). However, all results reported in this paper were obtained without the EM-refinement.

One important aspect of clustering is the stopping criterion, i.e. the decision whether a specific group of cells should be divided into subclusters or not. We suggest a heuristic approach to automatically decide if the computed normalized similarity matrix S^* indicates that there are two (or more) different clusters of cells. We fit a Gaussian mixture model with 1, 2, 3 or 4 components to the smallest k eigenvectors of S^* and compare their likelihood using the Bayesian information criterion (BIC). If the model with only one component is preferred by BIC over the models with 2, 3 or 4 components we do not split the data further. We further do not accept the split if the resulting subclone has too few cells (we used 500 in our experiments). We also require that the mean within-cluster coverage is at least 9, the lowest coverage sufficient for a reliable variant call (see Supplementary Material S6).

3 Results

3.1 SECEDO recovers tumor subclones with average precision of 97% on simulated data

In order to test the performance of our method, we simulated a dataset consisting of 7250 cells divided into nine groups of various sizes: one group of healthy cells and eight groups of tumor cells. The genome of the healthy cells was created using Varsim 0.8.4 (Mu *et al.*, 2014) based on the GRCh38.p13 human reference genome. Common variants from dbSNP 20180418 (Sherry *et al.*, 2001) (3 000 000 single-nucleotide polymorphisms, 100 000 small insertions, 100 000 small deletions, 50 000 multi-nucleotide polymorphisms, 50 000 complex variants) were added to the genome. The genome of the tumor cells was built by adding 2500–20 000 of both coding and non-coding SNVs (subclonal SNV fraction of 3–27%; Dentro *et al.*, 2021), randomly chosen from the COSMIC v94 (Catalogue Of Somatic Mutations In Cancer) database (Tate *et al.*, 2018), to the parent genome, in addition to 250 small insertions, 250 small deletions, 200 multi-nucleotide variants and 200 complex variants (Fig. 3, left). Paired-end reads, with each mate of length 100 bp, were simulated using ART 2.5.8 (Huang *et al.*, 2011) at an average coverage of 0.05× per cell and with the error profile of Illumina HiSeq 2000 machines. The reads were then aligned using Bowtie 2.4.4 (Langmead and Salzberg, 2012) and filtered using Samtools 1.12 (Li, 2011) to select for reads mapped only in proper pair, non-duplicate and only primary alignments.

For efficiency reasons, we build the pileup files used by the Bayesian filtering using our own implementation rather than existing tools that are not optimized for use on thousands of cells simultaneously (e.g. Samtools, which currently does not offer a multi-threaded pileup creation). We eliminate bases with read qualities below 30, reads with a mapping quality below 30 and loci where all bases are identical. The pileup creation, distributed on 23 commodity machines (one for each chromosome) using 20 threads each, takes about 70 min (down from 72 h when using Samtools' pileup creation on the same machines). We ran SECEDO on the resulting pileup files on an Intel(R) Xeon(R) Gold 6140 CPU @ 2.30GHz using 20 threads and 32GB of RAM. The filtering, clustering and VCF generation took 21 min. Since the time complexity of the most expensive step, the similarity matrix computation, is quadratic in the pooled coverage (for every filtered position), the running time decreases rapidly with each level of the tree. Consistently, half of the total running time was spent performing the top-level clustering, where the filtering step kept about 1 in 16 000 loci. Somewhat counter-intuitively, the number of filtered loci approximately doubled at each level as we traveled down the clustering tree. This is due to the fact that previously rejected loci may be classified as informative when looking at a subset of the data, particularly loci containing reads from smaller clusters. At the same time, the discriminative power of the Bayesian filtering degrades as the mean pooled coverage decreases (from 248 at the root to 20 at the leaves), such that a larger proportion of loci that are not relevant are let through.

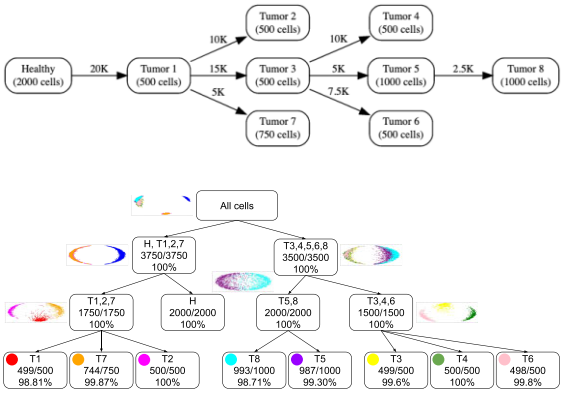

Fig. 3. Clustering a synthetic dataset with nine unequally sized subclones totaling 7250 cells. Top: Theoretical phylogenetic tree of the dataset. Edge labels indicate the number of additional SNVs in each subclone relative to the parent, node labels indicate the number of cells in each subclone. Bottom: Recursive clustering by SECEDO. Each node corresponds to one SECEDO clustering step; the first row indicates the subclones assigned to that node, the second row the number of recovered cells out of the total and the third row indicates the clustering precision (correctly clustered cells relative to total cells in cluster). The scatter plots above parent nodes depict the second and third eigenvectors of the similarity matrix Laplacian. For leaf nodes, SECEDO correctly determined that further clustering is not desirable

SECEDO was able to recover all nine subclones with an average precision of 97.45% (Fig. 3, right). Note that SECEDO is not attempting to reconstruct the evolutionary history of the tumor, but merely trying to efficiently find a grouping of cells that reflect the current subclonal structure and enable downstream tasks like variant calling. Therefore, the clustering tree reconstructed by SECEDO does not reflect the actual developmental process that gave rise to the given population of cancer cells; indeed, the SECEDO clustering tree differs from the true phylogenetic tree of the population (Fig. 3).

In order to show the potential of the resulting clusters for somatic variant calling, we identified the most likely genotype of each cluster using a simple MAQ-based approach (Li *et al.*, 2008) (Supplementary Material S6) and generated VCF files for each cluster against the GRCh38 human reference genome. Similarly to other variant callers that remove germline variants (Cibulskis *et al.*, 2013), we then removed the ground-truth variants that were present in the healthy cells and compared the remaining SNVs against the ground truth SNVs provided by Varsim for each cluster. SECEDO was able to detect 92.11% of the somatic SNVs (versus 77.79% when calling variants on the unclustered cells) with a 52.41% average precision (see Supplementary Table S2).

3.2 SECEDO is able to correctly group cells starting at 0.03× coverage and 500 cells per cluster

One practical question of crucial importance is how to determine if, given a dataset, SECEDO will be able to correctly cluster the cells for meaningful downstream processing. To answer this question, we conducted a series of experiments to determine the conditions under

which SECEDO can successfully be applied to a given dataset. There are three cluster attributes that affect SECEDOs ability to separate cell clusters: (i) the number of cells, (ii) the average per-cell coverage, and (iii) the number of SNVs in which the clones differ. In order to test the interplay of these three cluster attributes, we devised a series of synthetic datasets, each consisting of 1000 cells belonging to two groups. The sizes of the two groups were either equal (i.e. 500 cells in each group) or in ratio 1:3 (i.e. one cluster consisted of 250 cells and the other one of 750 cells). We further constructed a series of synthetic datasets consisting of 2000 cells being split equally among two groups (i.e. 1000 cells in each group). Then, for a given number of SNVs and given sizes of clusters, we gradually lowered the per-cell coverage until the algorithm was unable to cluster the cells correctly. The genome creation, reads simulation, and alignment were done as described in the previous section. For most parameter configurations, the currently achievable per-cell coverage of 0.05× is sufficient for SECEDO to correctly cluster the cells (see Fig. 4). Since SECEDO is able to discriminate between balanced clusters of 1000 cells that differ in as little as 2500 SNVs (equivalent to an SNV prevalence of ca 8.33×10^{-7}), the method can be applied to a wide variety of cancers, starting from those with very high mutation rates, such as melanoma (median prevalence of somatic SNVs ca 10^{-5}) down to pancreatic and breast cancer (median prevalence of somatic SNVs ca 10^{-6}) (Alexandrov *et al.*, 2013; Lawrence *et al.*, 2013). Note that there is a relationship between tumor mutational burden and SECEDOs ability to distinguish subclones. SECEDO is able to identify complex subclonal structures (such as in Fig. 3) in cancers with high mutational burden (e.g. melanoma), whereas in cancers with lower mutational burden (e.g. pancreatic and breast cancer) only major clones could be identified, as shown in the next section.

As expected, the discriminative power of SECEDO increases with the number of cells (Fig. 4), as well as with the per-cell coverage (Supplementary Fig. S6), since both act as a multiplying factor for the pooled coverage.

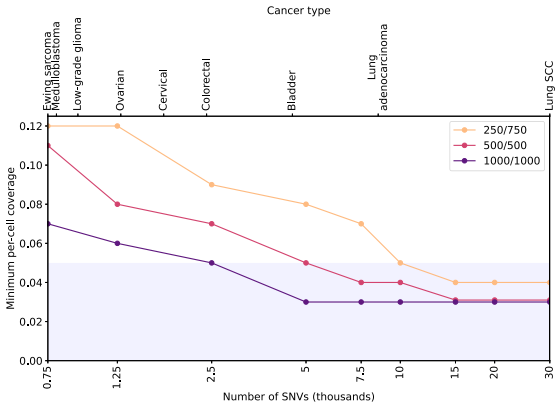

Fig. 4. Minimum required coverage for successful clustering (>90% precision and recall) of sub-clones differing in the given number of SNVs, in three scenarios: clustering 1000 cells, with a $(1/4, 3/4)$ split, with an equal $(1/2, 1/2)$ split, and clustering 2000 cells with an equal split. The shaded area marks the coverage currently achievable in practice. The top labels indicate the cancer type with median mutation rate closest to the given SNV density [cancer mutation rates according to Lawrence *et al.* (2013)]

3.3 SECEDO recovers dominant subclones in a breast cancer dataset, clearly outperforming state of the art

In order to test the performance of SECEDO on real data, we downloaded a publicly available 10X Genomics single-cell DNA sequencing dataset (https://github.com/ratschlab/secedo-evaluation/tree/main/breast_cancer) sequenced using an Illumina NovaSeq 6000 System. The dataset contains five tumor sections (labeled A–E) of a triple negative ductal carcinoma, each section containing roughly 2000 cells. The mean per-cell coverage in the dataset is 0.03×, with individual coverage ranging from 0.006× to 0.086×. CHISEL, the CNV-based clustering algorithm proposed by Zaccaria and Raphael (2021), identified three dominant clones in each of the sections, except for section A, which consists mainly of healthy cells.

We applied SECEDO separately to each of the tumor sections. The filtering step reduced the number of loci in each tumor section to roughly 1 000 000 bp (ca 0.03% of the original size); the average pooled coverage across the ≈2000 cells in each dataset ranged from 45 to 55. The number of clusters identified in each slice ranged between 3 and 10; it is likely that some of them are only artifacts. However, SECEDO was able to recover the three dominant clones in sections B, C, D, and E with high accuracy (96.68% recall, 66.59% precision) in the first two clustering steps. Note that we included cells that were unassigned to any clone by CHISEL, affecting our precision. The scatter plots of the second and third eigenvectors of the similarity matrix confirm that each tumor section, except for section A, indeed consists of three highly separable clusters (Fig. 5).

We compared SECEDOs performance to that of SBMClone (Myers *et al.*, 2020), the current state of the art in SNV-based clustering. As a metric for evaluation we used the Adjusted Rand Index (ARI) (Hubert and Arabie, 1985), measuring the similarity of the ground truth and data-derived clusterings. Since SBMClone was reported to work only at coverage ≥0.2×, and the coverage of the breast cancer dataset is 0.03×, we created higher coverage data *in silico* by merging sequencing data from cells reported to be in the same cluster by CHISEL. In addition, SBMClone requires a matched normal sample, so we again used the clustering in CHISEL to determine the healthy cells; from the variants determined using Varscan (https://github.com/raphael-group/chisel-data/blob/master/patientS0/snvs/cellmutations.tsv.gz) (Koboldt *et al.*, 2009), we removed all mutations that appeared in at least one healthy cell, and the remaining mutations were fed to SBMClone. SECEDO does not require a matched normal sample, so the sequencing data were used without this pre-processing. SECEDO correctly clustered (precision >96%) all cells at the original coverage (including the separation of healthy cells), and its performance remained relatively constant as

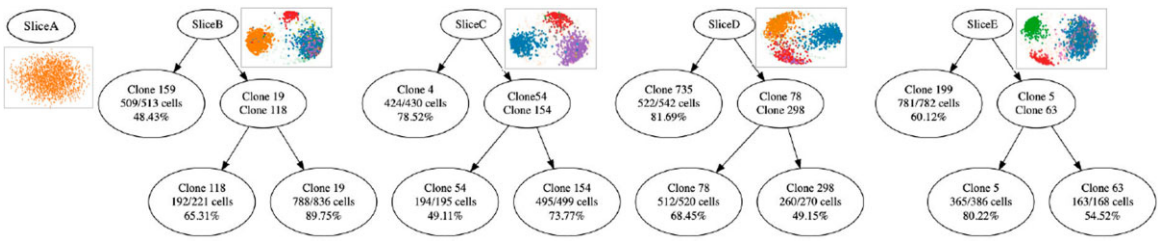

Fig. 5. Clustering of the five tumor sections in the 10x Genomics ductal carcinoma dataset. The first row in each node denotes the cluster name; for consistency, we used the same cluster numbering as CHISEL (https://github.com/raphael-group/chisel-data/). The second row denotes the number of cells recovered by SECEDO versus the total number of cells as identified by CHISEL. The last row denotes the precision of the clustering, i.e. the percentage of cells in the SECEDO cluster that match the originally reported cluster. The lower precision values are due to the fact that cells categorized by CHISEL as 'None' based on the CNV signature are assigned a category by SECEDO based on the genomic signature. The first section (SliceA) consists mainly of healthy cells, as reflected by the scatter plot of the second and third eigenvectors of the similarity matrix Laplacian

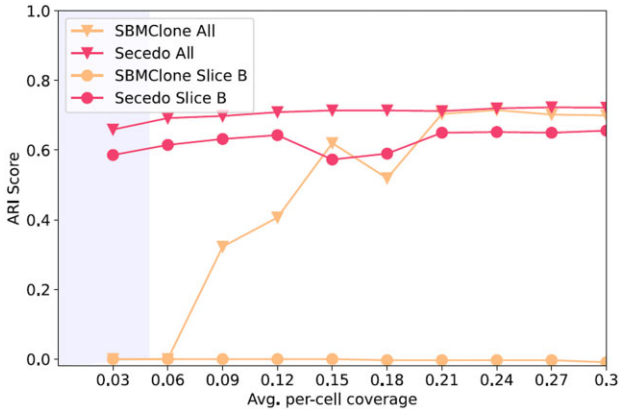

Fig. 6. Adjusted Rand Index (ARI) scores for the SECEDO and SBMClone clustering for Slice B and all slices of the breast cancer dataset at coverage ranging from 0.03× to 0.3×. Shaded area marks the average per-cell coverage achievable with current technology. Note that due to various factors such as cell merging and lack of ground truth the accuracy is not expected to be monotonically increasing

coverage increased. SBMClone was able to provide an approximate clustering starting at 3-fold the original coverage, and its performance matched SECEDOs at 7-fold the original coverage when combining data from all slices. For individual slices, SBMClone was not able to cluster the cells, irrespective of the coverage (Fig. 6).

We then called SNVs on each subclone of Slice B, as identified by SECEDO, independently, and on the entire slice. In order to call SNVs, we created a Panel of Normals from the cells categorized as normal by CHISEL based on the CNV profile (Clone19 in the leftmost tree of Fig. 5). We ran MuTect 1.1.4 (Cibulskis *et al.*, 2013) with the default settings, using dbSNP v20180418 (Sherry *et al.*, 2001) and Cosmic v94 (Tate *et al.*, 2018) as priors. The number of distinct SNVs in the two tumor subclones is more than double the number of variants that were called when pooling all cells together (Supplementary Fig. S7, left). The histogram of the allelic ratio for the sublconal and global SNVs shows a significant shift to the right for the subclonal SNVs, an indication that the clustering correctly identified and separated genetically similar cells, enabling the detection of twice as many SNVs at twice the allelic ratio (Supplementary Fig. S7, right).

4 Discussion

We introduced SECEDO, a method that is able to correctly identify SNV-based subclones in single-cell sequencing datasets with coverage as low as 0.03× per cell. This is a significant improvement in comparison to SBMClone, the current state-of-the-art method (Myers *et al.*, 2020), which, using the same data, was able to cluster the cells only after pooling data from all five datasets and artificially increasing the coverage by a factor of 7. This improvement in performance can likely be attributed to the fact that SECEDO takes into account the information on read phasing, as well as its efficient filtering of uninformative positions. We also note that unlike SBMClone, SECEDO does not require a matched normal sample for the identification of potential SNVs. We provide an efficient, well-tested, ready-to-use C++ implementation of SECEDO, which uses established data formats for both input and output, and can thus be easily incorporated into existing bioinformatics pipelines.

We demonstrated SECEDOs applicability to currently available single-cell sequencing data and find that SECEDO correctly clustered cells on a series of synthetic and five breast cancer datasets. CNA frequencies and patterns vary significantly across cancer types (Harbers *et al.*, 2021; Zack *et al.*, 2013), similarly to SNV frequency. Since SECEDO does not use copy-number information to cluster cells, it can infer sub-clones even in cancer types where CNAs do not vary or where the frequency of CNAs is generally low (e.g. pancreatic neuroendocrine tumors; Dentro *et al.*, 2021). It is also notable that not all CNAs affect the SNV profile of a cell. Thus, CNA-based clustering may lead to suboptimal grouping of cells, e.g. from a variant calling perspective. SECEDO is able to group cells with similar SNV profiles irrespective of their CNA profiles. This can lead to improvements in the precision and accuracy of the variant calling. Using the clusters identified by SECEDO, we were able to recover 92.11% of the SNVs present in the synthetic dataset using a simple variant caller. On Slice B of the breast cancer dataset, the number and the confidence of the called SNVs more than doubled after clustering using SECEDO, compared to calling variants on the entire slice.

While SECEDO enables accurate cell-clustering and variant calling, there are a number of areas for future improvement. First, SECEDO currently only uses single-nucleotide substitutions to cluster cells, which are known to be the most common type of mutations in adult and childhood cancers (Gröbner *et al.*, 2018; Lawrence *et al.*, 2014; Ma *et al.*, 2018). We expect that the clustering accuracy could be further improved if e.g. short insertions and deletions were additionally used. Second, the smallest subclones that SECEDO was able to detect had ≈200 cells. However, as technology inevitably improves and the sequencing coverage increases, SECEDOs resolution and variant calling quality will also proportionally increase.

We hope that SECEDO will facilitate new types of analyses and form the basis for future methodological development in the field of cancer research and treatment outcome prognosis.

Acknowledgement

The authors would like to thank Ximena Bonilla for her constructive feedback on the manuscript.

Funding

This work was supported by ETH core funding to G.R. (supporting D.D., S.S., A.K., K.-V.L.).

Conflict of Interest: none declared.

Data availability

The data underlying this article are available in Zenodo, at https://doi.org/10.5281/zenodo.6516955.

References

Alexandrov,L.B. *et al.* (2013). Signatures of mutational processes in human cancer. *Nature*, **500**, 415–421.

Arthur,D. and Vassilvitskii,S. (2006). k-means++: the advantages of careful seeding. *Technical Report 2006-13*. Stanford InfoLab.

Bohrson,C.L. *et al.* (2019). Linked-read analysis identifies mutations in single-cell DNA-sequencing data. *Nat. Genet.*, **51**, 749–754.

Bryc,K. *et al.* (2013). A novel approach to estimating heterozygosity from low-coverage genome sequence. *Genetics*, **195**, 553–561.

Cibulskis,K. *et al.* (2013). Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat. Biotechnol.*, **31**, 213–219.

Dempster,A.P. *et al.* (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B Methodol.*, **39**, 1–38.

Dentro,S.C. *et al.* (2021). Characterizing genetic intra-tumor heterogeneity across 2,658 human cancer genomes. *Cell*, **184**, 2239–2254.e39.

Dong,X. *et al.* (2017). Accurate identification of single-nucleotide variants in whole-genome-amplified single cells. *Nat. Methods*, **14**, 491–493.

Durante,M.A. *et al.* (2020). Single-cell analysis reveals new evolutionary complexity in uveal melanoma. *Nat. Commun.*, **11**, 496.

Gawad,C. *et al.* (2016). Single-cell genome sequencing: current state of the science. *Nat. Rev. Genet.*, **17**, 175–188.

Gröbner,S.N. *et al.* (2018). The landscape of genomic alterations across childhood cancers. *Nature*, **555**, 321–327.

Harbers,L. *et al.* (2021). Somatic copy number alterations in human cancers: an analysis of publicly available data from the cancer genome atlas. *Front. Oncol.*, **11**, 2877.

Hård,J. *et al.* (2019). Conbase: a software for unsupervised discovery of clonal somatic mutations in single cells through read phasing. *Genome Biol.*, **20**, 68.
Huang,W. *et al.* (2011). ART: a next-generation sequencing read simulator. *Bioinformatics*, **28**, 593–594.
Hubert,L. and Arabie,P. (1985). Comparing partitions. *J. Classif.*, **2**, 193–218.
Koboldt,D.C. *et al.* (2009). VarScan: variant detection in massively parallel sequencing of individual and pooled samples. *Bioinformatics*, **25**, 2283–2285.
Kuipers,J. *et al.* (2017). Advances in understanding tumour evolution through single-cell sequencing. *Biochim. Biophys. Acta Rev. Cancer*, **1867**, 127–138.
Lähnemann,D. *et al.* (2021). Accurate and scalable variant calling from single cell DNA sequencing data with ProSolo. *Nat. Commun.*, **12**, 6744.
Laks,E. *et al.* (2019). Clonal decomposition and DNA replication states defined by scaled single-cell genome sequencing. *Cell*, **179**, 1207–1221.e22.
Langmead,B. and Salzberg,S.L. (2012). Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.
Lawrence,M.S. *et al.* (2013). Mutational heterogeneity in cancer and the search for new cancer-associated genes. *Nature*, **499**, 214–218.
Lawrence,M.S. *et al.* (2014). Discovery and saturation analysis of cancer genes across 21 tumour types. *Nature*, **505**, 495–501.
Li,H. (2011). A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics*, **27**, 2987–2993.
Li,H. *et al.* (2008). Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res.*, **18**, 1851–1858.
Lloyd,S. (1982). Least squares quantization in PCM. *IEEE Trans. Inform. Theory*, **28**, 129–137.
Luquette,L.J. *et al.* (2019). Identification of somatic mutations in single cell DNA-seq using a spatial model of allelic imbalance. *Nat. Commun.*, **10**, 3908.
Ma,X. *et al.* (2018). Pan-cancer genome and transcriptome analyses of 1,699 paediatric leukaemias and solid tumours. *Nature*, **555**, 371–376.
Meyer,M. *et al.* (2012). A high-coverage genome sequence from an archaic Denisovan individual. *Science*, **338**, 222–226.
Mu,J.C. *et al.* (2014). VarSim: a high-fidelity simulation and validation framework for high-throughput genome sequencing with cancer applications. *Bioinformatics*, **31**, 1469–1471.
Myers,M.A. *et al.* (2020). Identifying tumor clones in sparse single-cell mutation data. *Bioinformatics*, **36** (**Suppl. 1**), i186–i193.
Navin,N. *et al.* (2011). Tumour evolution inferred by single-cell sequencing. *Nature*, **472**, 90–94.
Ng,A.Y. *et al.* (2001). On spectral clustering: analysis and an algorithm. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS'01*, pp. 849–856, MIT Press, Cambridge, MA.
Porter,M.A. *et al.* (2009). Communities in networks. *Not. Am. Math. Soc.*, **56**, 1082–1097.
Sherry,S.T. *et al.* (2001). dbSNP: the NCBI database of genetic variation. *Nucleic Acids Res.*, **29**, 308–311.
Singer,J. *et al.* (2018). Single-cell mutation identification via phylogenetic inference. *Nat. Commun.*, **9**, 5144.
Stratton,M.R. *et al.* (2009). The cancer genome. *Nature*, **458**, 719–724.
Tate,J.G. *et al.* (2018). COSMIC: the catalogue of somatic mutations in cancer. *Nucleic Acids Res.*, **47**, D941–D947.
Velazquez-Villarreal,E.I. *et al.* (2020). Single-cell sequencing of genomic DNA resolves sub-clonal heterogeneity in a melanoma cell line. *Commun. Biol.*, **3**, 318.
Zaccaria,S. and Raphael,B.J. (2021). Characterizing allele- and haplotype-specific copy numbers in single cells with CHISEL. *Nat. Biotechnol.*, **39**, 207–214.
Zack,T.I. *et al.* (2013). Pan-cancer patterns of somatic copy number alteration. *Nat. Genet.*, **45**, 1134–1140.
Zafar,H. *et al.* (2016). Monovar: single-nucleotide variant detection in single cells. *Nat. Methods*, **13**, 505–507.